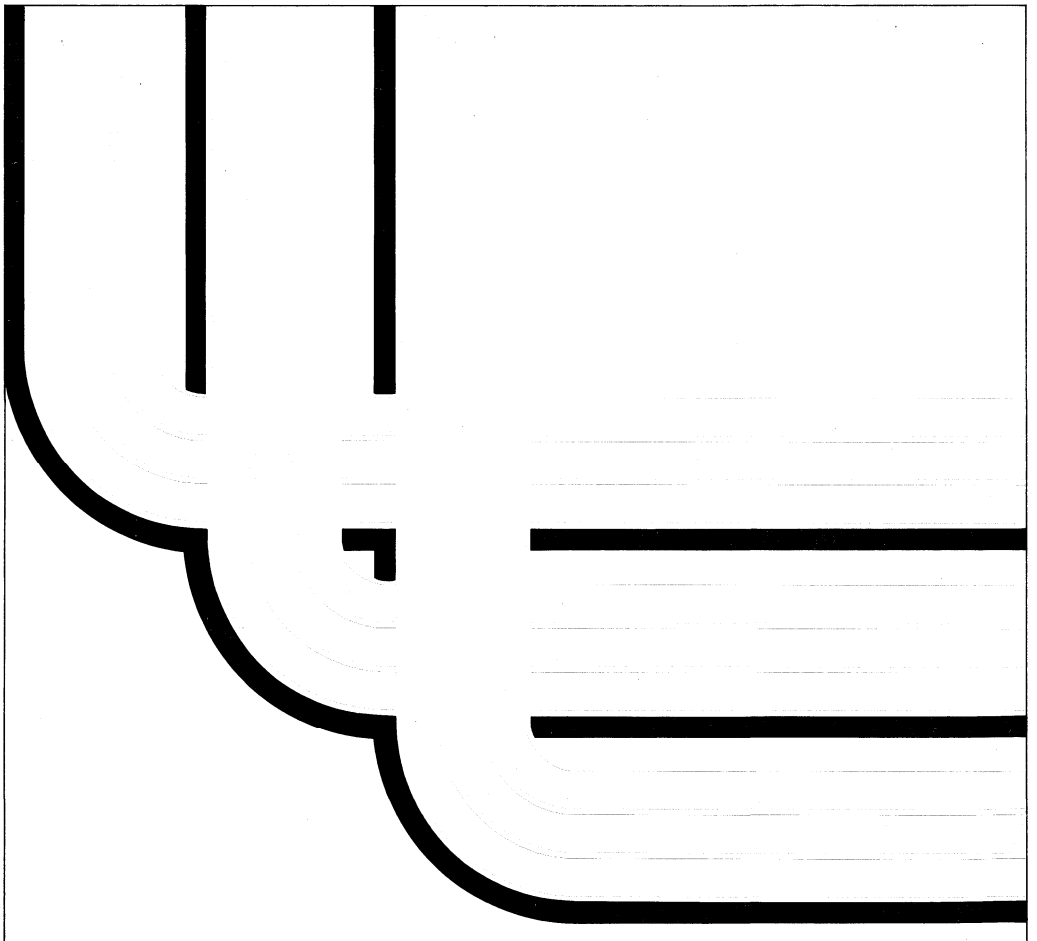
Application System/400

SC41-0030-02

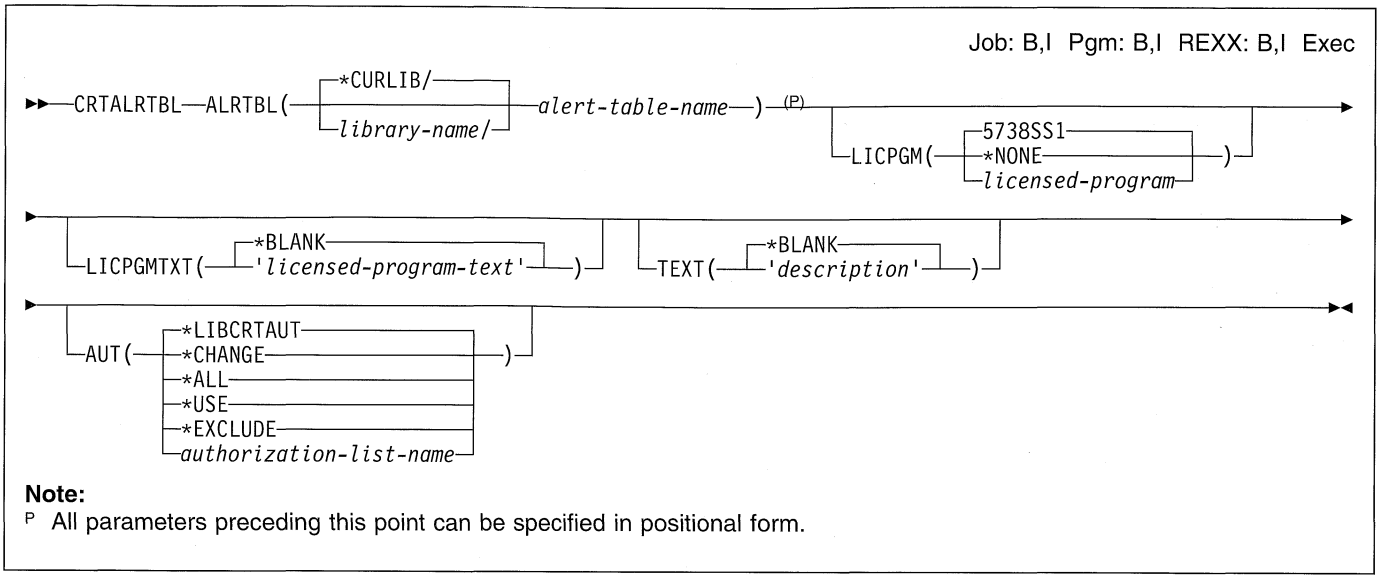
Programming:
Control Language Reference
CRTxxx through LODxxx Commands



System and Application Support

(2)

CRTALRTBL (Create Alert Table) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Alert Table (CRTALRTBL) command is used to create an alert table for storing alert description records. An alert table specifies the hardware and software products it is used for and defines alerts (problem notifications in a network). The typical user is the system or network programmer or operator responsible for network management. More information on alerts is in the *Alerts and DSNX Guide*.

Required Parameters

ALRTBL

Specifies the qualified name of the alert table that is created. An alert table has a one-to-one correspondence with a message file. To define an alert for a particular message, the name of the alert table must be the same as the name of a message file.

Note: The alert table and message file do not have to be in the same library. The alert table library does have to be in the library list of the job that causes the alert to be created.

The name of the alert table can be qualified by one of the following library values:

***CURLIB:** The alert table is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the alert table is created.

alert-table-name: Specify the name of the alert table that is created.

Optional Parameters

LICPGM

Specifies the licensed program that the alert table is identifying. The program identifier is included in the alert as product identification for the alert sender.

5738SS1: The licensed program for the OS/400 system is used.

***NONE:** There is no licensed program for this alert table. This value is allowed for products that do not have a licensed program.

licensed-program: Specify a 7-character identifier for the licensed program. This identifier retrieves release and level information for the program.

Note: The licensed program is not necessarily an IBM Licensed Program. Any 7-character identifier that is meaningful to the use of alerts can be specified (for example, USR1234). If the value is assigned to the system, the identifier and release level information are included in the alert. If the value is not known, only the identifier and text specified in LICPGMTXT are included in the alert.

LICPGMTXT

Specifies descriptive text for the licensed program (for example, the OS/400 system). This text is included in the alert as product identification for the alert sender.

***BLANK:** Text is not specified.

'licensed-program-text': Specify up to 30 characters of text that describe the licensed program.

TEXT

Specifies text that briefly describes the alert table. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

AUT

Specifies the authority given to users who do not have specific authority to the alert table, who are not on an authorization list, and whose user group has no specific authority to the alert table. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the alert table is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the alert table). The public authority is determined when the alert table is created. If the CRTAUT value for the library changes after the alert table is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the alert table.

***USE:** The user can perform basic operations on the alert table, such as running a program or reading a file. The user cannot change the alert table. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the alert table.

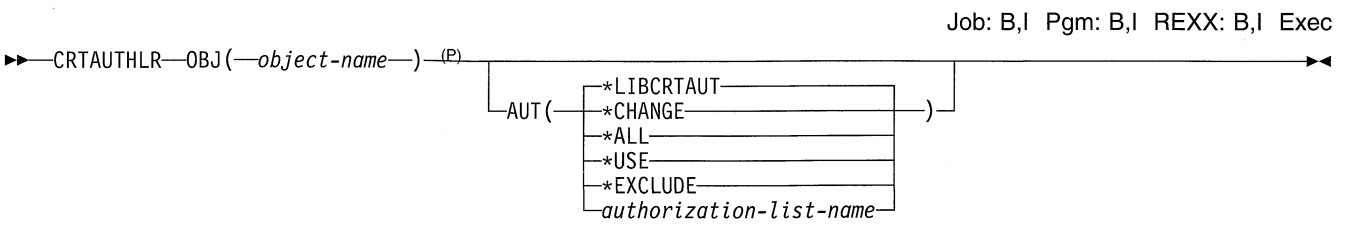
authorization-list-name: Specify the name of the authorization list used.

Example

```
CRTALRTBL  ALRTBL(ALRTBLLIB/ALRTBLNBR1)
  LICPGM(5728SS1)  LICPGMTXT('OS/400')
  TEXT('This is the first alert table created')
```

This command creates an alert table in the library ALRTBLLIB called ALRTBLNBR1. The public has *CHANGE authority to the table. The licensed program associated with this ALRTBL is the OS/400 system.

CRTAUTHLR (Create Authority Holder) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Authority Holder (CRTAUTHLR) command allows a user to create an authority holder to secure an object of type *FILE before it exists on the system. The file must be a program-described database file. When an object by the specified name is created, the authorities specified in the authority holder are linked to the newly created object.

The authority holder is associated with one specific object, object type, and library. This allows only users with the correct authority to access the object. The authority holder and associated object always have the same owner.

If the object has authorities associated with it, they are linked to the newly created authority holder. The owner of the object becomes the owner of the authority holder. Authority holders are located in library QSYS.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. The object type being secured by the new authority holder is limited to *FILE. The file must be a program-described database file.
3. The authority holder cannot be created for objects located in libraries QRCL, QRECOVERY, QSPL, QSYS, or QTEMP.

Required Parameter

OBJ

Specifies the qualified name of the database file that the authority holder secures when it is created.

Optional Parameter

AUT

Specifies the authority given to users who do not have specific authority to the object, who are not on an authorization list, and whose user group has no specific authority to the object.

***LIBCRTAUT:** The public authority for the object is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the object). The public authority is determined when the object is created. If the CRTAUT value for the library changes after the object is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the object.

***USE:** The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the object.

authorization-list-name: Specify the name of the authorization list used.

Examples

Example 1: Granting EXCLUDE Authority

```
CRTAUTHLR  OBJ(QGPL/FIL1)  AUT(*EXCLUDE)
```

By running this command, user ONE creates an authority holder with *EXCLUDE authority.

Example 2: Granting USE Authority

```
GRTOBJAUT  OBJ(QGPL/FIL1)  TYPE(*FILE)  USER(TWO)
  AUT(*USE)
```

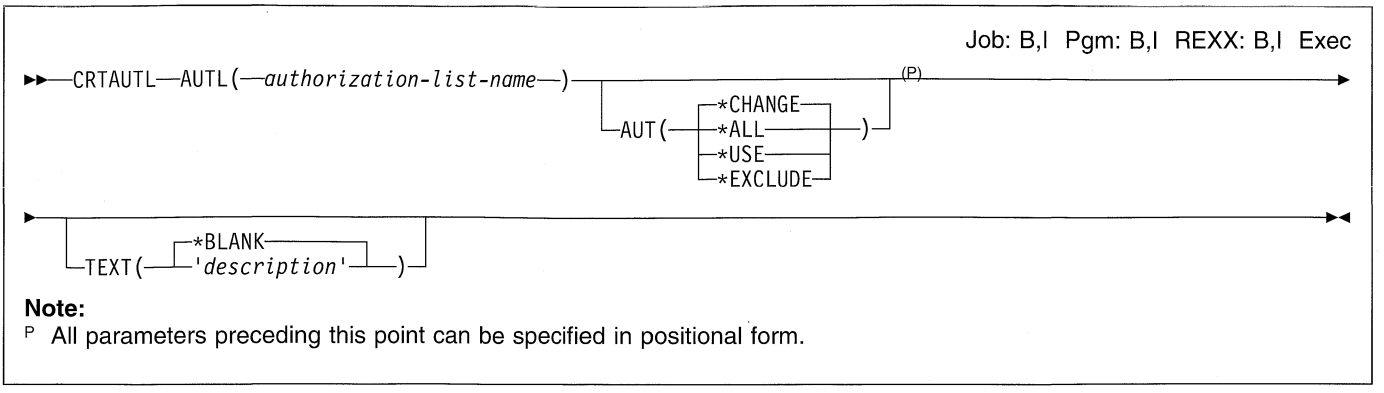
By running this command, user ONE grants *USE authority to user TWO for the specified authority holder.

Example 3: Granting Matching Authority

```
CRTSRCF  FILE(QGPL/FIL1)
```

By running this command, user ONE creates a file that has a matching authority holder. User ONE becomes the owner of the file with user TWO having *USE authority to QGPL/FIL1.

CRTAUTL (Create Authorization List) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Authorization List (CRTAUTL) command creates an authorization list. Authorization lists are used to grant to a set of users specified authorities to an object or to a set of objects. Each user has the authorities to all of the objects. When an authorization list is granted authority to an object, the users on the list get authority to the object. The authorities they receive are those specified for them in the authorization list.

A user's authority on an authorization list can be overridden by granting the user specific authority to the object. Specific authorities override the user's authority specified in the authorization list. Specific authorities override the user's group authority, if both are specified.

If public authority specified for the object is *AUTL, then the public authority specified on the authorization list is used.

Restrictions:

1. Authorization lists cannot be used to specify the authority for user profiles or other authorization lists.
2. Only one authorization list can be used to specify the authority for an object.

Required Parameter

AUTL

Specifies the name of the authorization list being created. Up to 10 alphanumeric characters can be used. The authorization list name follows the system naming conventions.

Optional Parameters

AUT

Specifies the authority given to users who do not have specific authority to the object, who are not on an authorization list, and whose user group has no specific authority to the object.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the object.

***USE:** The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the object.

TEXT

Specifies text that briefly describes the authorization list. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

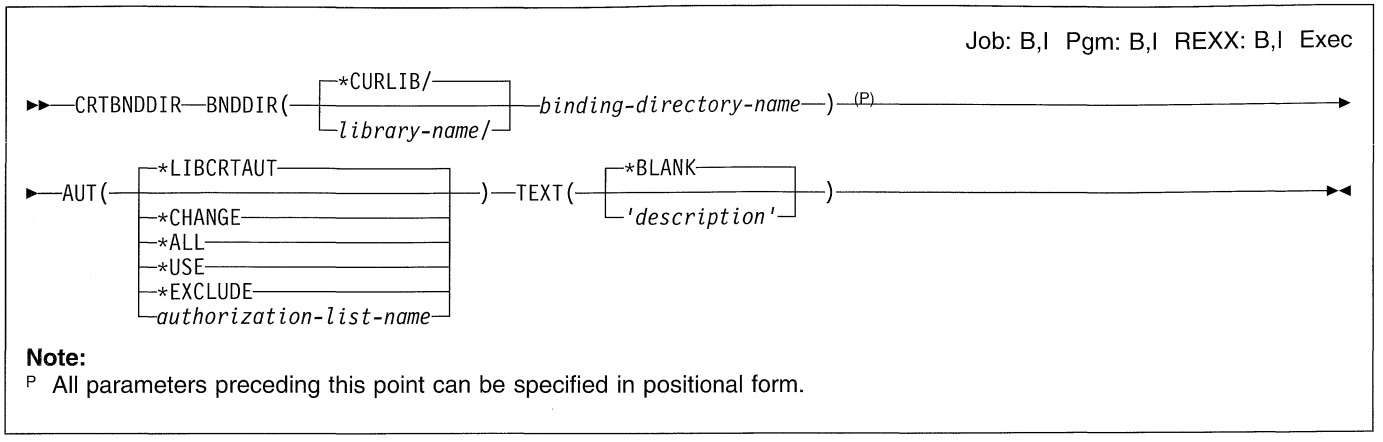
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTAUTL  AUTL(PROGMR)  AUT(*CHANGE)
  TEXT('Programmers authorization list')
```

This command creates an authorization list (PROGMR). If an object whose authority comes from the authorization list has specified USER(*PUBLIC) as AUT(*AUTL), the users who do not have specific authority and whose group does not have specific authority to the object are given *CHANGE authority for the object secured by this authorization list.

CRTBNDDIR (Create Binding Directory)



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Binding Directory (CRTBNDDIR) command creates a binding directory object in the specified library.

Restriction: You must have *READ and *ADD authority for the library where the binding directory is created.

Required Parameters

BNDDIR

Specifies a binding directory name to be created.

The name of the binding directory can be qualified by one of the following library values:

*CURLIB: The binding directory is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the binding directory is created.

binding-directory-name: Specify the name of the binding directory to be created.

AUT

Specifies the authority given to users who do not have specific authority to the binding directory, who are not on an authorization list, and whose user group has no specific authority to the binding directory.

*LIBCRTAUT: The public authority for the binding directory is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the binding directory). The public authority is determined when the binding directory is created. If the CRTAUT value for the library changes after the binding directory is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the binding directory.

***USE:** The user can perform basic operations on the binding directory, such as running a program or reading a file. The user cannot change the binding directory. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the binding directory.

authorization-list: Specify an authorization list name. The list must exist when the object is created. Users are granted authority to the object as specified by the list.

TEXT

Specifies text that briefly describes the binding directory object. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*BLANK: Text is not specified.

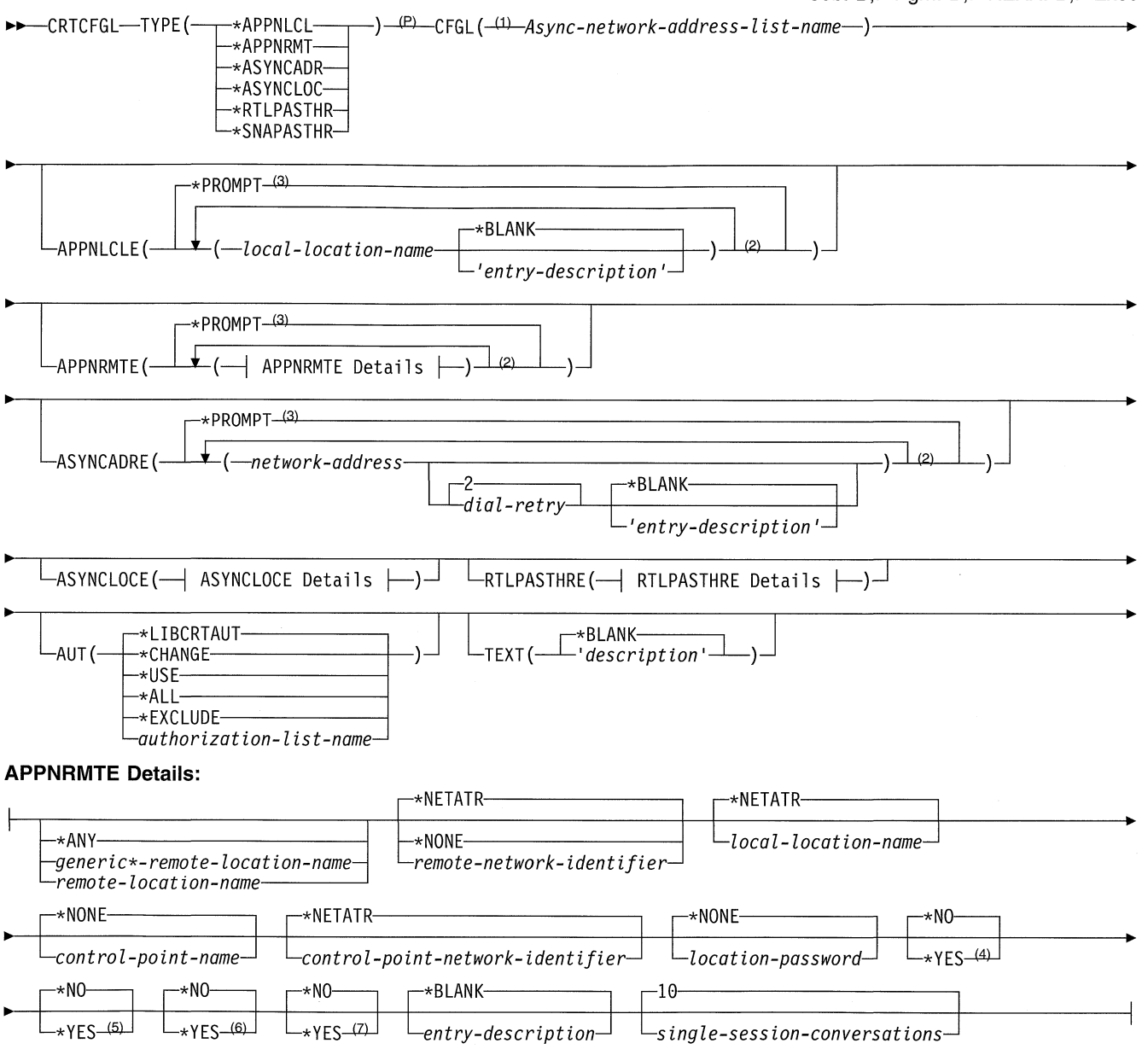
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTBNDDIR   BNDDIR(DISPLAYS)
```

This command creates the binding directory named DISPLAYS in the current library or QGPL.

CRTCFGL (Create Configuration List) Command



APPNRMTE Details:

Notes:

1 Required only for TYPE(*ASYNCADR)

2 A maximum of 50 repetitions

3 The value *PROMPT cannot be used in batch mode.

4 Location Security

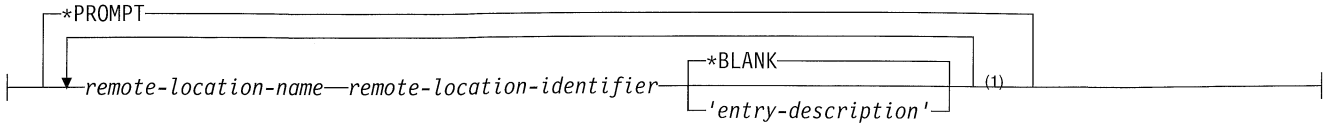
5 Single Session Location

6 Locally Controlled Session

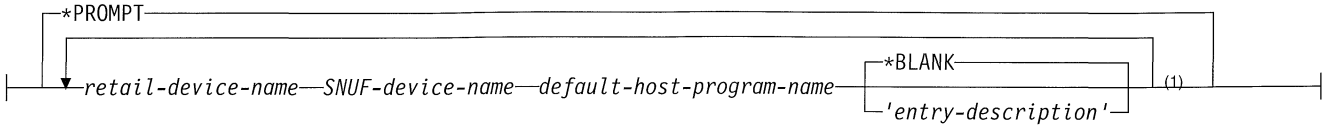
7 Preestablished Session

P All parameters preceding this point can be specified in positional form.

ASYNCLOCE Details:



RTLPASTHRE Details:



Note:
1 A maximum of 50 repetitions

Purpose

The Create Configuration List (CRTCFGL) command creates a configuration list.

Required Parameters

TYPE

Specifies the type of configuration list that will be created.

***APPNLCL:** An advanced peer-to-peer networking (APPN) local location list is created. Up to 476 APPN local location entries are allowed in the configuration list.

***APPNRMT:** An APPN remote location list is created. Up to 1898 APPN remote location entries are allowed in the configuration list.

***ASYNCADR:** An asynchronous network address list is created. Up to 294 asynchronous network address entries are allowed in the configuration list.

***ASYNCLOC:** An asynchronous remote location list is created. Up to 4995 asynchronous remote location entries are allowed in the configuration list.

***RTLPASTHR:** A retail pass-through list. Up to 450 retail pass-through entries are allowed in the configuration list.

***SNAPASTHR:** An SNA pass-through list. Up to 254 SNA pass-through entries are allowed in the configuration list. Only the text description for an SNA configuration list can be specified using this command.

Notes:

1. To add an entry to an SNA configuration list, use the Add Configuration List Entry (ADDCFGLE) command.
2. To add a device name and description to an existing entry, use the Change Configuration List Entry (CHGCFGLE) command.

CFGL

Specifies the name of the configuration list. This parameter is valid only when *ASYNCADR is specified on the TYPE parameter. Only one of the other configuration list types is allowed on a system. The list types have system-supplied names: QAPPNLCL, QAPPNRMT, QASYNCADR, QASYNCLOC, QRTLPASTHR, QSNAPASTHR.

Optional Parameters

APPNLCLE

Specifies the APPN local location entry. This value is required if *APPNLCL is specified for the TYPE parameter.

***PROMPT:** The value of *PROMPT allows the user to add, remove, and change entries using a full-screen entry display.

Element 1: Operation on a Local Entry

local-location-name: Specify the local location name residing on the local system. These names are used by the local APPN controller to determine whether a request being received is intended for its system or another system in the network. The local location name must be unique, and it cannot already exist as a remote location name used by configuration list QAPPNRMT or be specified on another system as a local location in the same APPN network.

Element 2: Describing a Local Entry

***BLANK:** Text is not specified.

'entry-description': Specify a short description for each local entry.

APPNRMTE

Specifies the APPN remote location entry. This value is required if *APPNRMT is specified for the TYPE parameter.

***PROMPT:** The value of *PROMPT allows the user to add, remove, and change entries using a full-screen entry display.

Element 1: Remote Location Name

***ANY:** The system potentially accepts all requests sent to it.

generic-remote-location-name:* Specify the generic name (part of a name followed by an asterisk) of the remote location(s) to be changed. The generic location name allows one directory entry to be defined for all locations, on a single control point, with a name that matches the characters preceding an *. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be created only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

remote-location-name: Specify the full name of a remote location.

Element 2: Remote Network Identifier

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

remote-network-identifier: Specify the network identifier of the network in which the remote location resides.

Element 3: Local Location Name

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the location name on the local system. This name is used by APPN to match a local/remote location pair entry.

Element 4: Control Point Name

***NONE:** There is no control point name.

control-point-name: Specify the name of the control point that provides network functions for the remote location. This field is required if the entry is either a generic name or *ANY.

Element 5: Control Point Network Identifier

***NETATR:** The LCLNETID value specified in the system network attributes is used.

control-point-network-identifier: Specify the network identifier of the network in which the control point resides.

Element 6: Location Password

***NONE:** There is no location password.

location-password: Specify the password used when establishing sessions on the local location/remote location name pair. It must be an even number of hexadecimal characters.

Element 7: Location Security

This value specifies whether the local location allows the remote location to verify user passwords when the remote location sends program start requests to the local location.

***NO:** The remote location is not allowed to verify user passwords when sending program start requests to the local location.

***YES:** The remote location is allowed to verify user passwords when sending program start requests to the local location.

Element 8: Single Session Location

This value specifies whether the connection between the local location and remote location is a single session connection.

***NO:** The connection between the local location and the remote location is not a single session connection.

***YES:** The connection between the local location and the remote location is a single session connection.

Element 9: Locally Controlled Session

This value specifies, for single connection sessions only, whether the connection between the local location and remote location is locally controlled.

***NO:** The single session connection is not locally controlled.

***YES:** The single session connection is locally controlled.

Element 10: Preestablished Session

This value specifies whether the single session is automatically bound when the mode is started between the local location and remote location.

***NO:** The single session connection is not automatically bound between the local and remote location.

***YES:** The single session connection is automatically bound between the local and remote location.

Element 11: Remote Entry Description

***BLANK:** Text is not specified.

'entry-description': Specify up to 20 characters of text, enclosed in apostrophes, for the entry description.

Element 12: Number of Single-Session Conversations

10: The number of single session conversations allowed for this device description is ten.

single-session-conversations: Specify the number of conversations allowed for a single session of this device description. Valid values range from 1 through 512.

Note: The combination of remote location name, remote network identifier, and local location

name must be unique. Also, the remote location name cannot already exist as a local location in configuration list QAPPNLCL, or as the current value for either LCLLOCNAME or CPNAME network attribute.

ASYNCADRE

Specifies the asynchronous network address entry. This value is required if *ASYNCADR is specified for the TYPE parameter.

Element 1: Operation on a Network Address Entry

***PROMPT:** The value of *PROMPT allows the user to add, remove, and change entries using a full-screen entry display.

network-address: Specify the X.25 network address. This value must contain only digits ranging from 0 to 9.

Element 2: Number of Dial Retries

2: The default number of retries is two.

dial-retry: Specify the number of times dialing is retried (because errors occur) before the next number on the list is dialed. The valid number of dial retries ranges from 1 through 255.

Element 3: Description of Network Address Entry

***BLANK:** Text is not specified.

'entry-description': Specify a short description for each remote network address entry. Specify no more than 20 characters of text, enclosed in apostrophes.

ASYNCLOCE

Specifies the asynchronous remote location entry. This value is required if *ASYNCLOC is specified for the TYPE parameter.

***PROMPT:** The value of *PROMPT allows the user to add, remove, and change entries using a full-screen entry display.

Element 1: Remote Location Name

This name, in combination with the remote location identifier, is used to determine whether an incoming call is accepted. This value must be unique on the system.

remote-location-name: Specify the full name of a remote location.

Element 2: Remote Location Identifier

remote-location-identifier: Specify an identifier which, when combined with the remote location name, determines whether an incoming call is accepted. This identifier must be the same as the remote system's local identifier.

Element 3: Description of Remote Location Entry

***BLANK:** Text is not specified.

'entry-description': Specify a short description for each remote location entry.

RTLPASTHRE

Specifies the retail pass-through entry. This value is required if TYPE(*RTLPASTHR) is specified. Up to of 50 entries can be specified for this parameter.

The values for the retail-pass-through entry can be specified as a single value (*PROMPT) or as a list of values (elements).

***PROMPT:** The value of *PROMPT allows the user to add, remove, and change entries using a full-screen entry display.

Element 1: Retail Device Name

retail-device-name: Specify the name of the retail device to use for the pass-through session. This must be a unique value.

Element 2: SNA Upline Facility Device Name

SNUF-device-name: Specify the name of the host device to use for the pass-through session. This must be a unique value. retail controller.

Element 3: Default Host Program Name

default-host-program-name: Specify the name of the program to be started on the host if a program name was not specified by the retail controller.

Element 4: Text Description

***BLANK:** Text is not specified.

'entry-description': Specify a short description for each retail pass-through entry.

AUT

Specifies the authority given to users who do not have specific authority to the configuration list, who are not on an authorization list, and whose user group has no specific authority to the configuration list.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the configuration list, such as running a program or reading a file. The user cannot change the configuration list. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change

the object, and perform basic functions on the object. The user can change ownership of the configuration list.

***EXCLUDE:** The user cannot access the configuration list.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the configuration list. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

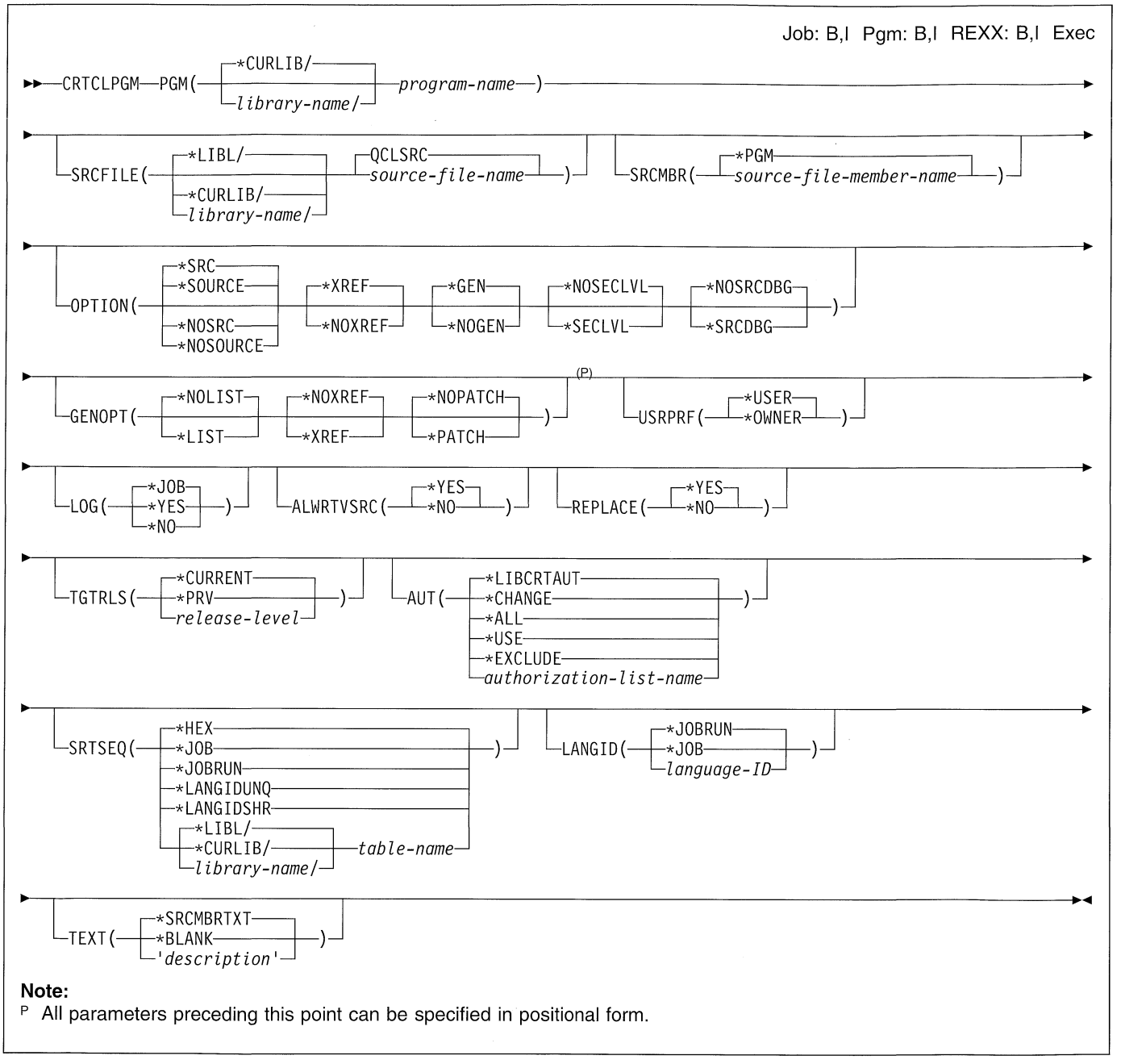
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCFGL   TYPE(*APPNLCL)
```

This command brings up a full-screen entry display that allows the user to add new entries to configuration list QAPPNLCL.

CRTCLPGM (Create Control Language Program) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Control Language Program (CRTCLPGM) command creates a CL program from the specified CL source program. The command is valid in batch and interactive jobs, and in both compiled and interpreted CL.

Restriction: The amount of auxiliary storage occupied by a compiled program varies with the number of commands in the program, the kinds of functions performed by the commands (for example: display, create, add, and call), and the kinds of parameter values specified (variables versus constants). Some combinations of these factors can cause the system internal size limits for the program to be exceeded (an unlikely occurrence). When the limits are exceeded, the program must be rewritten, usually as several programs instead of one program.

Required Parameter

PGM

Specifies the qualified name of the compiled program.

The name of the program can be qualified by one of the following library values:

***CURLIB:** The program is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the program is created.

program-name: Specify the name of the program being created.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file that contains the CL source program being compiled. The program is known by the name given in the PGM parameter.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCLSRC: The IBM-supplied source file, QCLSRC, that contains the CL source program being compiled is used.

source-file-name: Specify the name of the source file that contains the CL source program being compiled. The source file can be a database file, a device file, or an inline data file.

SRCMBR

Specifies the name of the member of the source file that contains the CL source program being compiled.

***PGM:** The CL source program being compiled is in the member of the source file that has the same name as that specified in the PGM parameter for the compiled program.

source-file-member-name: Specify the name of the member that contains the CL source program, if the member name is not the same as the name of the program being created.

OPTION

Specifies the types of output lists created when this command is processed and whether a program is created. Up to three of the following values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value will be used.

Note: The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

Source Listing Options

***SRC** or ***SOURCE:** The compiler creates a list of the source input used to compile the program.

***NOSRC** or ***NOSOURCE:** A complete compiler source list is not created; only compiler errors are listed.

Cross-Reference Options

***XREF:** A cross-reference listing of the source program is produced.

***NOXREF:** No cross-reference list of references to variables and data items in the source is created.

Program Creation Options

***GEN:** The compiler creates a program that can run after the program is compiled.

***NOGEN:** A program object is not created. The compiler checks only the syntax of the statements.

Second-Level Message Text Options

***NOSECLVL:** No second level message text is printed.

***SECLVL:** Second-level message text is printed.

Source-Level Debug Options

***NOSRCDBG:** Source-level error and debug information is not generated.

***SRCDBG:** The compiler generates source-level error and debug information. An event file is created even if the compiler completes the process without error.

GENOPT

Specifies whether the intermediate representation of a program, a cross-reference list of objects defined in the intermediate representation of a program, an attribute list from the intermediate representation of a program, and the program template are printed, and whether a program patch area is included in the compiled program. This information may be required if a problem occurs in RPG/400. This parameter also specifies whether program optimization is done.

Note: Up to three of the following values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value will be used. The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

***NOLIST:** An intermediate representation of the program listing is not created.

***LIST:** An intermediate representation of the program is included in the compiler listing.

***NOXREF:** No cross-reference list of variable and data item references in the intermediate representation of the program is created.

***XREF:** A cross-reference listing of all objects defined in the intermediate representation of the program is produced.

***NOPATCH:** No space is reserved in the compiled program for a program patch area.

***PATCH:** Space is reserved in the compiled CL program for a program patch area.

USRPRF

Specifies whether the authority checking done while this program is running includes only the user who is running the program (*USER) or both the user running the program and the program owner (*OWNER). The profiles of the program user or both the program user and the program owner are used to control which objects can be used by the program, including the authority the program has for each object. Only the program owner or a user with QSECOFR authority can change the user profile attribute.

Note: This parameter will be ignored if REPLACE(*YES) is specified.

***USER:** The program runs under the user profile of the program's user.

***OWNER:** The user profiles of both the program's owner and the program's user are used when the program is processed. The collective sets of object authority in both user profiles are used to find and access objects during program processing. Authority from the owning user profile's group profile is not included in the authority for the running program.

LOG

Specifies the logging options for a created CL program.

***JOB:** Logging of commands in a running CL program depends on the status of the job's logging flag (see the LOGCLPGM parameter of the Change Job (CHGJOB) command). To list the logged commands, the logging level of the jobs must be 3 or 4.

A *YES or *NO value takes precedence over any value specified in the CHGJOB command.

***YES:** Commands are logged in all cases.

***NO:** Commands are not logged.

ALWRTVSRC

Specifies whether the source for the CL program is saved with the program. This source can be retrieved later by using the Retrieve CL Source (RTVCLSRC) command.

***YES:** The source for the CL program is saved with the program.

***NO:** The source for the CL program is not saved with the program.

Notes:

1. If the RTVCLSRC command specifies a program that does not have the source saved with the program, a message is sent, and the RTVCLSRC command is not processed.
2. Specifying ALWRTVSRC(*NO) does not prevent commands from being logged while a CL program is running. To prevent logging of commands in the program, LOG(*NO) must be specified on the CRTCLPGM command.

REPLACE

Specifies whether an existing program is replaced if a program with the same name already exists in the specified library.

***YES:** Replace the existing program by moving it to the QRPLOBJ library.

Notes:

1. If a running CL program is recompiled with *YES specified for the REPLACE parameter, message queue errors may occur in the running CL program.
2. Specifying *YES for this parameter will cause the values of USRPRF (user profile) and AUT (authority) parameters to be ignored. The existing program is used as the source of authority, and the user profile attribute is copied from the existing program to the new program. Use the Change Program (CHGPGM) command to change the user profile and the Grant Object Authority (GRTOBJAUT) or Revoke Object Authority (RVKOBJAUT) commands to change the authority for the program.

***NO:** No replacement occurs.

TGTRLS

Specifies the release level of the operating system on which you intend to use the object being created.

The release level is specified in the format VxRxMx, where Vx is the version, Rx is the release, and Mx is the modification level. The release level V2R3M0 is Version 2, Release 3, Modification 0.

Note: To use the object on the target system, you must save the object to the target release level specified on the create command and then restore it on the target system.

If your system is running V2R3M0 and you want to create a program object for distribution to a V2R2M0 system you must:

1. Create the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
2. Save the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
3. Restore the program on the V2R2M0 system. The program object can also be restored to a V2R3M0 or later system.

***CURRENT:** The object is to be used on the release of the operating system currently running on your system. If V2R3M0 is running on the system, *CURRENT means that you intend to use the object on a system with V2R3M0 installed. You also can use the object on a system with any later release of the operating system installed.

***PRV:** The object is to be used on the previous release with modification level 0 of the operating system. If V2R3M0 is running on the system, *PRV means that you intend to use the object on a system with V2R2M0 installed. You also can use the object on a system with any later release of the operating system installed.

release-level: Specify the release level in the format VxRxMx. The object can be used on a system with the specified release or with any later release of the operating system installed.

Valid values depend on the current version, release, and modification level, and they change with each new release. Valid values for release V2R3M0 are: V2R3M0 and V2R2M0.

AUT

Specifies the authority given to users who do not have specific authority to the command language program, who are not on an authorization list, and whose user group has no specific authority to the command language program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: This parameter will be ignored if REPLACE(*YES) is specified.

***LIBCRTAUT:** The public authority for the command language program is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the command language program). The public authority is determined when the command language program is created. If the CRTAUT value for the library changes after the command language program is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the command language program.

***USE:** The user can perform basic operations on the command language program, such as running a program or reading a file. The user cannot change the command language program. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the command language program.

authorization-list-name: Specify the name of the authorization list used.

SRTSEQ

Specifies the sort sequence table to be used for string comparisons for this CL program. The sort sequence is used with the language identifier and the coded character set identifier of the job to determine the sort sequence table to use.

Note: SRTSEQ(*HEX) must be specified when a previous release is specified on the TGTRLS parameter.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.

***JOB:** The sort sequence used is the SRTSEQ associated with the job when the CL program is created.

***JOBRUN:** The sort sequence used is the SRTSEQ associated with the job when the CL program is run.

***LANGIDUNQ:** A unique weight sort table is used. The sort sequence table must contain a unique weight for multiple characters and is the unique weight table associated with the language specified on the LANGID parameter.

***LANGIDSHR:** A shared weight sort table is used. The sort sequence table can contain the same weight for multiple characters and is the shared weight table associated with the language specified on the LANGID parameter.

The name of the sort sequence table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify the name of the sort sequence table to be used with this CL program.

LANGID

Specifies the language identifier to be used when SRTSEQ(*LANGIDUNQ) or SRTSEQ(*LANGIDSHR) is specified.

***JOBRUN:** The language ID used is the LANGID associated with the job when the CL program is run.

***JOB:** The language ID used is the LANGID associated with the job when the CL program is created.

language-ID: Specify the language identifier to be used by the job.

TEXT

Specifies text that briefly describes the compiled CL program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member being used to create the CL program. If the source file is an inline file or a device file, the text is blank.

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Compiling a Program to be Run by Any System User

```
CRTCLPGM   PAYROLL TEXT('Payroll Program')
```

This command calls the CL compiler to compile a program named PAYROLL. The source program is in the default source file QCLSRC in the member PAYROLL. A compiler listing is created. The program is processed under the program user's user profile and can be run by any system user.

Example 2: Compiling a Program to be Run by an Authorized User

```
CRTCLPGM   PGM(PARTS)  SRCFILE(MYLIB/PARTDATA)
  AUT(*EXCLUDE)
  TEXT('This program displays all parts data')
```

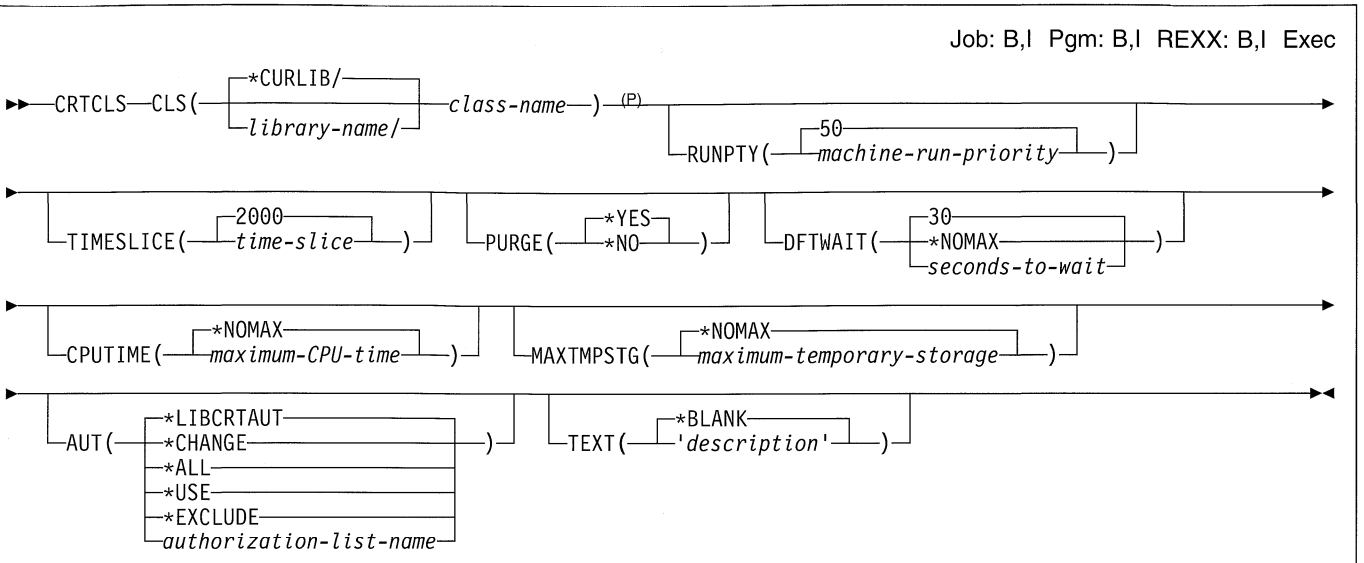
This command creates a CL program named PARTS and stores it in the current library. The source for the program is in the PARTS member of the source file PARTDATA in the library MYLIB. A compiler listing is created. This program can be processed under the profile of the user that is running the program, who could be the owner or another user to which the owner has granted specific authorization by name in the Grant Object Authority (GRTOBJAUT) command.

Example 3: Compiling a Program to be Run on a Previous Release System

```
CRTCLPGM   PGM(MYPGM)  SRCFILE(MYLIB/MYDATA)
  TGTRLS(*PRV)
```

This command creates a CL program that can be saved for a previous release system, restored on that system, and run on that system.

CRTCLS (Create Class) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Class (CRTCLS) command creates a class object and specifies the attributes to be contained in the class. The class defines the processing parameters for jobs that use the class; the class used by a job is specified in the subsystem description routing entry that is used to start the job.

Required Parameter

CLS

Specifies the qualified name of the class. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

The name of the class can be qualified by one of the following library values:

***CURLIB:** The class is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the class is created.

class-name: Specify the name of the class.

Optional Parameters

RUNPTY

Specifies the priority of running jobs using the class being created. Machine priority of running is a value, ranging from 1 (highest priority) through 99 (lowest priority), that represents the importance of the job when it competes with other jobs in the class for the machine resources. This value represents the relative, not absolute, importance of the job. For example, a job with a priority of running of 25 is *not* twice as important as one with a priority of running of 50.

50: Jobs that use this class have a priority of running of 50.

machine-run-priority: Specify the priority of running of jobs using this class.

TIMESLICE

Specifies the maximum amount of processor time (in milliseconds) given to a job using this class before other jobs are given the opportunity to run. The time slice establishes the amount of time needed by the job to accomplish a meaningful amount of processing. At the end of the time slice, the job may be put in an inactive state so that other jobs can become active in the storage pool.

2000: A maximum running time of 2,000 milliseconds is allocated to each job each time it is allowed to process.

time-slice: Specify the maximum amount of time (in milliseconds) that each job in this class can have to run when it is given processing time. Valid entries range from 1 through 9999999 (that is, 9,999,999 milliseconds or 9999.999 seconds).

PURGE

Specifies whether the job is marked eligible to be moved out of main storage and put into auxiliary storage at the end of a time slice or when there is a long wait (such as waiting for a work station user's response).

***YES:** The job is eligible to move out of main storage and into auxiliary storage.

***NO:** The job is not to move out of main storage. However, when some of main storage is needed for other jobs running in the same storage pool, pages belonging to this job are moved (eight at a time) to auxiliary storage to accommodate pages needed for other jobs. When this job runs again, its pages are returned to main storage as they are needed.

DFTWAIT

Specifies the default maximum wait time (in seconds) that processing of a job is held up until a system instruction that performs a wait finishes running. This default wait time is used when a wait time is not otherwise specified for a given situation. Normally, this would be the amount of time the system user is willing to wait for the system before the request is canceled.

If the wait time for any one instruction is exceeded, an error message can be displayed or it can be automatically handled by a Monitor Message (MONMSG) command.

30: An instruction wait has up to 30 seconds in which to complete.

***NOMAX:** There is no limit on the time the system waits for the completion of the operation.

seconds-to-wait: Specify a value, ranging from 1 through 9999999 seconds, that specifies the maximum time that the system waits for the system instruction to finish running.

CPUTIME

Specifies the maximum processing unit time (in milliseconds) that a job's routing step using this class has to completely run the entire job. If not finished before the maximum time is used, running of the job is ended.

***NOMAX:** There is no limit on the time the system waits for the completion of the operation.

maximum-CPU-time: Specify the maximum amount of processing unit time (in milliseconds) that the job's routing step has in which to finish running. Valid entries range from 1 through 9999999 (that is, 9,999,999 milliseconds or 9999.999 seconds).

MAXTMPSTG

Specifies the maximum amount of temporary (auxiliary) storage (in kilobytes) that a job in this class can use for processing. This temporary storage is used for storage required by the program itself and by implicitly created internal system objects used to support the job. It is not storage in the QTEMP library. If the maximum temporary storage is exceeded by a job, the job is ended. This parameter does not apply to the use of permanent storage, which is controlled through the user profile.

***NOMAX:** The system maximum is used.

maximum-temporary-storage: Specify a value in kilobytes (ranging from 1 through 9999999) that specifies the maximum amount of temporary storage of a job in this class.

AUT

Specifies the authority given to users who do not have specific authority to the class, who are not on an authorization list, and whose user group has no specific authority to the class.

***LIBCRTAUT:** The public authority for the class is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the class). The public authority is determined when the class is created. If the CRTAUT value for the library changes after the class is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the class.

***USE:** The user can perform basic operations on the class, such as running a program or reading a file. The user cannot change the class. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the class.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the class being created. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

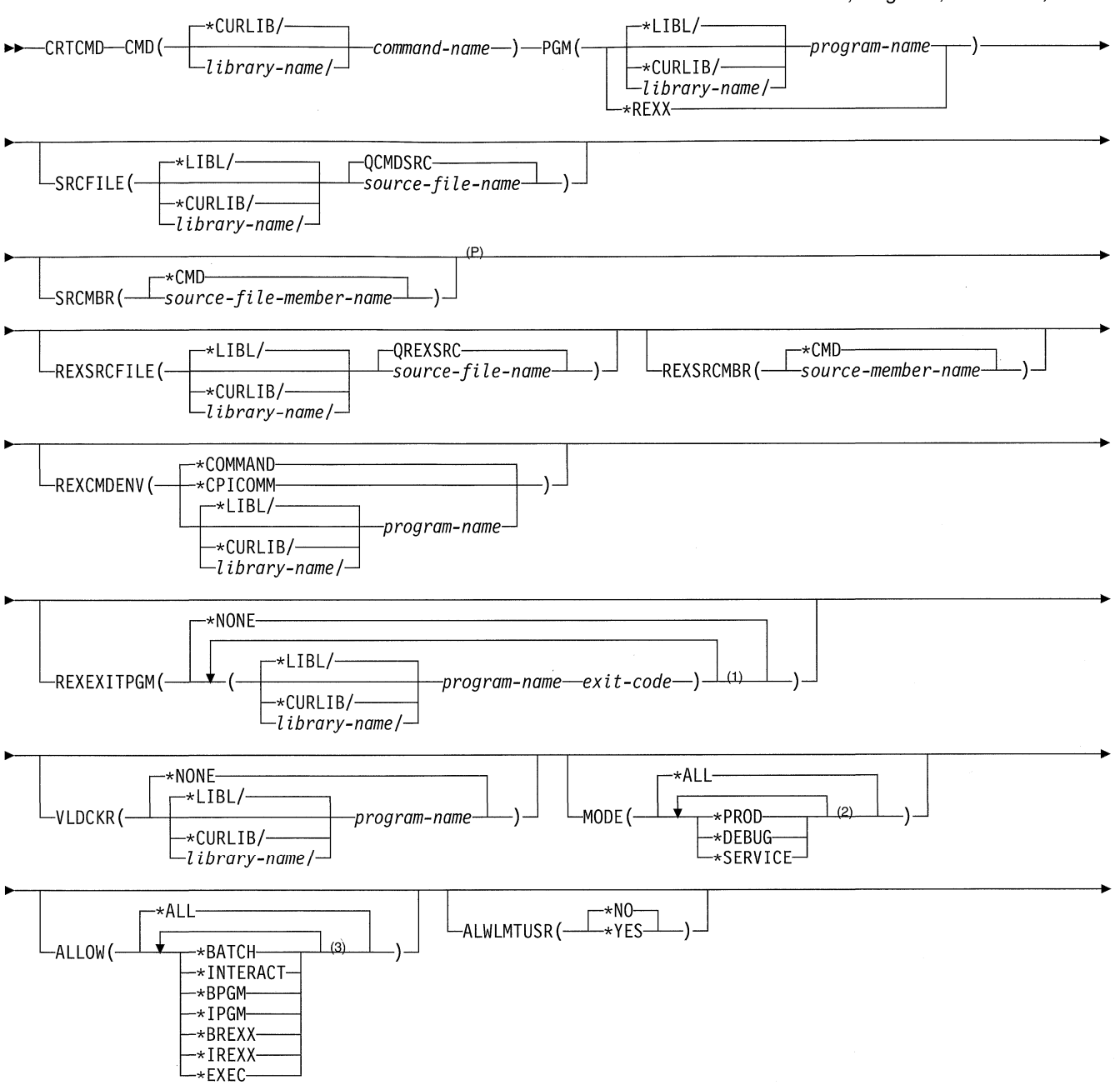
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCLS   CLS(CLASS1)  RUNPTY(60)  TIMESLICE(900)
  TEXT('This class for all batch jobs
  from Dept 4836')
```

This command creates a class called CLASS1. The class is stored in the current library specified for the job. The user text 'This class for all batch jobs from Dept 4836' describes the class. The attributes of this class provide a machine priority of running of 60 and a time slice of 900 milliseconds. If the job has not finished running at the end of a time slice, it is eligible to be moved out of main storage until it is allocated another time slice. The defaults for the other parameters are assumed.

CRTCMD (Create Command) Command



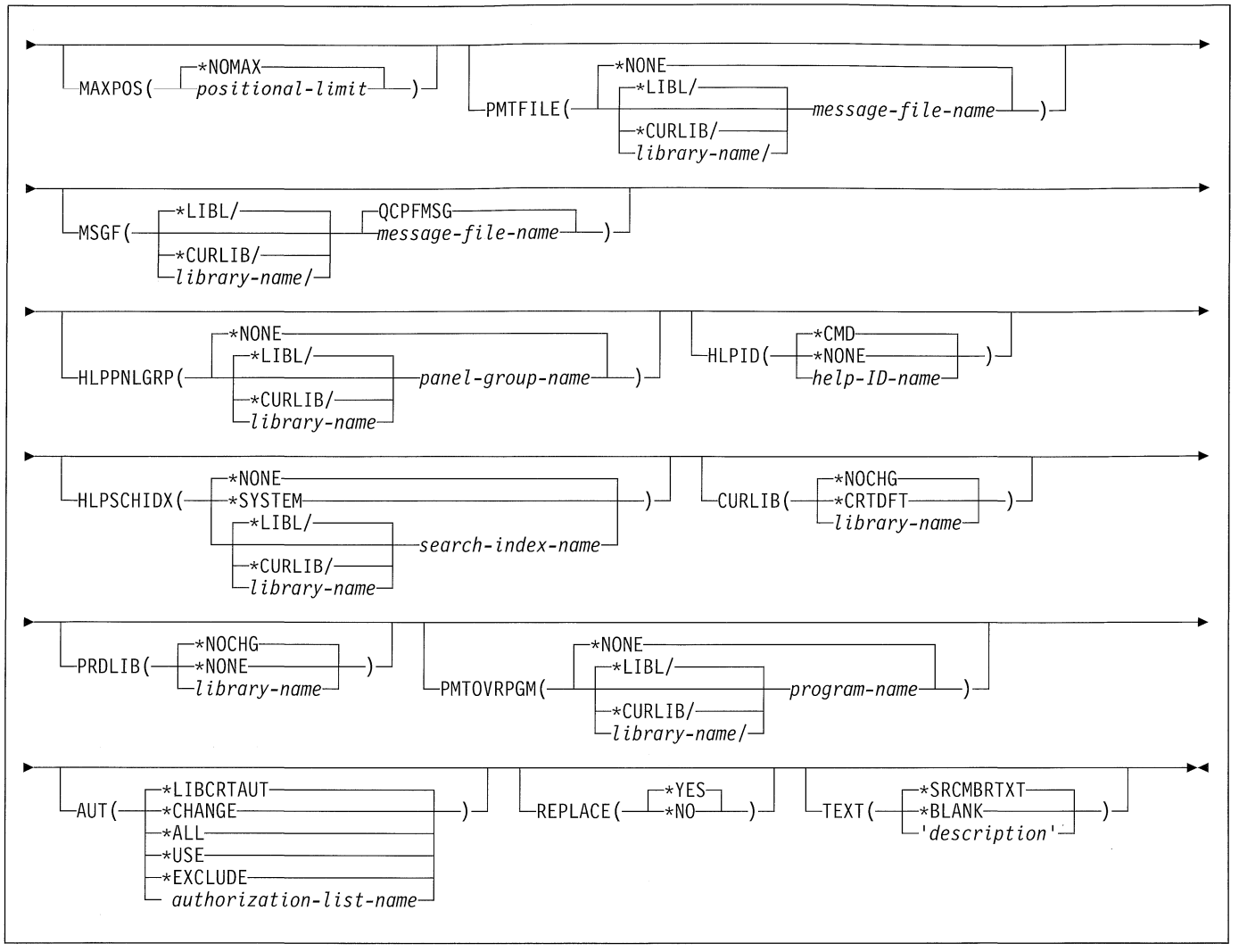
Notes:

| P All parameters preceding this point can be specified in positional form.

1 A maximum of 8 repetitions

2 A maximum of 3 repetitions

3 A maximum of 7 repetitions



Purpose

The Create Command (CRTCMD) command creates a new user-defined command (that is, a command definition) that can use the command processing support used by IBM-supplied commands. The command definition is an object that can be stored in the general purpose library (the QGPL library) or in a user library. The CRTCMD command also can be used to update an existing command; for example, to change the name of one of its parameter keywords. To do this, the command must first be deleted by the Delete Command (DLTCMD) command and then created again by the CRTCMD command. However, some of the values can be changed by the Change Command (CHGCMD) command.

To create a command, a set of command definition statements are entered into a source file. The CRTCMD command is used to process the source file and create a command definition object. The following command definition statements are used as needed:

- Command statement (CMD): One CMD statement is needed for each command being defined.
- Parameter statement (PARM): One PARM statement is required for each command parameter in the command being defined. It defines the parameter to be passed to the command processing program (CPP).
- Element statement (ELEM): An ELEM statement further defines a parameter that is a list of values. One statement is required for each possible element of the list.
- Qualifier statement (QUAL): A QUAL statement is required to describe each part of a qualified name that can be accepted for a parameter (defined in a PARM statement) or for an element in a list of values (defined in an ELEM statement).
- Dependent statement (DEP): The DEP statement indicates which parameters are interdependent.
- Prompt control statement (PMTCTL): The PMTCTL statement describes the conditions used to select a parameter for prompting.

Descriptions of the five command definition statements are in Chapter 5, "Command Definition Statements." Additional information about command definitions is in the *CL Programmer's Guide*.

Restriction: The CRTCMD command can be used only to create the command definition of an actual CL command. That is, it cannot be used to create definitions of *statements*, such as the command definition statements themselves.

Required Parameters

CMD

Specifies the qualified name of the command being created.

The name of the command can be qualified by one of the following library values:

***CURLIB:** The command is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the command is created.

command-name: Specify the name of the command being created.

PGM

Specifies the qualified name of the compiled program.

Note: The CPP is not required to exist until command run time.

The name of the CPP can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program that is to process the command.

Other Single Values

***REXX:** The CPP for this command is the REXX procedure identified by the REXSRCMBR and REXSRCFILE parameters.

The parameters passed to the CPP are the ones defined by the command definition statements in the source file specified by the SRCFILE parameter.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file that contains the command definition statements.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCMDSRC: The IBM-supplied source file, QCMDSRC, contains the command definition statements.

source-file-name: Specify the name of the source file that contains the command definition statements for the command being created.

SRCMBR

Specifies the name of the source file member that contains the command definition statements used to create the command.

***CMD:** When the file specified in the SRCFILE parameter is a database file, the name of the source file member is the name specified in the CMD parameter of this command.

source-file-member-name: Specify the name of the source file member that contains the command definition statements used to create the command.

REXSRCFILE

Specifies the qualified name of the source file containing the REXX procedure that is the command processing program.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QREXSRC: The IBM-supplied source file, QREXSRC, contains the source member that is used.

source-file-name: Specify the name of the source file containing the source member (the member that holds the REXX procedure) that is used.

REXSRCMBR

Specifies the source member containing the REXX procedure that is the command processing program.

***CMD:** The source member name is the same as the command name (the name specified on the CMD parameter).

source-member-name: Specify the name of the source member that contains the REXX procedure.

REXCMDENV

Specifies the command environment that is active when the REXX CPP starts to run. If the AS/400 system control language (CL) environment is not used, a

program can be specified to process commands found in the REXX procedure. The REXX interpreter calls this program to process commands encountered in the procedure. This environment can be changed with the REXX ADDRESS instruction.

***COMMAND:** The AS/400 system control language (CL) command environment is used.

***CPICOMM:** The Common Programming Interface (CPI) for Communications command environment is used. CPICOMM is the command environment used for CL commands that are imbedded within a REXX procedure.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program to process commands found in the REXX procedure.

REXEXITPGM

Specifies the exit programs to be used when the REXX interpreter is started. This parameter can be specified as a single value (*NONE) or as a list of two values (elements).

***NONE:** There are no exit programs for this call of the CPP.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

Element 1: Program Name

program-name: Specify the name of the exit program.

Element 2: Exit Code

exit-code: Specify one of the following values for the exit code:

Exit-code	Description
2	The associated program is called whenever an external function or subroutine has been called by the REXX program. The exit program is then responsible for locating and calling the requested routine.
3	The associated program is called whenever the interpreter is going to call a command. The exit program is responsible for locating and calling the command given the command string and the current environment name.
4	The associated program is called whenever a REXX instruction or function attempts an operation on the REXX external data queue.
5	The associated program is called when session input or output operations are attempted.
7	The associated program is called after running each clause of the REXX procedure to determine whether it must be halted.
8	The associated program is called after running each clause of the REXX program to check if tracing must be turned on or off.
9	The associated program is called before interpretation of the first instruction of a REXX procedure (including REXX procedures called as external functions and subroutines).
10	The associated program is called after interpretation of the last instruction of a REXX procedure (including REXX procedures called as external functions and subroutines).

VLDCKR

Specifies, each time the command is used, the qualified name of a program that performs additional user-defined validity checking on the parameters in the command being created. The same parameters that are passed to the CPP (these are identified in the PGM parameter) are also passed to the validity checking program. The validity checker performs additional user-defined validity checking beyond that specified by the command definition statements in the source file, and beyond normal control language syntax checking. More information on validity checking is in the *CL Programmer's Guide*.

***NONE:** No separate user-defined validity checking is done for this command. All validity checking is done by the command analyzer and the command processing program.

The name of the user-defined validity checking program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the user-defined validity checking program that checks the validity of the command being created. The validity checker is not required to exist until command run time.

MODE

Specifies the modes of operating environment to which the newly defined command applies. One or more of the modes can be specified.

***ALL:** The command is valid in all the types of operation: production, debug, and service modes.

***PROD:** The command is valid for production mode operations.

***DEBUG:** The command is valid for debug mode operations.

***SERVICE:** The command is valid for service mode operations.

ALLOW

Specifies where the CRTCMD command can be processed. One or more of the following options can be specified.

***ALL:** The command can be processed in a batch input stream, in a CL program, in a REXX procedure, or when processed interactively. It can also be passed to the system program QCMDEXC (or QCAEXEC) for processing.

Note: User-created commands created in releases prior to REXX must be modified to work in REXX.

***BATCH:** The command can be processed in a batch input stream, external to a compiled CL program.

***INTERACT:** The command can be processed interactively, external to a compiled CL program.

***BPGM:** The command can be processed in a compiled CL program that is called from batch entry.

***IPGM:** The command can be processed in a compiled CL program that is called from interactive entry.

***BREXX:** The command can be used in a REXX procedure that is run in batch job.

***IREXX:** The command can be used in a REXX procedure that is run in an interactive job.

***EXEC:** The command can be processed as a parameter on the CALL command, and can be passed as a character string to the system program QCMDEXC (or QCAEXEC) for processing. If *EXEC is specified, either *BATCH or *INTERACT must also be specified.

ALWLMTUSR

Specifies whether a user whose profile is set for limited capabilities is allowed to use the command by typing it in the command line on a menu.

***NO:** A user whose profile is set for limited capabilities is not allowed to use the command by typing it in the command line on a menu.

***YES:** A user whose profile is set for limited capabilities is allowed to use the command by typing it in the command line on a menu.

MAXPOS

Specifies the maximum number of parameters that can be coded positionally for this command. This parameter value must be greater than the number of nonconstant required parameters and less than the total number of nonconstant parameters. Parameters of TYPE(*ZEROELEM), parameters with the CONSTANT attribute, and lists and qualified names whose ELEMs and QUALs have the CONSTANT attribute or are of TYPE(*ZEROELEM) are not included in the number of parameters that can be coded positionally for this command.

***NOMAX:** The system maximum is used.

positional-limit: Specify the maximum number of parameters that can be coded positionally for this command. Valid values range from 0 through 75.

PMTFILE

Specifies the qualified name of the message file from which the prompt text for the command is retrieved.

***NONE:** No message file is needed for the prompt text. The text, if any, is supplied in the definition statements that define the command.

The name of the message file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

message-file-name: Specify the name of the message file to use.

MSGF

Specifies the qualified name of the message file from which messages identified on the DEP statements used to define the command are retrieved. These messages are those that can be sent if the command, while running, finds a parameter dependency error. The messages referred to by this MSGF parameter are those whose message identifiers are specified in the MSGID parameter of one or more DEP definition statements, but whose identifiers do not have the 3-character prefix, CPF. QCPFMSG is always used for messages that have the CPF prefix.

The name of the message file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCPFMSG: Dependency error messages are retrieved from the IBM-supplied CPF message file, QCPFMSG.

message-file-name: Specify the name of the message file to use.

HLPPNLGRP

Specifies the qualified name of the help panel group for this command.

***NONE:** No help panel group is specified.

Note: If *NONE is specified on this parameter and a value is specified on the HLPID parameter, an error message is issued when the command is changed.

The name of the panel group can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

panel-group-name: Specify the name of the help panel group for this command.

HLPID

Specifies the name of the general help module for the names of the help identifiers for this command.

***CMD:** The name of the command is the name of the general help module.

***NONE:** No help identifier is specified.

Note: If *NONE is specified on this parameter and a value is specified on the HLPID parameter, an error message is issued when the command is changed.

help-ID-name: Specify the name of the general help module for the help identifiers for this command.

HLPSCHIDX

Specifies the qualified name of the help search index to be used when the search index function key is pressed.

***NONE:** No search index is specified.

***SYSTEM:** The system search index, QHSS1, is used.

The name of the search index can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

search-index-name: Specify the name of the search index to be used when the search index function key is pressed.

CURLIB

Specifies the name of the library being used as the current library during the processing of this command.

Note: This library is also the current library when the validity checker program (if any) is processed for the command.

***NOCHG:** The current library does not change for the processing of this command. If the current library is changed during processing of the command, the change remains in effect after command processing is complete.

***CRTDFT:** No current library is active during processing of the command. The current library that was active before command processing began is restored when processing is completed.

If *CURLIB was specified as the to-value for any single values or special values for this command, or for any command processed while no current library is active, the QGPL library is used as the current library.

library-name: Specify the name of the library used as the current library. The library need not exist when the command is created, but must exist when the command is processed. When command processing is completed, the current library is restored to its previous value. If the current library is changed during command processing by the Change Library List (CHGLIBL) command or Change Current Library (CHGCURLIB) command, the change is effective only until the command is processed. QTEMP cannot be specified for the current library.

PRDLIB

Specifies the name of the product library that is in effect during the processing of the command.

***NOCHG:** The product library does not change when command processing starts. If the product library is changed during the processing of the command, the change remains in effect after command processing is complete.

***NONE:** There is no product library in the job's library list. The product library is restored to its previous value on completion of command processing.

library-name: Specify the name of the library used as the current library. The library need not exist when the command is created, but must exist when the command is processed. When command processing is completed, the current library is restored to its previous value. If the current library is changed during command processing by the Change Library List (CHGLIBL) command or

Change Current Library (CHGCURLIB) command, the change is effective only until the command is processed. QTEMP cannot be specified for the current library.

PMTOVRPGM

Specifies the qualified name of the prompt override program (POP) that replaces (on the prompt display) the default values with the current actual values for the parameter. If a POP is specified, the key parameters (specified as KEYPARM(*YES) on the PARM statement in the command definition source) are the only parameters visible on the initial prompt display. When values are input for the key parameters, the remaining parameters are shown on the display with the actual values instead of the default values.

***NONE:** No prompt override program is specified.

Note: If *NONE is specified when key parameters exist in the command definition source (KEYPARM(*YES) specified on the PARM statement), a warning message is issued when the command is created, and KEYPARM(*NO) is assumed for all parameters.

The name of the prompt override program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the prompt override program for the command.

AUT

Specifies the authority given to users who do not have specific authority to the command, who are not on an authorization list, and whose user group has no specific authority to the command. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the command is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the command). The public authority is determined when the command is created. If the CRTAUT value for the library changes after the command is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the command.

***USE:** The user can perform basic operations on the command, such as running a program or reading a file. The user cannot change the command. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the command.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies the replacement of an existing command with a new version of the same name, type, and library, without first issuing a DLTCMD command to remove the old one.

The old version is renamed, moved to library QRPLOBJ, and then cleared on the next IPL of the system.

***YES:** If the create operation is successful, existing commands are replaced by new versions of the same command.

***NO:** No replacement occurs.

TEXT

Specifies text that briefly describes the command and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: The text in the command description can be displayed by the Display Object Description (DSPOBJD) command.

***SRCMBRTXT:** The text is taken from the source file member being used to create the CL command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** No text is specified.

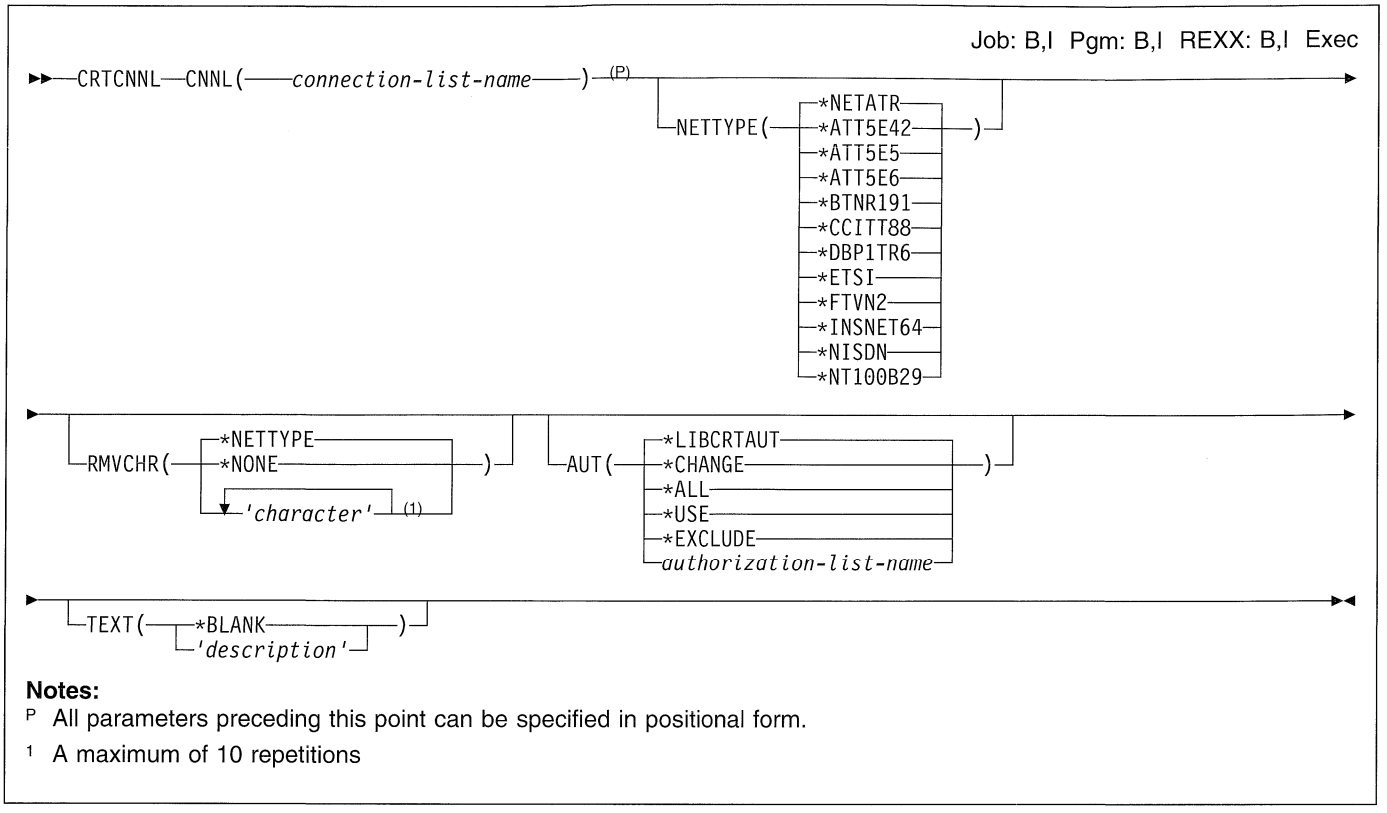
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCMD   CMD(PAYROLL)  PGM(PAY076)
  SRCFILE(PAYSOURCE)  AUT(*EXCLUDE)
```

The command named PAYROLL is created from the source file PAYSOURCE. The command is private and calls the command processing program (CPP) named PAY076. It is a valid command when entered in a batch input stream, when compiled in a control language program, when entered interactively, or when passed to the QCMDEXC program.

CRTCNNL (Create Connection List) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 10 repetitions

Purpose

The Create Connection List (CRTCNNL) command is used to create a connection list. A connection list consists of up to 256 connection list entries and is used to manage incoming and outgoing calls. Users specify a network type and the removal of extra characters for all of the entries of a list. To add entries to a connection list, use the Add Connection List Entry (ADDCNNLE) Command.

Required Parameter

CNNL

Specifies the name of the connection list being created.

Optional Parameters

NETTYPE

Specifies the type of Integrated Services Digital Network (ISDN) used by the connection list. This parameter is used to determine the defaults for other network-dependent parameters. Those parameters include: Remote Number Type, Remote Numbering Plan, Remote Subaddress Type, Local Number Type, Local Numbering Plan, Local Subaddress Type, Transit Network Plan, and Transit Network Type.

Note: The defaults that are set for those parameters can be viewed by using the Add Connection Lists Entry (ADDCNNLE) command, or by checking the *ISDN Guide*.

***NETATR:** The network type specified on the default network type parameter (DFTNETTYPE) in the network attributes is used. The values of the parameters in the network attributes for the system can be displayed by using the Display Network Attributes (DSPNETA) command.

***ATT5E42:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T 5ESS version 5E4.2 switching equipment.

***ATT5E5:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T 5ESS version 5E5 switching equipment.

***ATT5E6:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T 5ESS version 5E6 switching equipment.

***BTNR191:** This value is used when attaching to an ISDN in the United Kingdom controlled by British Telecomm.

***CCITT88:** The ISDN default values recommended by the International Telegraph and Telephone Consultative Committee (CCITT) in 1988 are used.

***DBP1TR6:** This value is used when attaching to an ISDN controlled by Germany's PTT (Deustche Bundes Poste 1TR6).

***ETSI:** This value is used when attaching to an ISDN that uses the European Telecommunications Standards Institute (ETSI, also known as EuroISDN) standard.

***FTVN2:** This value is used when attaching to version 2 of the ISDN controlled by France's post telephone and telegraph administration (PTT) (France Telecom Numeris VN2).

***INSNET64:** This value is used when attaching to the INSNET64 ISDN controlled by Japan's Nippon Telephone and Telegraph Public Corporation (NTT).

***NISDN:** This value is used when attaching to an ISDN that uses the National ISDN-1 or National ISDN-2 standard for North America.

***NT100B29:** This value is used when attaching to an ISDN in the United States or Canada that uses Northern Telecom DMS100 Version BCS-29 switching equipment.

RMVCHR

Specifies up to 10 characters that are removed from remote and local numbers. Extra characters are removed from numbers before sending or comparing those numbers. The ability to shorten numbers prior to their use by the system means extra characters can be inserted in the numbers to help organize and read them.

***NETTYPE:** A default value is used based on the network type specified on the NETTYPE parameter when the connection list was created using the Create Connection List (CRTCNNL) command.

***NONE:** No characters are removed.

'character': Specify a character to be removed. Up to 10 characters can be specified.

AUT

Specifies the level of public authority for this object.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations on the object except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence, specify the security for the object, change the object, and perform basic functions on the object. The user can transfer ownership of the object.

***USE:** The user can perform basic operations on the connection list, such as running a program or reading a file. The user cannot change the connection list. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the connection list.

authorization-list-name: Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

TEXT

Specifies text that briefly describes the connection list. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

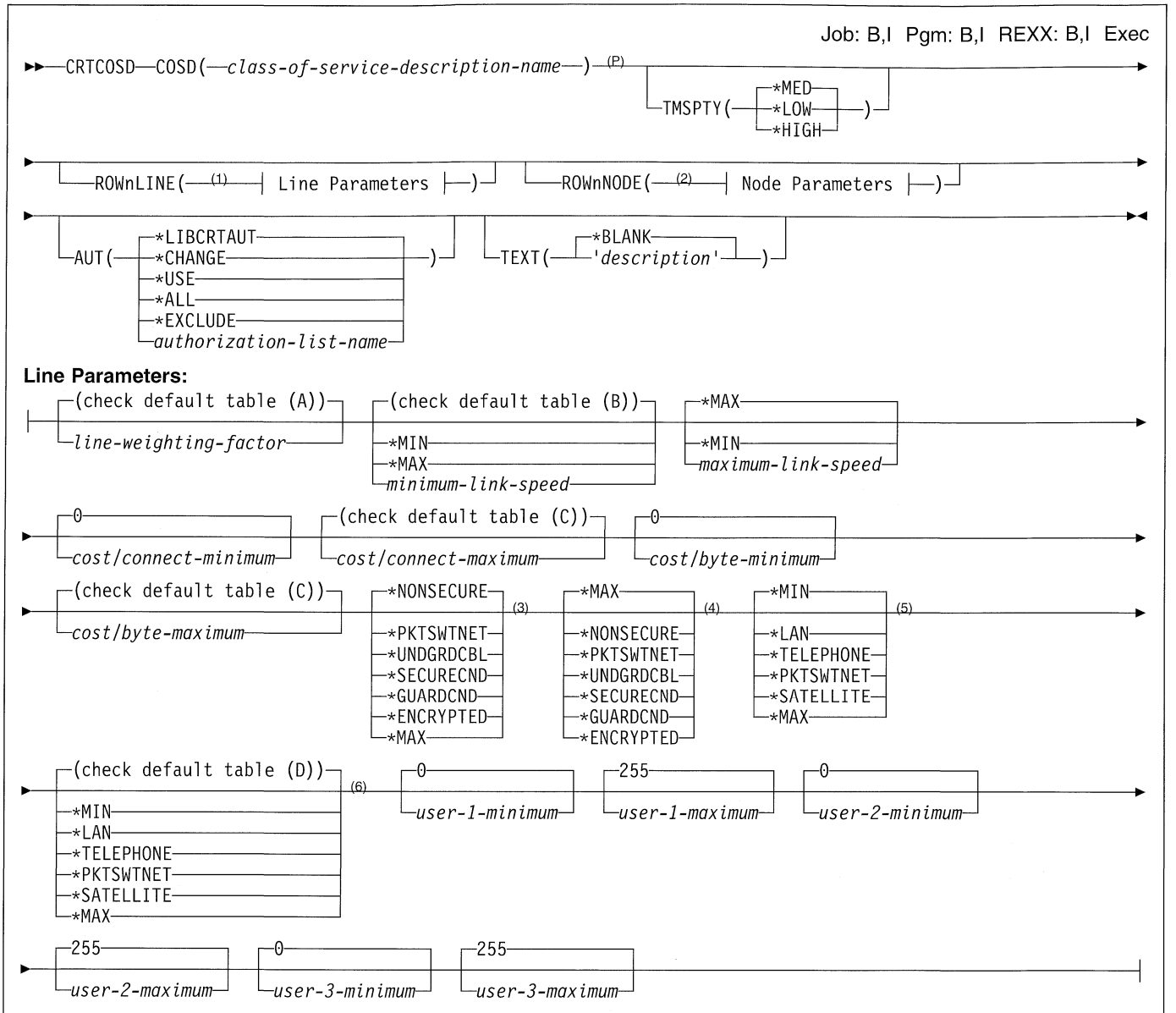
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCNNL   CNNL(CHICAGO)
  ENTTYPE(*CCITT88)
```

This command creates a connection named CHICAGO with a network type of *CCITT88. Any entries added or changed (using the ADDCNNLE or CHGCNNLE commands) that have parameters specified with the value *NETTYPE have values set to work on a *CCITT88 type ISDN.

CRTCOSD (Create Class-of-Service Description) Command



Notes:

1 n can be a number from 1 to 8. Table 23 shows the defaults for the row line parameters.

2 n can be a value from 1 to 8. Table 24 shows the defaults for the row node parameters.

P All parameters preceding this point can be specified in positional form.

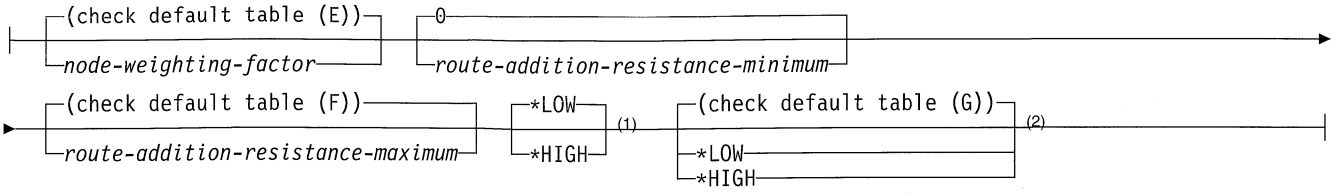
3 Minimum Security

4 Maximum Security

5 Minimum Propagation Delay

6 Maximum Propagation Delay

Node Parameters:



Notes:
1 Minimum Congestion
2 Maximum Congestion

Purpose

The Create Class-of-Service Description (CRTCOSD) command creates a class-of-service description. A class-of-service description determines the nodes and transmission groups that are eligible to be included in advanced program-to-program communications (APPC) or Advanced Peer-to-Peer Networking (APPN) support. More information on class-of-services is in the *APPN Guide* and the *APPC Programmer's Guide*

Table 23. Default Table for Row/Lines 1–8

Line	(A)	(B)	(C)	(D)
ROW1LINE	30	4M	0	*LAN
ROW2LINE	60	56000	0	*TELEPHONE
ROW3LINE	90	19200	0	*TELEPHONE
ROW4LINE	120	9600	0	*TELEPHONE
ROW5LINE	150	19200	0	*PKTSWTNET
ROW6LINE	180	9600	128	*PKTSWTNET
ROW7LINE	210	4800	196	*MAX
ROW8LINE	240	*MIN	255	*MAX

Table 24. Default Table for Row/Nodes 1–8

Node	(E)	(F)	(G)
ROW1NODE	5	31	*LOW
ROW2NODE	10	63	*LOW
ROW3NODE	20	95	*LOW
ROW4NODE	40	127	*LOW
ROW5NODE	80	159	*LOW
ROW6NODE	100	191	*LOW
ROW7NODE	120	223	*HIGH
ROW8NODE	150	255	*HIGH

Required Parameter

COSD
Specifies the name of the class-of-service description created. This name ranges from 1 through 8 characters.

Optional Parameters

TMSPTY
Specifies the transmission priority for this class-of-service description. The priority levels are listed below.

***MED:** Medium transmission priority is given for this class-of-service description.

***LOW:** The lowest transmission priority is given for this class-of-service description.

***HIGH:** The highest transmission priority is given for this class-of-service description.

ROW1LINE - ROW8LINE
Specifies the list of line-related criteria used for the first through the eighth rows of the class-of-service description. These rows describe the attributes of the line connection between two nodes in the APPN network. The rows are examined in order (from 1 through 8) to define a network routing path.

Element 1: Line Weight Factor

line-weighting-factor: Specify the relative weight of lines for this row. The weight, which ranges from 0 through 255, indicates the relative cost of a line connection. Refer to (A) in Table 23 for the default. This value varies for each ROWxLINE.

Element 2: Minimum Link Speed

***MIN:** The minimum link speed is specified.

***MAX:** The maximum link speed is specified.

minimum-link-speed: This is the minimum link speed for a line connection that is accepted by this row criteria. Valid values are *MIN, 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 4M, 10M, 16M, or *MAX. Refer to (B) in Table 23 for the default. This value varies for each ROWxLINE.

Element 3: Maximum Link Speed

***MAX:** The maximum link speed is specified.

***MIN:** The minimum link speed is specified.

maximum-link-speed: This is the maximum link speed for a line connection that is accepted by this row criteria. See the list of valid values under minimum link speed in Element 2.

Element 4: Minimum Cost Per Connect Time

cost/connect-minimum: Specify the minimum relative cost per connect time that is accepted by this row criteria. Valid costs range from 0 through 255. The default value is 0.

Element 5: Maximum Cost Per Connect Time

cost/connect-maximum: Specify the maximum relative cost per connect time that is accepted by the row criteria. Valid costs range from 0 through 255. Refer to (C) in Table 23 for the default. This value varies for each ROWxLINE.

Element 6: Minimum Cost Per Byte

cost/byte-minimum: Specify the minimum relative cost per byte that is accepted by this row criteria. Valid costs range from 0 through 255. The default value is 0.

Element 7: Maximum Cost Per Byte

cost/byte-maximum: Specify the maximum relative cost per byte that is accepted by this row criteria. Valid costs range from 0 through 255. Refer to (C) in Table 23 for the default. This value varies for each ROWxLINE.

Element 8: Minimum Security Level

Minimum Security: This is the minimum security level that is accepted by this row criteria. Valid values are in order from least secure to most secure. The default value is *NONSECURE.

*NONSECURE	Nonsecured line
*PKTSWTNET	Packet switched network
*UNDGRDCBL	Underground cable
*SECURECND	Secure conduit
*GUARDCND	Guarded conduit
*ENCRYPTED	Encrypted line
*MAX	Guarded conduit, protected against physical and radiation tapping.

Element 9: Maximum Security Level

Maximum Security: This is the maximum security level that is accepted by this row criteria. Valid values are in order from least to most secure. The default value is *MAX.

*MAX	Guarded conduit, protected against physical and radiation tapping.
*NONSECURE	Nonsecured line
*PKTSWTNET	Packet switched network
*UNDGRDCBL	Underground cable
*SECURECND	Secure conduit
*GUARDCND	Guarded conduit
*ENCRYPTED	Encrypted line

Element 10: Minimum Propagation Delay

Minimum Propagation Delay: This is the minimum propagation delay that is accepted by this row criteria. The default value is *MIN.

*MIN	Minimum propagation delay
*LAN	Local area network
*TELEPHONE	Telephone propagation delay
*PKTSWTNET	Packet switched network propagation delay
*SATELLITE	Satellite propagation delay
*MAX	Maximum propagation delay

Element 11: Maximum Propagation Delay

Maximum Propagation Delay: This is the maximum propagation delay that is accepted by this row criteria. Refer to (D) in Table 23 for the default. This value will vary for each ROWxLINE.

The values are:

*MIN	Minimum propagation delay
*LAN	Local area network
*TELEPHONE	Telephone propagation delay
*PKTSWTNET	Packet switched network propagation delay
*SATELLITE	Satellite propagation delay
*MAX	Maximum propagation delay

Element 12: User's First Minimum Line Connection Criteria

user-1-minimum: Specify your own line connection criteria. Valid values range from 0 through 255 with a default value of 0.

Element 13: User's First Maximum Line Connection Criteria

user-1-maximum: Specify your own line connection criteria. Valid values range from 0 through 255 with a default value of 255.

Element 14: User's Second Minimum Line Connection Criteria

user-2-minimum: Specify your own line connection criteria. Valid values range from 0 through 255 with a default value of 0.

Element 15: User's Second Maximum Line Connection Criteria

user-2-maximum: Specify your own line connection criteria. Valid values range from 0 through 255 with a default value of 255.

Element 16: User's Third Minimum Line Connection Criteria

user-3-minimum: Specify your own line connection criteria. Valid values range from 0 through 255 with a default value of 0.

Element 17: User's Third Maximum Line Connection Criteria

user-3-maximum: Specify your own line connection criteria. Valid values range from 0 through 255 with a default value of 255.

ROW1NODE - ROW8NODE

Specifies the list of node-related criteria used for the first through eight rows of the class-of-service description. These rows describe the attributes of a node in the APPN network. The rows are examined in order (from 1 to 8) to define a network routing path.

Element 1: Node Weighting Factor

node-weighting-factor: Specify the relative weight of nodes for this row. The weight, which ranges from 0 through 255, indicates the relative cost of passing through a node in the network. Refer to (E) in Table 24 for the default. This value varies for each ROWxNODE.

Element 2: Minimum Route Additional Resistance

route-additional-resistance-minimum: Specify the minimum route additional resistance accepted by this row criteria. Valid values range from 0 through 255 with a default of 0.

Element 3: Maximum Route Additional Resistance

route-additional-resistance-maximum: Specify the maximum route additional resistance accepted by this row criteria. Refer to (F) in Table 24 for the default. This value varies for each ROWxNODE.

Element 4: Minimum Congestion Level

***LOW** congestion-minimum: Low congestion is tolerated.

Element 5: Maximum Congestion Level

***HIGH** congestion-maximum: High congestion is tolerated. Refer to (G) in Table 24 for the default. This value varies for each ROWxNODE.

AUT

Specifies the authority given to users who do not have specific authority to the class-of-service description, who are not on an authorization list, and whose user group has no specific authority to the class-of-service description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the class-of-service description, such as running a program or reading a file. The user cannot change the class-of-service description. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the class-of-service description.

***EXCLUDE:** The user cannot access the class-of-service description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

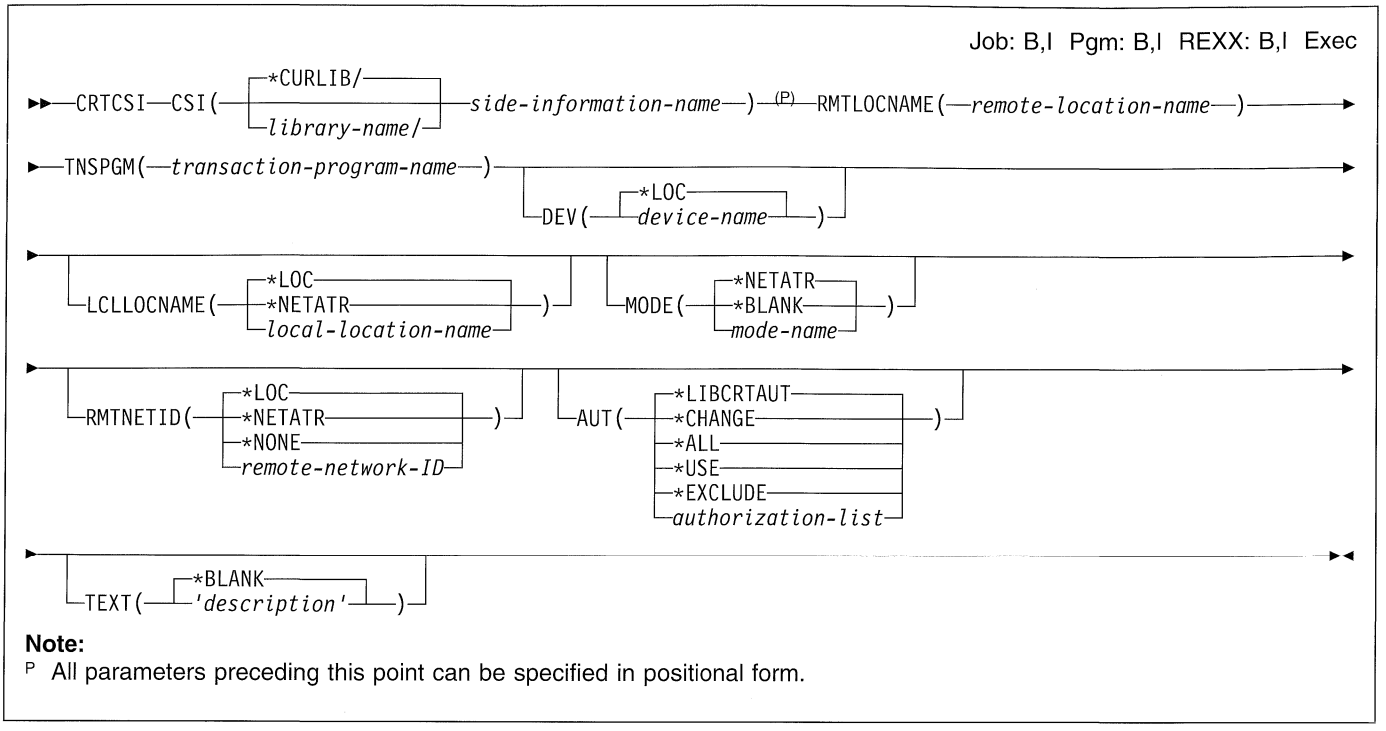
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCOSD  COSD(COSD1)
  ROW1LINE(15 9600 19200 20 30
  50 75 *SECURECND *GUARDCND
  *PKTSWTNET *SATELLITE
  100 200 50 100 25 50)
  TEXT('customized Row1line')
```

This command creates COSD1. All values are specified for Row1Line, and defaults are accepted for all the other rows.

CRTCSI (Create Communications Side Information) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Communications Side Information (CRTCSI) command is used to create a side information object in a specific library. The side information object name must be unique for the library in which the side information object is to be created. More information on how the system uses the RMTLOCNAME, DEV, LCLLOCNAME, and RMTNETID parameters to select an APPC device description is in the *APPC Programmer's Guide*.

Required Parameters

CSI

Specifies the name of the side information object to be created. An object name must be specified. The side information object is accessed by the symbolic destination name (sym_dest_name) parameter on the Initialize_Conversation (CMINIT) call.

The name of the side information object can be qualified by one of the following library values:

***CURLIB:** The side information object is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the side information object is created.

side-information-name: Specify the name of the object that contains the desired side information. This is the CPI-Communications symbolic destination name (sym_dest_name).

RMTLOCNAME

Specifies the remote location name that is associated with the symbolic destination name. The Common Programming Interface (CPI) -Communications partner_LU_name, which consists of the remote network identifier and the remote location, determines the remote logical unit (LU) of the remote program.

TNSPGM

Specifies the name (up to 64 characters) of the transaction program on the remote system to be started. This is the CPI-Communications TP_name. The user can enter SNA service transaction program names by specifying hexadecimal input. For example, '03F0F0F4' is a valid input format for an SNA service transaction program.

Optional Parameters

DEV

Specifies the name of the device description used for the remote system.

***LOC:** The device associated with the remote location is used. If several devices are associated with the remote location, the system determines which device is used.

device-name: Specify the name of a communications device associated with the remote location. If the device name is not valid for the remote location, a message is

sent when the program device entry is acquired. More information on device names is in the *APPC Programmer's Guide*.

LCLLOCNAME
Specifies the local location name.

***LOC:** The device associated with the remote location is used. If several devices are associated with the remote location, the system determines which device is used.

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the name of the local location. Specify the local location if the remote location is also indicated by a specific local location name.

MODE
Specifies the mode used to control the session. This name is the same as the CPI-Communications mode_name.

***NETATR:** The mode name specified in the network attributes is used.

***BLANK:** Eight blank characters are used.

mode-name: Specify a mode_name for the remote location.

Note: The values SNASVCMG and CPSVCMG cannot be specified.

RMTNETID
Specifies the remote network identifier used with the remote location. The CPI-Communications partner_LU_name, which consists of the remote network identifier and the remote location, locates the LU in the network.

***LOC:** The remote network ID for the remote location is used.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** The remote network has no name.

remote-network-ID: Specify a remote network ID.

AUT
Specifies the authority given to users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.

***LIBCRTAUT:** Authority for the object is taken from the CRTAUT parameter of the specified library. This value is determined at create time. If the CRTAUT value for the library changes after the object is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations on the object except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence, specify the security for the object, change the object, and perform basic functions on the object. The user cannot transfer ownership of the object.

***USE:** The user can perform basic operations on the communications side information, such as running a program or reading a file. The user cannot change the communications side information. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the communications side information.

authorization-list: Specify the name of the authorization list whose authority is used for the side information.

TEXT
Specifies text that briefly describes the object and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

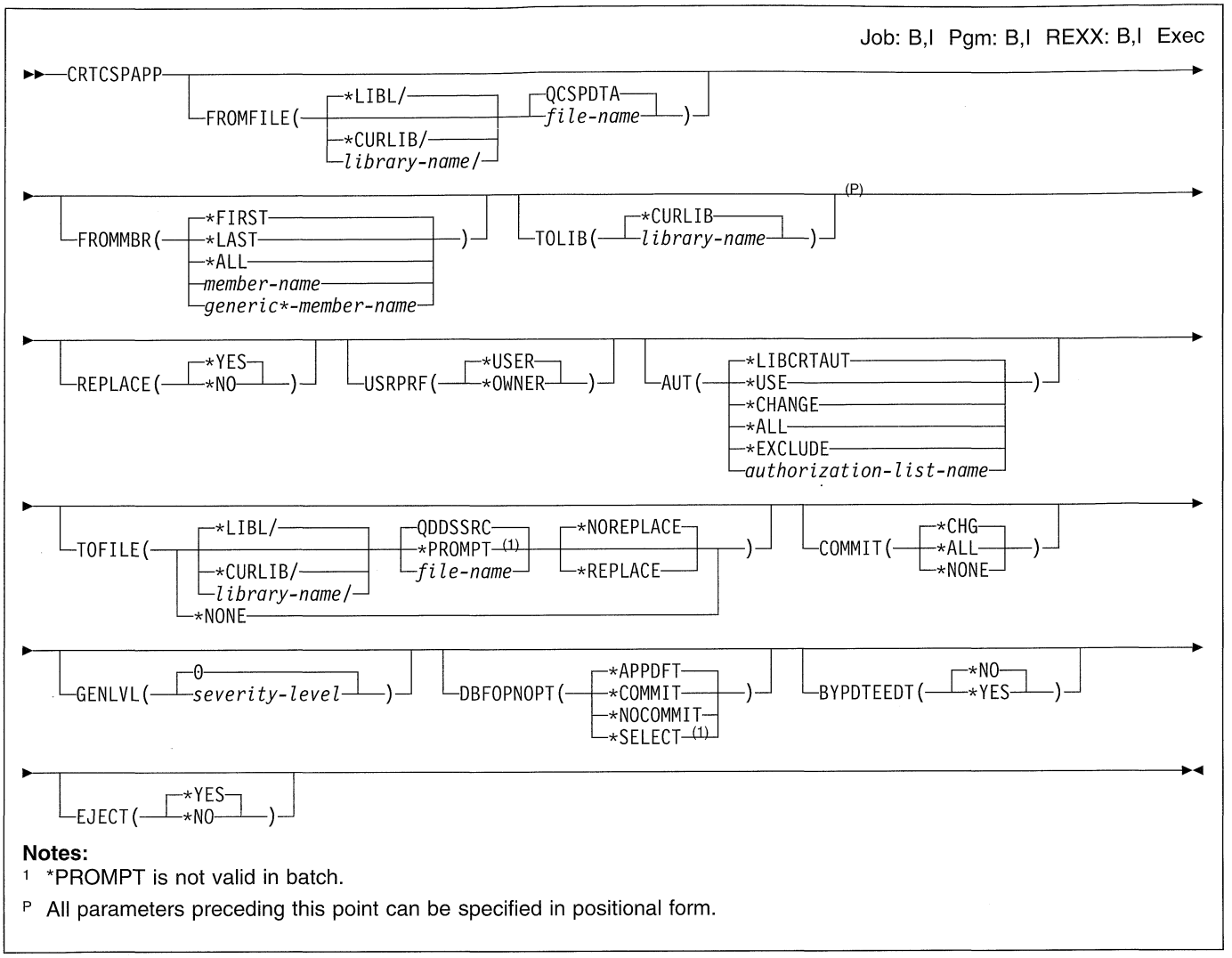
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCSI   CSI(QGPL/SIDEOBJ)  RMTLOCNAME(APPC2)
  TNSPGM(TESTNOP)  AUT(*CHANGE)
```

This command creates a communications side information object named SIDEOBJ in library QGPL. The remote location name is set to APPC2, the transaction program name to TESTNOP. The authority to the object is *CHANGE.

CRTCSPAPP (Create CSP/AE Application Objects) Command



Notes:
1 *PROMPT is not valid in batch.
P All parameters preceding this point can be specified in positional form.

Purpose

The Create CSP/AE Application Objects (CRTCSPAPP) command creates a single object or group of objects extracted from a Cross System Product/Application Development (CSP/AD) system (using the application load file utility Extract function). The types of objects that can be created or replaced are *CSPMAP, *CSPTBL, and *PGM (attribute CSPAE). Data description specifications (DDS) source members for the objects can be created or replaced.

The *from-file* must be either the supplied default file for this command (QGPL/QCSPDTA), or it must have been created on the system using the standard AS/400 command CRTPF (Create Physical File). The *from-file* members must be moved to the AS/400 system using a method defined by System Application Architecture* (SAA*). The *from-file* contains records of a fixed length of 272 bytes (284 in a source file) consisting of a 10-byte key, 6 unused bytes, and 256 bytes of formatted data.

The input object can contain any number and combination of CSP/AD programs, map groups, and tables. Applications can contain data description specifications (DDS) and Structured Query Language (SQL) records that require additional processing and user input. The DDS processing is optional, but the SQL processing is always performed if SQL records are found.

If the object to be created already exists in the *to-library* on the AS/400 system, the value of the REPLACE keyword determines the correct action. The host CSP/AD system provides the names of objects created during application development and generation. The *to-library* is the destination library for all objects created or replaced.

If the command is running in the batch subsystem, then DDS members are created using the default values from the CSP/AD application, and the Create DDS Source Member display does not appear.

If DDS records are found while an application is being created interactively and TOFILE(*PROMPT) is specified, then the Create DDS Source Member display appears, requesting DDS destination information. The default reply is to press Enter. Processing then continues, and a DDS source member is created.

If the user choose not to create DDS source members, the corresponding CSP/AE application program may not be affected. The CSP/AE application the user is creating does not require DDS members, because the application contains internal record definitions for the databases that it uses. Before a CSP/AE application can run, a database with the correct record format must exist on the AS/400 system. The DDS information helps the user to create the physical or logical files, containing no data, that the application requires. Using available OS/400 commands, the user can create a DDS source member in a file, and then create a database file from that DDS source information. If the user is putting a new version of an existing application on the system using the CRTCSPAPP command, and the record definitions for the new and existing applications match, then there may be no need to recreate the DDS information for the new version.

When the CRTCSPAPP command runs, a spooled file named QPAECRT is sent to the current job's output queue. This file contains log data pertaining to the objects created and any conditional processing such as "DDS Ignored."

Optional Parameters

FROMFILE

Specifies the qualified name of the *from-file* that is to be used to create the CSP/AE objects. The file must contain members extracted from a CSP/AD system using the ALF utility Extract function. The file contains fixed length records of 272 bytes (284 in a source file).

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCSPDTA: The IBM-supplied from-file, QCSPDTA in the QGPL library is used. This file has the correct record length for the data that CSP/AD generates for the OS/400. It contains the data that CSP/AD creation produced when the application was generated for OS/400.

file-name: Specify the name of the from-file used to create the CSP/AE objects. It contains the data that CSP/AD creation produced when the application was generated for OS/400.

FROMMBR

Specifies the file member name in the from-file used. A name, generic name, or one of three special names are used to select the file member from which objects are created.

***FIRST:** The first member in the file is used to create objects.

***LAST:** The last member in the file is used to create objects.

***ALL:** All members in the file are used to create objects.

member-name: Specify the name of the member from which objects are created.

generic-member-name:* Specify the generic name of the member. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be created only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

TOLIB

Specifies the name of the library where CSP/AE application objects are created.

***CURLIB:** Created objects are stored in the current library.

library-name: Specify the name of the library where the created CSP/AE objects are stored.

REPLACE

Specifies whether a created object replaces an existing object that has the same name as the created object.

***YES:** If an existing object in the specified library has the same name as the object to be created, it is replaced by the new object.

***NO:** If an existing object in the specified library has the same name as the object to be created, the object in the library is not replaced. No new object is created.

USRPRF

Specifies the user profile under which the compiled program is run. The profile of either the program owner or the program user is used to run the program and control which objects can be used by the program, including the authority the program has for each object.

***USER:** The program runs under the user profile of the program's user.

***OWNER:** The profiles of both the application owner and the user are used when the application runs.

AUT

Specifies the authority given to users who do not have specific authority to the CSP/AE, who are not on an authorization list, and whose user group has no specific authority to the CSP/AE.

***LIBCRTAUT:** The public authority for the CSP/AE is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the CSP/AE). The public authority is determined when the CSP/AE is created. If the CRTAUT value for the library changes after the CSP/AE is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the CSP/AE, such as running a program or reading a file. The user cannot change the CSP/AE. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user has *CHANGE, object existence, and object management authorities to the objects.

***EXCLUDE:** The user cannot access the CSP/AE.

authorization-list-name: Specify the name of the authorization list.

TOFILE

Specifies the qualified name of the file that receives the copied records.

Note: The values that act as processing control values are *NONE and *PROMPT.

Element 1: Specifying the Source File to Receive DDS Members

The name of the DDS source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The default file is used to receive the data description specifications (DDS) members.

***PROMPT:** The Create DDS Source Member display is shown if DDS records are found in the *from-file*. The display appears for each DDS record found, and the user can choose to change the DDS processing options.

file-name: Specify the source file name to receive the DDS members.

Element 2: Replacing a Source Member

***NOREPLACE:** An existing DDS source member with the same name is not replaced.

***REPLACE:** An existing DDS source member with the same name is replaced with the newly created object. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Single Value

***NONE:** The DDS records found in the *from-file* are ignored. The DDS source member display does not appear. If DDS records are ignored, DDS members are not created for CSP/AE application files found. A complete create operation for the CSP/AE application must be performed to recover the DDS records and create the source members.

COMMIT

Specifies whether SQL statements in the application program are run under commitment control. This parameter may affect files referred to by non-SQL processes. A value other than *NONE causes the SQL/400 program to start a commitment control cycle automatically.

***CHG:** Only the updated, deleted, and inserted rows are locked until the end of the transaction.

***ALL:** All rows selected, updated, deleted, and inserted are locked until the end of the transaction.

***NONE:** Commitment control is not used.

GENLVL

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

0: The default value of 0 (zero) is used.

Some suggested values are listed below.

0 The default severity level.
10 The level value for warnings.
20 The level value for general error messages.
30 The level value for serious error messages.
40 The level value for system detected error messages.

severity-level: Specify a value ranging from 0 through 40.

DBFOPNOPT

Specifies whether files used by non-SQL processes can be opened with commitment control.

***APPDFT:** Files used by non-SQL processes are opened with commitment control if the following conditions are true:

- A commitment control cycle is active when the file is opened
- The file is opened for writing
- The COMMIT parameter value is *CHG or *ALL
- The application uses explicit commitment control logic

***COMMIT:** All files used by non-SQL processes are opened with commitment control.

***NOCOMMIT:** All files used by non-SQL processes are opened without commitment control.

***SELECT:** The Open Database Files display is shown. The display lists every non-SQL file used by the application. The database file open option can be changed individually for every file using this display.

BYPDTEEDT

Specifies whether data edit checks occur for date fields on maps if the field is filled with blanks or erased with the Field Exit key.

***NO:** Data edit checks are not bypassed when a date field is cleared. CSP/AE produces an error message stating that an incorrect date format was entered in the field.

***YES:** Data edit checks are bypassed when a date field is cleared. The blank date field is accepted by CSP/AE. The map variable receives a zero value.

EJECT

Specifies whether the first floating print map in a CSP/AE application is printed at the top line of the floating area or on the next available line in the floating area.

***YES:** The first floating print map in the application is printed at the top of the floating area.

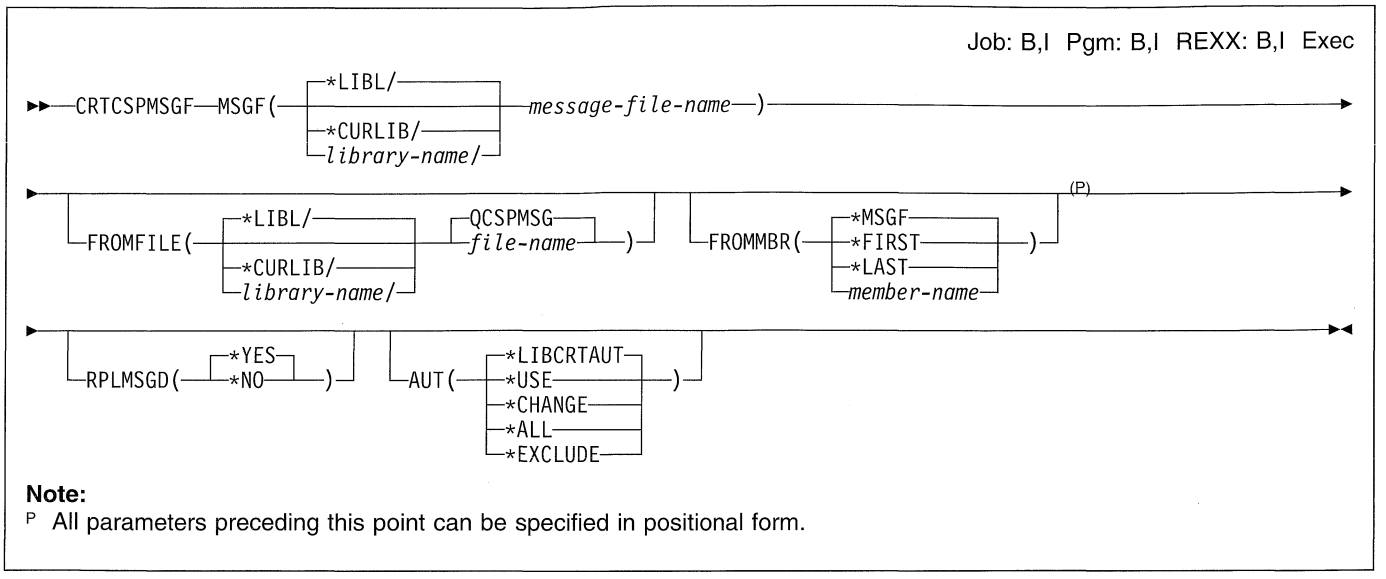
***NO:** The first floating print map in the application is printed at the next available floating area line.

Example

```
CRTCSPAPP   FROMFILE(VMNET/QCSPDTA)
  FROMMBR(PAYROLL)  TOLIB(CSPLIB)  REPLACE(*NO)
  AUT(*CHANGE)  DBFOPNOPT(*NOCOMMIT)
```

This command creates CSP/AE objects on the AS/400 system using member PAYROLL of file QCSPDTA in library VMNET and stores the objects in library CSPLIB. If any objects having the same names already exist in the target library, they are not replaced. *CHANGE authority for the objects is given to the users. All files used by non-SQL processes are opened without commitment control.

CRTCSPMSGF (Create CSP/AE User Message File) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Create CSP/AE User Message File (CRTCSPMSGF) command updates or creates a single message file object extracted from a Cross System Product/Application Development (CSP/AD) system. The AS/400 message file object can be updated with all messages from the CSP/AD user message file, or with only part of them.

The *from-file* must contain 256-byte fixed length records. These records must be formatted in the form of CSP/AD user messages. Each record in the input message file contains space for three formatted messages. These messages will be reformatted to conform to the AS/400 *MSGF object type and then stored in an AS/400 message file.

If the destination message file exists on the AS/400 system, messages from the *from-file* will be added to the message file only where messages having the same message number do not already exist. Existing messages in the message file are replaced according to the value of the RPLMSGD parameter. RPLMSGD(*YES) allows an existing message in the destination message file to be replaced; RPLMSGD(*NO) prevents an existing message from being replaced.

Required Parameter

MSGF

Specifies the qualified name of an OS/400 message file and the library in which the message file is or will be stored. The name is restricted to three characters. These three characters should be the same as the first three characters of the message file name that was defined for the application in CSP/AD.

| The name of the message file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

message-file-name: Specify the name of the message file to use.

Optional Parameters

FROMFILE

Specifies the qualified name of the from-file. The file must have members containing CSP/AD user messages. The from-file may be a data file defined with 256-byte records or a source file defined with 268-byte records.

| The name of the from-file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCSPMSG: The IBM-supplied message file, QCSPMSG, can be used to receive a user message file coming from a CSP/AD system. This file has the correct record length for the contents of an extracted CSP/AD user message file. It should contain the data that VSAM REPRO produced when the user message file was extracted on the CSP/AD system.

file-name: Specify the name of a file containing CSP/AE user messages. The file should contain the data that VSAM REPRO produced when the user message file was extracted on the CSP/AD system.

FROMMBR

Specifies the input file member name. The member normally would come from a user message file on a CSP/AD system.

***MSGF:** The member having the same name as the message file is created or updated.

***FIRST:** The first member in the *from-file* is used.

***LAST:** The last member in the *from-file* is used.

member-name: Specify the name of the member in the *from-file*.

RPLMSGD

Specifies if an incoming message replaces an existing message having the same message number in the destination message file.

***YES:** The existing message in the message file is replaced by an incoming message having the same message number.

***NO:** An existing message in the message file having the same number as an incoming message is not replaced.

AUT

Specifies the authority given to users who do not have specific authority to the CSP/AE user message file, who are not on an authorization list, and whose user group has no specific authority to the CSP/AE user message file.

***LIBCRTAUT:** The public authority for the CSP/AE user message file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the CSP/AE user message file). The public authority is determined when the CSP/AE user message file is created. If the CRTAUT value for the library changes after the CSP/AE user message file is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the CSP/AE user message file, such as running a program or reading a file. The user cannot change the CSP/AE user message file. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** Users have the user *CHANGE, object existence, and object management authorities to the object.

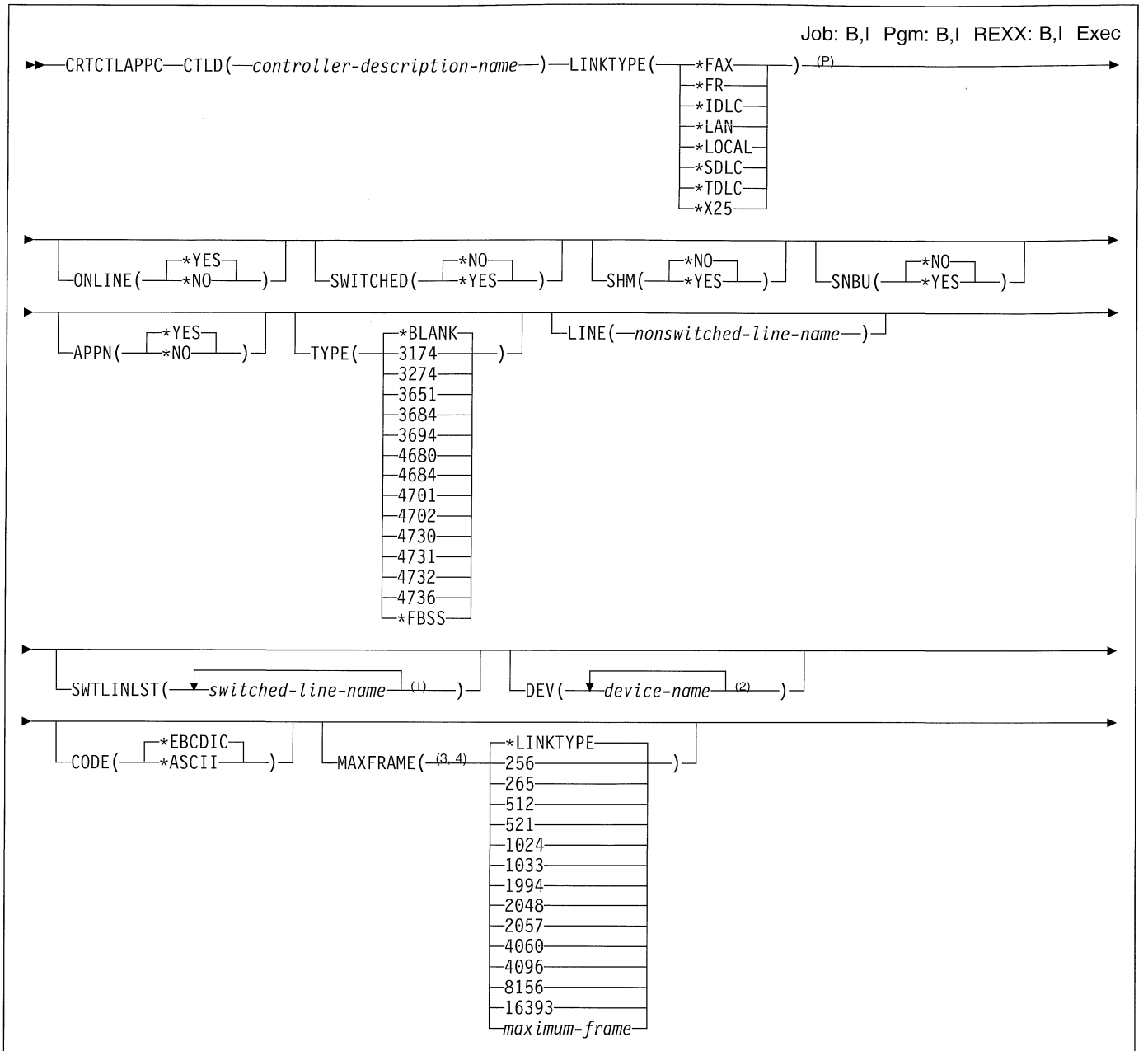
***EXCLUDE:** The user cannot access the CSP/AE user message file.

Example

```
CRTCSPMSGF   MSGF(PRODUCTION/PRL)
  FROMFILE(VMNET/QCSPMSG)  FROMMBR(*MSGF)
  RPLMSGD(*NO)  AUT(*CHANGE)
```

This command puts new messages from the member PRL of the file QCSPMSG in the library VMNET into the message file PRL in the library PRODUCTION. If matching message numbers are found in the message file, they are not replaced. Users have *CHANGE authority to the message file.

CRTCTLAPPC (Create Controller Description (APPC)) Command



Notes:

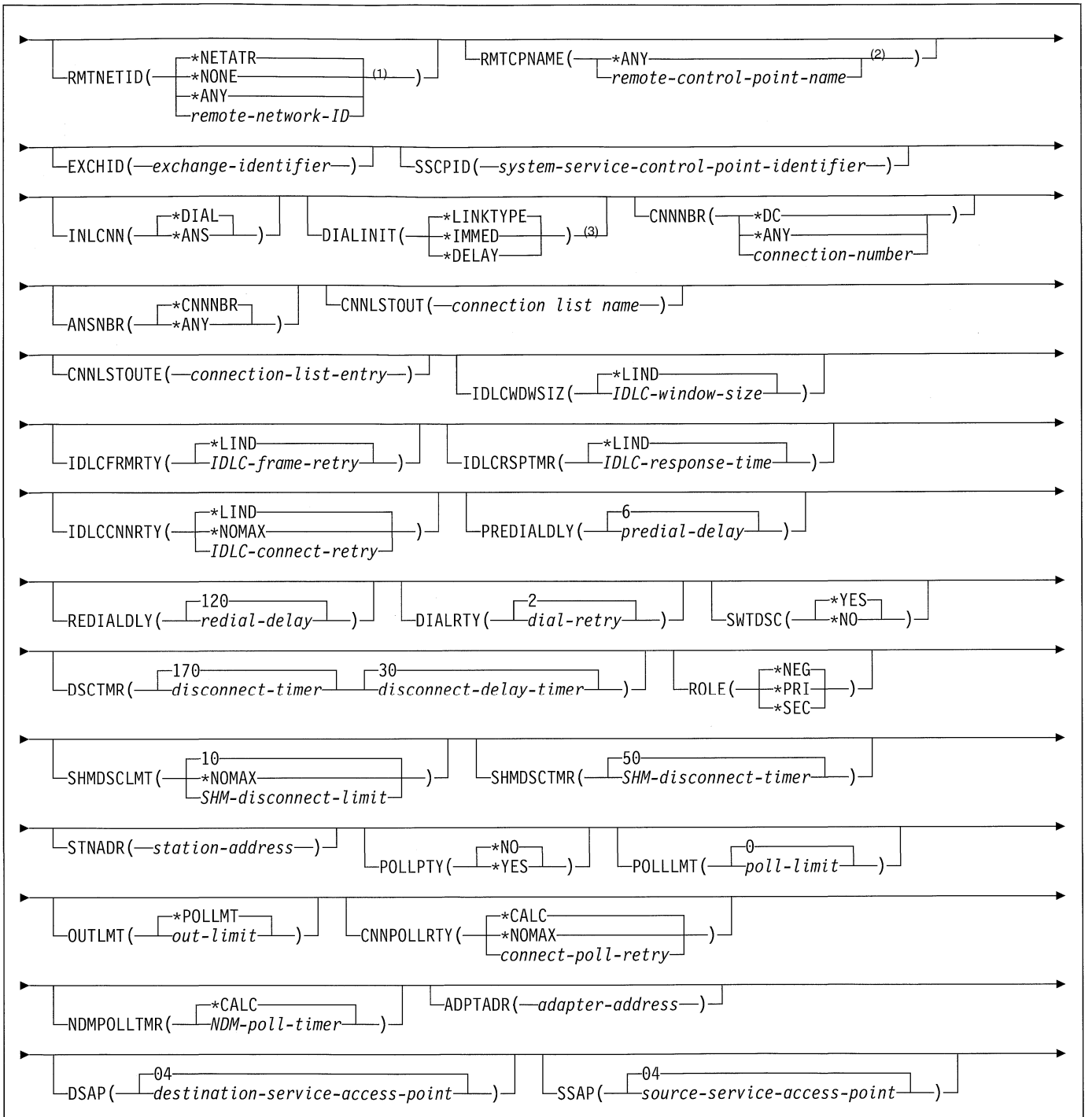
P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions

2 A maximum of 254 repetitions

3 If LINKTYPE is *LAN, valid values range from 265 through 16393.

4 For DDI LANs, valid values range from 265 through 4444.

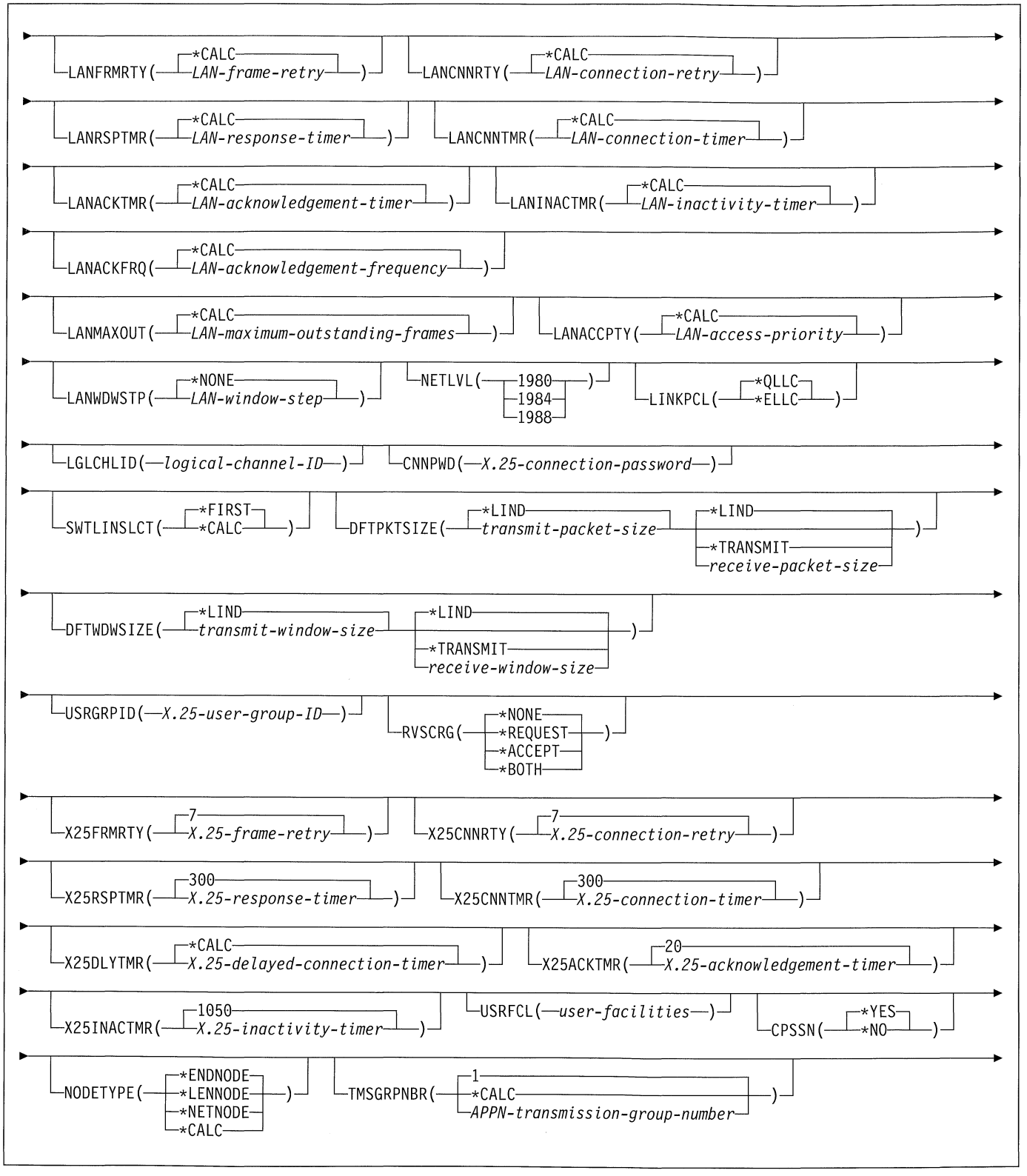


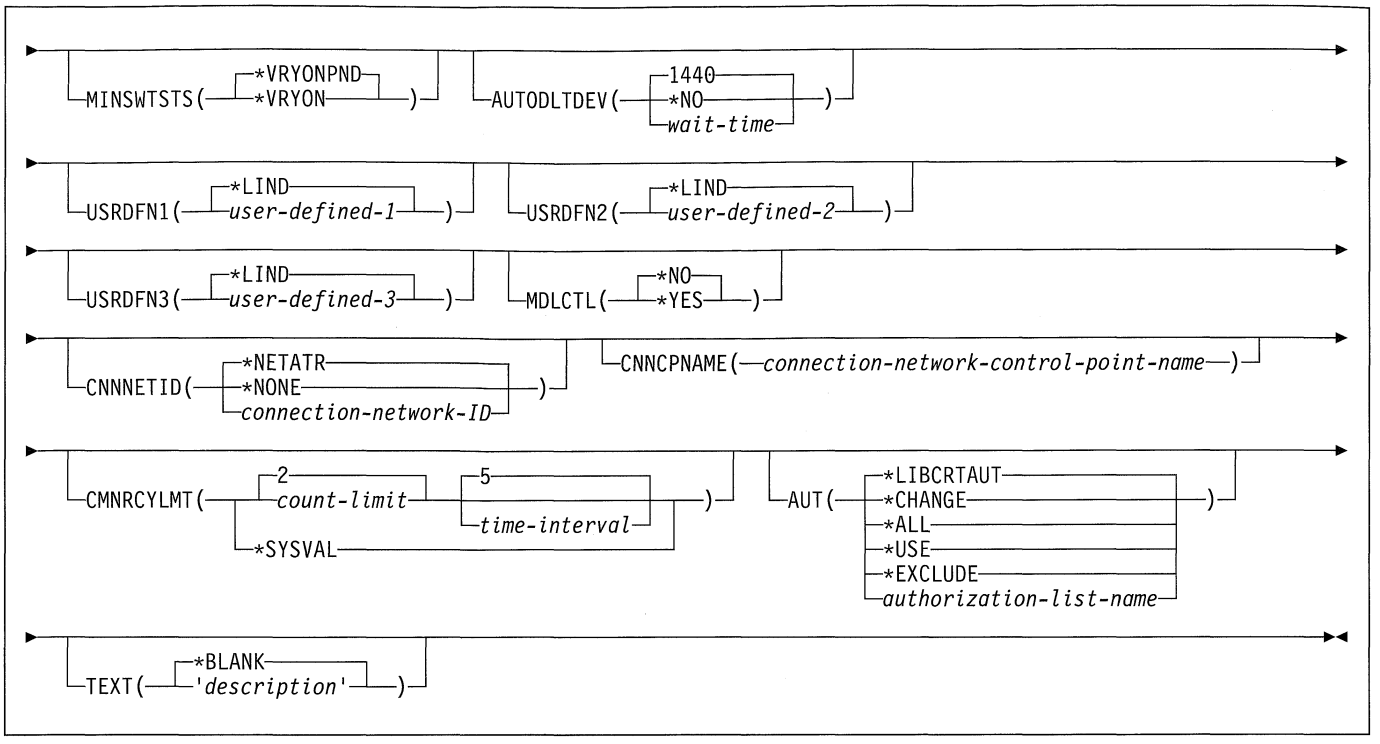
Notes:

1 If RMTNETID(*ANY) is specified, RMTCPNAME(*ANY) must also be specified.

2 RMTCPNAME(*ANY) cannot be specified if LINKTYPE is *IDLC and SWITCHED or SNBU is *YES; if LINKTYPE is *SDLC and SWITCHED or SNBU is *YES; if NODETYPE is *LENNODE; or if MDLCTL is *YES.

3 This parameter is valid only if INLCNN(*DIAL) is specified.


LANFRMRTY(*CALC LAN-frame-retry)
LANCNNRTY(*CALC LAN-connection-retry)
LANRSPTMR(*CALC LAN-response-timer)
LANCNNTMR(*CALC LAN-connection-timer)
LANACKTMR(*CALC LAN-acknowledgement-timer)
LANINACTMR(*CALC LAN-inactivity-timer)
LANACKFRQ(*CALC LAN-acknowledgement-frequency)
LANMAXOUT(*CALC LAN-maximum-outstanding-frames)
LANACCPTY(*CALC LAN-access-priority)
LANWDWSTP(*NONE LAN-window-step)
NETLVL(1980 1984 1988)
LINKPCL(*QLLC *ELLC)
LGLCHLID(logical-channel-ID)
CNNPWD(X.25-connection-password)
SWTLINSLCT(*FIRST *CALC)
DFTPKTSIZE(*LIND transmit-packet-size *LIND *TRANSMIT receive-packet-size)
DFTWDWSIZE(*LIND transmit-window-size *LIND *TRANSMIT receive-window-size)
USRGRPID(X.25-user-group-ID)
RVSCRG(*NONE *REQUEST *ACCEPT *BOTH)
X25FRMRTY(7 X.25-frame-retry)
X25CNNRTY(7 X.25-connection-retry)
X25RSPTMR(300 X.25-response-timer)
X25CNNTMR(300 X.25-connection-timer)
X25DLYTMR(*CALC X.25-delayed-connection-timer)
X25ACKTMR(20 X.25-acknowledgement-timer)
X25INACTMR(1050 X.25-inactivity-timer)
USRFCL(user-facilities)
CPSSN(*YES *NO)
NODETYPE(*ENDNODE *LENNODE *NETNODE *CALC)
TMSGRPNBR(1 *CALC APPN-transmission-group-number)



Purpose

The Create Controller Description (APPC) (CRTCTLAPPC) command creates a controller description for an advanced program-to-program communications (APPC) controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

LINKTYPE

Specifies the type of line to which this controller is attached.

***FAX:** This controller is attached to a local fax line. If *FAX is specified, the following criteria must be met:

- The following parameters can be specified: online at IPL (ONLINE), attached devices (DEV), authority (AUT), and text description (TEXT)
- The SWITCHED and APPN parameters must not be specified or have *NO specified
- APPN support is not provided; therefore, all attached devices must have APPN(*NO) specified

***FR:** This controller is attached to a frame relay line.

***IDLC:** This controller is attached to an integrated services digital network (ISDN) data link control (IDLC) line.

***LAN:** This controller is attached to a distributed data interface (DDI), Ethernet, or token-ring local area network line.

***LOCAL:** This controller uses no communication lines since all communications are local. If *LOCAL is specified, the following criteria must be met:

- SWITCHED, SNBU, and APPN must remain unspecified or be *NO
- The LINE, MAXFRAME, RMTNETID, RMTCPNAME, EXCHID, CPSSN, NODETYPE, TMSGRPNBR, SSCPID, and ROLE parameters must not be specified
- Parameters relating to SDLC, X.21 short-hold mode, X.25, or LAN configuration must not be specified
- Since LINKTYPE(*LOCAL) controllers must have APPN(*NO) specified, any attached devices must also have APPN(*NO) specified

***SDLC:** This controller is attached to a synchronous data link control (SDLC) line.

***TDLC:** This controller is attached to a twinaxial data link control (TDLC) line.

***X25:** This controller is attached to an X.25 line.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

SWITCHED

Specifies whether this controller is attached to a switched line, a token-ring network, Ethernet LAN, or an

X.25 switched virtual circuit (SVC). *NO must be specified for APPC controllers attached to a TDLC line.

***NO:** This controller is attached to a nonswitched line or to an X.25 permanent virtual circuit (PVC) connection.

***YES:** This controller is attached to a switched line, an X.25 switched virtual circuit (SVC), or a LAN line.

Note: If LINKTYPE is *LAN, the SWITCHED parameter value must be *YES or must not be specified.

SHM

Specifies whether this controller is used for X.21 short-hold mode. To specify *YES, LINKTYPE(*SDLC) and SWITCHED(*YES) must also be specified.

***NO:** This controller is not used for X.21 short-hold mode.

***YES:** This controller is used for X.21 short-hold mode.

SNBU

Specifies, for controllers supporting the switched network backup (SNBU) feature, whether the SNBU feature is activated or deactivated. Both the local and remote modems must support the SNBU feature to perform a valid activation.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote system's modem has the SNBU feature.

APPN

Specifies whether the local system uses advanced peer-to-peer networking (APPN) functions when communicating with this controller. *YES must be specified for APPC controllers attached to a TDLC line.

***YES:** The local system uses APPN functions.

***NO:** The local system does not use APPN functions.

TYPE

Specifies the type of controller for this description. This parameter is valid only when *NO is specified on the APPN parameter.

***BLANK:** No T2.0 controller is specified.

3174: This description represents a 3174 controller.

3274: This description represents a 3274 controller.

3651: This description represents a 3651 controller.

3684: This description represents a 3684 controller.

3694: This description represents a 3694 controller.

4680: This description represents a 4680 controller.

4684: This description represents a 4684 controller.

4701: This description represents a 4701 finance controller.

4702: This description represents a 4702 finance controller.

4730: This description represents a 4730 personal banking machine.

4731: This description represents a 4731 personal banking machine.

4732: This description represents a 4732 personal banking machine.

4736: This description represents a 4736 self-service transaction machine.

***FBSS:** This description represents a Financial Branch System Services (FBSS) controller.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

Note: The associated line must be varied off before this command is entered. For X.25, this parameter is used only for permanent virtual circuits (PVC).

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

DEV

Specifies the names of devices attached to this controller. The device descriptions must already have been created. Up to 254 devices can be specified.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

***LINKTYPE:** The maximum frame size depends on the type of link being used. One of the following values (measured in bytes) is used if *LINKTYPE is specified:

```
*SDLC -521     *LAN-16393     *TDLC-4105
*IDLC-2048     *X25-1024      *FR  -1590
```

maximum-frame: Specify the maximum frame size for the controller. The frame size that can be used depends on the type of line being used. Valid frame sizes for each line type are shown below.

Link Type	Frame Size (in bytes)
*FR	265 - 8182
*IDLC	265 - 8196
*LAN	265 - 16393 (265 - 4444 for DDI LANs)
*SDLC	265, 521, 1033, 2057
*X25	256, 265, 512, 521, 1024, 1033, 2048, 4096

Note: The numeric values listed for *LINKTYPE are valid only if TYPE(*BLANK) is specified when the controller is created.

RMTNETID
Specifies the name of the remote network in which the adjacent control point resides.

Note: If MDLCTL(*YES) is specified, parameters RMTNETID, RMTCPNAME, and ADPTADR are optional. More information on model controllers is in the MDLCTL parameter description and the *APPN Guide*.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

***ANY:** The system determines which remote network identifier is used.

remote-network-ID: Specify the remote network identifier.

RMTCPNAME
Specifies the name of the remote control point. This control point name must match the remote system's local control point name. The remote control point name can be seen on the remote system with the DSPNETA command.

Note: If MDLCTL(*YES) is specified, parameters RMTNETID, RMTCPNAME, and ADPTADR are optional. For more information on model controllers see the MDLCTL parameter and the *APPN Guide*.

***ANY:** The system determines the name of the remote control point used.

remote-control-point-name: Specify the remote control point name.

EXCHID
Specifies the exchange identifier of this controller. The controller sends (exchanges) its identifier to another location when a connection is established. The 8-digit hexadecimal identifier contains three digits for the block number and five digits for the identifier of the specific controller. This parameter is required for synchronous data link control (SDLC) switched or switched network backup (SNBU) connections.

Note: This parameter is required for both SDLC switched lines and SNBU lines when either:

- Both APPN(*YES) and NODETYPE(*LENNODE) *are* specified.
- APPN(*NO) is specified and RMTCPNAME *is not* specified.

SSCPID
Specifies the system service control point (SSCP) of the host system.

Note: The SSCP identifier is a 12-digit hexadecimal value ranging from 000000000001 to FFFFFFFFFFFF.

INLCNN
Specifies the method used to establish a connection with this controller.

***DIAL:** The connection is made by a call initiated from the AS/400 system.

For X.25 connection, the line attached to the controller requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH(*SVCOUT or *SVCBOTH) for the connection to succeed.

***ANS:** The connection is made by the AS/400 system when it answers an incoming call from this controller. If a call is received from the remote controller and all necessary conditions are met, the incoming call is answered by the system.

For X.25 connections, the line to which the controller attaches requires switched virtual circuits (SVC) configured on the LGLCHLE parameter of type IN or BOTH(*SVCIN or *SVCBOTH) for the connection to succeed. The line can be changed using the Change Line Description (X.25) (CHGLINX25) command.

DIALINIT
Specifies the method used to make the initial dial on a switched line between the system and the remote controller.

***LINKTYPE:** The type of dial connection initiated is specified on the LINKTYPE parameter. For LAN or SDLC short-hold mode connections, the default is to dial the connection immediately upon vary on of the controller description. For all other link types, the default is to delay the dial.

***IMMED:** The dial connection is initiated immediately upon vary on of the controller description.

***DELAY:** The dial connection is delayed until a job is initiated that requests the use of the remote controller resources.

CNNNBR
Specifies, for switched lines, the telephone number used to establish a switched connection on this line description. The value specified on this parameter is sent to the autocall unit if automatic calling is used to establish a switched connection, or it is sent to the V.25 bis modem if V.25 bis automatic dialing is used to establish the connection. When manual dialing is used to establish a switched connection on the line, this number is displayed to the system operator. For X.25, this

number is the remote data terminal equipment (DTE) address for the switched virtual circuit (SVC) provided by the X.25 network.

***DC:** For X.21 circuit switched connections, a direct call is used to connect to the controller.

***ANY:** Calls are accepted from any network address.

connection-number: Specify a telephone number, an X.25 network address, or an X.21 connection number.

ANSNBR

Specifies the X.25 network address from which to accept calls.

***CNNNBR:** Calls from the X.25 network address specified by the connection number (CNNNBR) parameter are accepted.

***ANY:** Calls are accepted from any X.25 network address.

CNNLSTOUT

Specifies, for ISDN switched connections, the name of a connection list object that contains the ISDN assigned numbers for a dial out operation to the ISDN.

CNNLSTOUTE

Specifies, for ISDN switched connections, the entry name from the connection list that is used to make a call to the ISDN. The connection list must be specified on the CNNLSTOUT parameter.

IDLCWDWSIZ

Specifies the window size for transmission to and reception from controllers attached to the IDLC line.

***LIND:** The window size specified in the line description is used.

IDLC-window-size: Specify the window size. Valid values range from 1 through 31.

IDLCFRMRTY

Specifies the maximum number of attempts to transmit before an error is reported.

***LIND:** The number of attempts specified in the line description is used.

IDLC-frame-retry: Specify the number of attempts. Valid values range from 0 through 100.

IDLCRSPTMR

Specifies the amount of time, in tenths of a seconds to wait before retransmitting a frame if acknowledgement has not been received.

***LIND:** The time specified in the line description is used.

IDLC-response-timer: Specify the amount of time to wait before retransmitting an unacknowledged frame of data. Valid values range from 10 through 100 tenths of a second. For example, 10 seconds equals 100 tenths of a second.

IDLCCNNRTY

Specifies the number of times to attempt re-transmission at connection time.

***LIND:** The number of attempts specified in the line description is used.

***NOMAX:** There is no disconnect limit.

IDLC-response-retry: Specify the number of attempts. Valid values range from 1 through 100.

PREDIALDLY

Specifies how long to wait (in 0.5 second intervals) before dialing.

Note: This parameter can be specified only if *YES is specified on either the SWITCHED or SNBU parameter, and LINKTYPE(*SDLC) and SHM(*NO) are both specified.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY

Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful. This parameter can be specified only if *YES is specified on either the SWITCHED or SNBU parameter, and if both LINKTYPE(*SDLC) and SHM(*NO) are specified.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

DIALRTY

Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

Note: This parameter can be specified only if *YES is specified on either the SWITCHED or SNBU parameter, and LINKTYPE(*SDLC) and SHM(*NO) are both specified.

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of times the dialing is tried.

SWTDSC

Specifies whether the switched connection to this controller is disconnected when the last session is unbound.

***YES:** The switched connection is disconnected when the last device is varied off.

***NO:** This switched connection is not dropped when the last session is unbound.

DSCTMR

Specifies options for controlling the time (in seconds) before an inactive connection is dropped (Element 1), or the amount of time to delay the automatic disconnection

(Element 2). If the user does not want the line to drop, specify *NO for the SWTDSC parameter.

Element 1: Minimum Connect Timer

170: A connection is dropped when it has been inactive for 170 seconds.

disconnect-timer: Specify a time to wait before disconnecting. Valid values range from 0 through 65535 seconds.

Element 2: Disconnect Delay Timer

30: The disconnection is delayed for 30 seconds.

disconnect-delay-timer: Specify a value to delay link take down after the last session on the controller is terminated. Valid values range from 0 through 65535 seconds.

Note: For objects migrated from V2R1M1 or previous systems, the disconnect delay timer has a value of zero. This value is interpreted by the system as 30 seconds.

ROLE

Specifies the data link role of the remote controller on this connection.

The primary station is the controlling station and the secondary station is the responding station. The primary station controls the communications link by sending commands to the secondary station, and the secondary station responds to the commands.

***NEG:** The AS/400 system and the remote system negotiate to determine which system is the primary station.

***PRI:** The remote system (controller) is the primary station on this communications line.

***SEC:** The remote system (controller) is a secondary station on this communications line.

SHMDSCLMT

Specifies the number of nonproductive responses (RR or RNR) that are required from the remote station before the connection can be suspended for this X.21 short-hold mode connection. This parameter is used only if *YES is specified on the SHM parameter.

10: Ten nonproductive responses must be received before the connection can be suspended.

***NOMAX:** There is no disconnect limit.

SHM-disconnect-limit: Specify a number ranging from 1 through 254, indicating the number of nonproductive responses that must be received before the connection can be suspended.

SHMDSCTMR

Specifies the minimum length of time that the primary system maintains the connection to the remote system for this X.21 short-hold mode controller, in tenths of a second. This parameter is used only if *YES is specified on the SHM parameter.

50: The primary system maintains the connection to the remote system for a minimum of 5 seconds.

SHM-disconnect-timer: Specify a value ranging from 2 through 3000, indicating the minimum length of time, in tenths of a second, that the primary maintains the connection to the remote system.

STNADR

Specifies, for SDLC or TDLC link types, the station address used when communicating with the controller. A hexadecimal value ranging from 01 through FE is used.

Note: If *SEC is specified on the ROLE parameter, this is the station address of the remote controller. If *PRI or *NEG is specified on the ROLE parameter, this is the local station address.

POLLPTY

Specifies, for an SDLC secondary or negotiable controller, whether this controller has priority when polled. This parameter can be specified only if SHM is *NO.

***NO:** This controller does not have polling priority.

***YES:** This controller has polling priority.

POLLLMT

Specifies, for an SDLC secondary or negotiable controller, the number of consecutive polls issued to the same controller when the poll results in receiving frames. This parameter can be specified only if *NO is specified on the SHM parameter.

0: No subsequent polls are attempted.

poll-limit: Specify a value ranging from 0 through 4 for the number of consecutive polls.

OUTLMT

Specifies the number of times SDLC allows the consecutive transmission of the maximum number of frames to a station before allowing transmission to another station.

***POLLLMT:** The value specified on the POLLLMT parameter is used.

out-limit: Specify a value ranging from 0 through 4 for the number of consecutive transmissions.

CNNPOLLRTY

Specifies, for an SDLC secondary or negotiable controller, the number of connect poll retries before reporting the error. This parameter can be specified only if *NO is specified on the SHM parameter.

***CALC:** The number of retries is 7 if the controller is switched, and *NOMAX if the controller is nonswitched.

***NOMAX:** There is no disconnect limit.

connect-poll-retry: Specify a value ranging from 0 through 65534 connect poll retries.

NDMPOLLTMR

Specifies the minimum interval at which a secondary station must be polled if a poll from the primary to the

secondary (which is in normal response mode) does not result in receiving appropriate response.

This prompt is valid only if the link type is *SDLC and the controller role is secondary or negotiable and *NO is specified on the SHM parameter.

***CALC:** The system determines the value to use.

NDM-poll-timer: Specify a value ranging from 1 through 3000 in 0.1-second intervals, or specify 0 to indicate no timer.

ADPTADR

Specifies the 12-character adapter address of the remote controller. This is the address to which the system sends data when it communicates with the remote controller. This value can be obtained from the remote controller's configuration record. Valid values range from hex 000000000001 through hex FFFFFFFFFFFF.

DSAP

Specifies the logical address of the SNA destination service access point (DSAP) to which this system sends data when it communicates with the remote controller. This address allows the controller and the system to route the data that comes from this system. The value must match the value specified on the source service access point (SSAP) parameter in the remote controller's configuration record.

04: The destination service access point is the default 04.

destination-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the destination service access point.

SSAP

Specifies the source service access point. This is the logical address this system uses when it sends data to the remote controller. This address allows the remote controller and the system to route the data that comes from this system. It must match the value assigned to the destination service access point (DSAP) prompt in the remote controller's configuration record.

04: The system uses the logical address of 04.

source-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the source service access point.

LANFRMRTY

Specifies the number of times a frame is transmitted if there is no acknowledgement from the remote controller in the time period specified by the LANRSPTMR parameter. This value is only used after a successful connection has been made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-frame-retry: Specify a value ranging from 0 through 254 for the number of times a frame is transmitted before an acknowledgement is received.

LANCNNRTY

Specifies the number of times a transmission is attempted before an acknowledgement is received. This value is used at connection time (unlike LANFRMRTY which is used *after* a connection has been made).

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-retry: Specify a value ranging from 0 through 254 for the number of times the transmission is attempted before an acknowledgement is received.

LANRSPTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link after a connection is made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-response-timer: Specify a value ranging from 1 through 254 in 0.1- second intervals for the amount of time the system waits before an inoperative condition occurs.

LANCNNTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link at connection time.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANACKTMR

Specifies the length of time the system waits before sending an acknowledgement for received frames.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgement-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals, or specify 0 to indicate no delay. If 0 is specified for this parameter, 0 must also be specified for the LANACKFRQ parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKFRQ parameter.

LANINACTMR

Specifies the length of time used to determine an inactive condition for the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-inactivity-timer: Specify a value ranging from 1 through 255 in 0.1-second intervals for the length of time used to determine an inactive condition for the controller.

LANACKFRQ
Specifies the maximum number of frames that are received before an acknowledgement is sent to the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgement-frequency: Specify a value ranging from 0 through 127 for the number of frames received. If 0 is specified for this parameter, 0 must also be specified for the LANACKTMR parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKTMR parameter.

LANMAXOUT
Specifies the maximum number of frames that can be sent before an acknowledgement is received from the remote system.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-maximum-outstanding-frames: Specify a value ranging from 1 through 127 for the number of frames that can be sent before an acknowledgement is received.

LANACCPTY
Specifies the priority granted to the sending system for sending frames. The larger the number, the higher the priority. This parameter is only used when the controller attaches to TRLAN.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-access-priority: Specify a value ranging from 0 through 3 to indicate how soon a frame can be sent on a local area network (LAN).

LANWDWSTP
Specifies whether to reduce to 1 the maximum number of frames outstanding to the remote system during network congestion. This parameter indicates the number of frames that must be successfully received by the remote system before the number of maximum outstanding frames can be increased by 1. The increase continues this way until the maximum number of outstanding frames reaches the value specified by the LAN maximum outstanding frames (LANMAXOUT) parameter.

***NONE:** The number of outstanding frames is not reduced during network congestion.

LAN-window-step: Specify a value from 1 to 127 for the number of frames that must be successfully received by the remote system before the maximum number of outstanding frames can be increased by 1.

NETLVL
Specifies the level of X.25 support implemented by the network and remote DTE. The level is specified by giving the year of the CCITT standard implemented.

Note: Use of the lower value of the remote DTE or the network level is suggested; for example, if the remote DTE is using the CCITT standard of 1980 and the network 1984, specify 1980 for this prompt.

1980: The 1980 standard is used.

1984: The 1984 standard is used.

1988: The 1988 standard is used.

LINKPCL
Specifies the link level protocol used on the X.25 network to communicate with this controller.

***QLLC:** The qualified logical link control (QLLC) protocol is used.

***ELLC:** The enhanced logical link control (ELLC) protocol is used.

LGLCHLID
Identifies the logical channel ID in the X.25 line description that this permanent virtual circuit (PVC) uses.

The valid entry is xyy. Where:

```
x  = the logical group number derived from the
     user network subscription.
yy = the logical channel number derived from
     the user subscription.
```

CNNPWD
Specifies, for X.25 switched virtual circuit (SVC) controllers, the password used when connecting to this controller.

Note: The password for each controller can consist of any alphanumeric characters represented by the hexadecimal range from 40 through FF. When less than 8 bytes are specified, the end of the password field is padded with blanks. A password of all blank is valid. All incoming call requests must match this password to establish a connection.

SWTLINSLCT
Specifies the method that is used to select lines from an X.25 switched line list.

***FIRST:** The lines are selected in the order in which they are specified.

***CALC:** The system determines the value to use.

DFTPKTSIZE
Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

***LIND:** The value specified in the line description is used as the default packet size.

transmit-packet-size: Specify a default packet size for transmission. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***LIND:** The value specified in the line description is used as the default packet size.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

DFTWDWSIZE

Specifies the default window size used by the X.25 network.

Element 1: Transmit Window Size

***LIND:** The value specified in the line description is used as the default window size.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

Element 2: Receive Window Size

***LIND:** The value specified in the line description is used as the default window size.

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

USRGRPID

Specifies the closed user-group ID for contacting this X.25 switched virtual circuit (SVC) controller. Specify a value ranging from 0 through 99, as provided by the network subscription. This parameter is not valid for permanent virtual circuit (PVC) connections. It is valid only for SVC circuit outgoing call operations and is ignored for SVC incoming call connections.

RVSCRG

Specifies whether reverse charges are accepted or requested when contacting this controller.

***NONE:** No reverse charging for network tariff billing is accepted.

***REQUEST:** Charges are requested on outgoing call request packets.

***ACCEPT:** Reverse charging for network tariff billing is accepted on incoming requests.

***BOTH:** Both incoming and outgoing requests are accepted.

X25FRMRTY

Specifies the maximum number of times that a logical link control (LLC) protocol data unit is sent after the response timer ends when connected to this controller. The value used in this prompt depends on the quality of service provided by the network and the connection to that network; that is, the frequency of lost link protocol data units.

7: The frame is sent seven times.

X.25-frame-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25CNNRTY

Specifies, for X.25 controllers specifying *SEC or *NEG on the ROLE parameter, the maximum number of times that an LLC is sent after the connect response timer ends when connecting to this controller.

7: The frame is sent seven times.

X.25-connection-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25RSPTMR

Specifies the time allowed to return an acknowledgement when an LLC is sent while connecting to this controller.

300: The time allowed to return an acknowledgement is 30 seconds.

X.25-response-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25CNNTMR

Specifies, for X.25 controllers specifying *SEC or *NEG on the ROLE parameter, the time allowed for an acknowledgement to be returned when an LLC is sent while connecting to this controller.

300: The time allowed for an acknowledgement to be returned is 30 seconds.

X.25-connection-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25DLYTMR

Specifies, for X.25 controllers specifying *SEC or *NEG on the ROLE parameter, how often to try establishing a connection to the controller.

***CALC:** The system determines the value to use.

X.25-delayed-connection-timer: Specify a value ranging from 1 through 32767 in 0.1-second intervals. The system retries the connection indefinitely at the specified interval.

X25ACKTMR

Specifies, for X.25 links using an enhanced logical link control (ELLC) link protocol, the time period to delay sending acknowledgements for received LLC.

20: The time allowed to delay sending an acknowledgement is 2 seconds.

X.25-acknowledgement-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals, or specify 0 to indicate no delay in sending an acknowledgement.

X25INACTMR
Specifies, for X.25 links using an ELLC link protocol, the time period used to determine an inactive condition for the controller.

1050: An inactive condition occurs after 105 seconds.

X.25-inactivity-timer: Specify a value ranging from 1 through 2550 0.1-second intervals.

USRFCL
Specifies a string of hexadecimal digits sent to the X.25 network to request additional services. The AS/400 system allows up to 218 hexadecimal characters.

CPSSN
Specifies whether this controller supports APPN control-point to control-point (CP to CP) sessions.

***YES:** This controller supports CP to CP sessions.

***NO:** This controller does not support CP to CP sessions.

NODETYPE
Specifies the type of APPN node this controller represents.

***ENDNODE:** This node is an end node in an APPN network.

***LENNODE:** This node is a low-entry networking node in an APPN network.

***NETNODE:** This node is a network node in an APPN network.

***CALC:** The system determines the value to use.

TMSGRPNBR
Specifies the APPN transmission group number for this controller.

1: The transmission group number is 1.

***CALC:** The system determines the value to use.

APPN-transmission-group-number: Specify a value ranging from 1 through 20 for the transmission group number.

MINSWTSTS
Specifies, for the switched connection, the minimum status that APPN requires for a controller to be considered available for routing.

***VRYONPND:** APPN considers the controller available for routing if the status is vary on pending, varied on, or active.

***VRYON:** Indicates that APPN considers the controller available for routing only if the status is varied on or active.

AUTODLTDEV
Specifies the number of minutes an automatically created device remains when the last session is unbound before the device description is varied off and deleted.

1440: The system will automatically vary off and delete the automatically-configured idle device descriptions after 1440 minutes (24 hours).

***NO:** The system will not automatically vary off and delete the automatically-configured idle device descriptions.

wait-time: Specify the number of minutes to wait before deleting the automatically-configured idle device descriptions for this controller. Valid values range from 1 to 10,000.

USRDFN1
Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

***LIND:** The user-defined value specified in the line description is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2
Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

***LIND:** The user-defined value specified in the line description is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3
Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

***LIND:** The user-defined value specified in the line description is used.

user-defined-3: Specify a value ranging from 0 through 255.

MDLCTL
Specifies whether this controller description is a *model* for automatically created controller descriptions. Values on the model description, such as timer delays, retry limits, and frame size, are used for new controller descriptions that are automatically created and configured when communications with a remote system is started. The new controller must be attached to one of the SINGLE line descriptions in the switched line list (SWTLINLST parameter) of the model controller.

A model controller description is not attached to any devices, and only one controller description can be varied on for each line description.

For more information on model controllers, see the *APPN Guide*.

***NO:** This controller description is not used as a model controller description.

***YES:** This controller description is used as a model for automatically created controller descriptions.

Note: This parameter is only valid if the parameter LINKTYPE is *LAN.

CNNNETID

Specifies the connection network identifier of this controller description. If a value is specified for this parameter (other than *NONE), this controller description represents this connection to the connection network.

Note: This parameter is only valid if MDLCTL(*YES) is specified.

***NETATR:** The LCLNETID value specified in the system network attributes is used.

***NONE:** No network connection identifier is used.

connection-network-ID: Specify the network connection identifier that represents this controller description to the network.

CNNCPNAME

Specifies the connection network control point name for this controller. If this value and a CNNNETID value (other than *NONE) are specified, this controller description represents this connection to the connection network.

Note: This parameter is only valid if MDLCTL(*YES) is specified.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the object and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

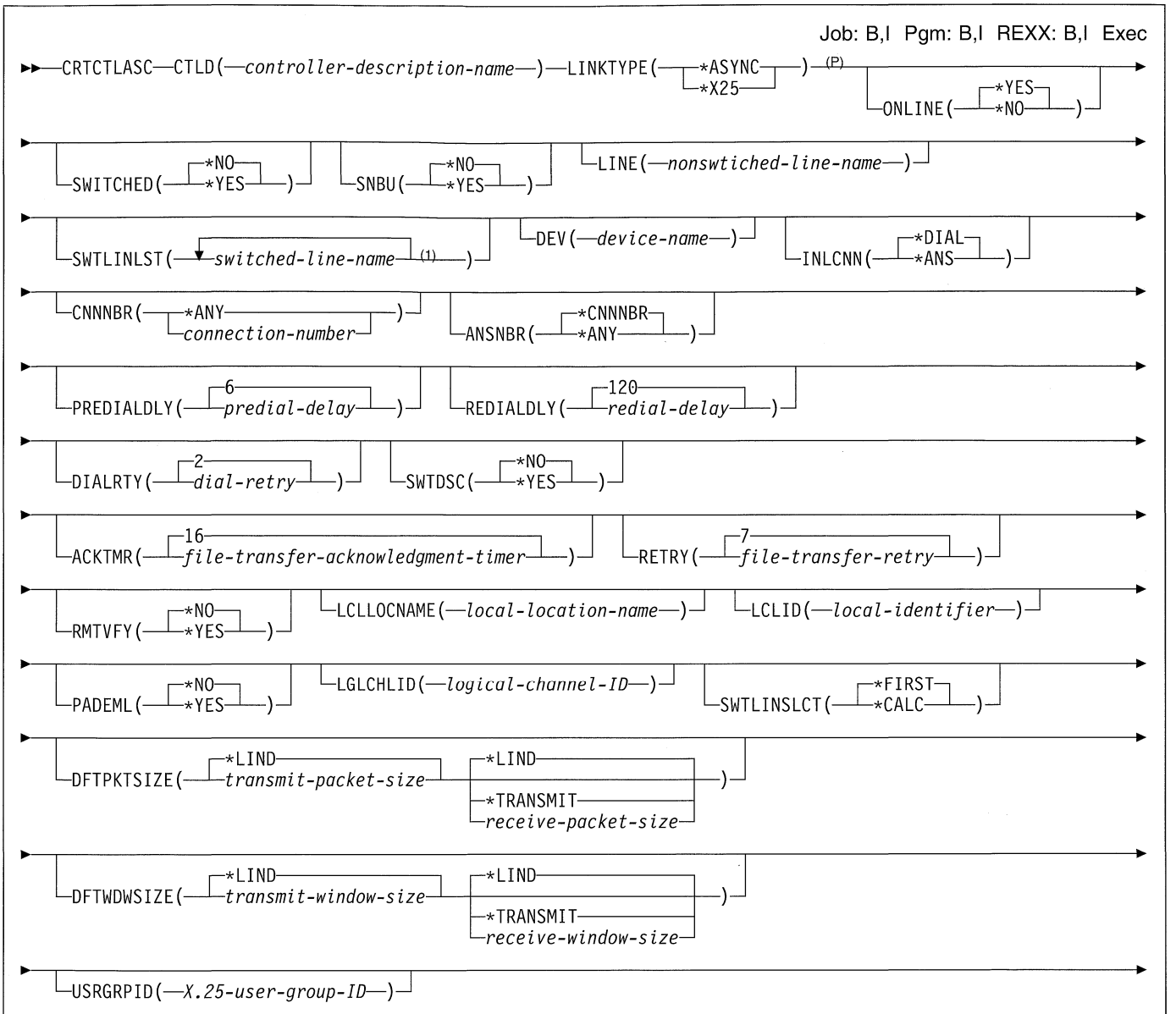
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLAPPC   CTLD(OUTLAND)  LINKTYPE(*SDLC)
  APPN(*NO)  LINE(OUTLINE)  RMTNETID(*NONE)
  STNADR(C1)
```

This command configures an APPC controller description attached to a nonswitched SDLC line with a station address of C1.

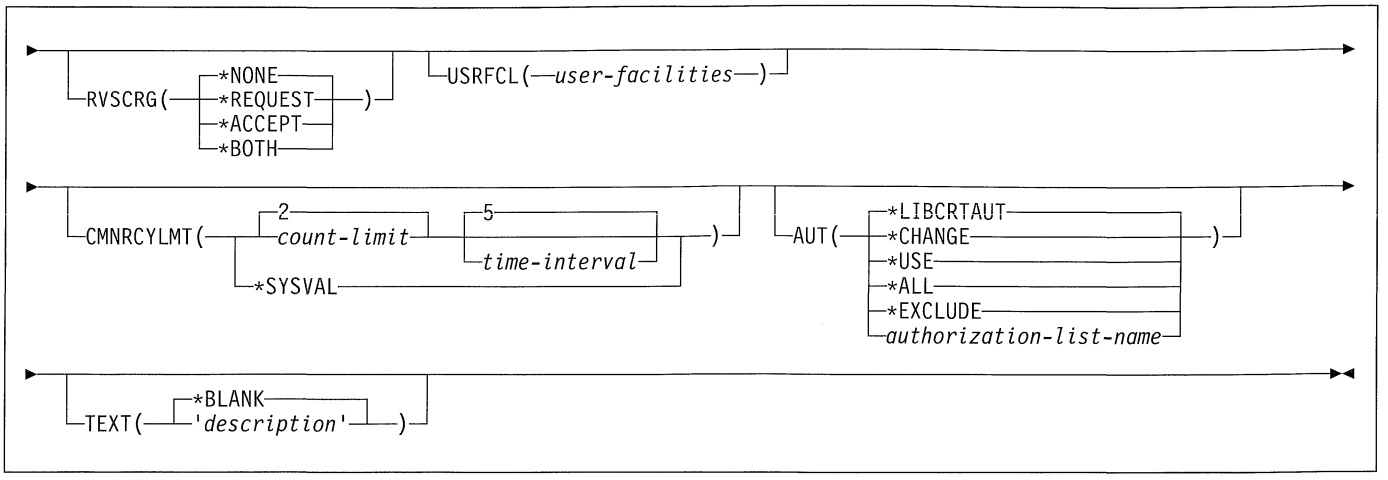
CRTCTLASC (Create Controller Description (Async)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions



Purpose

The Create Controller Description (Async) (CRTCTLASC) command creates a controller description for an asynchronous controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

LINKTYPE

Specifies the type of line to which this controller is attached.

***ASYNC:** This controller is attached to an asynchronous line.

***X25:** This controller is attached to an X.25 line.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

SWITCHED

Specifies whether this controller is attached to a switched line or to an X.25 switched virtual circuit (SVC).

***NO:** The controller is attached to a nonswitched line or, for X.25, to a permanent virtual circuit (PVC).

***YES:** The controller is attached to a switched line or, for X.25, to a switched virtual circuit (SVC).

SNBU

Specifies, for controllers supporting the switched network backup (SNBU) feature, whether the SNBU feature is activated or deactivated. Both the local and remote modems must support the SNBU feature to perform a valid activation.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote modem has the SNBU feature.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

Note: This parameter can be specified only if the user specified *NO for the switched line (SWITCHED parameter) for asynchronous communications controllers. If this controller is attached to an X.25 permanent virtual circuit (PVC), specify the name of the line on which that PVC is configured.

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

DEV

Specifies the name of a device attached to this controller. Asynchronous communications supports only one device attached to each controller. The device name must be the same as that specified when the associated device description was created. The device description must already exist.

INLCNN

Specifies the method used to establish a connection with this controller.

***DIAL:** The connection is made by a call initiated from the AS/400 system.

For X.25 connection, the line attached to the controller requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH(*SVCOUT or *SVCBOTH) for the connection to succeed.

***ANS:** The connection is made by the AS/400 system when it answers an incoming call from this controller. If a call is received from the remote controller and all necessary conditions are met, the incoming call is answered by the system.

For X.25 connections, the line to which the controller attaches requires switched virtual circuits (SVC) configured on the LGLCHLE parameter of type IN or BOTH(*SVCIN or *SVCBOTH) for the connection to succeed. The line can be changed using the Change Line Description (X.25) (CHGLINX25) command.

CNNNBR

Specifies the telephone number to dial to connect to this controller.

***ANY:** Any X.25 SVC incoming call can use this controller.

connection-number: Specify the connection number used to call this controller. This could be a telephone number or an X.25 network address.

ANSNBR

Specifies the X.25 network address from which to accept calls.

***CNNNBR:** Calls from the X.25 network address specified by the connection number (CNNNBR) parameter are accepted.

***ANY:** Calls are accepted from any X.25 network address.

PREDIALDLY

Specifies how long to wait (in 0.5 second intervals) before dialing.

Note: Predial delay can be specified only if switched line or switched network backup is *YES and the link type is *ASYNC for asynchronous controllers.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY

Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful.

Note: Redial delay can be specified only if switched line or switched network backup is *YES and the link type is *ASYNC for asynchronous controllers.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a number ranging from 1 through 254 in 0.5- second intervals, or specify 0 to indicate no delay.

DIALRTY

Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

Note: Dial retries can be specified only if switched line or switched network backup is *YES and the link type is *ASYNC for asynchronous controllers.

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of retries.

SWTDSC

Specifies whether the switched connection to this controller is made inoperative when the last device is varied off.

***NO:** The switched connection is not dropped when the last device is varied off.

***YES:** The switched connection is disconnected when the last device is varied off.

ACKTMR

Specifies the time allowed for an acknowledgment on an asynchronous file transfer.

16: The default value is 16 seconds.

file-transfer-acknowledgment-timer: Specify a value ranging from 16 through 65535 seconds for an acknowledgment on an Async file transfer.

RETRY

Specifies the number of retries on an Async file transfer.

7: Seven is the default for number of retries.

file-transfer-retry: Specify a value ranging from 1 through 255 for the number of retries.

RMTVFY

Specifies whether the remote system requires verification of local location name and local ID. The remote system requires verification if a generic controller and device are configured to accept calls from any X.25 network address.

***NO:** The remote system does not require verification of local location name and local ID.

***YES:** The remote system requires verification of the local location name and local ID.

LCLLOCNAME

Specifies the local location name.

Note: This name must be the same as that specified by the remote system in its remote location list.

LCLID

Specifies the local ID which, when combined with the local location name, identifies the controller to a remote system. This ID must be the same ID as that specified by the remote system in its remote location list.

PADEML

Specifies whether this controller emulates an X.25 packet assembler/disassembler (PAD). This PAD emulation follows CCITT recommendations for X.3, X.28, and X.29. PAD emulation is valid only for asynchronous

controllers for which the initial connection(INLCNN) is set to *DIAL.

***NO:** This controller does not emulate an X.25 packet assembler/disassembler (PAD).

***YES:** This controller emulates an X.25 packet assembler/disassembler (PAD).

LGLCHLID

Specifies the logical channel ID used for the X.25 permanent virtual circuit (PVC) to this controller. Specify the logical channel ID.

The valid entry is xyy, where:

x = the logical group number, derived from the network subscription.

yy = the logical channel number, derived from the user's subscription. The logical channel identifier must be one of the PVC logical channel identifiers that was defined in the X.25 line description. There is no default for this parameter.

SWTLINSLCT

Specifies the method that is used to select lines from an X.25 switched line list.

***FIRST:** The lines are selected in the order in which they are specified.

***CALC:** The system determines the value to use.

DFTPKTSIZE

Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

***LIND:** The value specified in the line description is used as the controller.

transmit-packet-size: Specify a default packet size for transmission. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***LIND:** The value specified in the line description is used as the controller.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

DFTWDWSIZE

Specifies the default window size used by the X.25 network.

Element 1: Transmit Window Size

***LIND:** The value specified in the line description is used as the controller.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

Element 2: Receive Window Size

***LIND:** The value specified in the line description is used as the controller.

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

USRGRPID

Specifies the closed user-group ID for contacting this X.25 switched virtual circuit (SVC) controller. Specify a value ranging from 0 through 99, as provided by the network subscription. This parameter is not valid for permanent virtual circuit (PVC) connections. It is valid only for SVC circuit outgoing call operations and is ignored for SVC incoming call connections.

RVSCRG

Specifies whether reverse charges are accepted or requested when contacting this controller.

***NONE:** No reverse charging for network tariff billing is accepted.

***REQUEST:** Charges are requested on outgoing call request packets.

***ACCEPT:** Reverse charging for network tariff billing is accepted on incoming requests.

***BOTH:** Both incoming and outgoing requests are accepted.

USRFCL

Specifies a string of hexadecimal digits sent to the X.25 network to request additional services. The AS/400 system allows up to 218 hexadecimal characters.

Note: Values entered in this field are determined by the supplier of the network subscription. Do not include coding of facilities shown through keywords: packet size, window size, user group identifier, and reverse charging.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Values

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the object and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

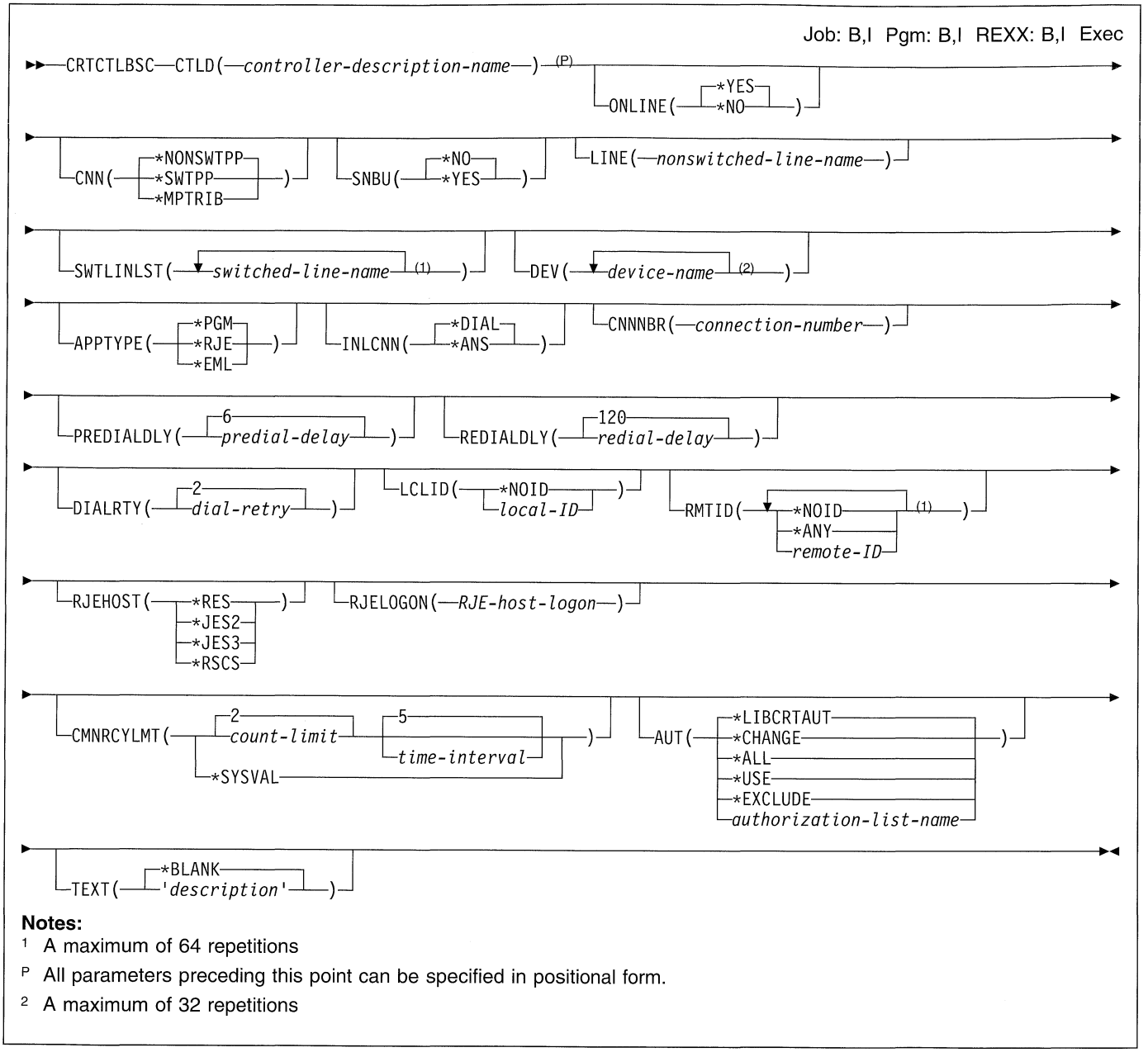
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLASC   CTLD(ASCCTL)  LINKTYPE(*ASYNC)
  LINE(ASCLIN)
```

This command creates an asynchronous controller attached to a nonswitched asynchronous line.

CRTCTLBSC (Create Controller Description (BSC)) Command



Notes:

1 A maximum of 64 repetitions

P All parameters preceding this point can be specified in positional form.

2 A maximum of 32 repetitions

Purpose

The Create Controller Description (BSC) (CRTCTLBSC) command creates a controller description for a binary synchronous communications (BSC) controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameter

CTLD

Specifies the name of the controller description being created.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

CNN

Specifies the type of line connection used.

Note: This value must match the value specified for the CNN parameter in the CRTLINBSC command.

***NONSWTPP:** A nonswitched point-to-point connection is used. This value cannot be selected if *EML is chosen for the APPTYPE parameter.

***SWTPP:** A switched point-to-point connection is used. This value cannot be selected if *EML is chosen for the APPTYPE parameter.

***MPTRIB:** A multipoint tributary connection is used. This value cannot be selected if *RJE is chosen for the APPTYPE parameter.

SNBU

Specifies, for controllers supporting the switched network backup (SNBU) feature, whether the SNBU feature is activated or deactivated. Both the local and remote modems must support the SNBU feature to perform a valid activation.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote modem has the SNBU feature.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

DEV

Specifies the names of the devices attached to this controller. The device descriptions must already exist.

Use this parameter only when the associated device descriptions have already been created. The maximum number of devices that can attach to the controller is 32 when specifying CNN(*MPTRIB), 24 when specifying APPTYPE(*RJE), and 1 for all other cases.

APPTYPE

Specifies the application type being used.

Note: This value must match the value specified on the APPTYPE parameter in the CRTLINBSC command.

***PGM:** The application is a user-written program.

***RJE:** The application is a BSC Remote Job Entry (RJE).

***EML:** The application is a 3270 device emulation.

INLCNN

Specifies the method used to establish a connection with this controller.

***DIAL:** The connection is made by a call initiated from the AS/400 system.

For X.25 connection, the line attached to the controller requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH(*SVCOUT or *SVCBOTH) for the connection to succeed.

***ANS:** The connection is made by the AS/400 system when it answers an incoming call from this controller. If a call is received from the remote controller and all necessary conditions are met, the incoming call is answered by the system.

For X.25 connections, the line to which the controller attaches requires switched virtual circuits (SVC) configured on the LGLCHLE parameter of type IN or BOTH(*SVCIN or *SVCBOTH) for the connection to succeed. The line can be changed using the Change Line Description (X.25) (CHGLINX25) command.

CNNNBR

Specifies the telephone number to dial to connect to this controller.

PREDIALDLY

Specifies how long to wait (in 0.5 second intervals) before dialing.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY

Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a number ranging from 1 through 254 in 0.5- second intervals, or specify 0 to indicate no delay.

DIALRTY

Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of retries.

LCLID

Specifies the local ID used to identify the system to the remote controller.

***NOID:** The AS/400 system sends a null identifier.

local-ID: The local ID must be from 4 to 30 hexadecimal characters long and must contain an even number of characters. If only four characters are specified, the first two and last two must be the same, such as F3F3 or 8484.

RMTID

Specifies a list of identifiers for remote BSC controllers. Each entry is from 4 to 30 hexadecimal characters long and must contain an even number of characters. Up to 64 remote controller ID's can be specified.

***NOID:** The AS/400 system accepts a null identifier.

***ANY:** The AS/400 system accepts any identifier sent by the remote controller. It must be the last, or only, entry in the list.

remote-ID: Specify the identifier or a list of identifiers (64 maximum) used by remote BSC controllers. The identifier must be specified in hexadecimal form and cannot contain any BSC control characters.

RJEHOST

Specifies the subsystem type of the host to which the Remote Job Entry (RJE) is connected.

***RES:** This value specifies that the host is an RES (Remote Entry System).

***JES2:** This value specifies that the host is a JES2 (Job Entry Subsystem 2).

***JES3:** This value specifies that the host is a JES3 (Job Entry Subsystem 3).

***RSCS:** This value specifies that the host is an RSCS (Remote Spooling Communications System).

Note: This parameter is required only when APPTYPE(*RJE) is specified.

RJELOGON

Specifies the sign-on (logon) text for the RJE host system.

Note: This parameter is required only when APPTYPE(*RJE) is specified. Specify the sign-on information required by the host system.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

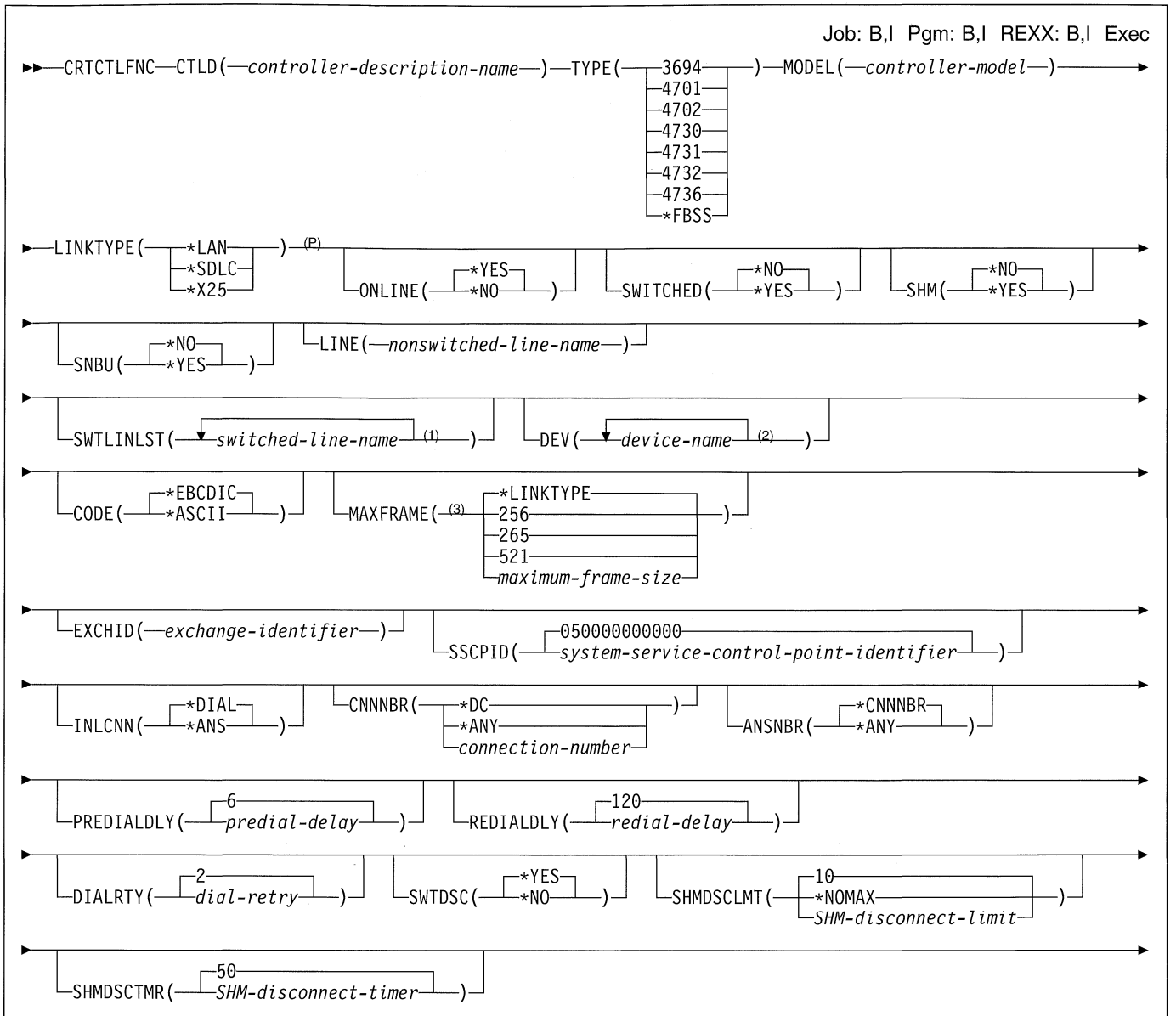
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLBSC   CTLD(BSC1)  CNN(*SWTPP)
  CNNNBR(1234567)  LCLID(020202)
  RMTID(*ANY)  SWTLINLST(BSC1)
  INLCNN(*ANS)
```

This command creates a BSC controller on a switched line that accepts calls from any remote location.

CRTCTLFNC (Create Controller Description (Finance)) Command



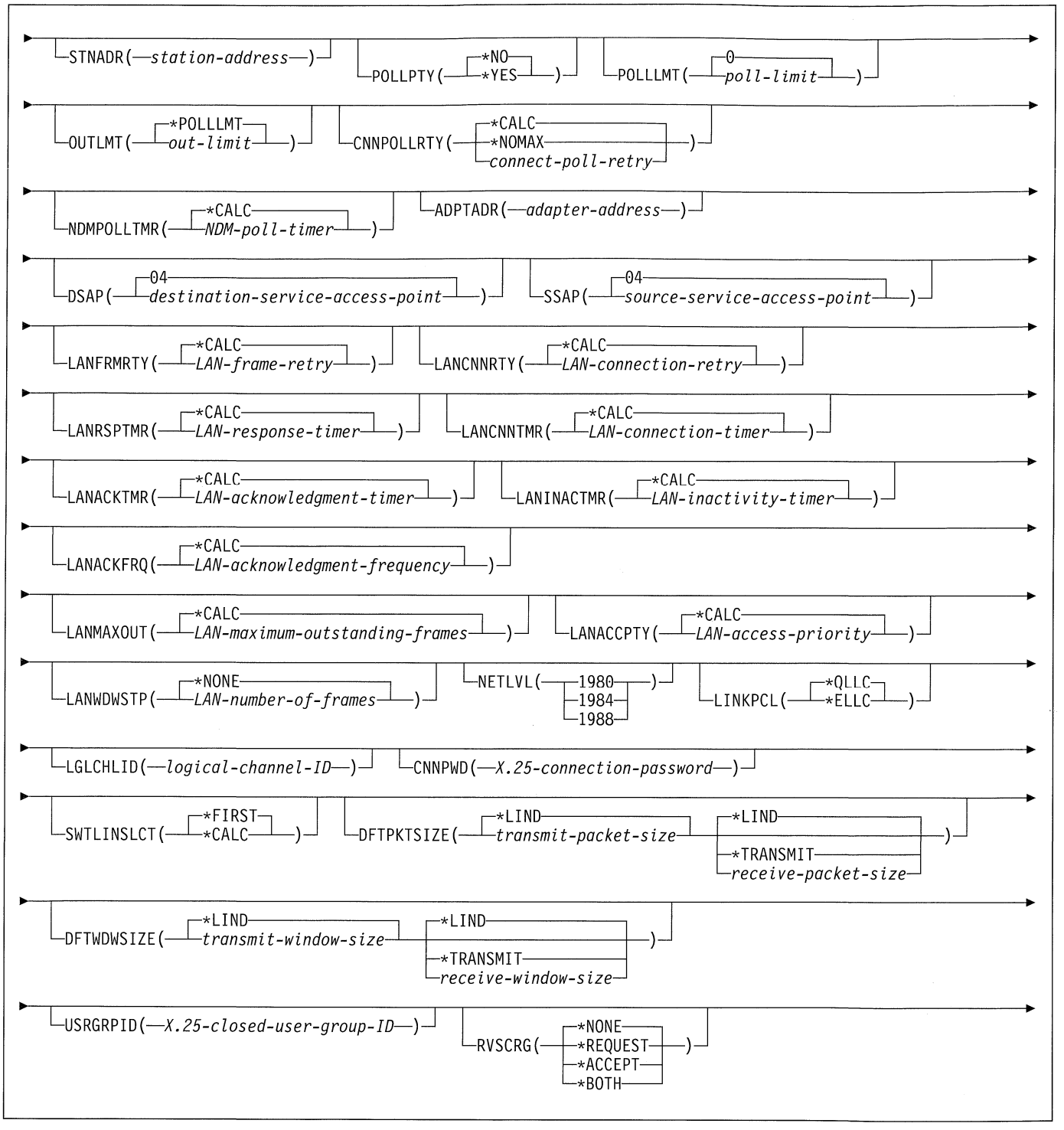
Notes:

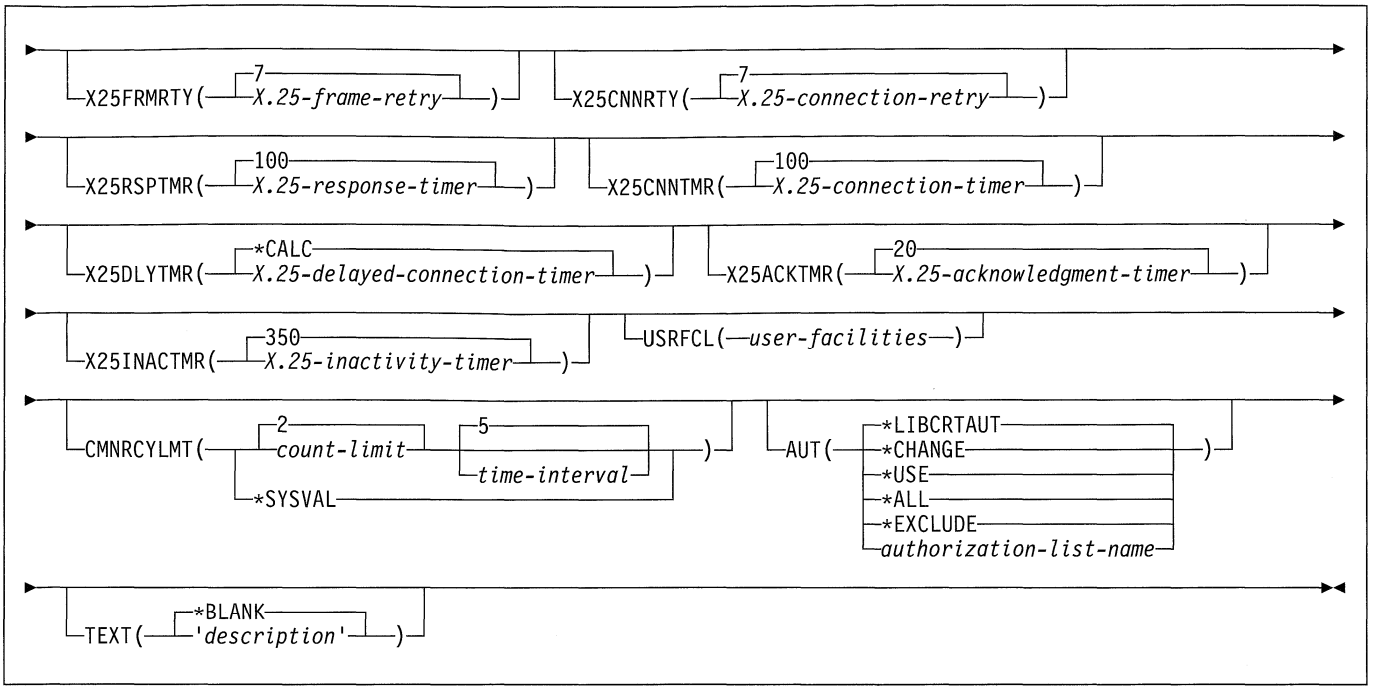
P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions

2 The number of reptitions are dependent on the controller type; see the table under the DEV parameter description.

3 If LINKTYPE(*LAN) is specified and TYPE(*FBSS) was specified on the CRTCTLFNC command, any value ranging from 265 through 521 can be specified on the MAXFRAME parameter.


STNADR(station-address)
POLLPTY(*NO *YES)
POLLLMT(0 poll-limit)
OUTLMT(*POLLLMT out-limit)
CNNPOLLRTY(*CALC *NOMAX connect-poll-retry)
NDMPOLLTMR(*CALC NDM-poll-timer)
ADPTADR(adapter-address)
DSAP(04 destination-service-access-point)
SSAP(04 source-service-access-point)
LANFRMRTY(*CALC LAN-frame-retry)
LANCNNRTY(*CALC LAN-connection-retry)
LANRSPTMR(*CALC LAN-response-timer)
LANCNNTMR(*CALC LAN-connection-timer)
LANACKTMR(*CALC LAN-acknowledgment-timer)
LANINACTMR(*CALC LAN-inactivity-timer)
LANACKFRQ(*CALC LAN-acknowledgment-frequency)
LANMAXOUT(*CALC LAN-maximum-outstanding-frames)
LANACCPTY(*CALC LAN-access-priority)
LANWDWSTP(*NONE LAN-number-of-frames)
NETLVL(1980 1984 1988)
LINKPCL(*QLLC *ELLC)
LGLCHLID(logical-channel-ID)
CNNPWD(X.25-connection-password)
SWTLINSLCT(*FIRST *CALC)
DFTPKTSIZE(*LIND transmit-packet-size *LIND *TRANSMIT receive-packet-size)
DFTWDWSIZE(*LIND transmit-window-size *LIND *TRANSMIT receive-window-size)
USRGRPID(X.25-closed-user-group-ID)
RVSCRG(*NONE *REQUEST *ACCEPT *BOTH)



Purpose

The Create Controller Description (Finance) (CRTCTLFNC) command creates a controller description for a finance controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

TYPE

Specifies the type of controller.

3694: A 3694 check processor is used.

4701: A 4701 finance controller is used.

4702: A 4702 finance controller is used.

4730: A 4730 personal banking machine is used.

4731: A 4731 personal banking machine is used.

4732: A 4732 personal banking machine is used.

4736: A 4736 personal banking machine is used.

***FBSS:** A Finance Branch System Services (FBSS) controller is used.

MODEL

Specifies the model number of the controller that is described. This number tells the system features of the controller. The only valid model number is 0.

LINKTYPE

Specifies the type of line to which this controller is attached.

***LAN:** This controller is attached to a local area network (LAN) line.

***SDLC:** This controller is attached to a synchronous data link control (SDLC) line.

***X25:** This controller is attached to an X.25 line.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

SWITCHED

Specifies whether this controller is attached to a switched line or, for an X.25 connection, to a switched virtual circuit (SVC).

***NO:** This controller is attached to a nonswitched line or, for an X.25 connection, to a permanent virtual circuit (PVC).

***YES:** This controller is attached to a switched line or, for an X.25 connection, to a switched virtual circuit (SVC).

SHM

Specifies whether this controller is used for an X.21 short-hold mode. If *YES is specified, LINKTYPE(*SDLC) and SWITCHED(*YES) must also be specified.

***NO:** This controller is not used for X.21 short-hold mode.

***YES:** This controller is used for X.21 short hold mode.

SNBU

Specifies, for controllers supporting the switched network backup (SNBU) feature, whether the SNBU feature is activated or deactivated. Both the local and remote modems must support the SNBU feature to perform a valid activation.

Note: This parameter is not used for X.25 lines.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote modem has the SNBU feature.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

Note: If this controller is attached by an X.25 permanent virtual circuit (PVC), specify the name of the line on which that PVC is configured.

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

DEV

Specifies the names of devices attached to this controller. The device descriptions must already exist. The maximum number of device descriptions that can be attached to this controller and active (varied on) is shown in the table below:

Table 25. Maximum Number of Device Descriptions that can be Attached and Active

Controller Type	Maximum Attached	Maximum Active
4701,4702	255	120
*FBSS	255	240
4730	3	3
4731,4732,4736	2	2

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

***LINKTYPE:** The following frame sizes (measured in bytes) are specified if *LINKTYPE is used:

```
*SDLC - 265      *LAN - 521      *X25 - 256
```

maximum-frame-size: Specify a maximum frame size for the controller. The frame size that can be used depends on the controller and type of line being used. Valid frame sizes are shown below.

Table 26. Maximum Frame Sizes (in bytes)

Controllers	Link Type	Frame Sizes
*FBSS	*LAN	265 - 521
4701, 4702, *FBSS	*SDLC	265, 521, 1033
4701, 4702, *FBSS	*X25	256, 265, 512, 521
3694, 4730, 4731, 4732, 4746	*SDLC	265

EXCHID

Specifies the exchange identifier of this controller. The controller sends (exchanges) its identifier to another location when a connection is established. The 8-digit hexadecimal identifier contains three digits for the block number and five digits for the identifier of the specific controller. This parameter is required for synchronous data link control (SDLC) switched or switched network backup (SNBU) connections.

SSCPID

Specifies the system service control point (SSCP) of the host system.

050000000000: The default value is 050000000000.

system-service-control-point-identifier: Specify the system service control point identifier (SSCPID) as a 12-digit hexadecimal value.

INLCNN

Specifies the method used to make the initial connection on a switched line between the system and the remote controller. For X.25 lines, this parameter specifies the type of switched virtual circuit (SVC) connection that is made over the X.25 line.

***DIAL:** The connection is made by a call initiated from the AS/400 system.

For X.25 connection, the line attached to the controller requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH(*SVCOUT or *SVCBOTH) for the connection to succeed.

***ANS:** The connection is made by the AS/400 system when it answers an incoming call from this controller. If a call is received from the remote controller and all necessary conditions are met, the incoming call is answered by the system.

For X.25 connections, the line to which the controller attaches requires switched virtual circuits (SVC) configured on the LGLCHLE parameter of type IN or BOTH(*SVCIN or *SVCBOTH) for the connection to succeed. The line can be changed using the Change Line Description (X.25) (CHGLINX25) command.

CNNNBR
Specifies the telephone number to dial to connect to this controller.

***DC:** For X.21 circuit switched connections, a direct call is used to connect to the controller.

***ANY:** Calls are accepted from any network address.

connection-number: Specify the connection number. This could be a telephone number, an X.25 network address, or an X.21 connection number, depending on the type of controller and the type of line to which it is attached.

ANSNBR
Specifies the X.25 network address from which to accept calls.

***CNNNBR:** Calls from the X.25 network address specified by the connection number (CNNNBR) parameter are accepted.

***ANY:** Calls are accepted from any X.25 network address.

PREDIALDLY
Specifies how long to wait (in 0.5 second intervals) before dialing.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY
Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

DIALRTY
Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of dial retries.

SWTDSC
Specifies whether the switched connection to this controller is made inoperative when the last device is varied off.

***YES:** The switched connection is disconnected when the last device is varied off.

***NO:** The switched connection is not dropped when the last device is varied off.

SHMDSCLMT
Specifies the number of nonproductive responses (RR or RNR) that are required from the remote station before the connection can be suspended for this X.21 short-hold mode (SHM) connection. This parameter is used only if SHM(*YES) is specified.

10: Ten nonproductive responses must be received before the connection can be suspended.

***NOMAX:** There is no disconnect limit.

SHM-disconnect-limit: Specify a value ranging from 1 through 254, indicating the number of nonproductive responses that must be received before the connection can be suspended.

SHMDSCTMR
Specifies the minimum length of time that the primary system maintains the connection to the remote system for this X.21 short-hold mode controller, in tenths of a second. This parameter is used only if SHM(*YES) is specified.

50: The primary system maintains the connection to the remote system for a minimum of five seconds.

SHM-disconnect-timer: Specify a value ranging from 2 through 3000, indicating the minimum length of time, in tenths of a second, that the primary system maintains the connection to the remote system.

STNADR
Specifies the station address used when communicating with the controller.

Note: This is the station address of the remote controller. Specify a 2-character hexadecimal value ranging from 01 through FE.

POLLPTY
Specifies whether this controller has priority when polled.

***NO:** This controller does not have priority when polled.

***YES:** This controller has priority when polled.

POLLLMT
Specifies the number of consecutive polls issued to the same controller when the poll results in receiving frames.

0: No subsequent polls are attempted.

poll-limit: Specify a value ranging from 0 through 4.

OUTLMT
Specifies the number of times SDLC allows the consecutive transmission of the maximum number of frames to a station before allowing transmission to another station.

***POLLLMT:** The value specified on the POLLLMT parameter is used.

out-limit: Specify a value ranging from 0 through 4 for the number of consecutive transmissions.

CNNPOLLRTY
Specifies the number of connect poll retries before reporting the error.

***CALC:** The number of retries is 7 if the controller is switched, and *NOMAX if the controller is nonswitched.

***NOMAX:** There is no disconnect limit.

connect-poll-retry: Specify a value ranging from 0 through 65534 to indicate the number of retries.

NDMPOLLTMR
Specifies the slow poll interval for this controller when it is in normal disconnect mode (NDM).

***CALC:** The system determines the value to use.

NDM-poll-timer: Specify a value ranging from 1 through 3000 in 0.1-second intervals, or specify 0 to indicate no timer.

ADPTADR
Specifies the 12-character adapter address of the remote controller. This is the address to which the system sends data when it communicates with the remote controller. This value can be obtained from the remote controller's configuration record. Valid values range from hex 000000000001 through hex FFFFFFFFFFFF.

DSAP
Specifies the logical address of the SNA destination service access point (DSAP) to which this system sends data when it communicates with the remote controller. This address allows the controller and the system to route the data that comes from this system. The value must match the value specified on the source service access point (SSAP) parameter in the remote controller's configuration record.

04: The destination service access point is 04.

destination-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of four (for example, 04, 08, 0C, 10) to represent the destination service access point.

SSAP
Specifies the source service access point. This is the logical address this system uses when it sends data to the remote controller. This address allows the remote controller and the system to route the data that comes from this system. It must match the value assigned to the destination service access point (DSAP) prompt in the remote controller's configuration record.

04: The default value of 04 is used.

source-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of four (for example, 04, 08, 0C, 10) to represent the source service access point.

LANFRMRTY
Specifies the number of times a frame is transmitted if there is no acknowledgment from the remote controller in the time period specified by the LANRSPTMR parameter. This value is only used after a successful connection has been made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-frame-retry: Specify a value ranging from 0 through 254 for the number of times a frame is transmitted before an acknowledgement is received.

LANCNNRTY
Specifies the number of times a transmission is attempted before an acknowledgement is received. This value is used at connection time (unlike LANFRMRTY which is used *after* a connection has been made).

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-retry: Specify a value ranging from 0 through 254 for the number of times the transmission is attempted before an acknowledgement is received.

LANRSPTMR
Specifies the length of time the system waits before an inoperative condition occurs on a link after a connection is made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-response-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANCNNTMR
Specifies the length of time the system waits before an inoperative condition occurs on a link at connection time.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANACKTMR
Specifies the length of time the system waits before sending an acknowledgement for received frames.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgment-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals, or specify 0 to indicate no delay. If 0 is specified for this parameter, 0 must also be specified for the LANACKFRQ parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKFRQ parameter.

LANINACTMR
Specifies the length of time used to determine an inactive condition for the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-inactivity-timer: Specify a value ranging from 1 through 255 in 0.1-second intervals for the length of time used to determine an inactive condition for the controller.

LANACKFRQ

Specifies the maximum number of frames that are received before an acknowledgment is sent to the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgment-frequency: Specify a value ranging from 0 through 127 for the number of frames received. If 0 is specified for this parameter, 0 must also be specified for the LANACKTMR parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKTMR parameter.

LANMAXOUT

Specifies the maximum number of frames that can be sent before an acknowledgment is received from the remote system.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-maximum-outstanding-frames: Specify a value ranging from 1 through 127 for the number of frames that can be sent before an acknowledgement is received.

LANACCPTY

Specifies the LAN priority granted to the sending system for sending frames. The larger the number, the higher the priority. This parameter is only used when the controller is attached to TRLAN.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-access-priority: Specify a value ranging from 0 through 3 to indicate how soon a frame can be sent on a local area network.

LANWDWSTP

Specifies whether to reduce the number of outstanding frames that can be sent before an acknowledgement is received from the remote system during network congestion. This parameter (LAN Window Step) also indicates the number of frames that must be successfully received before the number of outstanding frames can be incremented to the maximum allowable value.

***NONE:** The number of outstanding frames is not reduced during network congestion.

LAN-number-of-frames: Specify the number of outstanding frames to be reduced before an acknowledgement is received from the remote system.

NETLVL

Specifies the level of the X.25 network used to reach this controller. The level is specified by giving the year of the standard used by X.25 network.

1980: The 1980 standard is used.

1984: The 1984 standard is used.

1988: The 1988 standard is used.

LINKPCL

Specifies the link level protocol used on the X.25 network to communicate with this controller.

***QLLC:** The QLLC protocol is used.

***ELLC:** The ELLC protocol is used.

LGLCHLID

Specifies the logical channel ID used for the X.25 permanent virtual circuit to this controller. The valid entry is in the format xyy, where:

x = the logical group number, derived from the network subscription
yy = the logical channel number, derived from the subscription

CNNPWD

Specifies, for X.25 switched virtual circuit (SVC) controllers, the password used when connecting to this controller.

SWTLINSLCT

Specifies the method that is used to select lines from an X.25 switched line list.

***FIRST:** The lines are selected in the order in which they are specified.

***CALC:** The system determines the value to use.

DFTPKTSIZE

Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

***LIND:** The value specified in the line description is used as the controller.

transmit-packet-size: Specify a default packet size for transmission. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***LIND:** The value specified in the line description is used as the controller.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

DFTWDWSIZE

Specifies the default window size used by the X.25 network.

Element 1: Transmit Window Size

***LIND:** The value specified in the line description is used as the controller.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

Element 2: Receive Window Size

***LIND:** The value specified in the line description is used as the controller.

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

USRGRPID

Specifies the closed user-group ID for contacting this X.25 switched virtual circuit (SVC) controller. Specify a value ranging from 0 through 99, as provided by the network subscription. This parameter is not valid for permanent virtual circuit (PVC) connections. It is valid only for SVC circuit outgoing call operations and is ignored for SVC incoming call connections.

RVSCRG

Specifies whether reverse charges are accepted or requested when contacting this controller.

***NONE:** No reverse charging for network tariff billing is accepted.

***REQUEST:** Charges are requested on outgoing call request packets.

***ACCEPT:** Reverse charging for network tariff billing is accepted on incoming requests.

***BOTH:** Both incoming and outgoing requests are accepted.

X25FRMRTY

Specifies the maximum number of times a frame is sent after the response timer ends when connected to this controller.

7: The frame is sent seven times.

X.25-frame-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25CNNRTY

Specifies the maximum number of times a frame is sent after the connect response timer ends when connecting to this controller.

7: The frame is sent seven times.

X.25-connection-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25RSPTMR

Specifies the time allowed for an acknowledgment to return when a frame is sent while connected to this controller.

100: Ten seconds is allowed for acknowledgment.

X.25-response-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25CNNTMR

Specifies the time allowed for an acknowledgment to be returned when a frame is sent while connecting to this controller.

100: The time allowed for acknowledgment is 10 seconds.

X.25-connection-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25DLYTMR

Specifies the time interval to try establishing a connection to the controller.

***CALC:** Uses the values specified in the X25CNNTMR and X25CNNRTY parameters to determine how often and how many times to try establishing the connection.

X.25-delayed-connection-timer: Specify a value ranging from 1 through 32767 in 0.1-second intervals. The system retries the connection indefinitely at the specified interval.

X25ACKTMR

Specifies the time period to delay sending acknowledgments for received frames.

20: The time allowed for acknowledgment is two seconds.

X.25-acknowledgment-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals, or specify 0 to indicate no delay.

X25INACTMR

Specifies the time period used to determine an inactive condition for the controller.

350: An inactive condition is detected after 35-seconds.

X.25-inactivity-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

USRFCL

Specifies a string of hexadecimal digits sent to the X.25 network to request additional services. The AS/400 system allows up to 218 hexadecimal characters.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

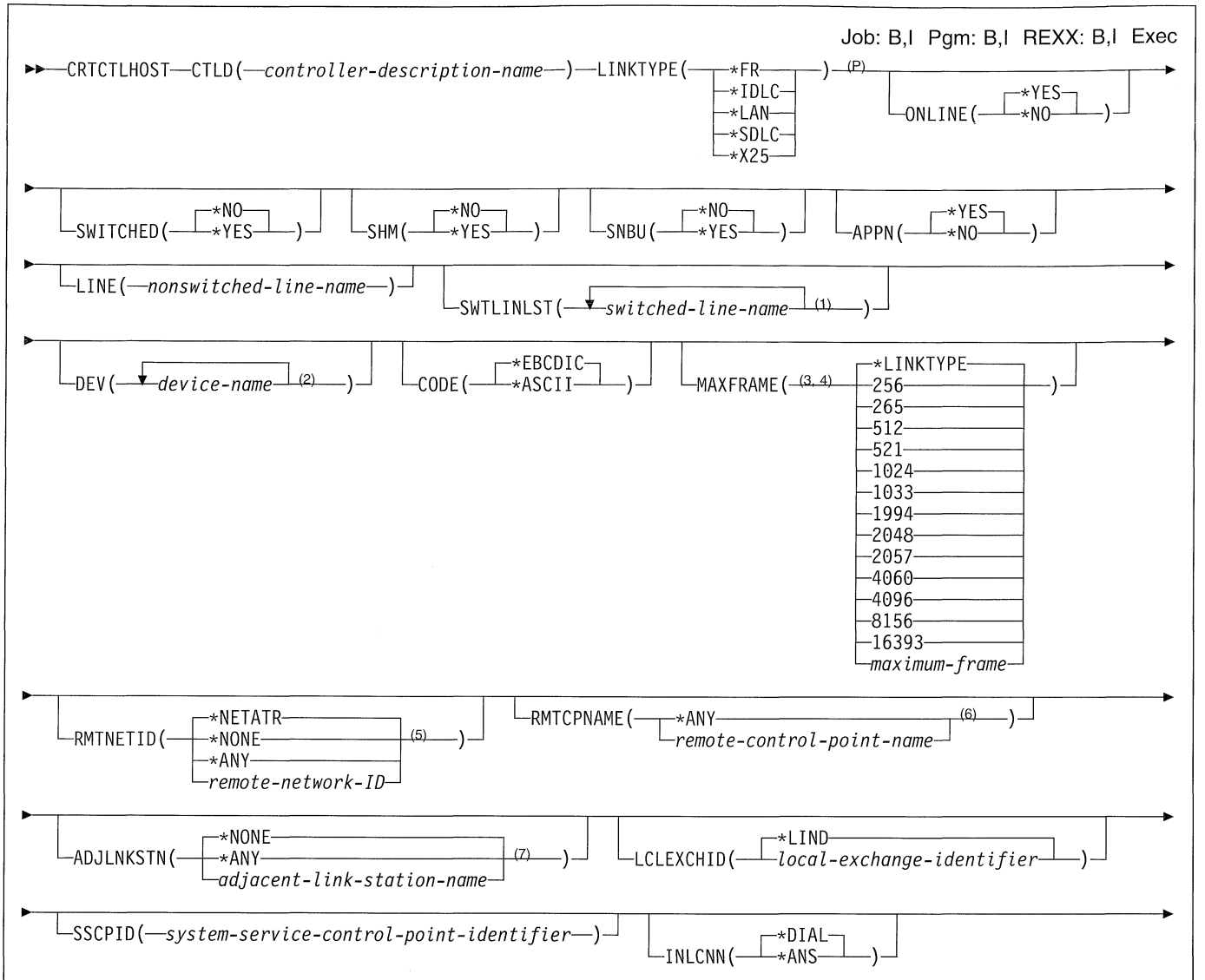
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLFNC   CTLD(FNC1)  TYPE(4701)
  MODEL(0)  LINKTYPE(*SDLC)
  LINE(FNCLINE)  STNADR(C1)
```

This command creates a 4701 finance controller attached to a nonswitched SDLC line with a station address of C1.

CRTCTLHOST (Create Controller Description (SNA Host)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions

2 A maximum of 254 repetitions

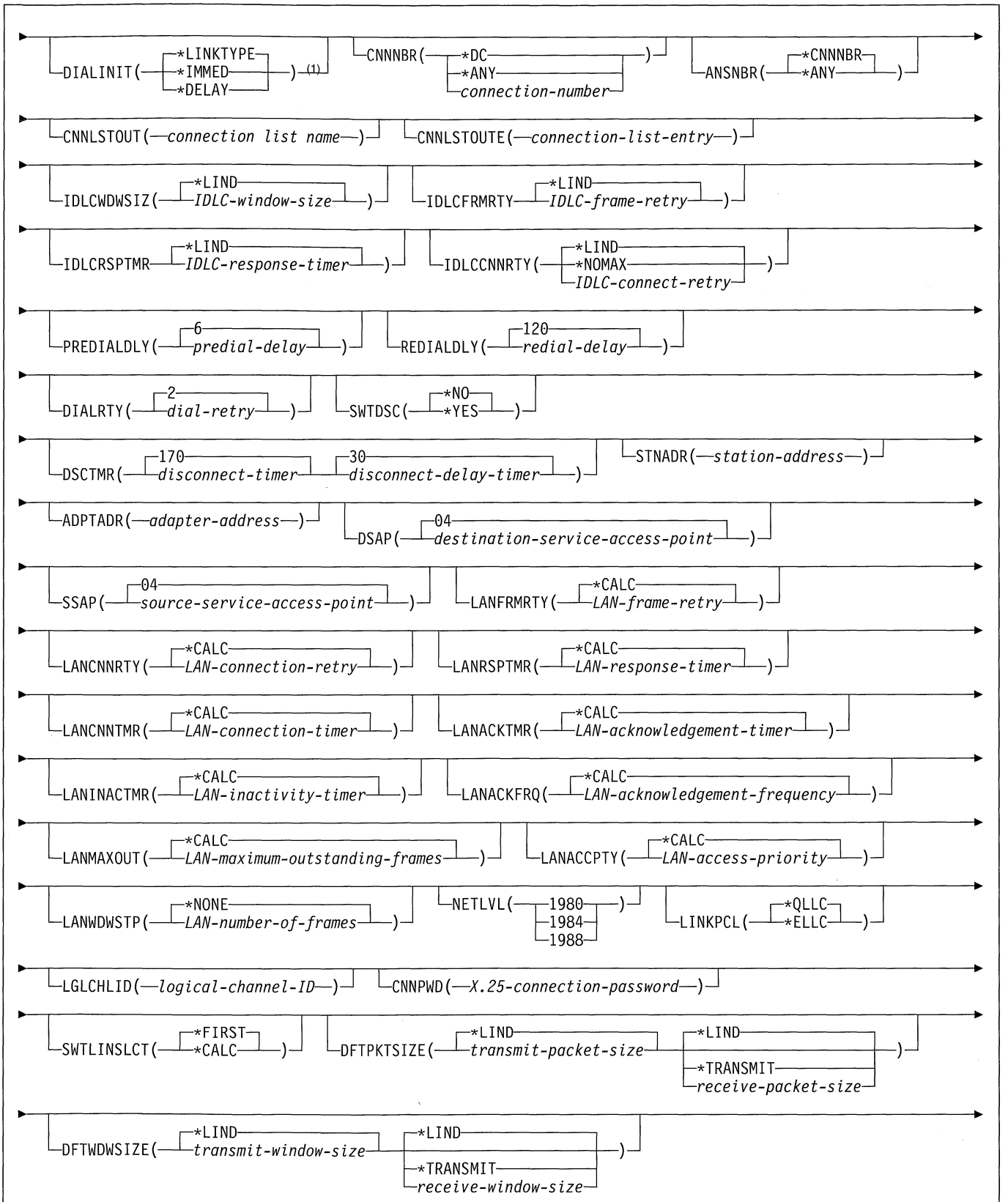
3 If LINKTYPE is *LAN, valid values range from 265 through 16393.

4 For DDI LANs, valid values range from 265 through 4444.

5 If RMTNETID(*ANY) is specified, RMTCPNAME(*ANY) must also be specified.

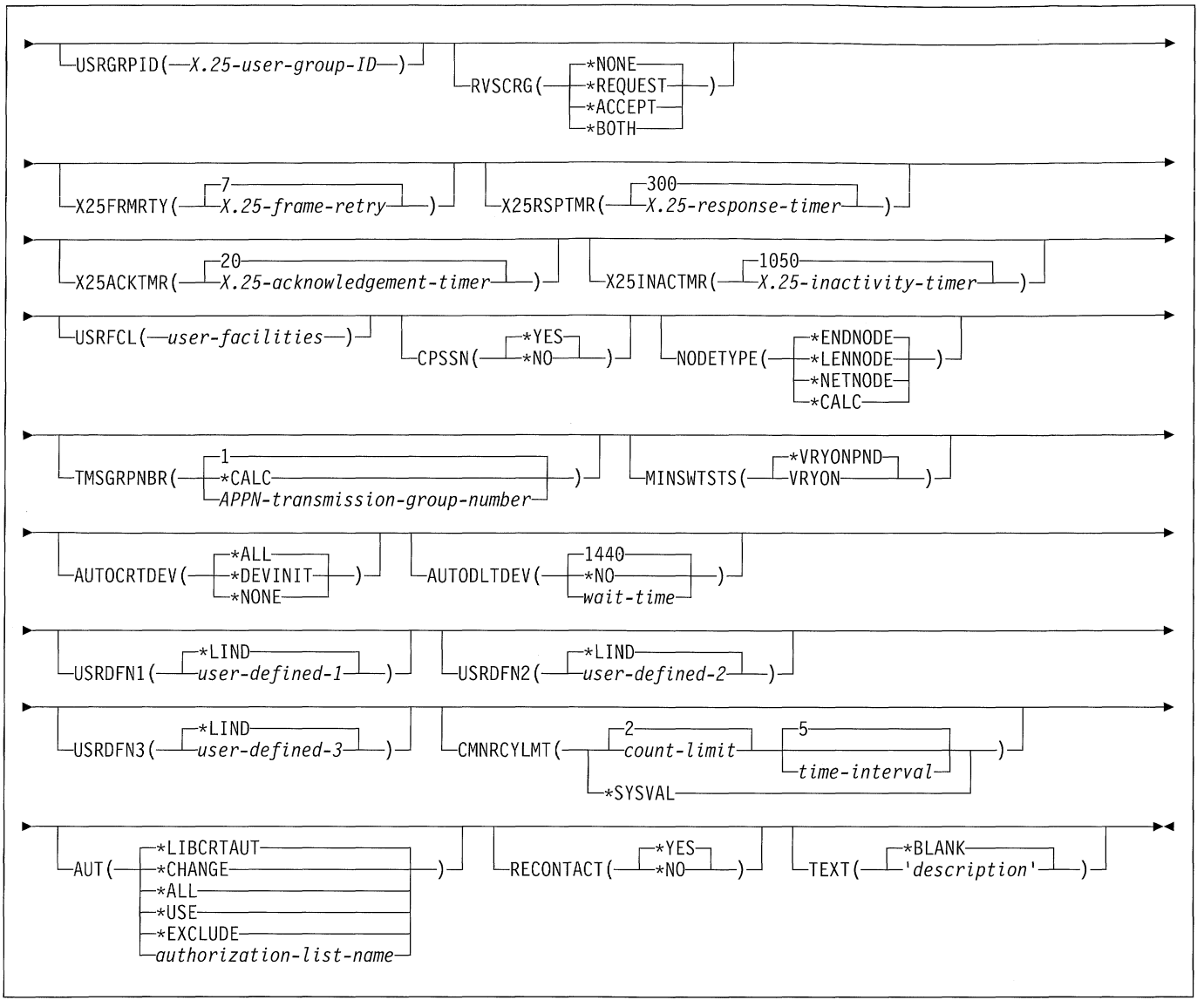
6 RMTCPNAME(*ANY) cannot be specified if LINKTYPE is *IDLC and SWITCHED or SNBU is *YES; if LINKTYPE is *SDLC and SWITCHED or SNBU is *YES; if NODETYPE is *LENNODE; or if MDLCTL is *YES.

7 ADJLNKSTN(*ANY) is only specified if LINKTYPE is *IDLC and SWITCHED is *YES, or LINKTYPE is *SDLC and SWITCHED is *YES. Also, ADJLNKSTN must be specified if RMTCPNAME is *ANY.



Note:

[1] This parameter is valid only if INLCNN(*DIAL) is specified.



Purpose

The Create Controller Description (SNA Host) (CRTCTLHOST) command creates a controller description for an SNA host system. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

LINKTYPE

Specifies the type of line to which this controller is attached.

***FR:** This controller is attached to a frame relay line.

***IDLC:** This controller is attached to an ISDN data link control (IDLC) line.

***LAN:** This controller is attached to a DDI, Ethernet, or token-ring local area network (LAN) line.

***SDLC:** This controller is attached to a synchronous data link control (SDLC) line.

***X25:** This controller is attached to an X.25 line.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

SWITCHED

Specifies whether this controller is attached to a switched line, an X.25 switched virtual circuit (SVC), or a

TRLAN line. *NO must be specified for APPC controllers attached to a TDLC line.

***NO:** This controller is attached to a nonswitched line or an X.25 permanent virtual circuit (PVC).

***YES:** This controller is attached to a switched line or an X.25 switched virtual circuit (SVC).

SHM

Specifies whether this controller is for X.21 short-hold mode. If *YES is specified on this parameter, LINKTYPE(*SDLC) and SWITCHED(*YES) must also be specified.

***NO:** This controller is not used for X.21 short-hold mode.

***YES:** This controller is used for X.21 short-hold mode.

SNBU

Specifies, for controllers supporting the switched network backup (SNBU) feature, whether the SNBU feature is activated or deactivated. Both the local and remote modems must support the SNBU feature to perform a valid activation.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote modem has the SNBU feature.

APPN

Specifies whether the local system uses advanced peer-to-peer networking (APPN) functions when communicating with this controller. If APPN functions are used, the user must specify a remote control point name and a remote network identifier.

***YES:** The local system uses APPN functions.

***NO:** The local system does not use APPN functions.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

DEV

Specifies the names of the devices attached to this controller. Use this parameter only when the associated device descriptions have already been created. The maximum number of devices that can attach to this controller is 254.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

***LINKTYPE:** The following values are used:

*SDLC-521	*LAN-16393	*FR-1590
*IDLC-2048	*X25-1024	

maximum-frame: Specify the maximum frame size for the controller. The frame size that can be used depends on the type of line being used. Valid frame sizes for each link type are shown below.

Link Type Frame Size (in bytes)

***FR**	265 - 8182
***LAN**	265 - 16393 (265 - 4444 for DDI LANs)
***IDLC**	265 - 8196
***SDLC**	265,521,1033,2057
***X25**	256,265,512,521,1024,1033,2048,4096

RMTNETID

Specifies the name of the remote network.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

***ANY:** The system determines which remote network identifier is used.

remote-network-ID: Specify the remote network identifier.

RMTCPNAME

Specifies the name of the remote control point.

***ANY:** The system determines the name of the remote control point to use.

remote-control-point-name: Specify the remote control point name.

ADJLNKSTN

Specifies the name of the adjacent link station. This name is used by the system to identify which switched controller description on the AS/400 system is used to establish a link to a host S/370 system. The adjacent link station name for the AS/400 system must match the name provided by the host system during link activation.

***NONE:** No adjacent link station name is specified.

***ANY:** The system determines which adjacent link station is used.

adjacent-link-station-name: Specify the adjacent link station name.

LCLEXCHID

Specifies the number used to identify the local system to the remote system.

Note: This parameter is necessary only if parallel connections are required between an AS/400 system and an SNA host system.

***LIND:** The AS/400 system uses the exchange identifier specified in the line description on the EXCHID parameter associated with this controller description.

local-exchange-identifier: Specify a local exchange identifier. The first three digits of this 8-digit hexadecimal number identify the block number and the remaining five digits identify the system.

SSCPID

Specifies the system service control point (SSCP) of the host system.

Note: The system service control point identifier (SSCPID) is a 12-digit hexadecimal value; the first 2 digits are hexadecimal 05. This parameter is required for SDLC switched and SNBU if APPN(*YES) and NODETYPE(*LENNODE) are specified, or if APPN(*NO) is specified but RMTCPNAME is not specified. This parameter must be specified if SHM(*YES) is also specified.

INLCNN

Specifies the method used to establish a connection with this controller.

***DIAL:** The connection is made by a call initiated from the AS/400 system.

For X.25 connection, the line attached to the controller requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH(*SVCOUT or *SVCBOTH) for the connection to succeed.

***ANS:** The connection is made by the local system answering an incoming call from this controller.

DIALINIT

Specifies the method used to make the initial dial on a switched line between the system and the remote controller.

***LINKTYPE:** The type of dial connection initiated is specified on the LINKTYPE parameter. For LAN or SDLC short-hold mode connections, the default is to dial the connection immediately upon vary on of the controller description. For all other link types, the default is to delay the dial.

***IMMED:** The dial connection is initiated immediately upon vary on of the controller description.

***DELAY:** The dial connection is delayed until a job is initiated that requests the use of the remote controller resources.

CNNNBR

Specifies the telephone number to dial to connect to this controller.

***DC:** For X.21 circuit switched connections, a direct call is used to connect to the controller.

***ANY:** Calls are accepted from any network address.

connection-number: Specify the connection number used to call this controller. This could be a telephone number, an X.25 network address, or an X.21 connection number, depending on the type of controller and the type of line to which it is attached.

ANSNBR

Specifies the X.25 network address from which to accept calls.

***CNNNBR:** Calls from the X.25 network address specified by the connection number (CNNNBR) parameter are accepted.

***ANY:** Calls are accepted from any X.25 network address.

CNNLSTOUT

Specifies, for ISDN switched connections, the name of a connection list object that contains the ISDN assigned numbers for a dial out operation to the ISDN.

CNNLSTOUTE

Specifies the entry name from the connection list that is used to make a call to the ISDN. The connection list must be specified on the CNNLSTOUT parameter.

IDLCWDWSIZ

Specifies the window size for transmission to and reception from the remote controllers.

***LIND:** The window size specified in the line description is used.

window-size: Specify the window size. Valid values range from 1 through 31.

IDLCFRMRTY

Specifies the maximum number of attempts to transmit before an error is reported.

***LIND:** The number of attempts specified in the line description is used.

IDLC-frame-retry: Specify the number of attempts. Valid values range from 0 through 100.

IDLCRSPTMR

Specifies the amount of time, in tenths of a second, to wait before retransmitting a frame if acknowledgement has not been received.

***LIND:** The time specified in the line description is used.

IDLC-response-timer: Specify an amount of time to wait before retransmitting an unacknowledged frame of data. Valid values range from 10 through 100 tenths of a second. For example, 10 seconds equals 100 tenths of a second.

IDLCCNNRTY
Specifies the number of times to attempt retransmission at connection time.

***LIND:** The number of attempts specified in the line description is used.

***NOMAX:** There is no disconnect limit.

IDLC-connect-retry: Specify the number of attempts. Valid values range from 1 through 100.

PREDIALDLY
Specifies how long to wait (in 0.5 second intervals) before dialing.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY
Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

DIALRTY
Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of dial attempts.

SWTDSC
Specifies whether the switched connection to this controller is disconnected when the last device in the host controller description becomes inactive.

***NO:** Specify *NO to keep this connection active even if the above conditions were met, or if the host system is configured to bring down the switched connection.

***YES:** The switched connection is disconnected when the last device is varied off.

DSCTMR
Specifies options for controlling the time (in seconds) before an inactive connection is dropped (Element 1), or the amount of time to delay the automatic disconnection (Element 2).

Note: The DISCNT parameter of the GROUP macro instruction in the NCP/VTAM definition can be used if the host system is to determine the disconnect characteristics.

Element 1: Minimum Connect Timer

170: A connection is dropped when it has been inactive for 170 seconds.

disconnect-timer: Specify a time to wait before disconnecting. Valid values range from 0 through 65535 seconds.

Element 2: Disconnect Delay Timer

30: The disconnection is delayed for 30 seconds.

disconnect-delay-timer: Specify a value to delay link take down after the last session on the controller is terminated. Valid values range from 0 through 65535 seconds.

Note: For objects migrated from V2R1M1 or previous systems, the disconnect delay timer has a value of zero. This value is interpreted by the system as 30 seconds.

STNADR
Specifies the station address used when communicating with the controller.

Note: This is the station address of the local system. Valid values range from 01 through FE.

ADPTADR
Specifies the 12-character adapter address of the remote controller. This is the address to which the system sends data when it communicates with the remote controller. This value can be obtained from the remote controller's configuration record. Valid values range from hex 000000000001 through hex FFFFFFFFFFFF.

DSAP
Specifies the logical address of the SNA destination service access point (DSAP) to which this system sends data when it communicates with the remote controller. This address allows the controller and the system to route the data that comes from this system. The value must match the value specified on the source service access point (SSAP) parameter in the remote controller's configuration record.

04: The destination service access point is 04.

destination-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the destination service access point.

SSAP
Specifies the source service access point. This is the logical address this system uses when it sends data to the remote controller. This address allows the remote controller and the system to route the data that comes from this system. It must match the value assigned to the destination service access point (DSAP) prompt in the remote controller's configuration record.

04: The source service access point is 04.

source-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the source service access point.

LANFRMRTY

Specifies the number of times a frame is transmitted if there is no acknowledgement from the remote controller in the time period specified on the LANRSPTMR parameter. This value is used only after a successful connection has been made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-frame-retry: Specify a value ranging from 0 through 254 for the number of times a frame is transmitted before an acknowledgement is received.

LANCNNRTY

Specifies the number of times a transmission is attempted before an acknowledgement is received. This value is used at connection time (unlike LANFRMRTY which is used after a connection has been made).

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-retry: Specify a value ranging from 0 through 254 for the number of times the transmission is attempted before an acknowledgement is received.

LANRSPTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link after a connection is made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-response-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANCNNTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link at connection time.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANACKTMR

Specifies the length of time the system waits before sending an acknowledgement for received frames.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgement-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals, or specify 0 to indicate no delay. If 0 is specified for this parameter, 0 must also be specified for the LANACKFRQ parameter; if a non zero value is specified for this parameter, a non zero value must also be specified for the LANACKFRQ parameter.

LANINACTMR

Specifies the length of time in which to determine an inactive condition for the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-inactivity-timer: Specify a value ranging from 1 through 255 in 0.1-second intervals for the length of time used to determine an inactive condition for the controller.

LANACKFRQ

Specifies the maximum number of frames that are received before an acknowledgement is sent to the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgement-frequency: Specify a value ranging from 0 through 127 for the number of frames received. If 0 is specified for this parameter, 0 must also be specified for the LANACKTMR parameter; if a non zero value is specified for this parameter, a non zero value must also be specified for the LANACKTMR parameter.

LANMAXOUT

Specifies the maximum number of frames that can be sent before an acknowledgement is received from the remote system.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-maximum-outstanding-frames: Specify a value ranging from 1 through 127 for the number of frames that can be sent before an acknowledgement is received.

LANACCPTY

Specifies the priority granted to the sending system for sending frames. The larger the number, the higher the priority.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-access-priority: Specify a value ranging from 0 through 3 to indicate how soon a frame can be sent on a local area network (LAN).

LANWDWSTP

Specifies whether to reduce the number of outstanding frames that can be sent before an acknowledgement is received from the remote system during network congestion. This parameter (LAN Window Step) also indicates the number of frames that must be successfully received before the number of outstanding frames can be incremented to the maximum allowable value.

***NONE:** The number of outstanding frames is not reduced during network congestion.

LAN-number-of-frames: Specify the number of outstanding frames to be reduced before an acknowledgement is received from the remote system.

NETLVL

Specifies the level of the X.25 network used to reach this controller. The level is specified by giving the year of the standard used by the X.25 network.

1980: The 1980 standard is used.

1984: The 1984 standard is used.

1988: The 1988 standard is used.

LINKPCL

Specifies the link level protocol used on the X.25 network to communicate with this controller.

***QLLC:** The QLLC protocol is used.

***ELLC:** The ELLC protocol is used.

LGLCHLID

Specifies the logical channel ID used for the X.25 PVC to this controller. Specify the logical channel ID. The valid entry is xyy. Where:

```
x =  the logical group number derived
       from user network subscription
yy = the logical channel number,
       derived from user subscription
```

CNNPWD

Specifies, for X.25 switched virtual circuit (SVC) controllers, the password used when connecting to this controller.

SWTLINSLCT

Specifies the method that is used to select lines from an X.25 switched line list.

***FIRST:** The lines are selected in the order in which they are specified.

***CALC:** The system determines the value to use.

DFTPKTSIZE

Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

***LIND:** The value specified in the line description is used as the default packet size.

transmit-packet-size: Specify a default packet size for transmission. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***LIND:** The value specified in the line description is used as the default packet size.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

DFTWDWSIZE

Specifies the default window size used by the X.25 network.

Element 1: Transmit Window Size

***LIND:** The value specified in the line description is used as the default window size.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

Element 2: Receive Window Size

***LIND:** The value specified in the line description is used as the default window size.

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

USRGRPID

Specifies the closed user-group ID for contacting this X.25 switched virtual circuit (SVC) controller. Specify a value ranging from 0 through 99, as provided by the network subscription. This parameter is not valid for permanent virtual circuit (PVC) connections. It is valid only for SVC circuit outgoing call operations and is ignored for SVC incoming call connections.

RVSCRG

Specifies whether reverse charges are accepted or requested when contacting this controller.

***NONE:** No reverse charging for network tariff billing is accepted.

***REQUEST:** Charges are requested on outgoing call request packets.

***ACCEPT:** Reverse charging for network tariff billing is accepted on incoming requests.

***BOTH:** Both incoming and outgoing requests are accepted.

X25FRMRTY

Specifies the maximum number of times that a link protocol data unit (LPDU) is sent after the response timer ends when connected to this controller.

7: The frame is sent seven times.

X.25-frame-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25RSPTMR

Specifies the time allowed to return an acknowledgement when a link protocol data unit (LPDU) is sent while connected to this controller.

300: The time allowed for acknowledgement is 30 seconds.

X.25-response-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25ACKTMR
Specifies the time period to delay sending acknowledgements for received LPDU.

20: Two seconds are allowed for acknowledgement.

X.25-acknowledgement-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals, or specify 0 to indicate no delay.

X25INACTMR
Specifies the time period in which to determine an inactive condition for the controller.

1050: An inactive condition is detected after 105 seconds.

X.25-inactivity-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

USRFCL
Specifies a string of hexadecimal digits sent to the X.25 network to request additional services. The AS/400 system allows up to 218 hexadecimal characters.

CPSSN
Specifies whether this controller supports control-point to control-point (CP to CP) sessions.

***YES:** This controller supports CP to CP sessions.

***NO:** This controller does not support CP to CP sessions.

NODETYPE
Specifies the type of APPN node this controller represents.

***ENDNODE:** This node is an end node in an APPN network.

***LENNODE:** This node is a low-entry networking node in an APPN network.

***NETNODE:** This node is a network node in an APPN network.

***CALC:** The system determines the value to use.

TMSGRPNBR
Specifies the APPN transmission group number for this controller.

1: The transmission group number is one.

***CALC:** The system determines the value to use.

APPN-transmission-group-number: Specify a value ranging from 1 through 20 for the transmission group number.

MINSWTSTS
Specifies, for the switched connection, the minimum status that APPN requires for a controller to be considered available for routing.

***VRYONPND:** APPN considers the controller available for routing if the status is vary on pending, varied on, or active.

***VRYON:** Indicates that APPN considers the controller available for routing only if the status is varied on or active.

AUTOCRTDEV
Specifies which devices are automatically created.

***ALL:** All devices that can be automatically created for this controller, except APPC devices are automatically created.

***DEVINIT:** Only session printer and display devices started by the SNA host controller (device-initiated) are automatically created.

***NONE:** No devices are automatically created.

AUTODLTDEV
Specifies the number of minutes an automatically created device can remain when the last session is unbound before the device description is varied off and deleted.

1440: The system will automatically vary off and delete the automatically-configured idle device descriptions after 1440 minutes (24 hours).

***NO:** The system will not automatically vary off and delete the automatically-configured idle device descriptions.

wait-time: Specify the number of minutes to wait before deleting the automatically-configured idle device descriptions for this controller. Valid values range from 1 to 10000.

USRDFN1
Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

***LIND:** The user-defined value specified in the line description is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2
Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

***LIND:** The user-defined value specified in the line description is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3
Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

***LIND:** The user-defined value specified in the line description is used.

user-defined-3: Specify a value ranging from 0 through 255.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

RECONTACT

Specifies whether to send a re-contact request to a remote system when the system is varied off.

Note: This parameter is valid only for X.25 and SDLC leased lines (if *X25 or *SDLC is specified on the LINKTYPE parameter and *NO is specified on the SWITCHED parameter).

***YES:** A request for re-contact with the remote system is sent. When this value is specified, a status of pending is shown for the remote system.

***NO:** A request for re-contact with the remote system is not sent. When this value is specified, a status of inactive is shown for the remote system.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

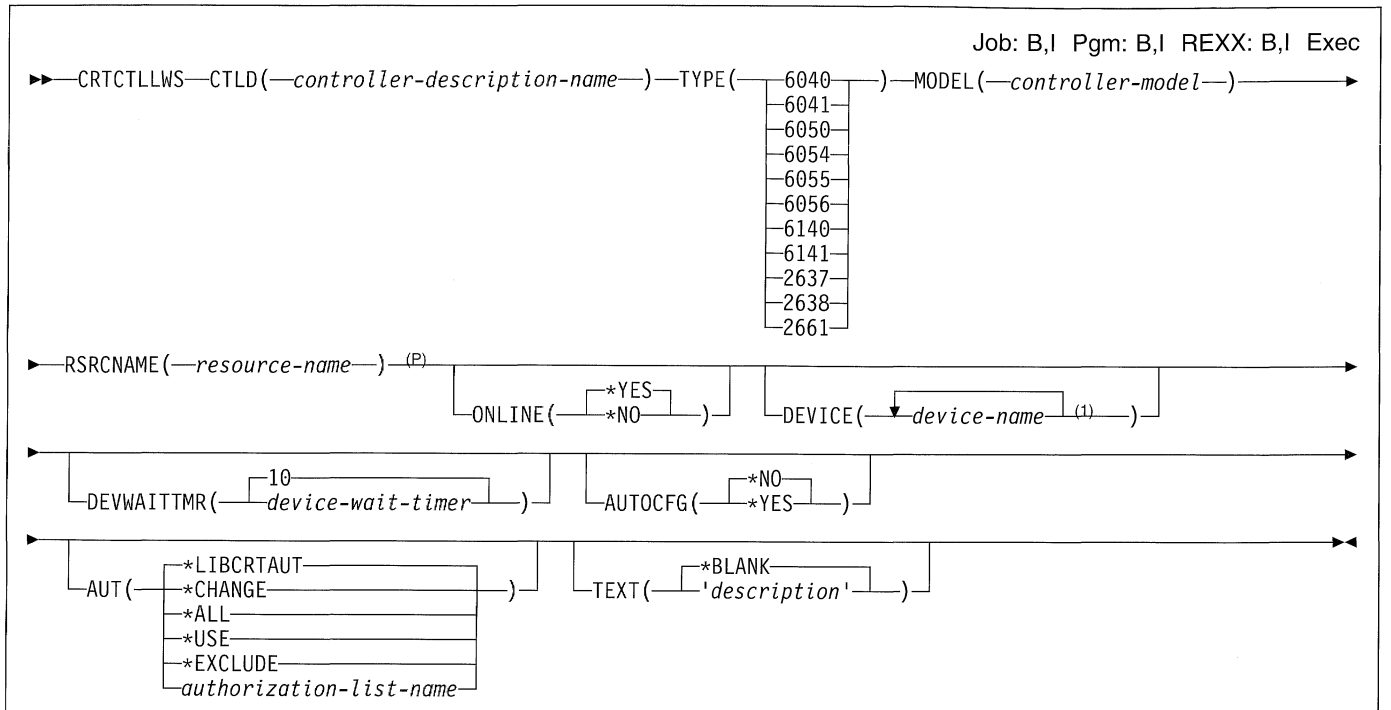
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLHOST   CTLD(HOST1)  LINKTYPE(*LAN)
  SWTLINLST(LAN1)  RMTCPNAME(CPX)
  ADPTADR(056000000011)
```

This command creates a host controller on a local area network.

CRTCTLLWS (Create Controller Description (Local Work Station)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 56 repetitions

Purpose

The Create Controller Description (Local Work Station) (CRTCTLLWS) command creates a controller description for a local work station controller. More information about using this command is in the *Device Configuration Guide*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

TYPE

Specifies the type of work station controller for this description.

6040: The local work station controller for the 9406 system is used.

6041: The ASCII work station controller for the 9406 system is used.

6050: The local work station controller for the 9402, 9404, or 9406 system is used.

6054: The local work station controller for the 9402, 9404, or 9406 system is used.

6055: The local work station controller for the 9402, 9404, or 9406 system is used.

6056: The local work station controller for the 9402, 9404, or 9406 system is used.

6140: The local work station controller for the 9404 or 9406 system is used.

6141: The ASCII work station controller for the 9404 or 9406 system is used.

2637: The ASCII work station controller for the 9402 system is used.

2638: The local work station controller for the 9402 system is used.

2661: The local work station controller for the 9402 system is used.

MODEL

Specifies the model of the work station controller. The model number for the work station controller is one.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *LWS specified for the TYPE parameter to determine the resource name.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

DEVICE

Specifies the device description names of devices attached to this controller. The device descriptions must already exist. The maximum number of device descriptions that can be attached to each controller and the number of devices that can be active for each controller is shown below:

Controller	Maximum Number of Device Descriptions	Maximum Number of Active Devices
6040,6140	56	40
6041,6141	36	18
6050,6054, 6055,6056	56	56
2637	24	-
2638	56	-

DEVWAITTMR

Specifies the device wait timeout value. This is used to limit the amount of time that a subsystem takes for the work station input/output to complete. The timeout value that is used for each device is taken from the controller that it is attached to at vary on time.

A change in the DEVWAITTMR parameter value takes effect for attached devices when they are next varied on.

10: For local controllers, the subsystem waits a maximum of 10 seconds for the completion of the input/output of each work station attached to this controller.

device-wait-timer: Specify a value ranging from 2 through 600 that specifies the maximum number of seconds that the subsystem waits for work station input/output to complete for all work stations attached to this controller.

When selecting a value for this parameter, the types of devices attached to the controller should be taken into account. Locally attached work stations should have a low value for this parameter (10 seconds or less).

AUTOCFG

Specifies whether this controller description is the one to which automatically configured devices are attached when they are configured. Although there can be more than one controller description for each controller, only one description can be an automatic configuration controller. When new devices are automatically configured on that controller, they are attached to the automatic configuration controller description.

***NO:** This is not an automatic configuration controller description.

***YES:** This is an automatic configuration controller.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

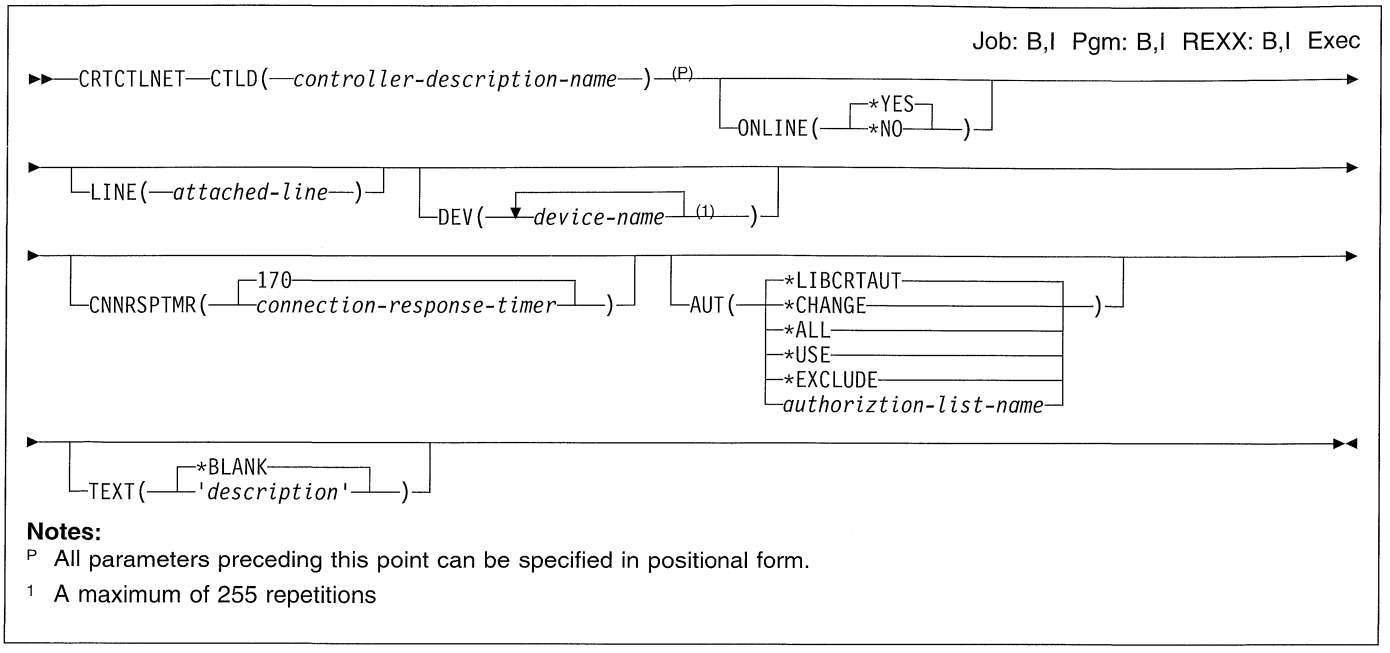
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLLWS   CTLD(CTL002)  TYPE(6040)  MODEL(1)
  RSRCNAME(CTL02)  ONLINE(*YES)
```

This command creates a 6040 Model 1 local work station controller description with the name CTL002. This controller is varied on at IPL time.

CRTCTLNET (Create Controller Description (Network)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 255 repetitions

Purpose

The Create Controller Description (Network) (CRTCTLNET) command creates a controller description for a network controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameter

CTLD

Specifies the name of the controller description being created.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

DEV

Specifies the names of up to 255 devices that are attached to this controller. The device descriptions must already exist.

CNNRSPTMR

Specifies the amount of time the system will wait before responding to an incoming connection request.

170: The system will wait 170 seconds before responding to an incoming connection request.

connection-response-timer: Specify the amount of time the system will wait before responding to a connection request. Valid values range from 1 to 3600 seconds.

AUT

Specifies the authority given to users who do not have specific authority to the controller description network, who are not on an authorization list, and whose user group has no specific authority to the controller description network.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description network.

***USE:** The user can perform basic operations on the controller description network, such as running a

program or reading a file. The user cannot change the controller description network. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description network.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

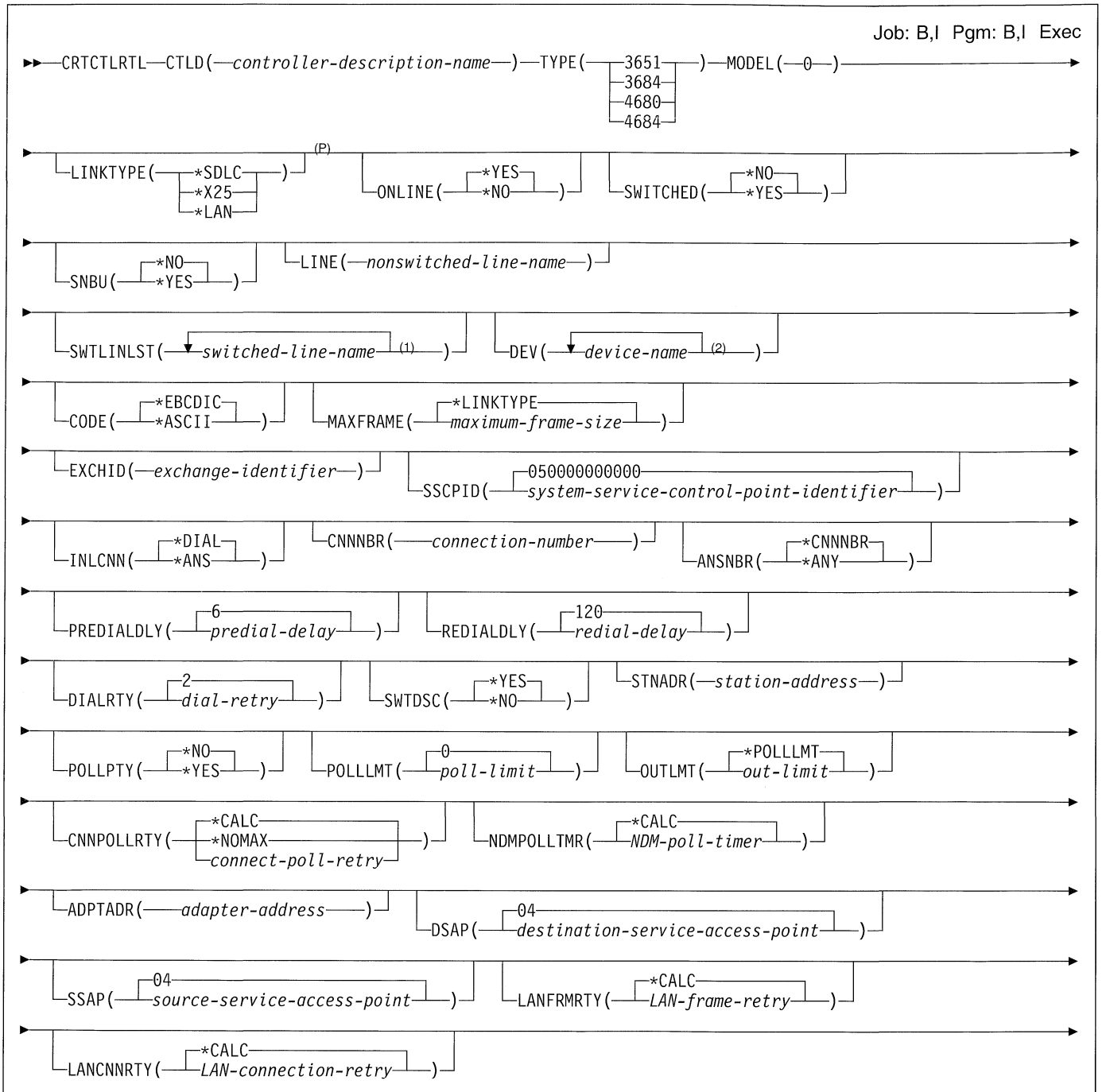
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLNET   CTLD(CTL0A)  LINE(LIN01)
```

This command creates a network controller description with the name CTL0A. An existing line, LIN01, is specified as the network line for the controller.

CRTCTLRTL (Create Controller Description (Retail)) Command

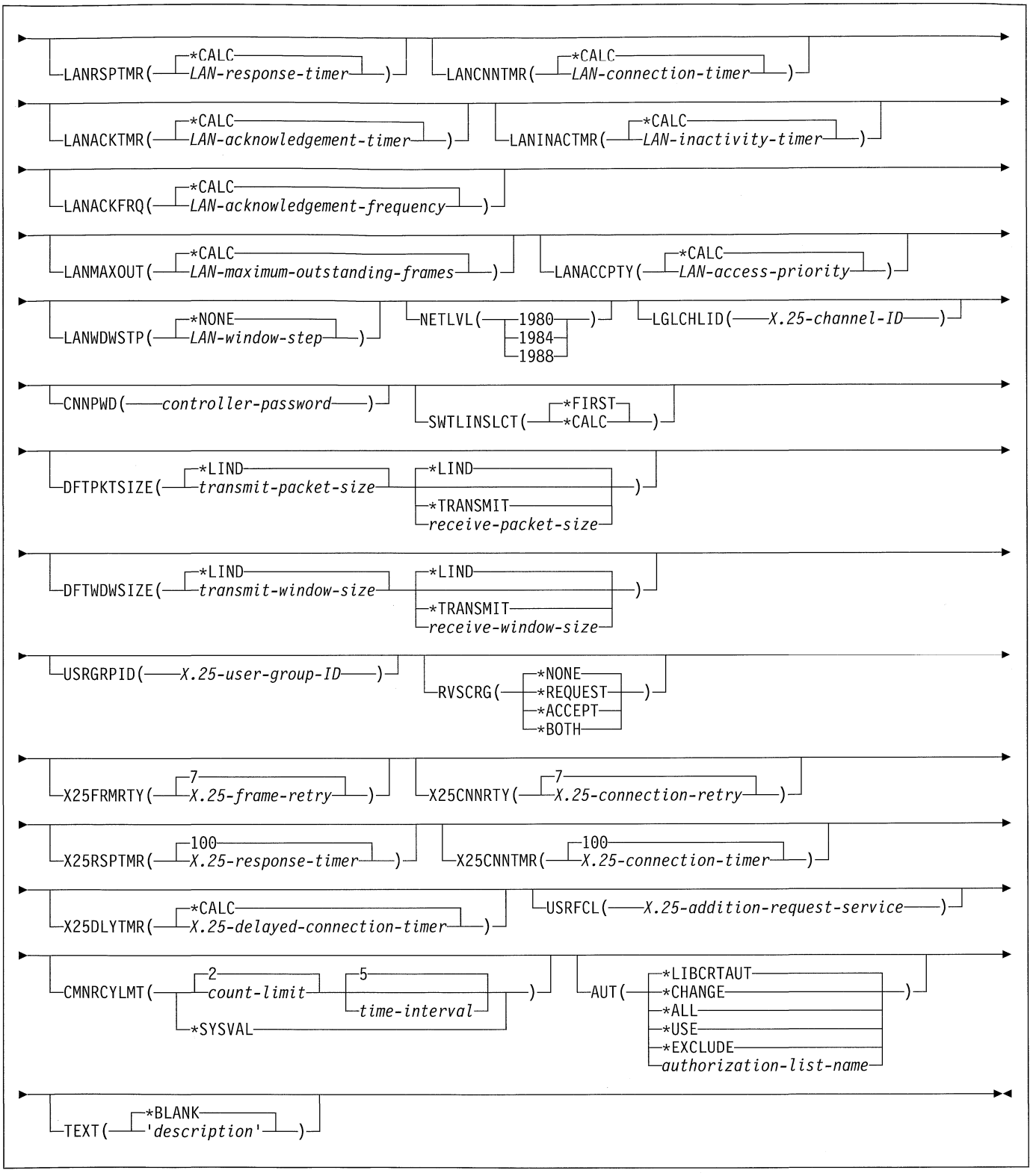


Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions

2 Number of repetitions are dependent on Controller Type; see the table under the DEV parameter description.



Purpose

The Create Controller Description (Retail) (CRTCTLRTL) command creates a controller description for a retail controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

TYPE

Specifies the type of controller.

3651: This description represents a 3651 controller.

3684: This description represents a 3684 controller.

4680: This description represents a 4680 controller.

4684: This description represents a 4684 controller.

MODEL

Specifies the model number of the controller represented by this description. Specify 0 for this parameter.

LINKTYPE

Specifies the type of line to which this controller is attached.

***SDLC:** This controller is attached to a synchronous data link control (SDLC) line.

***X25:** This controller is attached to an X.25 line.

***LAN:** This controller is attached to an Ethernet or token-ring local area network line.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

SWITCHED

Specifies whether this controller is attached to a switched line.

***NO:** This controller is attached to a nonswitched line.

***YES:** This controller is attached to a switched line.

SNBU

Specifies, for controllers attached to nonswitched lines only, whether the switched network backup (SNBU) feature is activated or deactivated. This feature lets the user bypass a broken nonswitched connection by establishing a switched connection. This parameter applies only if SWITCHED(*NO) and SNBU(*YES) are specified when the controller description is created.

Note: The Change Controller Description (Retail) (CHGCTLRTL) command must be used to activate the feature. This parameter is valid only when SWITCHED(*NO) is specified.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote modem has the switched backup feature. To activate the feature when the nonswitched connection is broken, specify ACTSNBU(*YES) on the CHGCTLRTL command.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

Note: The same line name can be used more than once. This parameter is valid only if SWITCHED(*YES) or SNBU(*YES) is specified.

DEV

Specifies the names the devices attached to this controller. The device descriptions must already exist. The maximum number of devices that can be attached to this controller and active (varied on) is shown in the table below:

Table 27. Maximum Number of Device Descriptions that Can be Attached and Active

Controller Type	Maximum Attached	Maximum Active
3651	14	14
3684	2	2
4680	84	40
4684	254	16

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

***LINKTYPE:** The frame size is 521 bytes for *SDLC, 1024 bytes for *X25, and 1994 bytes for *LAN.

maximum-frame-size: Specify 521, 1033, or 1994 bytes as the maximum frame size for this controller. Specify 1024 only if linktype is *X25. Specify 1994 only if linktype is *LAN.

EXCHID

Specifies the exchange identifier of this controller. The controller sends (exchanges) its identifier to another location when a connection is established. The 8-digit

hexadecimal identifier contains three digits for the block number and five digits for the identifier of the specific controller. This parameter is required for synchronous data link control (SDLC) switched or switched network backup (SNBU) connections.

Note: Identifiers must be specified for controllers with LINKTYPE(*SDLC) and either SWITCHED (*YES) or SNBU(*YES) specified. In all other cases the identifier is optional and is enforced only if specified.

SSCPID
Specifies the system service control point (SSCP) of the host system.

050000000000: The default value is 050000000000.

system-service-control-point-identifier: Specify the system service control point identifier (SSCPID) as a 12-digit hexadecimal value.

INLCNN
Specifies the method used to establish a connection with this controller.

***DIAL:** The connection is made by a call started from the local system.

For X.25, the line to which the controller attaches requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH (*SVCOUT or *SVCBOTH) for the connection to be successful.

***ANS:** The connection is made by the local system answering an incoming call from this controller.

CNNNBR
Specifies the telephone number to dial to connect to this controller.

connection-number: Specify a telephone number or a connection number depending on the type of controller and the type of line to which it is attached.

ANSNBR
Specifies the X.25 network address from which to accept calls.

***CNNNBR:** Calls from the X.25 network address specified by the connection number (CNNNBR) parameter are accepted.

***ANY:** Calls are accepted from any X.25 network address.

PREDIALDLY
Specifies how long to wait (in 0.5 second intervals) before dialing.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or 0 to indicate no delay.

REDIALDLY
Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or 0 to indicate no delay.

DIALRTY
Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

2: The default number of retries is two.

dial-retry: Specify a number ranging from 0 through 254 for the number of times to retry dialing.

SWTDSC
Specifies whether the switched connection to this controller is made inoperative when the last device is varied off.

***YES:** The switched connection is disconnected when the last device is varied off.

***NO:** The switched connection is not dropped when the last device is varied off.

STNADR
Specifies the station address used when communicating with the controller.

Note: This is the station address of the controller. Valid values range from 01 through FE. This is a required field for SDLC.

POLLPTY
Specifies whether this controller has priority when being polled.

***NO:** This controller does not have polling priority.

***YES:** This controller has polling priority.

POLLLMT
Specifies the number of consecutive polls that are issued to the same controller when the poll results in receiving frames.

0: No subsequent polls are attempted.

poll-limit: Specify a value ranging from 1 through 4 for the number of consecutive polls.

OUTLMT
Specifies the number of times SDLC allows the consecutive transmission of the maximum number of frames to a station before allowing transmission to another station.

***POLLLMT:** The value specified on the POLLLMT parameter is used.

out-limit: Specify a value ranging from 0 through 4 for the number of consecutive transmissions.

CNNPOLLRTY
Specifies the number of times to retry connecting to a controller before reporting an error.

***CALC:** The number of retries is 7 if the controller is switched, and *NOMAX if the controller is nonswitched.

***NOMAX:** There is no disconnect limit.

connect-poll-retry: Specify a value ranging from 0 through 65534 for the number of retries.

NDMPOLLTMR

Specifies the interval for polling this controller when it is in normal disconnect mode (NDM).

***CALC:** The system determines the value to use.

NDM-poll-timer: Specify a value ranging from 1 through 3000 in 0.1-second intervals, or 0 to indicate no timer.

ADPTADR

Specifies the 12-character adapter address of the remote controller. This is the address to which the system sends data when it communicates with the remote controller. This value can be obtained from the remote controller's configuration record. Valid values range from hex 000000000001 through hex FFFFFFFFFFFF.

DSAP

Specifies the logical address of the SNA destination service access point (DSAP) to which this system sends data when it communicates with the remote controller. This address allows the controller and the system to route the data that comes from this system. The value must match the value specified on the source service access point (SSAP) parameter in the remote controller's configuration record.

04: The destination SNA destination service access point is 04.

destination-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the SNA destination service access point.

SSAP

Specifies the source service access point. This is the logical address this system uses when it sends data to the remote controller. This address allows the remote controller and the system to route the data that comes from this system. It must match the value assigned to the destination service access point (DSAP) prompt in the remote controller's configuration record.

04: The SNA source service access point is 04.

source-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the SNA source service access point.

LANFRMRTY

Specifies the number of times a frame is transmitted if there is no acknowledgement * * from the remote controller in the time period specified on the LANRSPTMR parameter. This value is used only after a successful connection has been made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-frame-retry: Specify a value ranging from 0 through 254 for the number of times a frame is transmitted before an acknowledgement is received.

LANCNNRTY

Specifies the number of times a transmission is attempted before an acknowledgement is received. This value is used at connection time (unlike LANFRMRTY which is used after a connection has been made).

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-retry: Specify a value ranging from 0 through 254 for the number of times the transmission is attempted before an acknowledgement is received.

LANRSPTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link after a connection is made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-response-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANCNNTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link at connection time.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANACKTMR

Specifies the length of time the system waits before sending an acknowledgement for received frames.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgment-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals, or specify 0 to indicate no delay. If 0 is specified for this parameter, 0 must also be specified for the LANACKFRQ parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKFRQ parameter.

LANINACTMR

Specifies the length of time in which to determine an inactive condition for the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-inactivity-timer: Specify a value ranging from 1 through 255 in 0.1-second intervals for the length of time used to determine an inactive condition for the controller.

LANACKFRQ

Specifies the maximum number of frames that are received before an acknowledgment is sent to the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgment-frequency: Specify a value ranging from 0 through 127 for the number of frames received. If 0 is specified for this parameter, 0 must also be specified for the LANACKTMR parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKTMR parameter.

LANMAXOUT

Specifies the maximum number of frames that can be sent before an acknowledgment is received from the remote system.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-maximum-outstanding-frames: Specify a value ranging from 1 through 127 for the number of frames that can be sent before an acknowledgement is received.

LANACCPTY

Specifies the priority granted to the sending system for sending frames. The larger the number, the higher the priority.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-access-priority: Specify a value ranging from 0 through 3 to indicate how soon a frame can be sent on a local area network (LAN).

LANWDWSTP

Specifies whether to reduce to 1 the maximum number of frames outstanding to the remote system during network congestion. This parameter indicates the number of frames that must be successfully received by the remote system before the number of maximum outstanding frames can be increased by 1. The increase continues this way until the maximum number of outstanding frames reaches the value specified by the LAN maximum outstanding frames (LANMAXOUT) parameter.

***NONE:** The number of outstanding frames is not reduced during network congestion.

LAN-window-step: Specify a value from 1 to 127 for the number of frames that must be successfully received by the remote system before the maximum number of outstanding frames can be increased by 1.

NETLVL

Specifies the level of the X.25 network used to reach this controller. The level is specified by giving the year of the standard used by X.25 network.

1980: The 1980 Standard is used.

1984: The 1984 Standard is used.

1988: The 1988 standard is used.

LGLCHLID

Specifies the logical channel ID used for the X.25 permanent virtual circuit to this controller. The valid entry is in the format xyy, where:

x = the logical group number, derived from the network subscription
yy = the logical channel number, derived from the subscription

CNNPWD

Specifies the password used when connecting to this controller. This password for each controller can consist of any alphanumeric characters represented by the hexadecimal range from 40 through FF.

SWTLINSLCT

Specifies the method that is used to select lines from an X.25 switched line list.

***FIRST:** The lines are selected in the order in which they are specified.

***CALC:** The system determines the value to use.

DFTPKTSIZE

Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

***LIND:** The value specified in the line description is used as the default packet size.

transmit-packet-size: Specify a default packet size for transmission. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***LIND:** The value specified in the line description is used as the default packet size.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

DFTWDWSIZE

Specifies the default window size for transmission and reception.

Element 1: Transmit Window Size

***LIND:** The value specified in the line description is used as the default window size.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

Element 2: Receive Window Size

***LIND:** The value specified in the line description is used as the default window size.

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

USRGRPID

Specifies the closed user-group ID for contacting this X.25 switched virtual circuit (SVC) controller. Specify a value ranging from 0 through 99, as provided by the network subscription. This parameter is not valid for permanent virtual circuit (PVC) connections. It is valid only for SVC circuit outgoing call operations and is ignored for SVC incoming call connections.

RVSCRG

Specifies whether reverse charges are accepted or requested when contacting this controller.

***NONE:** No reverse charging for network tariff billing is accepted.

***REQUEST:** Charges are requested on outgoing call request packets.

***ACCEPT:** Reverse charging for network tariff billing is accepted on incoming requests.

***BOTH:** Both incoming and outgoing requests are accepted.

X25FRMRTY

Specifies the maximum number of times a frame is sent after the response timer ends when connected to this controller.

7: The frame is sent seven times.

X.25-frame-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25CNNRTY

Specifies the maximum number of times a frame is sent after the connect response timer ends when connecting to this controller.

7: The frame is sent seven times.

X.25-connection-retry: Specify a value ranging from 0 through 21 for the number of times a frame is sent.

X25RSPTMR

Specifies the time allowed for an acknowledgment to return when a frame is sent while connected to this controller.

100: Ten seconds is allowed for acknowledgment.

X.25-response-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25CNNTMR

Specifies the time allowed for an acknowledgment to be returned when a frame is sent while connecting to this controller.

100: The time allowed for acknowledgment is 10 seconds.

X.25-connection-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25DLYTMR

Specifies the time interval to try establishing a connection to the controller.

***CALC:** The system determines the value to use.

X.25-delayed-connection-timer: Specify a value ranging from 1 through 32767 in 0.1-second intervals. The system retries the connection indefinitely at the specified interval.

USRFCL

Specifies a string of hexadecimal digits sent to the X.25 network to request additional services. The AS/400 system allows up to 218 hexadecimal characters.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Values

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library

(DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

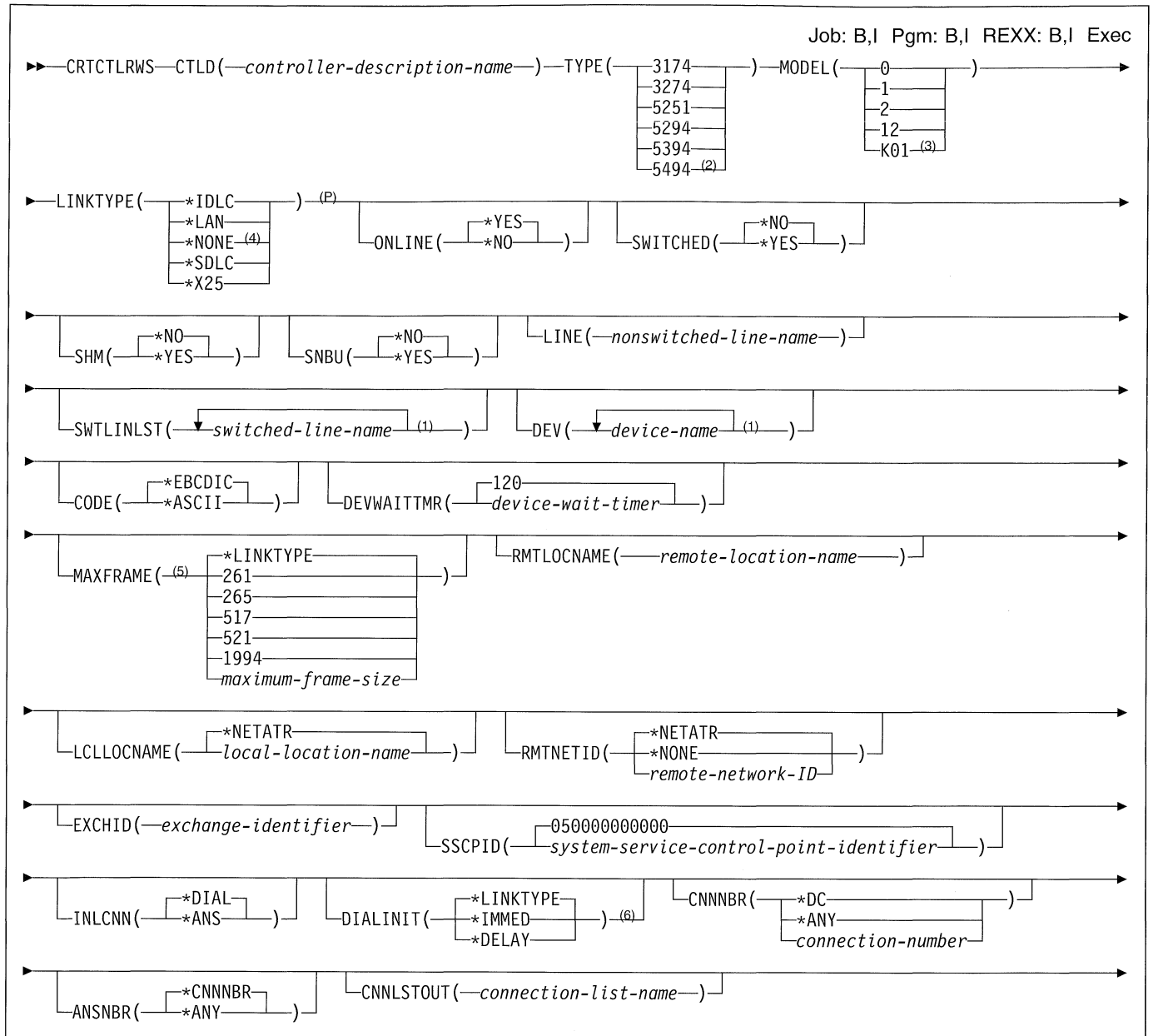
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLRTL   CTLD(CTLR05)  TYPE(3651)  MODEL(0)
  LINKTYPE(*SDLC)  SWITCHED(*YES)  SWTLINLST(SWITCHED1)
  EXCHID(00500005)  INLCNN(*ANS)  STNADR(E1)
  CNNNBR(2553217)
```

This command creates a 3651 Model 0 retail controller description called CTLR05. The controller is on a SDLC switched line at telephone number 255-3217. Connection to the controller is initiated by an incoming call to the IBM AS/400 system.

CRTCTLRWS (Create Controller Description (Remote Work Station)) Command



Notes:

1 A maximum of 64 repetitions

2 If TYPE is 5494, MODEL must be 1 or 2 and LINKTYPE must be *NONE.

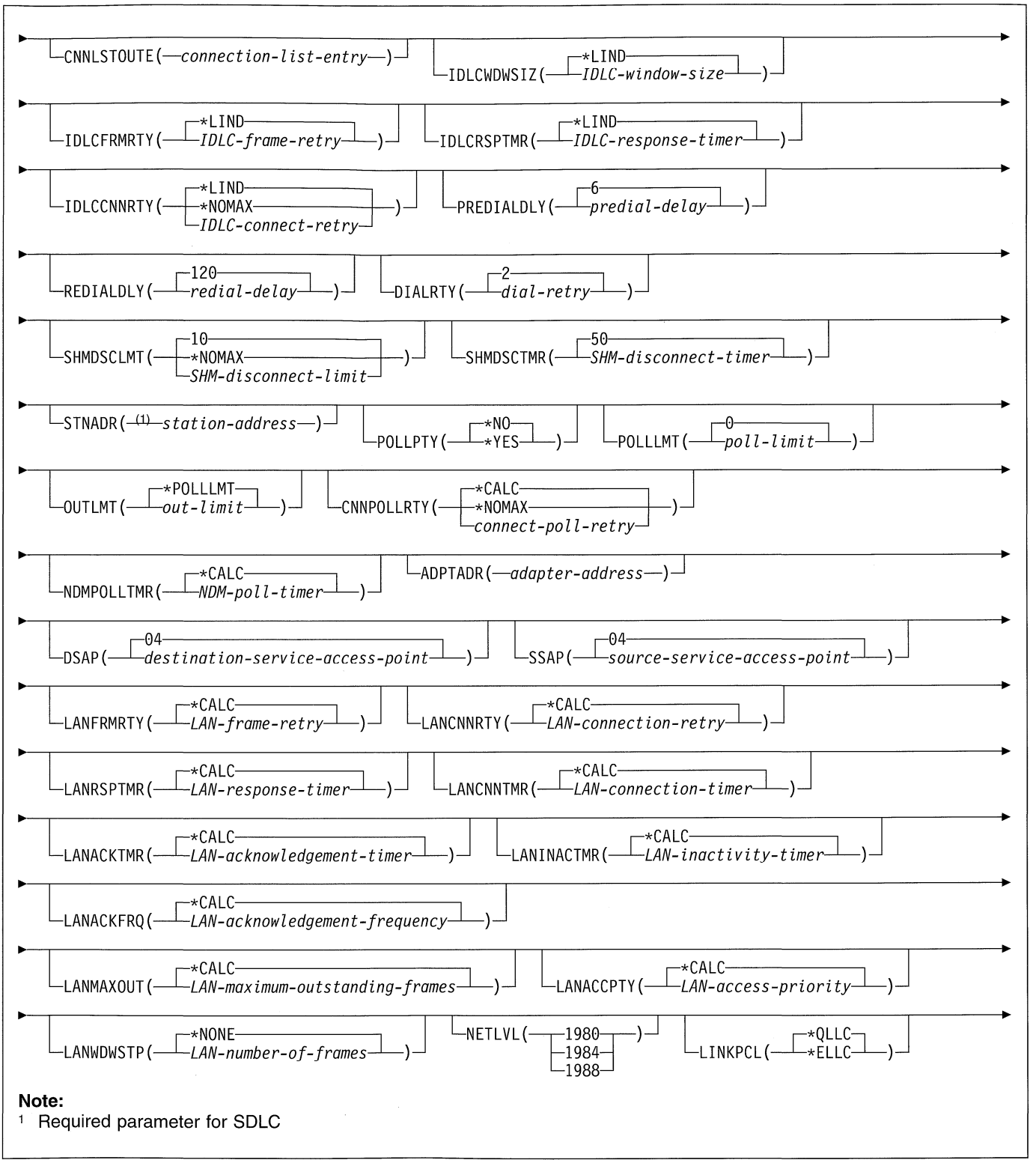
3 Model K01 is valid for DBCS systems.

4 If LINKTYPE is *NONE, TYPE must be 5394 or 5494.

P All parameters preceding this point can be specified in positional form.

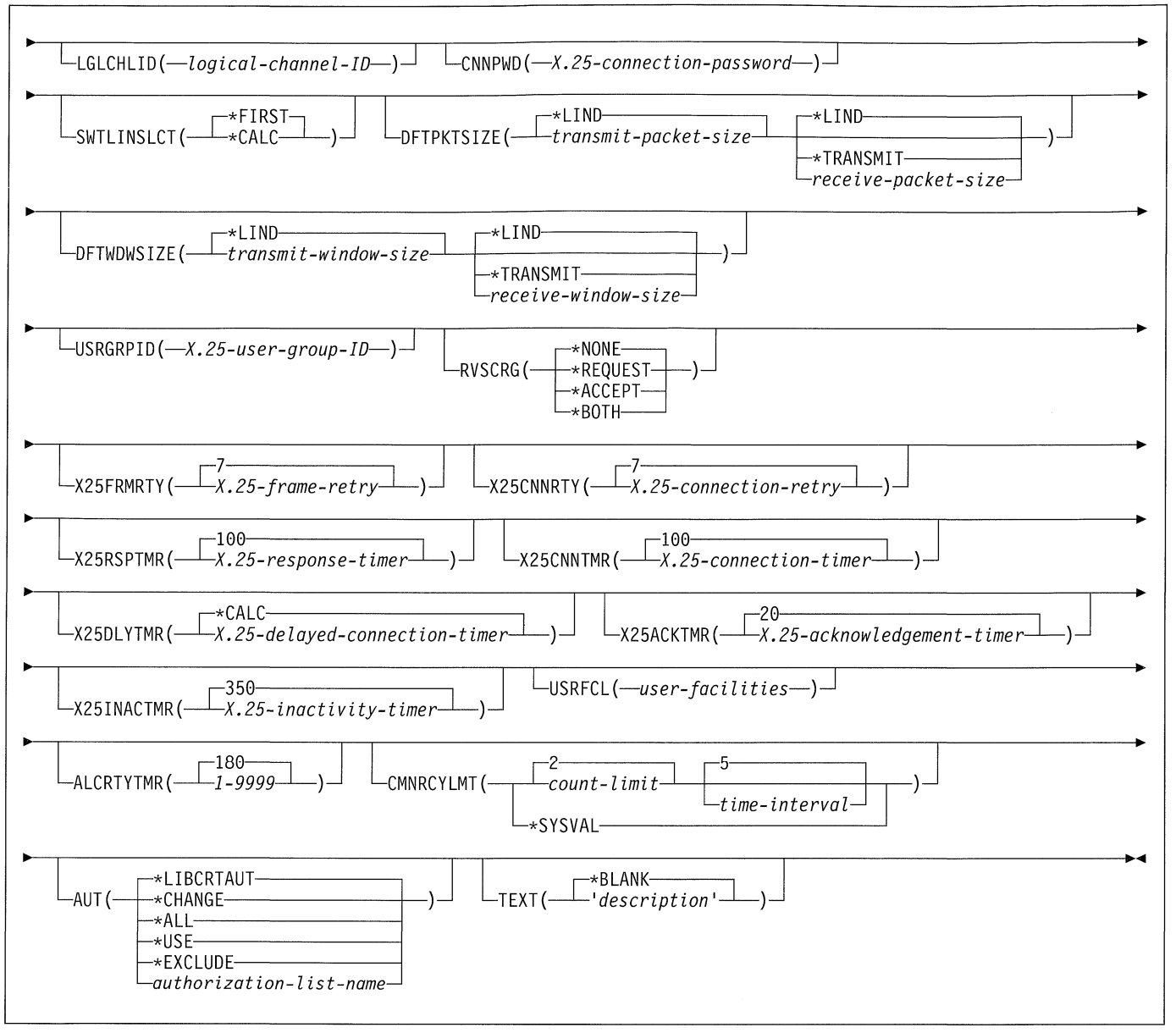
5 If the value of parameter LINKTYPE is *LAN or *TRLAN, any value ranging from 261 through 1994 can be specified.

6 This parameter is valid only if INLCNN(*DIAL) is specified.



Note:

[1] Required parameter for SDLC



Purpose

The Create Controller Description (Remote Work Station) (CRTCTLRWS) command identifies a remote work station controller and describes its characteristics to the system. For more information about using this command, see the *OS/400* Communications Configuration Reference*.

Required Parameters

CTLD

Specifies the name of the controller description being created.

TYPE

Specifies the type of controller for this description.

3174: This description represents a 3174 controller.

3274: This description represents a 3274 controller.

5251: This description represents a 5251 controller.

5294: This description represents a 5294 controller.

5394: This description represents a 5394 controller.

5494: This description represents a 5494 controller.

MODEL

Specifies the model number of the controller represented by this description.

0: The controller for this model number is 3174 or 3274.

1: The controller for this model number is 5294, 5394, or 5494.

2: The controller for this model number is 5394 or 5494.

12: The controller for this model number is 5251 Model 12.

K01: The controller for this model number is 5294 Model K01, which is valid for systems supporting DBCS (double-byte character set).

LINKTYPE

Specifies the type of line to which this controller is attached.

***IDLC:** The controller is attached to an ISDN data link control (IDLC) line. This value can only be specified for 3174 controllers.

***LAN:** This controller is attached to an ethernet or token-ring local area network line. This value can only be specified for 3174 controllers.

***SDLC:** The controller is attached to an synchronous data link control (SDLC) line.

***X25:** This controller is attached to an X.25 line.

***NONE:** This controller is not attached to a line.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

SWITCHED

Specifies whether this controller is attached to a switched line. For an X.25 line, this parameter specifies whether the controller is attached to a switched virtual circuit (SVC).

***NO:** This controller is attached to a nonswitched line or permanent virtual circuit (PVC).

***YES:** This controller is attached to a switched line, switched virtual circuit (SVC), or a local area network (LAN) line.

SHM

Specifies whether or not this controller is used for X.21 short-hold mode. To specify *YES, LINKTYPE(*SDLC) and SWITCHED(*YES) must also be specified.

***NO:** This controller is not used for X.21 short-hold mode.

***YES:** This controller is used for X.21 short-hold mode.

SNBU

Specifies, for controllers attached to nonswitched lines only, whether the switched network backup (SNBU) feature is activated or deactivated. This feature lets the user bypass a broken nonswitched connection by establishing a switched connection. This parameter applies only if SWITCHED(*NO) and SNBU(*YES) are specified when the controller description is created.

Note: The Change Controller Description Remote Work Station (CHGCTLRWS) command must be used to actually activate the feature. This parameter is valid only when SWITCHED(*NO) is specified.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The remote modem has the switched network backup feature. To activate the feature when the non-switched connection is broken, specify ACTSNBU(*YES) on the CHGCTLRWS command.

LINE

Specifies the name of the nonswitched lines to which this controller is attached. The line description must already exist.

Note: For X.25 attachment, this parameter applies to controllers attached to permanent virtual circuits (PVC). The associated line must be varied off before this command is entered.

SWTLINLST

Specifies the names of the switched lines to which this controller attaches. The line descriptions must already exist. Up to 64 switched line names can be specified.

Note: The same line name can be used more than once. This parameter is valid only if SWITCHED(*YES), SNBU(*YES), LINKTYPE(*LAN), or LINKTYPE(*TRLAN) is specified.

DEV

Specifies the names of up to 64 devices attached to this controller.

Note: Use this parameter only when the associated device descriptions have already been created.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

DEVWAITTMR

Specifies the device wait timeout value. This is used to limit the amount of time that a subsystem takes for the work station input/output to complete. The timeout value that is used for each device is taken from the controller to which it is attached at vary on time.

120: For the remote controller, the subsystem waits 120 seconds for the completion of the input/output of each work station attached to the controller.

device-wait-timer: Specify a value ranging from 2 through 600 seconds that the subsystem waits for the work station input/output to complete for all work stations attached to this controller.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

***LINKTYPE:** If *LINKTYPE is specified, the system uses one of the following values (sizes in bytes):

- For 3174 controllers, the following values are used:

```
*SDLC - 265    *IDLC  - 521
*X25 -  256    *LAN   - 1994
```

- For 5394 controllers, the following values are used:

```
*SDLC - 517
*X25  - 512
```

Note: A 5394 controller cannot be connected to local area network (*LAN).

maximum-frame-size: Specify the maximum frame size (PIU size) that can be used for this controller. The value specified for this parameter depends on the type of line and controller being used. The following values (measured in bytes) can be specified for this parameter:

- For 3174 controllers:

```
*SDLC          - 265
*X25           - 256, 265
*LAN           - 265 to 1994
*IDLC          - 265 to 1994
```

- For 5394 controllers:

```
*SDLC - 261, 517
*X25  - 256, 265, 512, 521
```

RMTLOCNAME

Specify the remote location name by which the controller is known to the local system.

LCLLOCNAME

Specifies the local location name.

***NETATR:** The local location name defined in the network attributes is used.

local-location-name: Specify a maximum of 8 characters for the local location name.

RMTNETID

Specifies the name of the remote network in which the adjacent control point resides. In general, the network identifier must be the same for all nodes in the network.

***NETATR:** The local network identifier defined in the network attributes is used as the remote network identifier.

***NONE:** No remote network identifier is used.

remote-network-ID: Specify a maximum of 8 characters for the remote network identifier.

EXCHID

Specifies the exchange identifier of this controller. The controller sends (exchanges) its identifier to another location when a connection is established. The 8-digit hexadecimal identifier contains three digits for the block number and five digits for the identifier of the specific controller. This parameter is required for synchronous data link control (SDLC) switched or switched network backup (SNBU) connections.

Note: Identifiers must be specified for controllers with LINKTYPE(*SDLC) and either SWITCHED(*YES) or SNBU(*YES). In all other cases the identifier is optional and is enforced only if specified.

For example, for the 5251 controller, the value is 020000xx, where 020 is the block number and 000xx is the controller identifier. The first 3 digits of the identifier are always 000, and xx represents the setting of the controller station address switches on the 5251 controller.

SSCPID

Specifies the system service control point (SSCP) of the host system.

050000000000: The default value is 050000000000.

system-service-control-point-identifier: Specify the system service control point identifier (SSCPID) as a 12-digit hexadecimal value.

INLCNN

Specifies the method used to make the initial connection on a switched line between the system and the remote controller. For X.25 lines, this parameter specifies the type of switched virtual circuit (SVC) connection that is made over the X.25 line.

***DIAL:** The connection is made by a call started from the local system.

For X.25, the line to which the controller attaches requires switched virtual circuits (SVCs) configured on the LGLCHLE parameter of type OUT or BOTH (*SVCOUT or *SVCBOTH) for the connection to be successful. If a call is received from the remote controller and all necessary conditions are met, the incoming call can be answered by the system. For X.25 lines, the line attached to the controller requires that switched virtual circuits (SVC) be configured on the LGLCHLE parameter of type IN or BOTH(*SVCIN or *SVCBOTH) for the connection to be successful.

***ANS:** The connection is made by the local system answering an incoming call from this controller.

DIALINIT

Specifies the method used to make the initial dial on a switched line between the system and the remote controller.

***LINKTYPE:** The type of dial connection initiated is specified on the LINKTYPE parameter. For LAN or SDLC short-hold mode connections, the default is to dial the connection immediately upon vary on of the con-

troller description. For all other link types, the default is to delay the dial.

***IMMED:** The dial connection is initiated immediately upon vary on of the controller description.

***DELAY:** The dial connection is delayed until a job is initiated that requests the use of the remote controller resources.

CNNNBR
Specifies the telephone number to dial to connect to this controller.

***DC:** For X.21 circuit switched connections, a direct call is used to connect to the controller.

***ANY:** Calls are accepted from any network address.

connection-number: Specify a telephone number, an X.25 network address, or an X.21 connection number depending on the type of controller and the type of line to which it is attached.

ANSNBR
Specifies the X.25 network address from which to accept calls.

***CNNNBR:** Calls from the X.25 network address specified by the connection number (CNNNBR) parameter are accepted.

***ANY:** Calls are accepted from any X.25 network address.

CNNLSTOUT
Specifies, for ISDN switched connections, the name of a connection list object that contains the ISDN assigned numbers for a dial out operation to the ISDN.

CNNLSTOUTE
Specifies for ISDN switched connections, the entry name from the connection list that is used to make a call to the ISDN. The connection list must be specified on the CNNLSTOUT parameter.

IDLCWDWSIZ
Specifies the window size for transmission to, and reception from, the remote controller.

***LIND:** The window size specified in the line description is used.

IDLC-window-size: Specify the window size. Valid values range from 1 through 31.

IDLCFRMRTY
Specifies the maximum number of attempts to transmit a frame before an error is reported.

***LIND:** The controller specified in the line description is used.

IDLC-frame-retry: Specify the number of attempts. Valid values range from 0 through 100.

IDLCRSPTMR
Specifies the amount of time, in tenths of a second, to wait before retransmitting a frame if acknowledgement has not been received.

***LIND:** The controller specified in the line description is used.

IDLC-response-timer: Specify the amount of time to wait. Valid values range from 10 through 100 tenths of a second. For example, 10 seconds equals 100 tenths of a second.

IDLCCNNRTY
Specifies the number of times to attempt retransmission at connection time.

***LIND:** The controller specified in the line description is used.

***NOMAX:** There is no disconnect limit.

IDLC-connect-retry: Specify the number of attempts. Valid values range from 1 through 100.

PREDIALDLY
Specifies how long to wait (in 0.5 second intervals) before dialing.

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY
Specifies how long to wait (in 0.5 second intervals) before re-dialing when the call attempt is unsuccessful.

120: The default value of 120 provides a 60-second delay.

redial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

DIALRTY
Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of times to retry dialing.

SHMDSCLMT
Specifies the number of nonproductive responses (RR or RNR) that are required from the remote station before the connection can be suspended for this X.21 short-hold mode connection. This parameter is used only if SHM(*YES) is specified.

10: Ten nonproductive responses must be received before the connection can be suspended.

***NOMAX:** There is no disconnect limit.

SHM-disconnect-limit: Specify a value ranging from 1 through 254, indicating the number of nonproductive responses that must be received before the connection can be suspended.

SHMDSCTMR

Specifies the minimum length of time that the primary system maintains the connection to the remote system for this X.21 short-hold mode controller, in tenths of a second. This parameter is used only if SHM(*YES) is specified.

50: The primary system maintains the connection to the remote system for a minimum of 5 seconds.

SHM-disconnect-timer: Specify a value, ranging from 2 through 3000, indicating the minimum length of time, in tenths of a second, that the primary system maintains the connection to the remote system.

STNADR

Specifies the station address used when communicating with the controller.

Note: This is the station address of the remote controller. Valid values range from 01 through FE. This is a required field for SDLC.

POLLPTY

Specifies whether this controller has priority when being polled.

***NO:** This controller does not have polling priority.

***YES:** This controller has polling priority.

POLLLMT

Specifies the number of consecutive polls that are issued to the same controller when the poll results in receiving frames.

0: No consecutive polls are attempted.

poll-limit: Specify a value ranging from 1 through 4 for the number of consecutive polls.

OUTLMT

Specifies the number of times SDLC allows the consecutive transmission of the maximum number of frames to a station before allowing transmission to another station.

***POLLLMT:** The value specified on the POLLLMT parameter is used.

out-limit: Specify a value ranging from 0 through 4 for the number of consecutive transmissions.

CNNPOLLRTY

Specifies the number of times the connect poll is retried before reporting an error.

***CALC:** The number of retries is 7 if the controller is switched, and *NOMAX if the controller is nonswitched.

***NOMAX:** There is no disconnect limit.

connect-poll-retry: Specify a value for the number of retries. Valid values range from 0 through 65534.

NDMPOLLTMR

Specifies the interval for polling this controller when it is in normal disconnect mode (NDM).

***CALC:** The system determines the value to use.

NDM-poll-timer: Specify a value ranging from 1 through 3000 in 0.1-second intervals, or specify 0 to indicate no timer.

ADPTADR

Specifies the 12-character adapter address of the remote controller. This is the address to which the system sends data when it communicates with the remote controller. This value can be obtained from the remote controller's configuration record. Valid values range from hex 000000000001 through hex FFFFFFFFFFFF.

DSAP

Specifies the logical address of the SNA destination service access point (DSAP) to which this system sends data when it communicates with the remote controller. This address allows the controller and the system to route the data that comes from this system. The value must match the value specified on the source service access point (SSAP) parameter in the remote controller's configuration record.

04: The destination SNA destination service access point is 04.

destination-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the SNA destination service access point.

SSAP

Specifies the source service access point. This is the logical address this system uses when it sends data to the remote controller. This address allows the remote controller and the system to route the data that comes from this system. It must match the value assigned to the destination service access point (DSAP) prompt in the remote controller's configuration record.

04: The SNA source service access point is 04.

source-service-access-point: Specify a hexadecimal value ranging from 04 through 9C, in increments of 4 (for example, 04, 08, 0C, 10) to represent the SNA source service access point.

LANFRMRTY

Specifies the number of times a frame is transmitted if there is no acknowledgement from the remote controller in the time period specified on the LANRSPTMR parameter. This value is used only after a successful connection has been made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-frame-retry: Specify a value ranging from 0 through 254 for the number of times a frame is transmitted before an acknowledgement is received.

LANCNNRTY

Specifies the number of times a transmission is attempted before an acknowledgement is received. This

value is used at connection time (unlike LANFRMRTY which is used after a connection has been made).

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-retry: Specify a value ranging from 0 through 254 for the number of times the transmission is attempted before an acknowledgement is received.

LANRSPTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link after a connection is made.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-response-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANCNNTMR

Specifies the length of time the system waits before an inoperative condition occurs on a link at connection time.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-connection-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals for the amount of time the system waits before an inoperative condition occurs.

LANACKTMR

Specifies the length of time the system waits before sending an acknowledgement for received frames.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgement-timer: Specify a value ranging from 1 through 254 in 0.1-second intervals, or specify 0 to indicate no delay. If 0 is specified for this parameter, 0 must also be specified for the LANACKFRQ parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKFRQ parameter.

LANINACTMR

Specifies the length of time in which to determine an inactive condition for the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-inactivity-timer: Specify a value ranging from 1 through 255 in 0.1-second intervals for the length of time used to determine an inactive condition for the controller.

LANACKFRQ

Specifies the maximum number of frames that are received before an acknowledgement is sent to the controller.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-acknowledgement-frequency: Specify a value ranging from 0 through 127 for the number of frames received. If 0 is specified for this parameter, 0 must also be specified for the LANACKTMR parameter; if a nonzero value is specified for this parameter, a nonzero value must also be specified for the LANACKTMR parameter.

LANMAXOUT

Specifies the maximum number of frames that can be sent before an acknowledgement is received from the remote system.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-maximum-outstanding-frames: Specify a value ranging from 1 through 127 for the number of frames that can be sent before an acknowledgement is received.

LANACCPTY

Specifies the priority granted to the sending system for sending frames. The larger the number, the higher the priority.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

LAN-access-priority: Specify a value ranging from 0 through 3 to indicate how soon a frame can be sent on a local area network (LAN).

LANWDWSTP

Specifies whether to reduce the number of outstanding frames that can be sent before an acknowledgement is received from the remote system during network congestion. This parameter (LAN Window Step) also indicates the number of frames that must be successfully received before the number of outstanding frames can be incremented to the maximum allowable value.

***NONE:** The number of outstanding frames is not reduced during network congestion.

LAN-number-of-frames: Specify the number of outstanding frames to be reduced before an acknowledgement is received from the remote system.

NETLVL

Specifies the level of the X.25 network used to reach this controller. The level is specified by giving the year of the standard used by the X.25 network.

1980: 1980 standard is used.

1984: 1984 standard is used.

1988: The 1988 standard is used.

LINKPCL

Specifies the link level protocol used on the X.25 network to communicate with this controller.

***QLLC:** The qualified logical link control (QLLC) protocol is used.

***ELLC:** The enhanced logical link control (ELLC) protocol is used.

LGLCHLID
Specifies the logical channel ID which is used for the X.25 PVC to this controller.

Specify xyy where x is the X.25 logical channel group number assigned to this controller, and yy is the X.25 logical channel number assigned to this controller.

CNNPWD
Specifies the X.25 network password used for password exchange with the X.24 Call Request and Incoming Call packets (Call User Data field). This parameter is not valid for permanent virtual circuit (PVC) connections.

This value cannot exceed eight characters. If fewer than eight characters are specified, the field is padded with blanks. The password on incoming call requests must match this password, or no connection is allowed. Valid passwords consist of hexadecimal values ranging from 40 through FF. The keyboard available to the user can only support a subset of these characters.

Note: This connection password is highly recommended for controllers that operate with the Enhanced Logical Link Control (ELLC) protocol LINKPCL(*ELLC) on switched virtual circuit (SVC) connections. This enhanced protocol supports reconnection of virtual circuits after network errors which disconnect the line with a clear signal; however, this reconnection is not allowed to proceed without the password validation procedure being used as a unique identifier for the controller.

SWTLINSLCT
Specifies the method that is used to select lines from an X.25 switched line list.

***FIRST:** The lines are selected in the order in which they are specified.

***CALC:** The system determines the value to use.

DFTPKTSIZE
Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

***LIND:** The value specified in the line description is used as the controller.

transmit-packet-size: Specify a default packet size for transmission. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***LIND:** The value specified in the line description is used as the controller.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

DFTWDWSIZE
Specifies the default window size used by the X.25 network.

Element 1: Transmit Window Size

***LIND:** The value specified in the line description is used as the controller.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus-8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

Element 2: Receive Window Size

***LIND:** The value specified in the line description is used as the controller.

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus-8 packet numbering. Valid values range from 1 through 15 for networks that use 128 packet numbering.

USRGRPID
Specifies the closed user-group ID for contacting this station as provided by the network subscription. This parameter is not valid for permanent virtual circuit (PVC) connections. It valid only for SVC circuits outgoing call operations and is ignored for SVC incoming call connections.

RVSCRG
Specifies whether reverse charges are accepted or requested when contacting this controller.

***NONE:** No reverse charging for network tariff billing is accepted.

***REQUEST:** Charges are requested on outgoing call request packets.

***ACCEPT:** Reverse charging for network tariff billing is accepted on incoming requests.

***BOTH:** Both incoming and outgoing requests are accepted.

X25FRMRTY
Specifies the maximum number of times a frame is sent after the response timer ends when connected to this controller.

7: The frame is sent seven times.

X.25-frame-retry: Specify a value ranging from 0 through 21 for the number of attempts.

X25CNNRTY
Specifies the maximum number of times that a frame is sent after the connect response timer ends when connecting to this controller.

7: The frame is sent seven times.

X.25-connection-retry: Specify a value ranging from 1 through 21 for the number in 0.1-second intervals, or specify 0 to indicate no timer.

X25RSPTMR
Specifies the time allowed for a response to be returned when a frame is sent while connected to this controller.

100: The time allowed for response is ten seconds.

X.25-response-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25CNNTMR
Specifies the time allowed for a response to be returned when a frame is sent while connecting to this controller.

100: The time allowed for a response is ten seconds.

X.25-connection-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25DLYTMR
Specifies how often to try establishing a connection to the controller.

***CALC:** The system determines the value to use.

X.25-delayed-connection-timer: Specify a value ranging from 1 through 32767 in 0.1-second intervals. The system retries the connection indefinitely at the specified intervals.

X25ACKTMR
Specifies the time period to delay sending responses for received frames.

20: The time allowed for a response is ten seconds.

X.25-acknowledgement-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

X25INACTMR
Specifies the time period used to determine an inactive condition for the controller.

350: An inactive condition is determined after 35 seconds.

X.25-inactivity-timer: Specify a value ranging from 1 through 2550 in 0.1-second intervals.

USRFCL
Specifies a string of hexadecimal digits sent to the X.25 network to request additional services. The AS/400 system allows up to 218 hexadecimal characters.

ALCRTYTMR
Specifies the length of time, in seconds, the system waits between attempts to establish an LU6.2 session.

180: The system waits 180 seconds.

1-9999: Specify a length of time in seconds. Valid values range from 1 through 9999.

CMNRCYLMT
Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT
Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

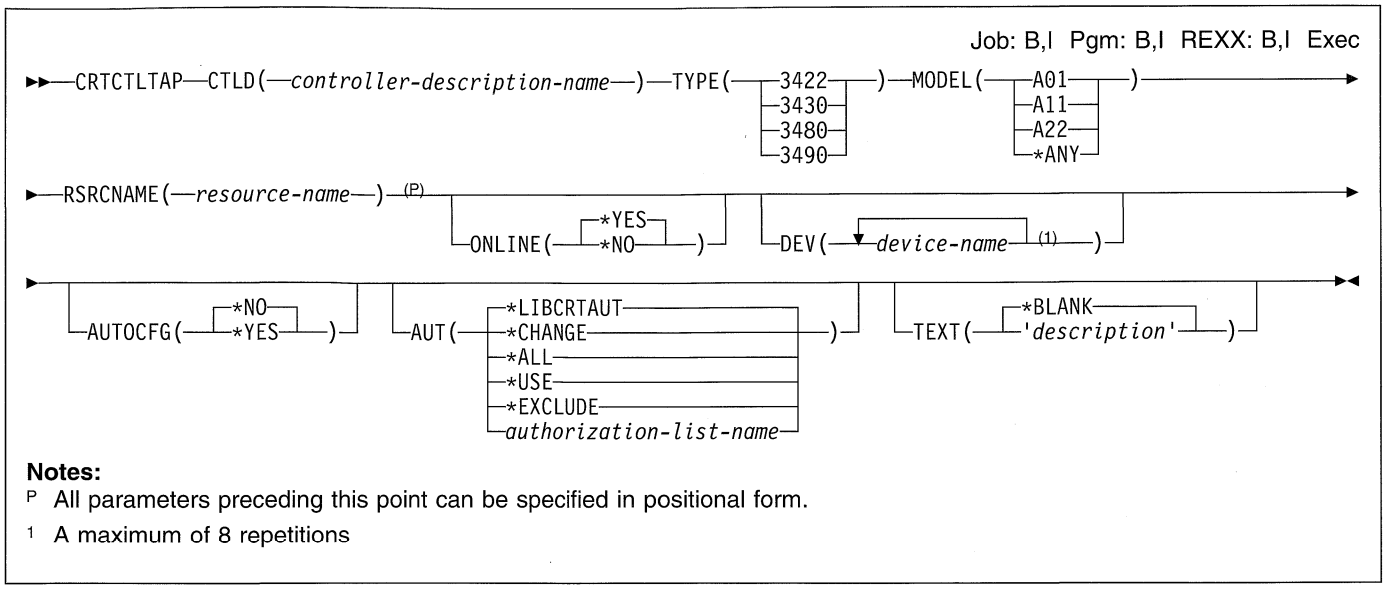
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLRWS   CTLD(CTL005)  TYPE(5294)  MODEL(1)
  LINKTYPE(*SDLC)  SWITCHED(*YES)
  EXCHID(04500003)  INLCNN(*ANS)  STNADR(03)
  CNNNBR(5551234)  TEXT('Chicago branch')
```

This command creates a 5294 Model 1 remote controller description called CTL005. The controller is on a SDLC switched line at telephone number 555-1234. Connection to the controller is initiated by an incoming call to the IBM AS/400 system.

CRTCTLTAP (Create Controller Description (Tape)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 8 repetitions

Purpose

The Create Controller Description (Tape) (CRTCTLTAP) command creates a controller description for a tape device.

Required Parameters

CTLD

Specifies the name of the controller description being created.

TYPE

Specifies the type of device this description represents.

3422: The controller is for a 3422 tape device.

3430: The controller is for a 3430 tape device.

3480: The controller is for a 3480 tape device.

3490: The controller is for a 3490 tape device.

MODEL

Specifies the model number of the controller represented by this description. Valid models for this parameter include A01, A11, A22, or *ANY.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the WRKHDWPRD command to determine the resource name.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

DEV

Specify the names of the devices that are attached to this controller (the device descriptions must already exist). The number of devices that can be specified for each controller is shown in the following table.

Controller	Maximum Number of Devices
3422, 3430	4
3480, 3490	8

AUTOCFG

Specifies whether this controller is automatically configured at IPL.

***NO:** This controller is not automatically configured at IPL.

***YES:** This controller is automatically configured at IPL.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management

authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

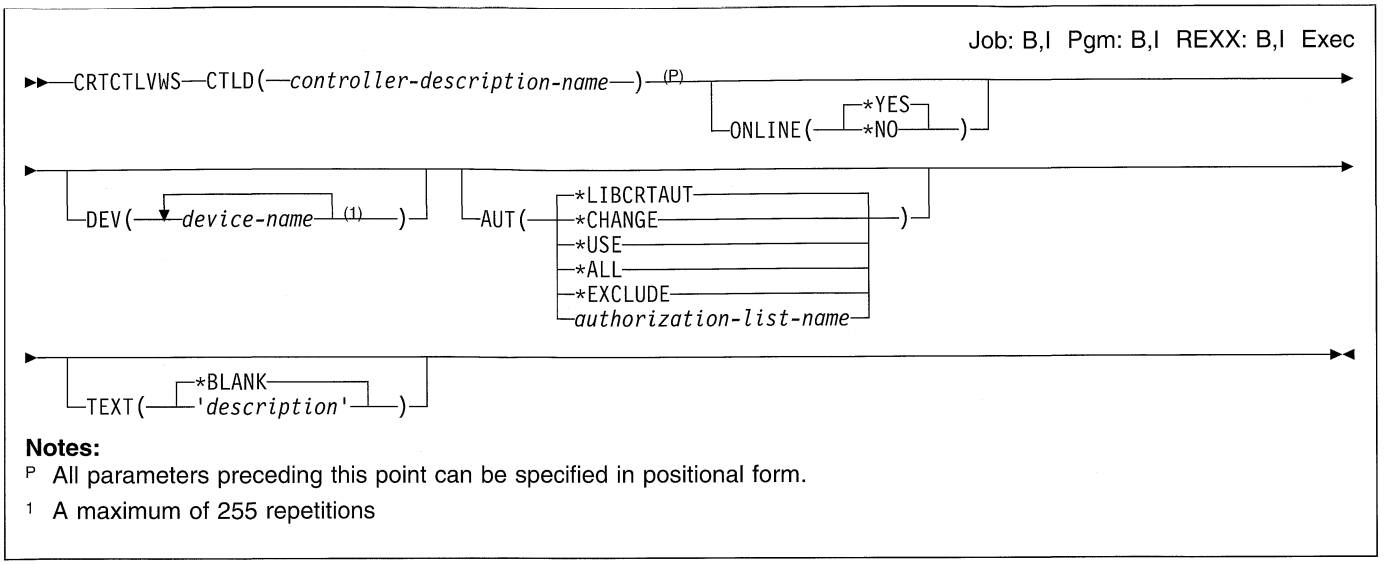
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLTAP   CTLD(TAP01)  TYPE(3422)  MODEL(A01)
  RSRCNAME(TAP01)
```

This command creates a controller description named TAP01 for a 3422 tape controller.

CRTCTLVWS (Create Controller Description (Virtual Work Station)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 255 repetitions

Purpose

The Create Controller Description (Virtual Work Station) (CRTCTLVWS) command creates a controller description for a virtual work station (pass-through) controller. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameter

CTLD

Specifies the name of the controller description being created.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The controller is automatically varied on at IPL.

***NO:** This controller is not automatically varied on at IPL.

DEV

Specifies the names of the virtual devices attached to this controller.

AUT

Specifies the authority given to users who do not have specific authority to the controller description, who are not on an authorization list, and whose user group has no specific authority to the controller description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the controller description, such as running a program or reading a file. The user cannot change the controller description. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the controller description.

***EXCLUDE:** The user cannot access the controller description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

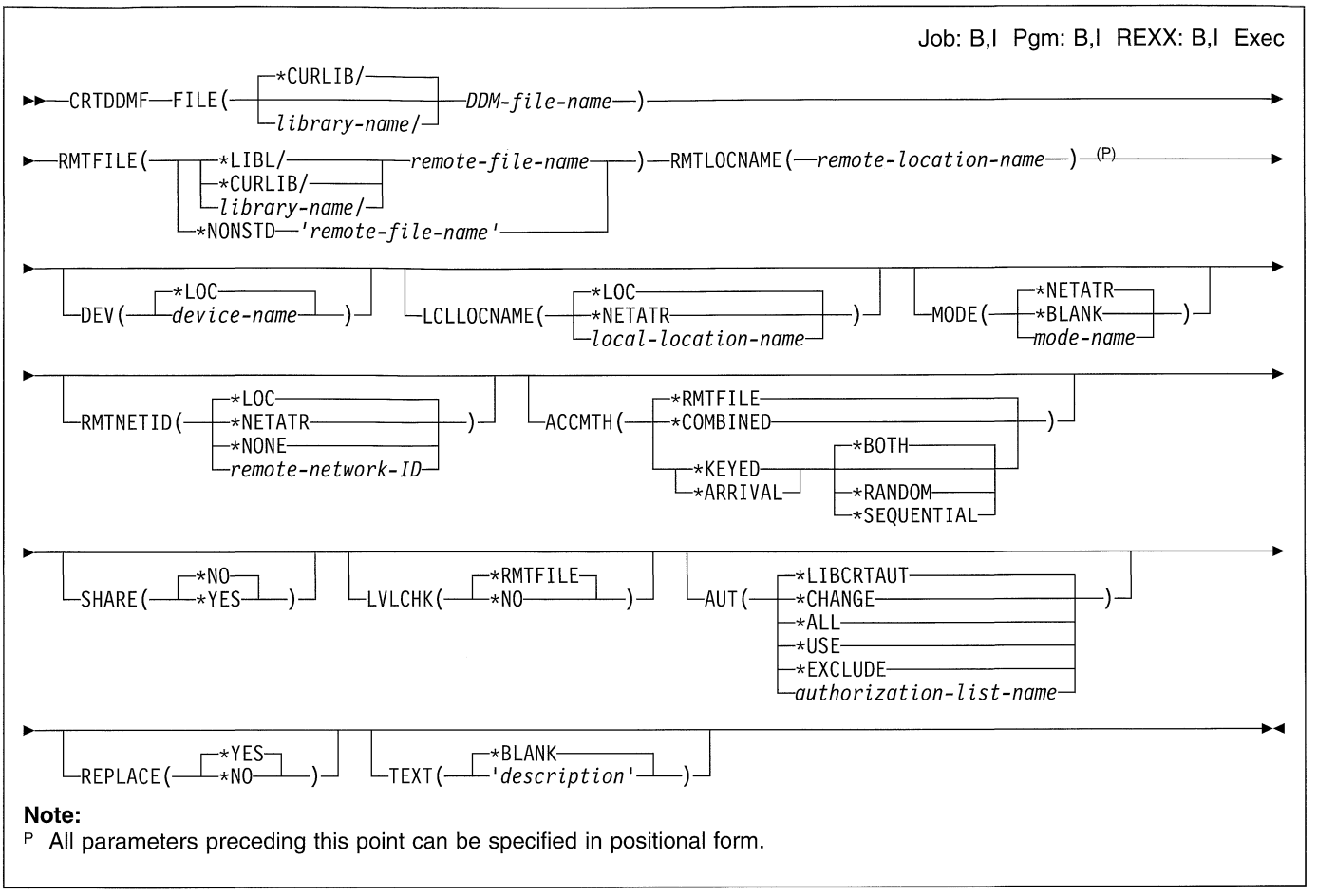
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTCTLVWS   CTLD(VRTCTL325)
  TEXT('S/325 virtual controller')
```

This command creates a controller description named VRTCTL325.

CRTDDMF (Create Distributed Data Management File) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Create Distributed Data Management File (CRTDDMF) command creates a distributed data management (DDM) file. The DDM file is used as a reference file by programs to access files located on a remote (target) system in the DDM network. Programs on the local (source) system know a remote file only by the DDM file's name, not by the remote file's actual name. (The DDM file name, however, can be the same as the remote file name.)

The DDM file (on the source system) contains the name of the remote file accessed and the name of the remote (target) system that contains the file. It can also specify the type of access method that is used to access records in the remote file.

Required Parameters

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the DDM file can be qualified by one of the following library values:

***CURLIB:** The DDM file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the DDM file is created.

DDM-file-name: Specify the name of the DDM file being created.

RMTFILE

Specifies the name of the remote file on the target system. This file name must be specified in code page 500. The file does not need to exist when the DDM file is created.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

Note: The library name is used only if the target system is an AS/400 system.

remote-file-name: Specify up to 10 characters for an AS/400 system file name, up to 10 characters for a System/38 simple file name, or up to 8 characters for a System/36 file name, that identifies the remote file accessed. No apostrophes, blanks, or any other special characters are allowed, and any lowercase characters are changed to uppercase.

- If the target system is an AS/400 system:
  - If *LIBL (the default library qualifier) is specified or assumed, the library list in the called job on the target system is searched to locate the file.
  - If *CURLIB is specified, the current library in the called job on the target system is searched to locate the file.
  - A member name can be specified as part of the remote file name, but it is considered a nonstandard name and the library/file (member) name parts must follow the value *NONSTD.
- If the target system is a System/38 system:
  - A qualified file name can be specified as part of the remote file name but is considered a nonstandard name and the full file name must follow the value *NONSTD.
  - A qualified file name and member name can be specified as part of the remote file name but is considered a nonstandard name and the full file name must follow the value *NONSTD.
  - If *LIBL is specified as the library value, the library list in the called job on the target system is searched to locate the file.
- If the target system is a System/36, the remote file name is the same as its System/36 file label, as used by System/36 OCL.

***NONSTD** *'remote-file-name':* For target systems that allow naming conventions other than those used by AS/400 system, System/38, and System/36, and when specifying a *member* name of a remote AS/400 system or qualified name or member name of a remote System/38 file, enter the value *NONSTD followed by up to 255 characters for the name of the remote file accessed; the name must be in the form required by the target system. The name must always be enclosed in apostrophes, and can contain lowercase letters, blanks, periods, or other special characters. The AS/400 system and System/38 names must be in uppercase (because they are not changed to uppercase if enclosed in apostrophes) and no blanks are allowed.

If the target system is a System/38, a file and library name can be specified enclosed in apostrophes following the value *NONSTD.

If the target system is an AS/400 system or System/38, a file name, library name, and member name can all be specified.

- If a member name is specified, the full file name must be enclosed in apostrophes and follow the value *NONSTD, and the member name *must* be enclosed in parentheses and immediately follow (with no space) either the library name or the file name.
- If the target system is an AS/400 system or System/38 and *LIBL is specified, the library list in the evoked job on the target system is used to search for the file.
- If the target system is an AS/400 system and *CURLIB is specified, the current library in the evoked job on the target system is used to search for the file.

Examples of specifying AS/400 system remote file member names are:

```
RMTFILE(*NONSTD 'CAR(JULY)')
RMTFILE(*NONSTD 'SALES/CAR(JULY)')
```

Examples of specifying System/38 remote file member names are:

```
RMTFILE(*NONSTD 'CAR.SALES')
RMTFILE(*NONSTD 'CAR.*LIBL')
RMTFILE(*NONSTD 'CAR(JULY)')
RMTFILE(*NONSTD 'CAR.SALES(JULY)')
```

RMTLOCNAME

Specifies the name of the remote location that is used with this object.

Note: Multiple DDM files can use the same remote location for the target system.

remote-location-name: Specify the name of the remote location that is associated with the target system. The remote location, which is used in accessing the target system, does not need to exist when the DDM file is created but must exist when the DDM file is opened.

More information on remote locations is in the *APPC Programmer's Guide*.

Optional Parameters

DEV

Specifies the name of the APPC device description on the source system that is used with this DDM file. The device description does not need to exist when the DDM file is created.

***LOC:** The device associated with the remote location is used. If several devices are associated with the remote location, the system determines which device is used.

device-name: Specify the name of a communications device associated with the remote location. If the device name is not valid for the remote location, a message is sent when the program device entry is acquired. More information on device names is in the *APPC Programmer's Guide*.

LCLLOCNAME

Specifies the local location name.

***LOC:** The device associated with the remote location is used. If several devices are associated with the remote location, the system determines which device is used.

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the name of the local location that is associated with the remote location. The local location name is specified only if the user indicates a specific local location for the remote location. If the local location name is not valid for the remote location, an escape message is sent when the DDM file is opened.

More information on local location names is in the *APPC Programmer's Guide*.

MODE

Specifies the mode name that is used with the remote location name to communicate with the target system.

***NETATR:** The mode name specified in the network attributes is used.

***BLANK:** Text is not specified.

mode-name: Specify the name of the mode that is used. If the mode name is not valid for any combination of remote location and local location, an escape message is sent when the DDM file is opened.

More information on mode names is in the *APPC Programmer's Guide*.

RMTNETID

Specifies the remote network ID in which the remote location resides that is used to communicate with the target system.

***LOC:** The remote network identifier (ID) associated with the remote location is used. If several remote network IDs are associated with the remote location, the system determines which remote network ID is used.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

remote-network-ID: Specify the remote network ID that is associated with the remote location. The remote network ID is specified only if the user indicates a specific remote network ID for the remote location. If the remote network ID is not valid for the remote location, an escape message is sent when the DDM file is opened.

More information on remote network IDs is in the *APPC Programmer's Guide*.

ACCMTH

Specifies, when the remote file is not on an AS/400 system or System/38 target system, the DDM access method used to open the remote file and access its records. Specifying a value other than *RMTFILE for this parameter may also improve performance when processing requests to remote files on a system other than an AS/400 system or System/38 targets.

This parameter is ignored when the target system is an AS/400 system or a System/38; the access to a remote AS/400 system and System/38 file is supported as if it were a local file.

***RMTFILE:** The source system selects the access method that is compatible with (a) the attributes of the remote file identified by the RMTFILE parameter and (b) the access methods supported by the target system for that file. For systems other than the AS/400 system and System/38 target systems, if this value is used and the source system cannot select an access method when the file is opened, a message is sent to the program user. A different value must then be specified for this parameter, using the CHGDDMF command, after someone at the target system has been contacted about the appropriate access method information for the file.

***COMBINED:** The DDM combined access method is used for the remote file. This access method combines the file processing capabilities of the *combined by key* (*KEYED *BOTH) and the *combined by record number* (*ARRIVAL *BOTH) access methods, as shown in the following table. The record can be selected with a key value or a record number. The position can then be set relatively or randomly by key value or by record number. If duplicate keys are present in the file, they are processed in the order defined by each target system's implementation of the DDM architecture.

Access Method

***KEYED or *ARRIVAL:** Specify a set of two values that indicates the access method that is used to access the remote file. If only the first value is specified (*KEYED or *ARRIVAL), the default for the second value is *BOTH, and either random or relative (sequential) selection can be requested.

The other possible values for the ACCMTH parameter are shown below. The remote file attributes (in the far left column) refer to the type of file on the target system. The local access method (in the last three columns) refers to the way in which the source AS/400 system or System/38 program intends to access the records in the remote file.

Remote File Attributes	Local Access Method		
	***SEQUENTIAL**	***RANDOM**	***BOTH**
***ARRIVAL**	Relative by record number	Random by record number	Combined by record number
***KEYED**	Relative by key	Random by key	Combined by key

The following explanations refer to the possible values for the ACCMTH parameter.

Relative by record number access method (*ARRIVAL *SEQUENTIAL): This method allows access to records relative to the current position in record number sequence. The record number is not specified to identify the record.

Random by record number access method (*ARRIVAL *RANDOM): This method allows access to records by specifying a record number in a random sequence determined by the requester.

Combined by record number access method (*ARRIVAL *BOTH): This method combines the capabilities of the relative by record number and random by record number access methods.

Relative by key access method (*KEYED *SEQUENTIAL): This method allows records in a keyed file to be accessed in key value sequence. Records can be accessed by moving forward or backwards in key sequence from the current record. The key value is not specified to identify the record.

Random by key access method (*KEYED *RANDOM): This method allows records in a keyed file to be accessed in a random sequence. Records are selected by their key value and not their position in the file.

Combined by key access method (*KEYED *BOTH): This method combines the capabilities of the relative by key and random by key access methods.

SHARE

Specifies whether the open data path (ODP) for the DDM file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

Operation considerations (regarding buffers and file position, for example) for SHARE(*YES) are the same as for database files.

LVLCHK

Specifies whether the record format level identifiers in the program are checked against those in the remote file when the DDM file is opened. If so, the record format identifiers in the program must match those in the remote file. If they do not match, an error message is sent to the requesting program and neither the DDM file nor the associated remote file is opened. Files that have an error while being opened are automatically closed. This parameter can be overridden by an Override with Database File (OVRDBF) command before the remote file is opened.

***RMTFILE:** The level identifiers of the record formats of the remote file (identified in the RMTFILE parameter) are checked at the time the DDM file is opened.

If the target system is *not* an AS/400 system and not a System/38, the source AS/400 system creates a level check value based on the record length of the remote file and any key fields used in it. The created values are then compared to the values in the program, and they must match before the remote file can be opened. This reduces the chances of the wrong file being selected.

Note: Before this can be done for a system other than an AS/400 system or a System/38, the program must be compiled (or recompiled) using the DDM file. During the compilation, the DDM file is used to establish communications with the target system, get the remote file's attributes from the target system, and generate the level identifier values so they can be included in the compiled program for later level checking.

***NO:** The level identifiers are not checked when the file is opened.

AUT

Specifies the authority given to users who do not have specific authority to the DDM file, who are not on an authorization list, and whose user group has no specific authority to the DDM file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the DDM file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the DDM file). The public authority is determined when the DDM file is created. If the CRTAUT value for the library changes after the DDM file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the DDM file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the DDM file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the DDM file.

***USE:** The user can perform basic operations on the DDM file, such as running a program or reading a file.

The user cannot change the DDM file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the DDM file.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether an existing file is replaced by the new DDM file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** An existing file is replaced by the DDM file being created.

***NO:** No replacement occurs.

TEXT

Specifies text that briefly describes the DDM file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

The following examples describe the creation of a DDM file.

Example 1: Creating a DDM File to Access a File at Another AS/400 System

```
CRTDDMF   FILE(SOURCE/SALES)
  RMTFILE(REMOTE/SALES)
  RMTLOCNAME(NEWYORK)
```

This command creates a DDM file named SALES, and stores it in the SOURCE library on the source system. This DDM file uses the remote location named NEWYORK to access a remote file named SALES stored in the REMOTE library on an AS/400 system in New York.

Example 2: Creating a DDM File to Access a File Member at Another IBM AS/400 System

```
CRTDDMF   FILE(SOURCE/SALES)
  RMTLOCNAME(NEWYORK)
  RMTFILE(*NONSTD 'REMOTE/SALES(APRIL)')
```

This command creates the same file as in the previous example, except that now it accesses a specific member in the remote SALES file; the member is named APRIL.

Example 3: Creating a DDM File to Access a File on a System/38

```
CRTDDMF   FILE(OTHER/SALES)
  RMTLOCNAME(CHICAGO)
  RMTFILE(*NONSTD 'PAYROLL.REMOTE')
```

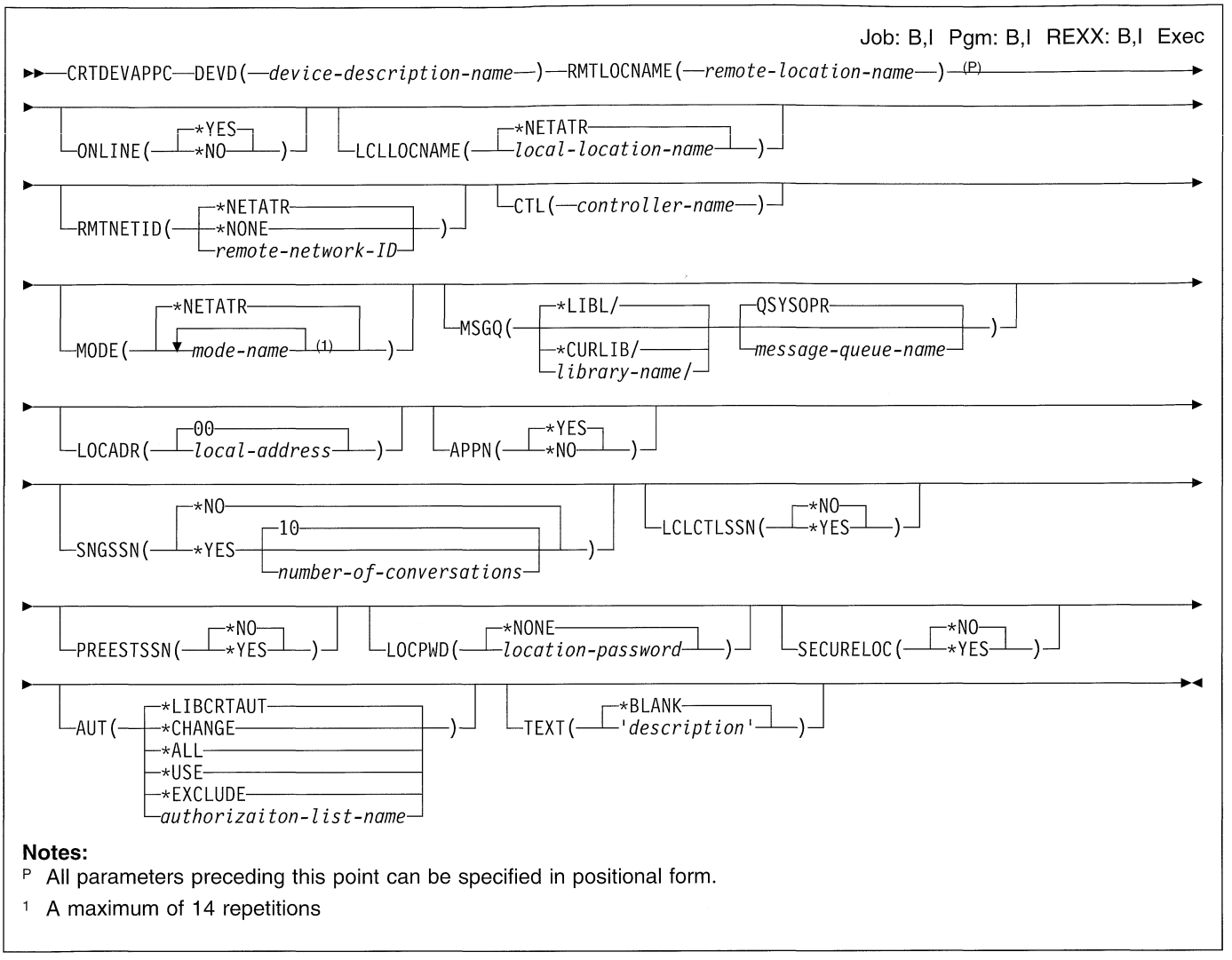
This command creates a DDM file named SALES, and stores it in the library OTHER on the source system. The remote location CHICAGO is used by the DDM file to access a remote file named PAYROLL in library REMOTE on a System/38.

Example 4: Creating a DDM File to Access a File on a System/36

```
CRTDDMF   FILE(OTHER/SALES)
  RMTFILE(PAYROLL)
  RMTLOCNAME(DENVER)  LVLCHK(*NO)
```

This command creates a DDM file named SALES, and stores it in the library OTHER on the source system. The remote location DENVER is used by the DDM file to access a remote file named PAYROLL on a System/36 in Denver. No level checking is performed between the PAYROLL file and the application programs that access it. Because the ACCMTH parameter was not specified, the access method for the target system is selected by the source AS/400 system when the DDM file is opened to access the remote file.

CRTDEVAPPC (Create Device Description (APPC)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 14 repetitions

Purpose

The Create Device Description (APPC) (CRTDEVAPPC) command creates a device description for an advanced program-to-program communications (APPC) device. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

RMTLOCNAME

Specifies the name of the remote location. The name can contain characters A through Z, 0 through 9, $, #, or @, but the first character cannot be 0 through 9.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

LCLLOCNAME

Specifies the local location name.

Note: The name cannot be the same as that specified for the RMTLOCNAME parameter. The combination of the LCLLOCNAME and RMTLOCNAME names must be unique for each device description attached to the same controller.

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the name (up to eight characters) by which the local system is known to the remote

device. The name can contain characters A through Z, 0 through 9, $, #, or @, but the first character cannot be 0 through 9.

RMTNETID
Specifies the name of the remote network.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

remote-network-ID: Specify the remote network ID. The name can contain characters A through Z, 0 through 9, $, #, or @, but the first character cannot be 0 through 9.

CTL
Specifies the name of the controller description to which this object is attached.

MODE
Specifies the names of the modes that define the sessions on this device.

***NETATR:** The mode name specified in the network attributes is used.

mode-name: Specify no more than eight characters to specify a mode name for this device. The name may contain characters A through Z, 0 through 9, $, #, and @. The first character may not be 0 through 9. The mode name cannot be CPSVCMG or SNASVCMG; these mode names are reserved for system use. Up to 14 mode names can be specified. When APPN(*YES) and LOCADR(00) are specified on the device, no modes should be added to the device description. APPN will automatically add the modes.

MSGQ
Specifies the qualified name of the message queue to which messages are sent.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QSYSOPR: Messages are sent to the QSYSOPR message queue.

message-queue-name: Specify the qualified name of the message queue to which operational messages are sent.

LOCADR
Specifies the local location address. Specify a 2-character hexadecimal value ranging from 00 through FF.

00: The 00 value is used.

local-address: Specify the address of the location of the APPC device.

APPN
This parameter specifies whether networking is used.

***YES:** Networking is used and dynamic definition of resources is allowable.

***NO:** Networking is not used. Mode names cannot be added to the device while in vary on status.

SNGSSN
Specifies whether a single session using this device description will be used between the local and remote locations.

***NO:** Multiple sessions are used. LOCADR(00) must be specified when SNGSSN(*NO) is specified.

Element 1: Single-Session Device Description

***YES:** This device description uses single sessions.

Element 2: Number of Single-Session Conversations

***10:** The number of single session conversations allowed for this device description is ten.

number-of-conversations: Specify the number of conversations allowed for a single session of this device description. Valid values range from 1 through 512.

LCLCTLSSN
Specifies whether single sessions are locally or remotely controlled. *YES is valid only if SNGSSN parameter was specified as *YES.

***NO:** The single session is remotely controlled.

***YES:** The single session is locally controlled.

PREESTSSN
Specifies whether the single session is established automatically when the mode is started with the remote location. The value of *YES is valid only if SNGSSN(*YES) and LCLCTLSSN(*YES) are both specified.

***NO:** The session is not automatically established when the mode is started.

***YES:** The session is automatically established when the mode is started.

LOCPWD
Specifies the password used to validate the session establishment request.

***NONE:** No password is used.

location-password: Specify the password to validate the session establishment request. An even number of hexadecimal characters must be specified. Up to 16 characters can be entered.

For example, a password consisting of the letter A would be input as C1, and an integer one would be input as F1 on the LOCPWD parameter.

SECURELOC

Specifies whether the local location allows the remote location to verify user passwords and to send an already verified indicator with the program start request.

***NO:** Security validation by the remote location is not accepted.

***YES:** Security validation by the remote location is accepted.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

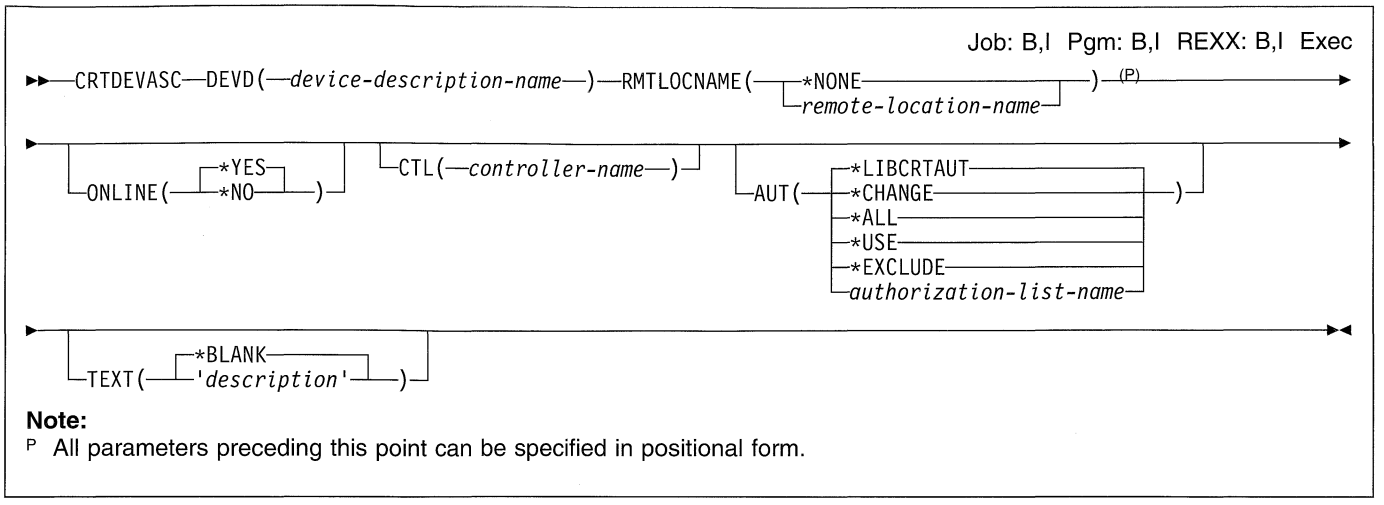
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVAPPC   DEVD(APPC1)  LOCADR(00)
  RMTLOCNAME(CHICAGO)  CTL(CTLAPPC01)  SNGSSN(*YES)
```

This command creates a device description for an APPC communications device named APPC1. The device has a location address of X'00' and is in Chicago, attached to controller CTLAPPC01. This device is limited to one session at a time.

CRTDEVASC (Create Device Description (Async)) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Async) (CRTDEVASC) command creates a device description for an asynchronous device. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

RMTLOCNAME

Specifies the remote location name of the system with which this object communicates.

***NONE:** The remote location is not specified.

remote-location-name: Specify the full name of a remote location.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

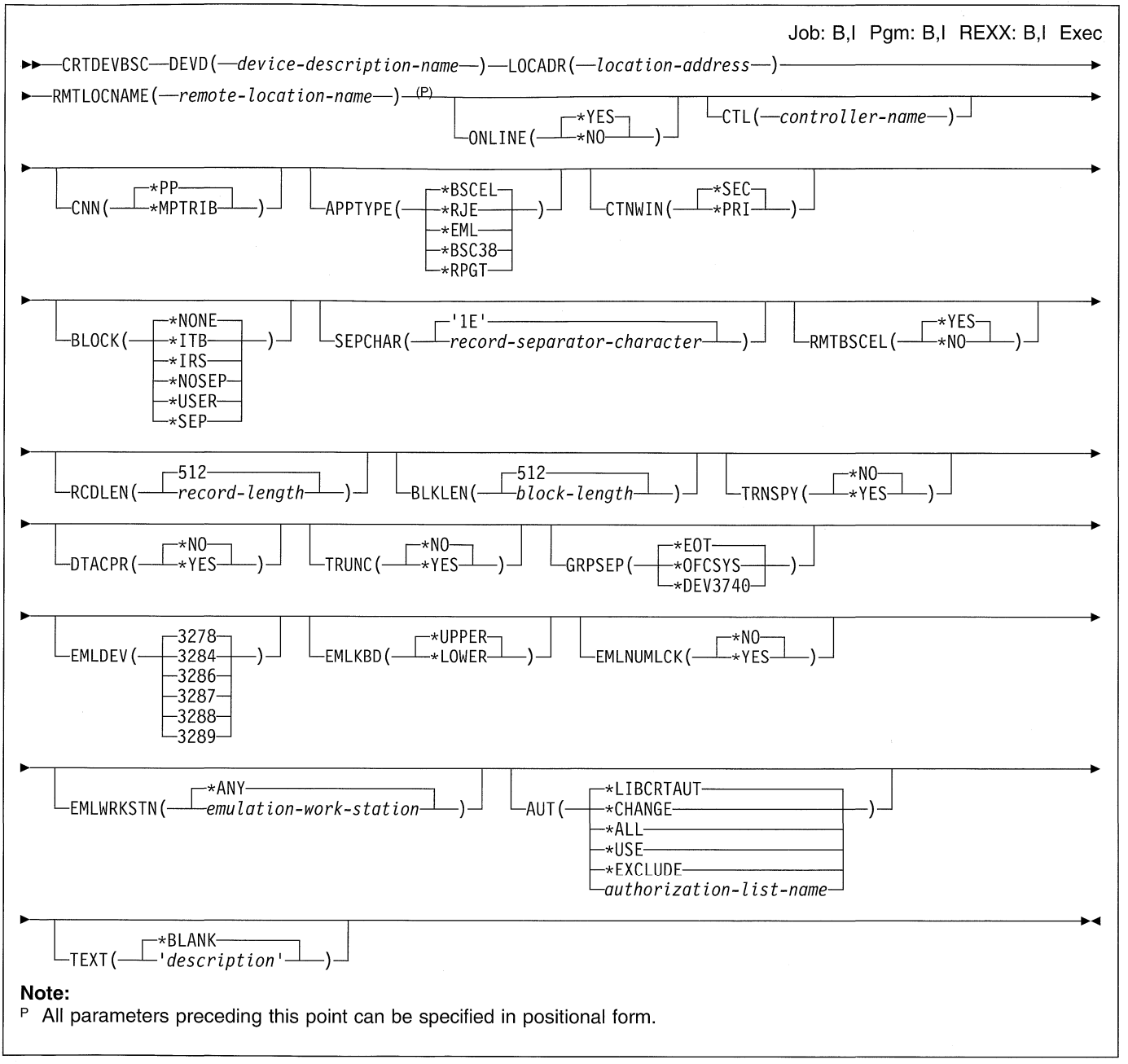
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVASC   DEVD(ASC001)  RMTLOCNAME(NYC)
  CTL(CTLASCNYC)
```

This command creates an asynchronous communications device. The device is located in New York City and is attached to controller CTLASCNYC.

CRTDEVBSC (Create Device Description (BSC)) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (BSC) (CRTDEVBSC) command creates a device description for binary synchronous communications (BSC). More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD
Specifies the name of the device description being created.

LOCADR
Specifies the local location address. Specify a 2-character hexadecimal value ranging from 00 through FE.

RMTLOCNAME
Specifies the remote location name of the system with which this object communicates.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

CNN

Specifies the type of line connection used.

***PP:** A point-to-point line is used.

***MPTRIB:** A multipoint tributary line is used.

APPTYPE

Specifies the application type being used.

***BSCEL:** The application type is Binary Synchronous Communications Equivalence Link (BSCEL). Specify this value to use the intersystem communications function (ICF) support to communicate with other BSC systems or devices. *PGM must be specified on the APPTYPE parameter for the CRTCTLBSC command.

***RJE:** The application type is the BSC Remote Job Entry (RJE). *RJE must be specified on the APPTYPE parameter for the CRTCTLBSC command.

***EML:** The application type is 3270 device emulation, using the 3270 Device Emulation Utility, or the 3270 program interface support provided in the System/38 environment. *EML must be specified on the APPTYPE parameter for the CRTCTLBSC command.

***BSC38:** The application type is a System/38 environment program. This parameter is specified for a device that communicates with other BSC systems or devices when the application uses a BSC device file or a mixed device file in the System/38 environment. *PGM must be specified on the APPTYPE parameter for the CRTCTLBSC command.

***RPGT:** The application uses RPGII with binary synchronous communications adapter (BSCA) devices and telecommunication specifications. *PGM must be specified on the APPTYPE parameter for the CRTCTLBSC command.

Note: This value can also be specified when the application uses ICF support, but does not need Evoke capability or program start request capability. If this value is specified, the value of the RMTBSCEL parameter is automatically set to *NO. If the application uses ICF, make sure there is no requirement for Evoke or program start request support before trying to use this value, or unpredictable results can occur.

CTNWIN

Specifies which BSC station is primary and which is secondary, in order to resolve contention for BSC point-to-point and multipoint lines.

***SEC:** The local AS/400 system is the secondary station, which stops when the other station needs the line.

***PRI:** The local AS/400 system is the primary station.

BLOCK

Specifies whether the system or the user blocks and deblocks transmitted records.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

With this parameter, one of the following conditions of record formatting can be specified:

No blocking/deblocking: The record format described in the data description specifications (DDS) is the format for both the record and the block.

User blocking/deblocking: The user must provide the BSC controls needed to describe the record format to the system.

System blocking of fixed-length records: The system uses fixed-length records, and blocks and deblocks records accordingly.

If the user specifies a parameter value other than *NONE or *USER, records are blocked as required by the system for output and deblocked on input. Blocking can be done with or without record separator characters. If TRNSPY(*YES) is specified, the records can be blocked without record separator characters by specifying BLOCK(*NOSEP), or the records can be transmitted one record at a time by specifying BLOCK(*NONE). By specifying BLOCK(*USER), records can be blocked to include the BSC transparency controls. If TRNSPY(*NO) is specified, all blocking options are valid.

Up to 512 records can be blocked for transmitting. When the system blocks and deblocks the records, record separator characters and control characters are not passed to the program as data.

***NONE:** No blocking or deblocking is done by the system.

***ITB:** The records are blocked or deblocked, based on the location of an ITB (intermediate text block) control character. For input files, a record is delimited by locating the next ITB character, an ETX (end-of-text), or ETB (end-of-transmission block) character. For output files, an ITB character is inserted after each record. If the ITB is the last character of the block, it is replaced by an ETX or an ETB character.

***IRS:** The records are blocked or deblocked, based on the location of an interrecord separator (IRS) character. For input files, a record is delimited by locating the next

IRS character. For output files, an IRS character is inserted after the record.

***NOSEP:** No record separator character is in the transmission block sent to or received from the device. The system blocks and deblocks the records by a fixed record length, as specified in the DDS format specifications, or record length in an RPGII telecommunications program.

***USER:** The program provides all the control characters (including record separator characters, binary synchronous communications (BSC) framing characters, and transparency characters) necessary to send records. More information about the device and binary synchronous communications equivalence link (BSCEL) support characteristics is in the *BSC Equivalence Link Programmer's Guide*.

***SEP:** Records are blocked or deblocked based on the location of a user-specified record separator character. For input files, a record is delimited by locating the next record separator character. For output files, a record separator character is inserted after the record.

If this option is selected, an option must also be specified on the SEPCHAR parameter.

SEPCHAR

Specifies a unique 1-byte record separator character. The record separator character must be specified as two hexadecimal characters.

'1E': The record separator character is '1E'.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

record-separator-character: Specify a record separator that **IS NOT** in the following list of BSC control characters.

EBCDIC	BSC Control
X'01'	SOH (Start of header)
X'02'	STX (Start of text)
X'03'	ETX (End of text)
X'10'	DLE (Data link escape)
X'1D'	IGS (Interchange group separator)
X'1F'	ITB (Intermediate text block)
X'26'	ETB (End-of-transmission block)
X'2D'	ENQ (Enquiry)
X'32'	SYN (Synchronization)
X'37'	EOT (End of transmission)
X'3D'	NAK (Negative acknowledgment)

RMTBSCEL

Specifies the type of BSCEL session with the remote system.

***YES:** The remote system can recognize BSCEL start and end commands, and BSCEL online messages.

***NO:** The remote system or device cannot recognize BSCEL commands and BSCEL online messages, but ICF support is desired.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

RCDLEN

Specifies the maximum record length allowed when communicating with this device. The maximum value for this parameter is 8192. The value must not exceed the block length value for this device and must not exceed the buffer size specified on the line description (MAXBUFFER) to which this device is attached.

512: The record length is 512.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

record-length: Specify a value (in bytes) that is at least the size of the largest record to be sent. Valid values range from 1 through 8192.

BLKLEN

Specifies the maximum block length allowed when communicating with this device. The maximum value for this parameter is 8192. This value must not exceed the buffer size specified on the line description (MAXBUFFER) to which this device is attached.

512: The block length is 512.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

block-length: Specify the maximum block length (in bytes) of records sent. The value must be at least the size of the largest record sent. Valid values range from 1 through 32767.

TRNSPY

Specifies whether text transparency is used when sending records. Text transparency permits the transmission of all 256 EBCDIC character codes. Use this feature when transmitting packed or binary data fields. The TRNSPY parameter is ignored if BLOCK(*USER) is specified. TRNSPY(*YES) cannot be specified if TRUNC(*YES) or DTACPR(*YES) is specified.

***NO:** Text transparency is not used.

***YES:** Text transparency is used that permits the transmission of all 256 EBCDIC character codes. *YES is a valid value only when BLOCK(*NONE), BLOCK(*NOSEP), or BLOCK(*USER) is specified. TRNSPY(*YES) is invalid if the line description specifies CODE(*ASCII).

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

DTACPR

Specifies whether data compression is performed.

Note: DTACPR(*YES) cannot be specified if TRNSPY(*YES) or TRUNC(*YES) is specified.

***NO:** No data compression or decompression occurs.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

***YES:** Data is compressed for output and decompressed for input.

TRUNC

Specifies whether trailing blanks are removed from output records. TRUNC(*YES) cannot be specified if BLOCK(*NOSEP), TRNSPY(*YES), or DTACPR(*YES) is specified. If BLOCK(*USER) is specified, the TRUNC parameter is ignored and trailing blanks are always removed from output records.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

***NO:** Trailing blanks are not removed from output records.

***YES:** Trailing blanks are removed from output records.

GRPSEP

Specifies a separator for groups of data, such as data sets and documents.

Note: This parameter is valid only when APPTYPE(*BSCEL) or APPTYPE(*RPGT) is specified.

***EOT:** An end of transmission (EOT) control character is used.

***OFCSYS:** A transmission block ending with an end of text (ETX) control character is used.

***DEV3740:** A null record (STX ETX) is used.

EMLDEV

Specifies the type of 3270 device being emulated. This parameter is valid only if *EML was specified on the APPTYPE parameter when the device was created. The change requested by this command becomes effective the next time either the Start 3270 Display Emulation (STREML3270) command or the Start Print Emulation (STRPRTEML) command accesses the device.

3278: This device description is used to emulate a 3278 display device.

3284: This device description is used to emulate a 3284 printer device.

3286: This device description is used to emulate a 3286 printer device.

3287: This device description is used to emulate a 3287 printer device.

3288: This device description is used to emulate a 3288 printer device.

3289: This device description is used to emulate a 3289 printer device.

EMLKBD

Specifies, only for 3270 device emulation applications, the type of 3278 display keyboard to be emulated. The specified value is used if EMLDEV(3278) was specified.

***UPPER:** A 3270 display device keyboard is emulated with uppercase characters only.

***LOWER:** A 3270 display device keyboard is emulated with uppercase and lowercase characters.

EMLNUMLCK

Specifies, only for 3270 device emulation applications, whether the numeric shift lock is set automatically for numeric input fields received from the host system. This parameter is for data entry keyboards only.

***NO:** 3270 emulation allows any data to be typed in the numeric input fields.

***YES:** 3270 emulation allows only numeric data to be typed in the numeric input fields. Numeric data that can be typed include the characters 0 through 9, and symbols + - , . and blank.

EMLWRKSTN

Specifies, only for 3270 device emulation applications, the device address of the emulation work station associated with an emulation device with a real display or printer device. The device address is reserved for use exclusively by that work station. If no device or *ANY is specified, any work station can use the emulation device.

***ANY:** Any work station can use the emulation device.

emulation-work-station: Specify the device address for the work station using this emulation device.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

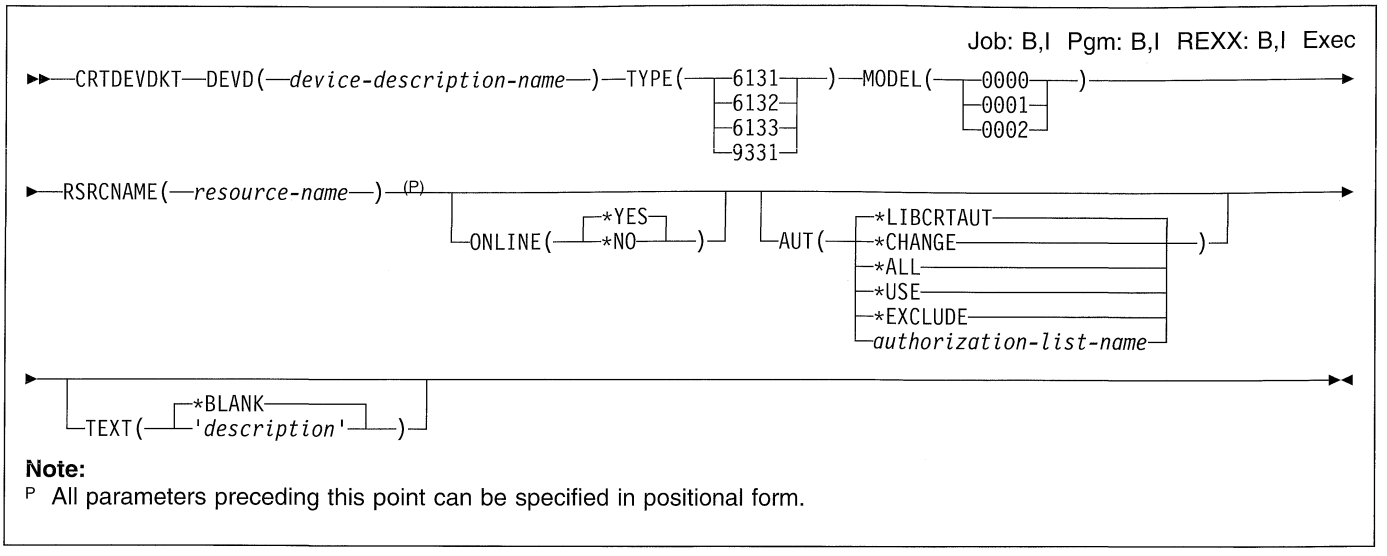
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVBSC   DEVD(BSC001)  LOCADR(27)
  RMTLOCNAME(BSC001LC)  CTL(CTLBSC001)  CNN(*PP)
  APPTYPE(*RJE)
```

This command creates a binary synchronous communications device named BSC001. Its address is X'27' and it is attached to the controller CTLBSC001. The connection type of the device is *PP and it uses the RJE application.

CRTDEVDKT (Create Device Description (Diskette)) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Diskette) (CRTDEVDKT) command creates a device description for a diskette device. More information about using this command is in the *Device Configuration Guide*.

Required Parameters

DEVD
Specifies the name of the device description being created.

TYPE
Specifies the type of device this description represents.

6131: The device type is 6131.

6132: The device type is 6132.

6133: The device type is 6133.

9331: The device type is 9331.

MODEL
Specifies the model number of the device for this description.

0000: This model number is for a 6131, 6132, or 6133 device type.

0001: This model number is for a 9331 device type only.

0002: This model number is for a 9331 device type only.

RSRCNAME
Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *STG specified on the TYPE parameter to determine the resource name.

Optional Parameters

ONLINE
Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

AUT
Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

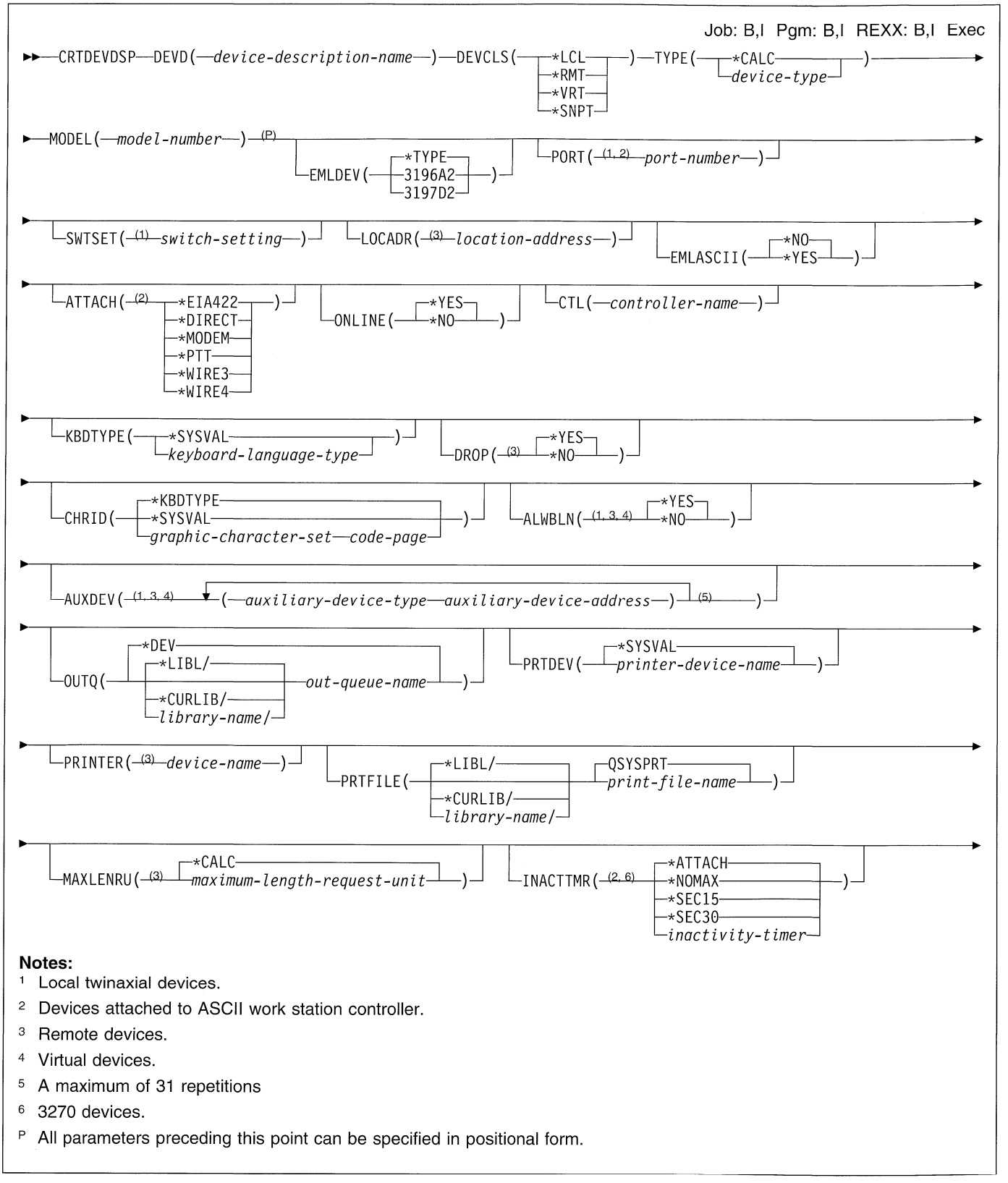
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVDKT   DEVD(DKT01)  TYPE(9331)  MODEL(0002)
  RSRCNAME(DKT01)
```

This command creates a device description for a 9331 Model 0002 diskette drive. The system resources manager (SRM) has assigned a resource name of DKT01.

CRTDEVDSP (Create Device Description (Display)) Command



Notes:

1 Local twinaxial devices.

2 Devices attached to ASCII work station controller.

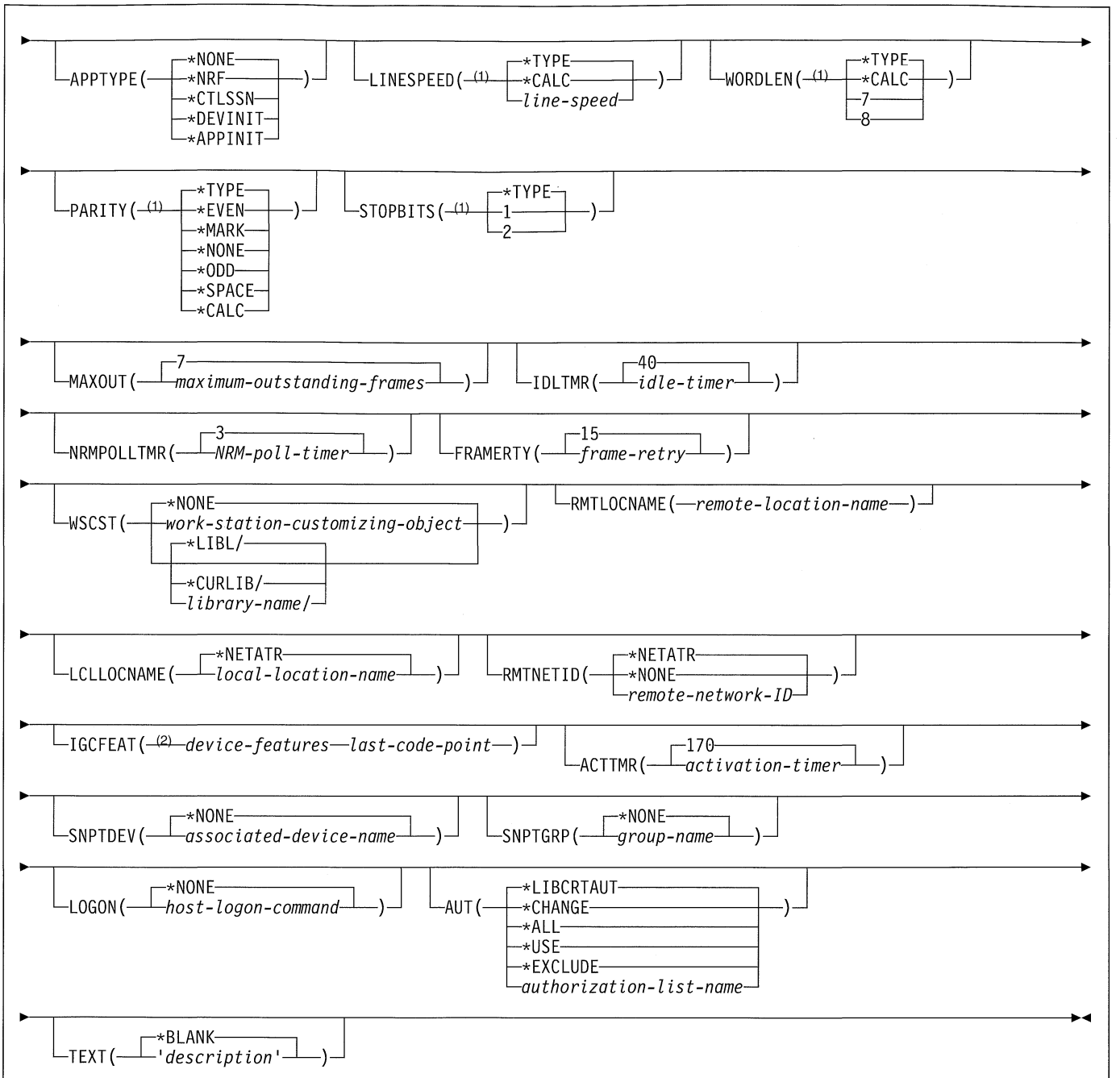
3 Remote devices.

4 Virtual devices.

5 A maximum of 31 repetitions

6 3270 devices.

P All parameters preceding this point can be specified in positional form.



Notes:

1 Devices attached to ASCII work station controller.

2 DBCS systems.

Purpose

The Create Device Description (Display) (CRTDEVDSP) command creates a device description for a display device. More information about using this command is in the *Device Configuration Guide* for local display stations and in the *OS/400* Communications Configuration Reference* for remote and virtual display stations.

Restrictions: If the display device is created as a shared-port device (*CALC is specified on TYPE parameter and/or the LINESPEED, WORDLEN, and PARITY parameters), the device cannot be changed (with the CHGDEVDSP command) to a nonshared-port device. Conversely, if the device is created as a non-port shared device, the device cannot be changed to a shared-port device. Additional information on sharing ports is in the *OS/400* Communications Configuration Reference* and the *Device Configuration Guide*.

Required Parameters

DEVD

Specifies the name of the device description being created.

DEVCLS

Specifies the device class for this display station.

***LCL:** This device description is for a display station connected to a local work station controller.

***RMT:** This device description is for a display station connected to a remote work station controller.

***VRT:** This device description is for a virtual display station. Through a virtual display station, users can access (pass through to) their own system from a remote system. A virtual display station allows the remote system or a personal computer to emulate the interface of the user's own system.

***SNPT:** This device description is for a display station connected to an SNA pass-through advanced program-to-program communications (APPC) controller. SNA pass-through support allows the user to connect this display station with host logical unit (LU) types 0 through 3 applications.

TYPE

Specifies the type of device this description represents.

***CALC:** The system determines the value to use.

device-type: Specify one of the following 4-character type codes that describes this type of device.

- 3179
- 3180
- 3196
- 3197
- 3277
- 3278
- 3279
- 3476
- 3477
- 3486
- 3487
- 5150 (any display station attached by a TDLC link)
- 5251
- 5291
- 5292
- 5555 (on systems supporting DBCS (double-byte character set))
- *NVT (TCP/IP TELNET Network Virtual Terminal)

The following type codes are valid for ASCII devices:

- 3101
- 3151
- 3161
- 3162
- 3163
- 3164
- D220 (Data General Dasher** D220)
- T910 (TeleVideo** 910)
- T925 (TeleVideo 925)
- T955 (TeleVideo 955)
- V100 (DEC** VT-100)
- V220 (DEC VT-220)
- W30 (Wyse** WY30**)
- W50 (Wyse WY50**)
- W60 (Wyse WY60**)
- *CALC

Note: The value *CALC allows ASCII devices to share a port. ASCII port sharing allows a port to be used by different ASCII device types with different parameters without being manually reconfigured.

The DEC VT-100 can be configured with a device class of *VRT.

MODEL

Specifies the model number of the device for this description. Specify one of the following model numbers for the device.

Type	Model
3179	2
3180	2
3196	A1, A2, B1, B2
3197	C1, C2, D1, D2, W1, W2
3277	0, *DHCF
3278	0, 4, *DHCF
3279	0, *DHCF
3476	EA, EC
3477	FA, FC, FD, FE, FG, and FW
3486	BA
3487	HA, HC, HG, and HW
5150	1, 2, A1
5251	11
5291	1, 2
5292	1 Color
5292	2 Graphics
5555	B01, E01, C01, F01, G01, G02
*NVT	0000

For ASCII devices only:

***ASCII:** Specify this value if the model number cannot be easily determined for this display station. The system assigns a model number (if the device has a model number). The model number (number assigned by the system) may not be the actual model number of the user's display station. It is recommended that the user determine the display station's model number if at all possible, and enter that number.

The possible choices for device model for each device type are as follows:

Type	Model
3101	23
3151	11(*ASCII), 31, 41
3161	11(*ASCII), 12
3162	11(*ASCII), 12, 31, 32

3163	11(*ASCII), 12
3164	11(*ASCII), 12

Type	Model
D220 (Data General Dasher D220)	*ASCII
T910 (TeleVideo 910)	*ASCII
T925 (TeleVideo 925)	*ASCII
T955 (TeleVideo 955)	*ASCII
V100 (DEC VT-100)	*ASCII
V220 (DEC VT-220)	*ASCII
W30 (Wyse WY30)	*ASCII
W50 (Wyse WY50)	*ASCII
W60 (Wyse WY60)	*ASCII
*CALC	*ASCII

Optional Parameters

EMLDEV

Specifies the twinaxial display emulation used by an IBM 3151 or IBM 3162 display. The available emulations are 3196A2, which permits an 80-column display, and 3197D2, which permits a 132-column display.

Note: 3197D2 emulation for an IBM 3151 requires the installation of a "cartridge for expansion" in addition to the appropriate setting of this parameter.

***TYPE:** The default value for this parameter is 3196A2.

3196A2: An 80-column display results.

3197D2: A 132-column display results.

PORT

Specifies, for local display stations, the port number for this device. Valid values range from 0 through 17.

For display stations attached to the ASCII work station controller, valid values range from 0 through 17 and indicate which port of the ASCII work station controller this display station is attached to. Without the 12-port expansion feature, only ports 0 through 5 are valid. With the 12-port expansion feature, ports 6 through 17 are added.

SWTSET

Specifies for local twinaxial display stations the switch setting or device address for this device. Valid values range from 0 through 6.

LOCADR

Specifies the local location address. Specify a 2-character hexadecimal value ranging from 00 through FE.

EMLASCII

Specifies, for ASCII devices, whether the device being configured is emulating a supported ASCII device type (TYPE parameter). When an ASCII device is configured, choices for the following parameters are restricted to the valid range for that device:

ATTACH
LINESPEED
WORDLEN
PARITY
STOPBITS

These restrictions may not apply to ASCII devices emulating supported ASCII device types because the emulating device may have a wider range of choices for these parameters than the emulated ASCII device.

When EMLASCII(*YES) is specified, these parameter restrictions are not enforced. The user is allowed to enter the full range of values for these parameters, but is also responsible for verifying that the values selected are valid for the device being configured.

***NO:** The device being configured is one of the supported ASCII device types (TYPE parameter), and the device-specific restrictions for the physical attachment, line speed, word length, and stop bits parameters are enforced.

***YES:** The device being configured is emulating one of the supported ASCII device types (TYPE parameter), and the device-specific restrictions for the physical attachment, line speed, word length, and stop bits parameters are not enforced.

ATTACH

Specifies the physical attachment of the display station to the ASCII work station controller. This is a required parameter for displays attached to an ASCII work station controller.

***EIA422:** The EIA-422 attachment (valid only for models 3101, 3151, 3161, 3162, 3163, and 3164) is used.

***DIRECT:** The EIA-232 direct attachment is used.

***MODEM:** The EIA-232 modem attachment is used.

***PTT:** The Post Telephone and Telegraph (PTT) attachment is used.

***WIRE3:** The EIA-232 3-wire attachment is used.

***WIRE4:** The EIA-232 4-wire attachment is used.

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

KBDTYPE

Specifies the country keyboard language identifier for this display station.

***SYSVAL:** The system uses the QKBDTYPE system value.

keyboard-language-type: Specify the 3-character country identifier (used for EBCDIC and ASCII) for this display station.

Table 28 shows the valid identifiers and the language the identifier represents. In addition, Table 28 shows the ASCII device groups, if applicable, for each language. The device group can be used to determine whether a country keyboard language identifier can be used with an ASCII device. Valid display devices and the associated ASCII device group are shown in Table 29. These two tables can be used to determine whether an ASCII display device can be used with a specified language or if an ASCII device can be used at all.

For example, assume a user wants to create or change a 3101 display device. Table 29 shows that a 3101 display supports ASCII device group A. Table 28 shows that the valid language identifiers that can be used with device group A include AGB, AGI, CAB, CAI, FAB, FAI, ITB, ITI, UKB, UKI, USB, and USI.

Table 28. Keyboard Mapping Table

Language/Country	Identifier	ASCII Device Group
Arabic X/Basic	CLB	D
Austria/Germany	AGB	A, B
Austria/Germany Multinational	AGI	A, B
Belgium Multinational	BLI	B
Brazilian Portuguese	BRB	
Canadian French	CAB	A, B
Canadian French Multinational	CAI	A, B
Cyrillic	CYB	
Denmark	DMB	B
Denmark Multinational	DMI	B
Finland/Sweden	FNB	B
Finland/Sweden Multinational	FNI	B
France (Azerty)	FAB	A, B
France (Azerty) Multinational	FAI	A, B
France (Qwerty)	FQB	
France (Qwerty) Multinational	FQI	
Greece	GNB	
Greece	GKB	
Hebrew	NCB	D
Iceland	ICB	
Iceland Multinational	ICI	
International	INB	
International Multinational	INI	
Italy	ITB	A, B
Italy Multinational	ITI	A, B
Japan English	JEB	
Japan English Multinational	JEI	
Japan Kanji (For PS/55* and 5295 display stations)	JKB	
Japan United States Basic	JUB	
Japan Katakana (For 5251, 5291, 5292, and 3180 Katakana display stations)	KAB	
Korea	KOB	
Latin-2/ROECE	ROB	
Netherlands	NEB	
Netherlands Multinational	NEI	
Norway	NWB	B
Norway Multinational	NWI	B
Portugal	PRB	B
Portugal Multinational	PRI	B

Table 28. Keyboard Mapping Table

Language/Country	Identifier	ASCII Device Group
Simplified Chinese	RCB	
Spain	SPB	B
Spain Multinational	SPI	B
Spanish Speaking	SSB	B
Spanish Speaking Multinational	SSI	B
Sweden	SWB	B
Sweden Multinational	SWI	B
Switzerland/French Multinational	SFI	B
Switzerland/German Multinational	SGI	B
Thai	THB	
Traditional Chinese	TAB	
Turkey	TKB	
United Kingdom	UKB	A, B
United Kingdom Multinational	UKI	A, B
United States/Canada	USB	A, B, C
United States/Canada Multinational	USI	A, B, C
Languages of the former Yugoslavia	YGI	

Table 29 shows the ASCII device groups that are supported for each ASCII display. Refer to Table 28 to determine which languages can be used with the ASCII device group. Note the ASCII device groups to which each language belongs and, in Table 29, note the display stations that support each ASCII device group:

Table 29. Display/Device Group Table

ASCII Display		ASCII Device Group
3101		A
3151		B
3161		B
3162		B
3163		B
3164		B
D220	(Data General Dasher** D220)	C
T910	(TeleVideo** 910)	C
T925	(TeleVideo 925)	C
T955	(TeleVideo 955)	C
V100	(DEC** VT-100)	C
V220	(DEC VT-220)	C, D
W30	(Wyse** WY30**)	C
W50	(Wyse WY50**)	C
W60	(Wyse WY60**)	C

Notes:

1. The following values (*SYSVAL, AGI, BLI, CAI, DMI, FNI, FAI, ITI, NWI, PRI, SPI, SSI, SWI, USB, or USI) are also allowed when DEVCLS(*RMT) is specified and when TYPE(3277), TYPE(3278), or TYPE(3279) is specified. Otherwise, this parameter is not valid when DEVCLS(*RMT) is specified.
2. This parameter is optional for the combination of DEVCLS(*LCL) and TYPE(5150).
3. This parameter is not valid for DEVCLS(*RMT) except as noted in the above note.

DROP

Specifies, for remote display stations, whether the line is disconnected by the system when all display stations on the line are no longer in use. When multiple display stations are attached to the same controller, the line is disconnected only if: (1) the device description for this device specifies DROP (*YES), or DROP (*YES) is specified on the SIGNOFF command when the user signs off the device; (2) all of the other display stations connected to the controller have signed off or are not in use; and (3) all printers attached to the controller are not in use.

The value specified in the device description can be overridden by a user signing off at the device if the DROP parameter on the SIGNOFF command is used.

***YES:** The switched line to the controller to which this device is attached is disconnected when this device and all other attached devices are no longer in use.

***NO:** The switched line is not disconnected from the controller when all of its attached devices are no longer in use.

CHRID

Specifies the character identifier (graphic character set and code page) that a work station display device supports. When a display file that was created with the CHRID DDS keyword is used with the device, the system converts data sent to and received from the device to ensure that the correct characters are shown and that the correct hexadecimal byte values are returned to the application program. More information about display file CHRID processing and the translation tables that are used to convert data sent to and received from the display are in the *Guide to Programming Displays*.

The CHRID value specified must be based on the attributes of the display device. Table 12 shows CHRID values that are appropriate for each work station display keyboard type. For some display devices, it is not necessary to specify the KBDTYPE parameter, but the KBDTYPE value for the equivalent keyboard can be used to determine the CHRID value for the device.

All characters included in the graphic character sets listed for the *Limited CHRID* values can be entered at the keyboard or displayed by the device. All characters associated with the *Full CHRID* values can be entered at the display station (including the use of hexadecimal representations), but the device may not be able to display all the characters.

Values shown in the *Limited CHRID* column must be used for 5291 and 5292 display stations; all other display stations must use the value shown in the *Full CHRID* column. All display stations that are to be used for OfficeVision/400, including 5291 and 5292 display stations, must use the value in the *Full CHRID* column.

Note: When using keyboard type values (KBDTYPE parameter) for devices that support both single-byte and double-byte data (JKB, KOB, RCB, and TAB), the CHRID values listed are for single-byte data only.

***KBDTYPE:** The system determines the best graphic character set and code page for the keyboard specified on the KBDTYPE parameter.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values that match the attributes of the display device. Valid values range from 1 through 32767.

Element 2: Code Page

code-page: Specify the code page set values that match the attributes of the display device. Valid values range from 1 through 32767.

ALWBLN

Specifies whether the (program controlled) blinking cursor is used. For 3179, 3180, 3196, 3197, 3476, 3477, 3486, 3487, 5251, 5291, and 5292 display devices, allowing the cursor to blink may distract the operator.

***YES:** Allows the cursor to blink for the 3179, 3180, 3196, 3197, 3476, 3477, 3486, 3487, 5251, 5252, 5291, and 5292 display devices.

***NO:** The blinking cursor is suppressed.

Note: For displays that have the capability of changing the blinking cursor attribute using the keyboard setup feature, the value for ALWBLN in the device description can be overridden.

AUXDEV

Specifies the device type and address of an additional device (if any) that is attached to the IEEE-488 port on the 5292 Model 2 device (or a PS/2 emulating a 5292 Model 2 graphics display). Up to 31 plotters can be attached to the same IEEE-488 AUXDEV port on the 5292 Model 2, but they must have different IEEE-488 addresses.

Element 1: Auxiliary Device Types

auxiliary-device-type: Specify the auxiliary device (IBM plotter) that is attached. Valid plotters include the following:

7371	6180	6184	6186-1
7372	6182	6185	6186-2

Element 2: Auxiliary Device Address

auxiliary-device-address: Specify the address ranging from 1 through 31 of the auxiliary device (plotter). Each plotter must have a unique address.

OUTQ

Specifies the qualified name of the output queue used by this user. The output queue must already exist.

***DEV:** The output queue specified on the PRTDEV parameter is used.

The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

output-queue-name: Specify the name of the output queue.

PRTDEV

Specifies the qualified name of the default printer device for this job. If OUTQ(*DEV) is specified, the file is placed on an output queue with the same name as the printer.

***SYSVAL:** The value specified in the system value QPRTDEV is used.

printer-device-name: Specify the name of a printer that is used to print the output for this user.

PRINTER

Specifies, for a remote display station, the device name of the printer associated with the display device. The device description of the work station printer named in this parameter must have already been created and must currently exist on the system. Both the printer and display station must be attached to the same controller. The relationship created by this parameter is used when a related printer (PRINT keyword in DDS) is referred to in a device file used to access this display station.

PRTFILE

Specifies an alternative printer device file used for processing the Print key on this display station.

The name of the printer file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QSYSPRT: If a file name is not specified, the precompiler printout is directed to the IBM-supplied printer file QSYSPRT.

print-file-name: Specify the name of the printer device file that performs Print key processing for this display station.

MAXLENRU

Specifies the maximum request unit (RU) length (in bytes) allowed.

***CALC:** The system determines the value to use.

maximum-length-request-unit: Specify the maximum length for incoming request units. Values 241 and 256 are valid only for X.25 devices.

INACTTMR

Specifies an inactivity timer (time-out) value for display devices. This parameter also specifies what happens when the time-out value is exceeded, dependent on other attributes of the device:

- For display stations attached to an ASCII work station controller, the user's job is canceled when the display station is inactive (no data is sent or received) for a period of time that exceeds the time-out value. The display station is automatically varied off and on again, resulting in a new sign-on display.
- For display devices connected using SNA pass-through (SNPT) support, the user is informed by a message to QSYSOPR and the session is ended when the amount of time that the device is not bound to a host application exceeds the time-out value. The user must reestablish the connection and session.
- For display devices with an application type value of *APPINIT, *DEVINIT, or *NRF, the session is ended when the device is inactive (the file opened against the device is closed and no additional requests to open files are received for the device) for a period of time that exceeds the time-out value.

Note: This timer is not used by devices allocated to a subsystem (normal interactive use) because the subsystem always has a file open for the device. The timer is used by batch jobs that open and close files for the device.

For a Post Telephone and Telegraph (*PTT) attachment, the valid values are *SEC15, *SEC30, and from 1 through 10 minutes. The default of *ATTACH maps to *SEC30 (30 seconds).

For a connection using SNA pass-through (*SNPT) device class support, the default of *ATTACH maps to *NOMAX.

For a device with an application type value of *APPINIT, *CLTSSN, *DEVINIT, or *NRF, the default of *ATTACH maps to 1 minute.

For all other attachments, valid values range from 1 through 30 minutes. The default of *ATTACH has a value of *NOMAX.

***ATTACH:** This value varies by the value specified on the physical attachment (ATTACH) parameter and certain values on the application type (APPTYPE) and device class (DEVCLS) parameters.

***NOMAX:** Maximum inactivity time is not tracked.

***SEC15:** A 15-second time-out period is used.

***SEC30:** A 30-second time-out period is used.

inactivity-timer: Specify a time-out value.

APPTYPE

Specifies the application type being used.

***NONE:** The device is not used for any application.

***NRF:** The device is used for the Network Routing Facility application.

| ***CTLSSN:** The device controls the sessions with
| *DEVINIT devices.

| ***DEVINIT:** The device starts (initiates) the session.

| ***APPINIT:** The application program starts (initiates) the
| session.

LINESPEED

Specifies, for display stations attached to the ASCII work station controller, the line speed (in bits per second) used to communicate over the interface (attachment) between the ASCII work station controller and the display station.

***TYPE:** The system uses the suggested setting for this device type. If *CALC is specified on the TYPE parameter, *TYPE defaults to 1200.

***CALC:** The system determines the value to use.

line-speed: Specify a line speed.

WORDLEN

Specifies, for display stations attached to an ASCII work station controller, the word length (in bits per character) used to communicate with this display station. All display stations support both 7-bit and 8-bit word lengths except for the 3101 and the D220, which only supports 7-bit word lengths. *TYPE specifies 8-bit word lengths for all displays stations, except for the 3101 and the D220 for which 7-bit word lengths are specified.

For modem and PTT attachments (ATTACH(*MODEM) or ATTACH(*PTT)), the word length specified must be the word length selected for the modem.
If *CALC is specified on the TYPE parameter, WORDLEN(*TYPE) defaults to eight.

***TYPE:** The system uses the suggested setting for this display station.

***CALC:** The system determines the value to use.

7: 7-bit word lengths are used.

8: 8-bit word lengths are used.

PARITY

Specifies, for display stations attached to an ASCII work station controller, the type of parity used to communicate with this display station. *TYPE specifies *EVEN for all display stations except for D220; *TYPE specifies *MARK for the D220.

For modem and PTT attachments (ATTACH(*MODEM) or ATTACH(*PTT)), the parity specified must be the parity selected for the modem. *TYPE specifies *EVEN parity for all display stations except the D220. For the D220, *TYPE specifies *MARK. If *CALC is specified on the TYPE parameter, PARITY(*TYPE) defaults to even parity.

***TYPE:** The system uses the suggested setting for this display station.

***EVEN:** Even parity is used.

***MARK:** Mark parity is used.

***NONE:** No parity bit is used.

***ODD:** Odd parity is used.

***SPACE:** Space parity is used.

***CALC:** The system determines the value to use.

STOPBITS

Specifies, for display stations attached to an ASCII work station controller, the number of stop bits used to communicate with this display station. All display stations support both 1 and 2 stop bits except for the DEC VT-100, which only supports 2, and the D220, which supports only 1. *TYPE specifies 1 stop bit for all display stations except DEC VT-100 (which is 2). For modem and PTT attachments (ATTACH(*MODEM) or ATTACH(*PTT)), the stop bits specified must be the stop bits selected for the modem.
If *CALC is specified on the TYPE parameter but not on the LINESPEED parameter, STOPBITS(*TYPE) defaults to one.

***TYPE:** The system uses the suggested setting for this display station.

1: One stop bit is used.

2: Two stop bits are used.

MAXOUT

Specifies the maximum number of frames that are sent sequentially to a remote system before the remote system (the 5150 work station) must respond.

This parameter is valid only if TYPE(5150) MODEL(A1) is specified.

7: The frame is sent seven times.

maximum-outstanding-frames: Specify a value ranging from 1 through 7 for the maximum number of frames sent before the remote system must respond.

IDLTMR

Specifies the time (in 0.1 second intervals) that the system waits for a response. If no response is received, error recovery procedures are started. This parameter is valid only if TYPE(5150) MODEL(A1) is specified.

40: The system waits 4.0 seconds.

idle-timer: Specify a value ranging from 10 through 250 in 0.1-second intervals.

NRMPOLLTMR

Specifies the interval (in 0.1 second intervals) for polling this device when it is in normal response mode (NRM).

3: Normal response mode polling is done at 0.3-second intervals.

NRM-poll-timer: Specify a value ranging from 2 through 100 in 0.1-second intervals.

This parameter is valid only if TYPE(5150) MODEL(A1) is specified.

FRAMERTY

Specifies the number of retries for an unanswered command frame or unacknowledged information frame. This parameter is valid only if TYPE(5150) MODEL(A1) is specified.

15: The default value is 15 retries.

frame-retry: Specify a value ranging from 5 through 64 for the number of retries.

WSCST

Specifies the qualified name of a work station customizing object to be created.

***NONE:** No work station customizing object is specified.

work-station-customizing-object: Specify the work station customizing object.

Note: If a work station customizing object is specified for the WSCST parameter, all country keyboard identifiers are valid for ASCII devices except for the following: FQB, FQI, INB, INI, JEB, JEI, JKB, JUB, KAB, KOB, RCB, and TAB. See Table 28 for a list of the country keyboard identifiers.

The name of the work station customizing object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

RMTLOCNAME

Specifies the remote location name of the system with which this object communicates.

Note: This parameter is required for APPTYPE(*APPINIT) devices. The remote location name for APPTYPE(*APPINIT) devices is the VTAM/NCP (Virtual Telecommunications Access Method/Network Control Program) name of the physical device.

LCLLOCNAME

Specifies the local location name. When this parameter is specified with APPTYPE(*CTLSSN) or APPTYPE(*APPINIT), the remote location name is the name of the independent logical unit (LU) in the network control program (NCP).

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the local location name.

RMTNETID

Specifies the name of the remote network identifier (ID). This parameter can be specified for APPTYPE(*APPINIT) devices.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

remote-network-ID: Specify the name of the remote network identifier.

IGCFEAT

Specifies which double-byte character set (DBCS) table is used, according to the device and language. This parameter is valid only for DBCS-capable devices.

Element 1: Features of the DBCS-Capable Device

device-features: Specify the device features from Table 31.

Element 2: Last Code Point

last-code-point: Specify the last code point from Table 31.

ACTTMR

Specifies, for switched lines, the amount of time (in seconds) that the SNA pass-through support waits for the device to respond to the activation request from the host AS/400 system. If the device does not respond within this time, it is considered not available.

170: Specifies the activate time of 170 seconds.

activation-timer: Specify a number ranging from 1 through 2550 indicating the number of seconds before the device is considered not available.

SNPTDEV

Specifies the name of the associated SNA pass-through device that is attached to a host or advanced program-to-program communications (APPC) controller.

***NONE:** No name is specified.

associated-device-name: Specify the name of a device that is attached to a host or an APPC controller that is associated with this device.

SNPTGRP

Specifies the name configured for a group of host devices in a configuration list. This indicates that this device is associated with any one of the devices in that group which is available.

***NONE:** No name is specified.

group-name: Specify the name configured for a group of host devices that must be associated with this device.

LOGON

Specifies the sign-on (logon) text. This parameter is allowed when DEVCLS(*SNPT) or APPTYPE(*NRF) is specified. APPTYPE(*NRF) specifies the logon string that is sent to the host system when a request is made to establish a session. DEVCLS(*SNPT) specifies the sign-on text that is sent to the host system after starting SNA pass-through support.

Specifies the logon string that is sent to the system services control point (SSCP) on the host network when the file is opened.

***NONE:** No text is sent to the host system.

host-logon-command: Specify text that is sent to the host system. The text must be enclosed in apostrophes if it contains blanks or other special characters. All apostrophes within the text must be represented by two apostrophes. A maximum of 256 characters can be specified.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Table 30 (Page 1 of 2). Character Identifier Values for Display Devices (CHRID Parameter)

Language/Country	Country Keyboard Language Identifier (KDBTYPE)	Limited CHRID	Full CHRID
International and US ASCII	INB	103 038	697 500
Multinational	AGI BLI CAI DMI FAI FNI FQI ICI INI ITI JEI NEI NWI PRI SFI SGI SPI SSI SWI UKI USI		697 500
Arabic	CLB		235 420
Austria/Germany	AGB	265 273	697 273
Belgium Multinational	BLI		697 500
Brazilian Portuguese	BRB		697 037
Canada/French	CAB	277 260	341 260
Cyrillic	CYB		960 880
Denmark/Norway	DMB NWB	281 277	697 277
Finland/Sweden	FNB SWB	285 278	697 278
France	FAB FQB	288 297	697 297
Greece	GNB GKB		925 875
Hebrew	NCB		941 424
Iceland	ICB		697 871
Italy	ITB	293 280	697 280
Japan/English	JEB	297 281	697 281
Japan/Kanji	JKB (For Personal System/55*, 5295 and 3477-J display stations)		1172 290
Japan/Katakana	KAB (For 5251, 5291, 5292, and 3180 Katakana display stations)		332 290
Korean	KOB		1173 833
Latin 2	ROB		959 870

Table 30 (Page 2 of 2). Character Identifier Values for Display Devices (CHRID Parameter)

Language/Country	Country Keyboard Language Identifier (KDBTYPE)	Limited CHRID	Full CHRID
Netherlands	NEB		697 037
Portugal	PRB	301 037	697 037
Simplified Chinese	RCB		1174 836
Spain	SPB	305 284	697 284
Spanish Speaking	SSB	309 284	697 284
Switzerland/French Multinational	SFI		697 500
Switzerland/German Multinational	SGI		697 500
Thai	THB		938 838
Traditional Chinese	TAB		101 037
Turkey	TKB		1152 1026
United Kingdom/English	UKB	313 285	697 285
United States/English	USB	101 037	697 037
Languages of the former Yugoslavia	YGI		959 870

Table 31. DBCS-Capable Device Features

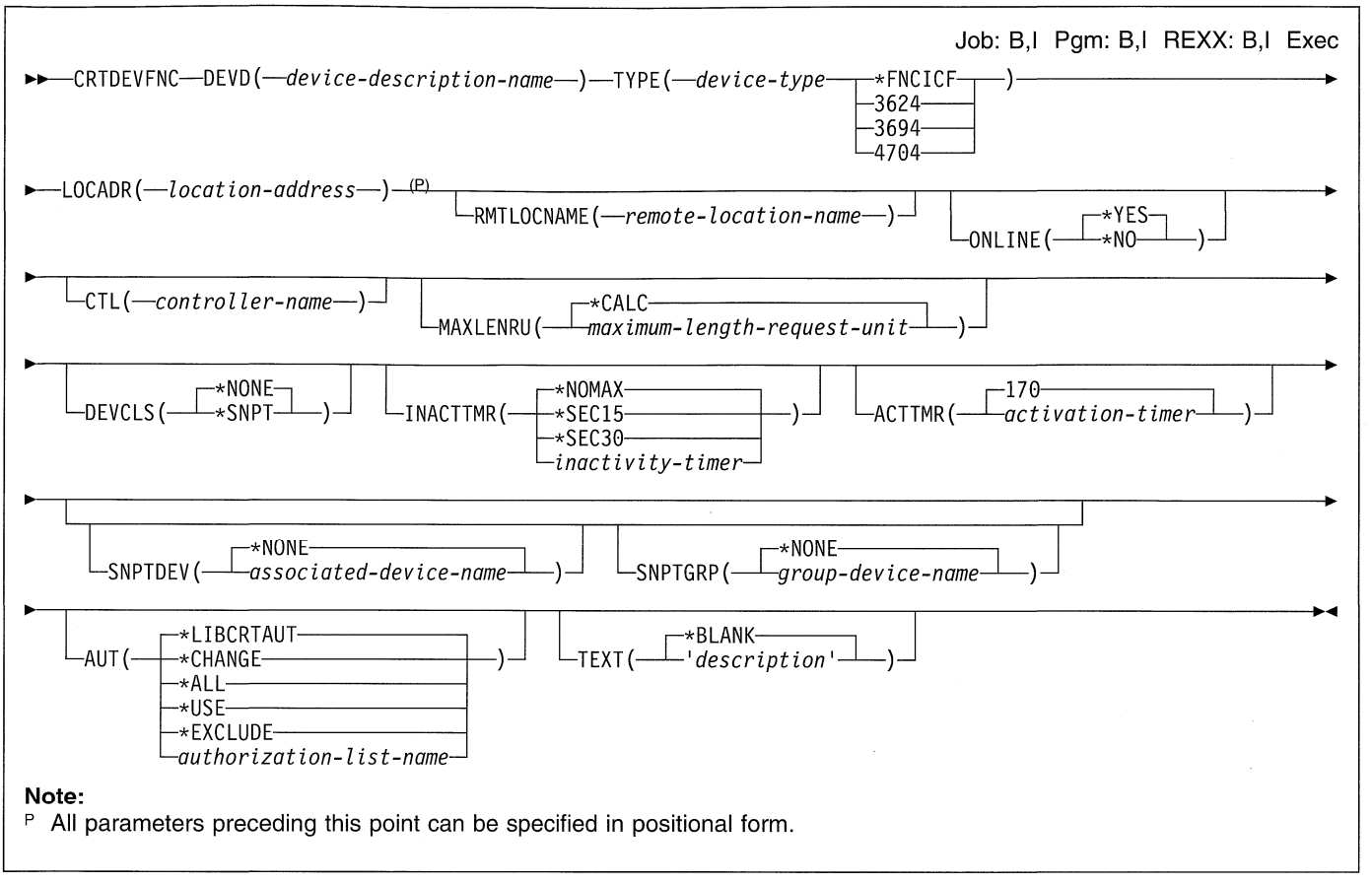
Language/ Device	Type of Physical DBCS-Capable Work Station	Configure as Type-Model	Configure with DBCS Feature
Japanese Display Stations	5295-001 Display Station	5555-B01	2424J4 55FE
	5295-002 Display Station	5555-B01	2424J4 68FE
	Personal System/55* running 5250PC	5555-B01	2424J4 68FE
	Personal System/55 running graphics-capable 5250PC	5555-G01	2424J4 68FE
	Personal System/55 running graphics-capable 5250PC	5555-G02	2424J4 68FE
	Personal System/55 running 5250PC/2	5555-E01	2424J0
Personal System/55 running PC Support/400 by OS/400 system	5555-B01	2424J0	
3270-type Display Stations*	3279-0	2424J0	
Korean Display Stations	5250-Type Display Stations	5555-B01	2424K0
	3270-type Display Stations*	3279-0	2424K0
Traditional Chinese Display Stations	5250-Type Display Stations	5555-B01	2424C0
	3270-type Display Stations*	3279-0	2424C0
Simplified Chinese Display Stations	5250-Type Display Stations	5555-B01	2424S0
	3270-type Display Stations*	3279-0	2424S0

Example

```
CRTDEVDSP   DEVD(DSP4)  TYPE(3180)  MODEL(2)
  DEVCLS(*LCL)  PORT(0)  SWTSET(4)  CTL(CTL01)
  KBDTYPE(USB)
```

This command creates a 3180 Model 2 local device description called DSP4. The display station is located on port 0 of the local work station controller CTL01. The device has an address of 4, with a United States/Canadian keyboard.

CRTDEVFNC (Create Device Description (Finance)) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Finance) (CRTDEVFNC) command creates a device description for a finance device. More information on this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

TYPE

Specifies the type of device this description represents.

***FNCICF:** This device is an intersystem communications function (ICF) finance device. Specify this value when using ICF finance communications. Actual device numbers (3624, 3694, or 4704) are not specified when using the ICF file interface.

3624: The device is a consumer transaction facility.

3694: The device is a document processor.

4704: The device is a finance display station.

LOCADR

Specifies the local location address. Specify a 2-character hexadecimal value ranging from 01 through FF.

Controllers and valid numbers of attachable devices are as follows:

3694	01-04
4701	01-FF
4702	01-FF
4730	01-03
4731	01-02
4732	01-02
4736	01-02
***FBSS**	01-FF

Note: The value 01 is only valid for ICF finance communications when TYPE(*FNCICF) is specified.

Optional Parameters

RMTLOCNAME

Specifies the remote location name of the system with which this object communicates.

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

MAXLENRU

Specifies the default maximum request unit (RU) length that can be sent or received by the local systems if the maximum size is not specified in the bind.

***CALC:** The system determines the value to use.

maximum-length-request-unit: Specify a value ranging from 8 to 4096 bytes for the maximum size of the request unit.

DEVCLS

Specifies the device class for this device.

***NONE:** This device description does not use SNA pass-through support.

***SNPT:** This device description is for a display station connected to an SNA pass-through advanced program-to-program communications (APPC) controller. SNA pass-through support allows the user to connect this display station with host logical unit (LU) types 0 through 3 applications.

INACTTMR

Specifies an inactivity timer (time-out) value. This parameter also specifies what happens when the time-out value is exceeded, dependent on other attributes of the device:

- For devices connected using SNA pass-through (SNPT) support, the user is informed by a message to QSYSOPR and the session is ended when the amount of time that the device is not bound to a host application exceeds the time-out value. The user must reestablish the connection and session.
- For all other attachments, valid values range from 1 through 30 minutes.

Note: This timer is not used by devices allocated to a subsystem (normal interactive use) because the subsystem always has a file open for the device. The timer is used by batch jobs that open and close files for the device.

***NOMAX:** Maximum inactivity time is not tracked.

***SEC15:** A 15-second time-out period is used.

***SEC30:** A 30-second time-out period is used.

inactivity-timer: Specify a time-out value in minutes.

ACTTMR

Specifies, for switched connections, the amount of time that the SNA pass-through support waits for the device to respond to the activation request from the host AS/400 system. If the device does not respond within this time, it is considered not available.

This parameter is valid only when *SNPT is specified for the DEVCLS parameter.

170: Specifies the activate time of 170 seconds.

activation-timer: Specify a number ranging from 1 through 2550 indicating the number of seconds before the device is considered not available.

SNPTDEV

Specifies the name of the associated SNA pass-through device that is attached to a host or advanced program-to-program communications (APPC) controller.

This parameter is valid only when *SNPT is specified for the DEVCLS parameter.

***NONE:** No name is specified.

associated-device-name: Specify the name of a device that is attached to a host or APPC controller that is associated with this device.

SNPTGRP

Specifies the name configured for a group of host devices in a configuration list (see CRTCFGL command). This indicates that this device is tied to any one of the devices in that group which is available.

This parameter is valid only when *SNPT is specified for the DEVCLS parameter.

***NONE:** No name is specified.

group-device-name: Specify the name configured for a group of host devices that must be associated with this device.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object.

The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Changing a Finance Device Description

```
CHGDEVFNC   DEVD(FNCDSP1)  ONLINE(*YES)
```

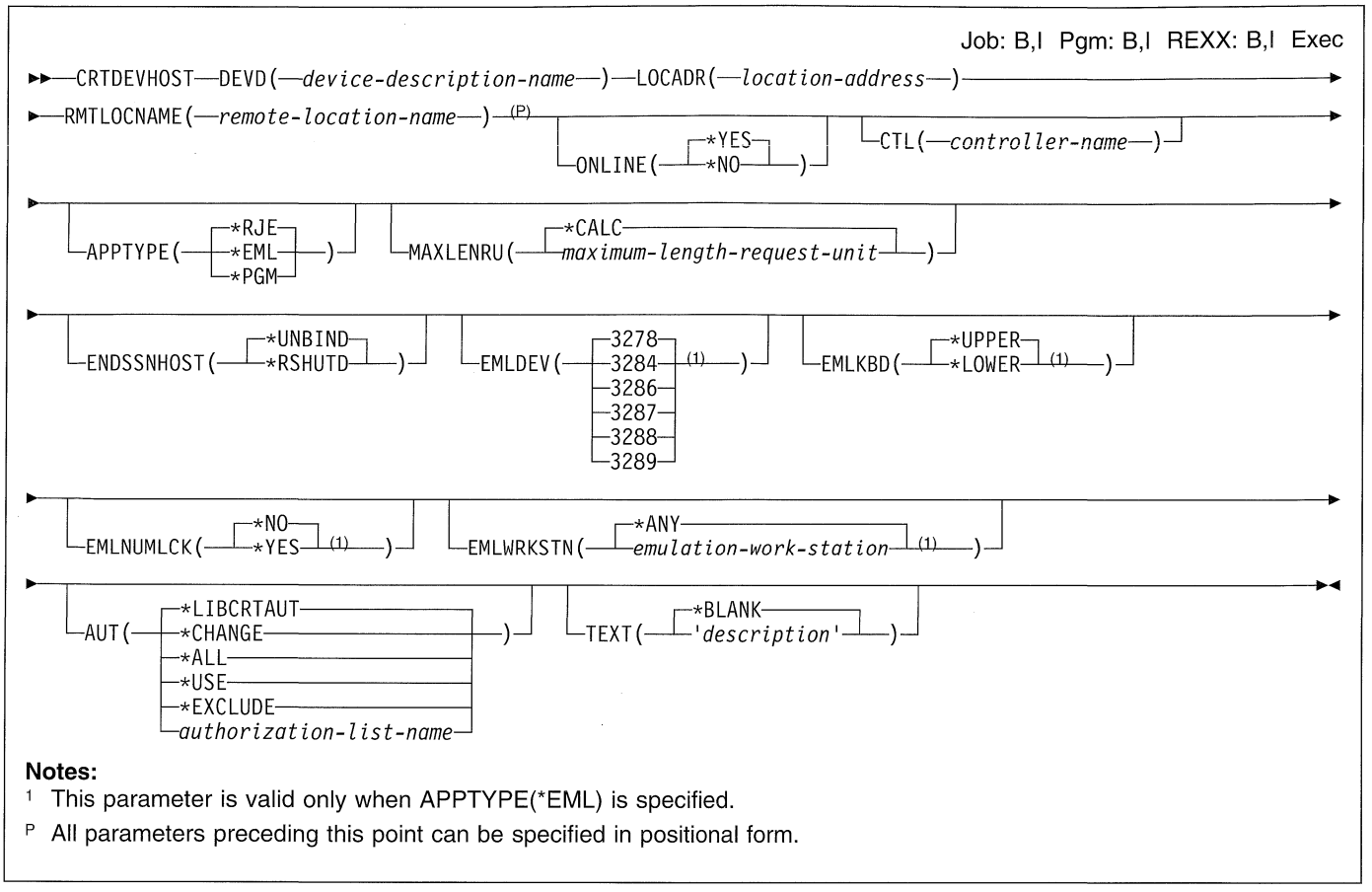
This command changes the device description for the finance device FNCDSP1 so that the device will be varied on at IPL.

Example 2: Creating a Finance Device Description

```
CRTDEVFNC   DEVD(FNCDSP1)  TYPE(*FNCICF)  LOCADR(01)
  CTL(FNCCTL1)
```

This command creates a finance device description named FNCDSP1. This is an ICF finance device with an address of X'01'. It is attached to the finance controller FNCCTL1.

CRTDEVHOST (Create Device Description (SNA Host)) Command



Notes:

1 This parameter is valid only when APPTYPE(*EML) is specified.

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (SNA Host) (CRTDEVHOST) command creates a device description for a Systems Network Architecture (SNA) host system device. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

LOCADR

Specifies the local location address. Specify a 2-character hexadecimal value ranging from 01 through FF.

RMTLOCNAME

Specifies the remote location name of the system with which this object communicates.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

APPTYPE

Specifies the application type being used.

***RJE:** This host device description is used by Remote Job Entry (RJE).

***EML:** This host device description is used by 3270 device emulation.

***PGM:** This host device description is used for program-to-program communications.

MAXLENRU

Specifies the default maximum request unit (RU) length that can be sent or received by the local systems if the maximum size is not specified in the bind.

***CALC:** The system determines the value to use.

maximum-length-request-unit: Specify, for SDLC devices, a value ranging from 256 through 4096 in 256-byte amounts to be used as the maximum length for incoming request units. Values 241, 245, 247, 497, 501 and 503 are valid only for X.25 devices.

ENDSSNHOST

Specifies how the host device ends a session with the host system.

***UNBIND:** The host device sends the SNA command requesting the AS/400 system to end the session.

***RSHUTD:** The host device sends the SNA command requesting the host system to end the session.

EMLDEV

Specifies the type of 3270 device being emulated. This parameter is valid only if *EML was specified on the APPTYPE parameter when the device was created. The change requested by this command becomes effective the next time either the Start 3270 Display Emulation (STREML3270) command or the Start Print Emulation (STRPRTEML) command accesses the device.

3278: This device description is used to emulate a 3278 display device.

3284: This device description is used to emulate a 3284 printer device.

3286: This device description is used to emulate a 3286 printer device.

3287: This device description is used to emulate a 3287 printer device.

3288: This device description is used to emulate a 3288 printer device.

3289: This device description is used to emulate a 3289 printer device.

EMLKBD

Specifies the type of 3278 display keyboard being emulated. A value can be specified for this parameter only if APPTYPE(*EML) is specified. The specified value is used if EMLDEV(3278) is specified.

***UPPER:** A 3270 display device keyboard is emulated with uppercase characters only.

***LOWER:** A 3270 display device keyboard is emulated with uppercase and lowercase characters.

EMLNUMLCK

Specifies whether numeric input fields allow only numeric data on a 5250 keyboard. The value can be specified for this parameter only if this is a 3270 device emulation application type (APPTYPE is *EML).

***NO:** 3270 emulation allows any data to be typed in the numeric input fields.

***YES:** 3270 emulation allows only numeric data to be typed in the numeric input fields. Numeric data that can be typed includes the characters 0 through 9 and the symbols + - , . and blank.

EMLWRKSTN

Specifies that this emulation device is associated with a real display or printer device and reserved for use exclusively by that work station. If no device or *ANY is specified, any work station can use the emulation device. A value can be specified for this parameter only if APPTYPE(*EML) is specified.

***ANY:** Any work station can use the emulation device.

emulation-work-station: Specify the device name for the work station that uses this emulation device.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

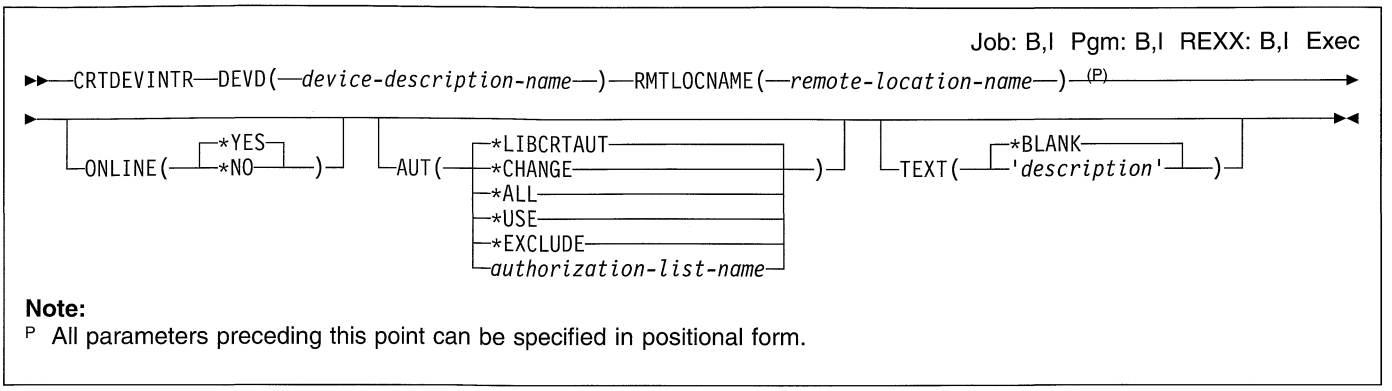
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVHOST   DEVD(COMMDSP1)  LOCADR(FE)
  RMTLOCNAME(SYS)  CTL(HOSTCTL001)
  APPTYPE(*EML)  EMLDEV(3284)
```

This command creates a device description for an SNA host communications device named COMMDSP1. The address of the device is X'FE'. It is attached to the controller HOSTCTL001 and communicates with SYS. This device uses 3270 emulation to emulate a 3284 printer.

CRTDEVINTR (Create Device Description (Intrasystem)) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Intrasystem) (CRTDEVINTR) command creates an intrasystem device description, which allows two programs running on the same system to communicate with each other.

Required Parameters

DEVD
Specifies the name of the device description being created.

RMTLOCNAME
Specifies the remote location name of the system with which this object communicates.

Optional Parameters

ONLINE
Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

AUT
Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT
Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

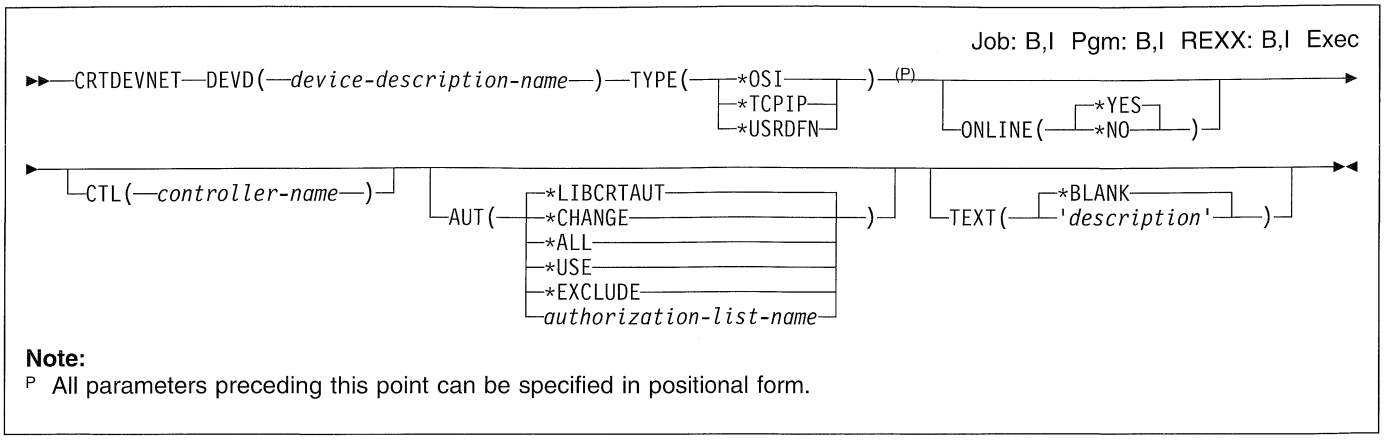
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVINTR   DEVD(INTRALOC)  RMTLOCNAME(INTRARMT)
```

This command creates a device description named INTRALOC for intrasystem communications with a remote location named INTRARMT.

CRTDEVNET (Create Device Description (Network)) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Network) (CRTDEVNET) command creates a device description for a network device. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

TYPE

Specifies the type of network associated with this device.

***OSI:** This device is attached to an Open System Interconnection (OSI) network.

***TCPIP:** This device is attached to a Transmission Control Protocol/Internet Productivity (TCP/IP) network.

***USRDFN:** This device is attached to a network that is supported by a program using the user-defined communications Applications Programming Interface.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

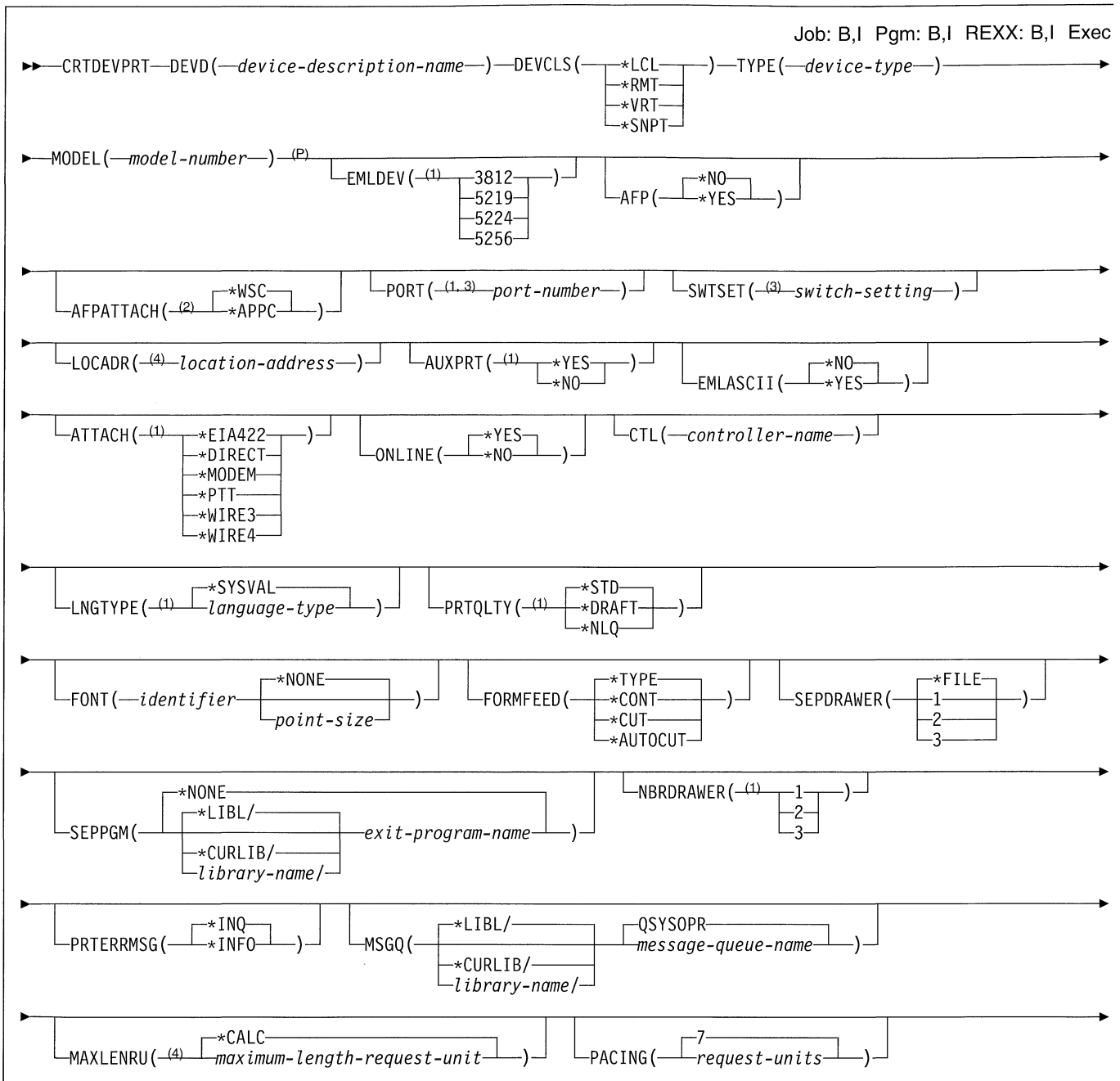
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVNET   DEVD(NETDEV02)  TYPE(*TCPIP)
```

This command creates a device description for a network device named NETDEV02.

CRTDEVPRT (Create Device Description (Printer)) Command



Notes:

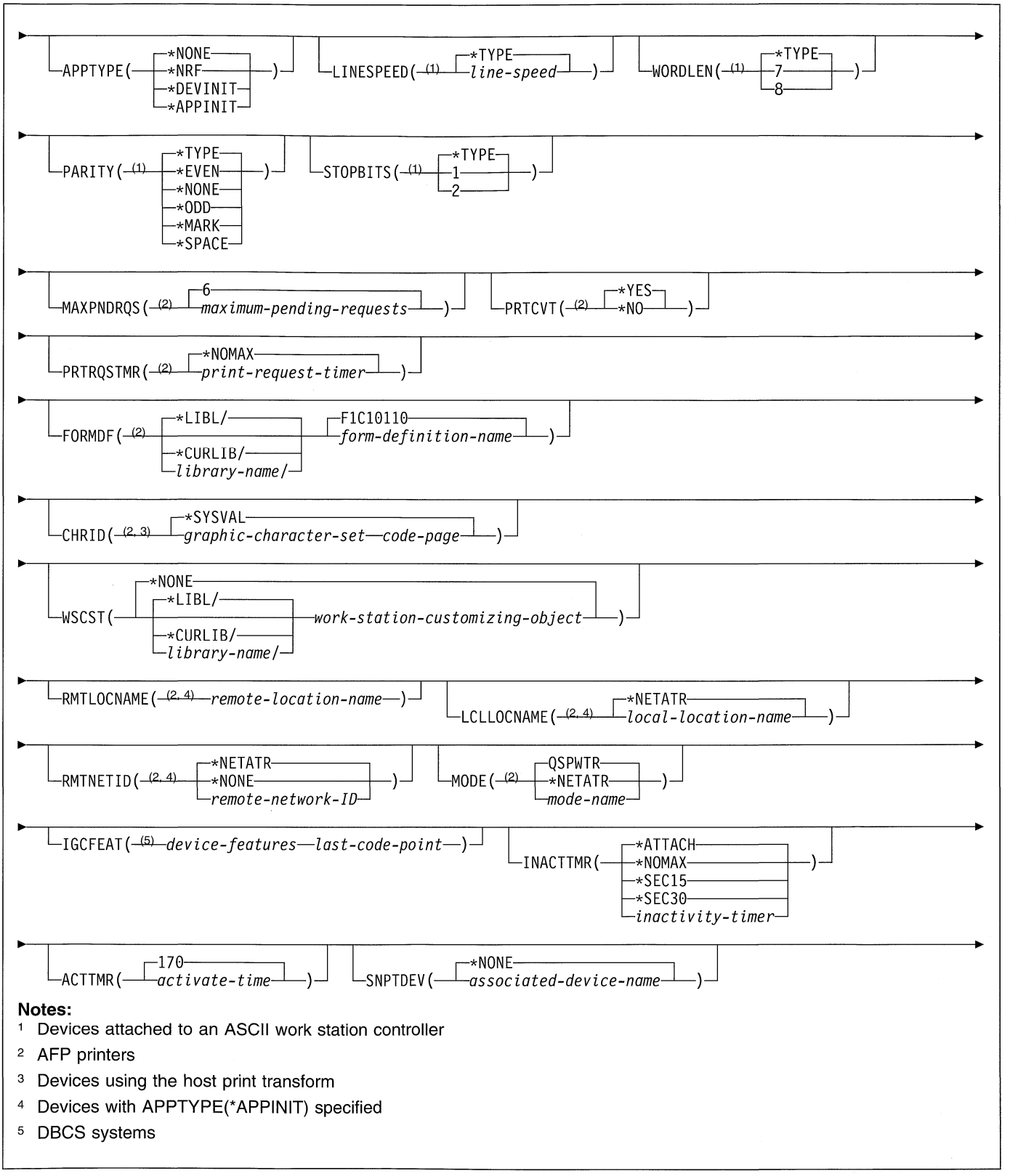
1 Devices attached to an ASCII work station controller

2 AFP printers

3 Local twinaxial devices

4 Remote devices

P All parameters preceding this point can be specified in positional form.



Notes:

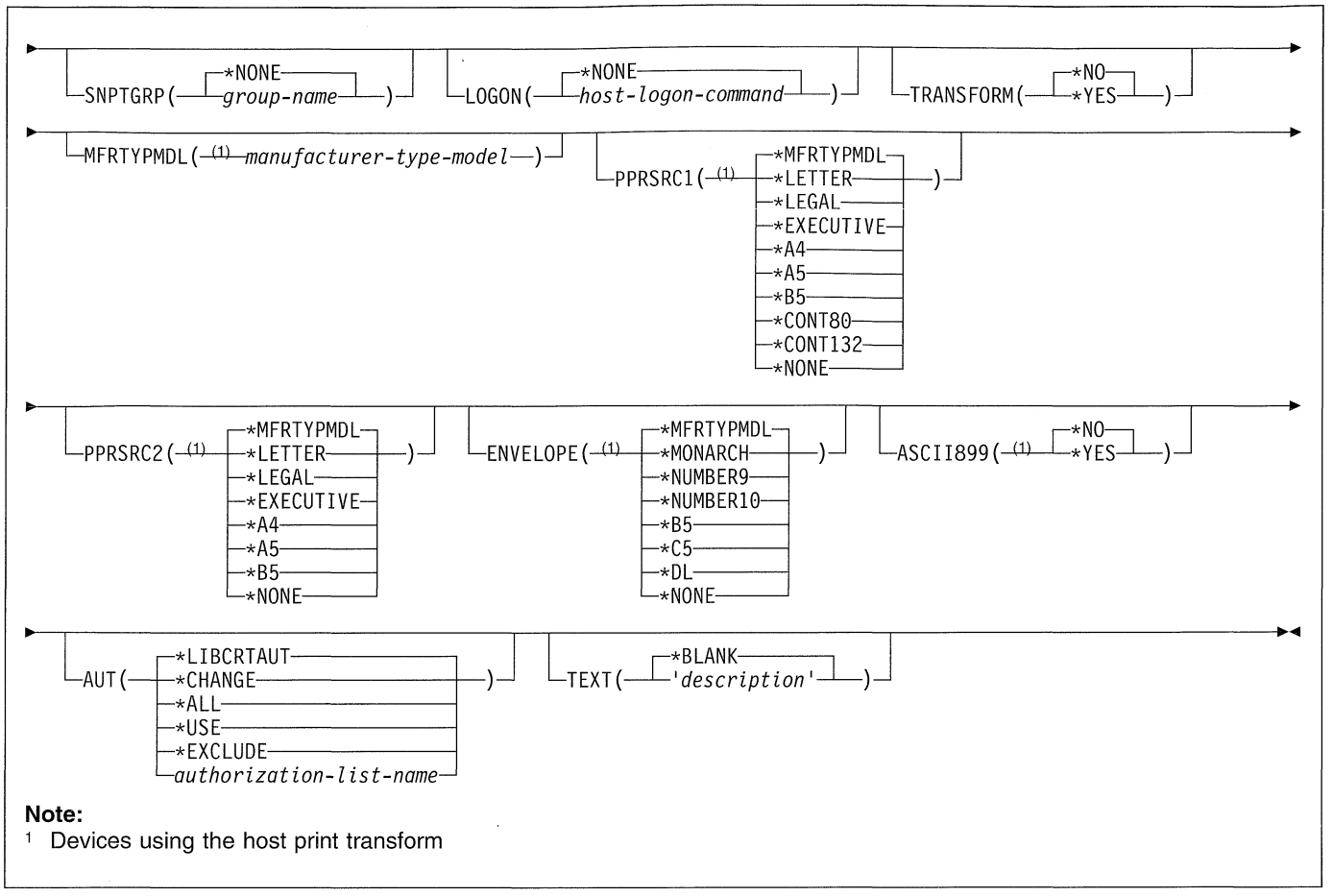
1 Devices attached to an ASCII work station controller

2 AFP printers

3 Devices using the host print transform

4 Devices with APPTYPE(*APPINIT) specified

5 DBCS systems



Note:
1 Devices using the host print transform

Purpose

The Create Device Description (Printer) (CRTDEVPRT) command creates a device description for a printer device. More information on this command is in the *Device Configuration Guide* for local printers and the *OS/400* Communications Configuration Reference* for remote and virtual printers.

Required Parameters

DEVD

Specifies the name of the device description being created.

DEVCLS

Specifies the device class for the printer.

***LCL:** The device description is for a printer connected to a local work station controller.

***RMT:** The device description is for a printer connected to a remote work station controller.

***VRT:** The device description is for a virtual printer.

***SNPT:** This device description is for a display station connected to an SNA pass-through advanced program-to-program communications (APPC) controller. SNA pass-through support allows the user to connect this display station with host logical unit (LU) types 0 through 3 applications.

TYPE

Specifies the type of device the description represents. Specify one of the following 4-character type codes that describes the type of device.

Note: *IPDS printers are described by a 5-character type code.

- 3287
- 3812
- 3900
- 4214
- 4234
- 4245
- 5219
- 5224
- 5225
- 5256
- 5262
- 5553 (DBCS-capable printer)
- 5583 (DBCS-capable printer)
- 6252
- *IPDS (includes all of the following printers: 3812, 3816, 3820, 3825, 3827, 3829, 3835, 3900, 4224, 4230, and 4234)

The following 4-character type codes are valid for ASCII:

Type	Description
4019	IBM 4019 LaserPrinter
4201	IBM Proprinter* II, IBM Proprinter III
4202	IBM Proprinter XL, IBM Proprinter II XL, IBM Proprinter III XL
4207	IBM Proprinter X24, IBM Proprinter X24E
4208	IBM Proprinter XL24, IBM Proprinter XL24E
4216	IBM Personal Pageprinter
4224	IBM 4224 Printer
4234	IBM 4234 Printer
5204	IBM Quickwriter*
6252	IBM Impact Writer

MODEL

Specifies the model number of the device. This number indicates to the system the operational characteristics of the device. Specify one of the following model numbers (ranging 1 through 4 characters) that matches the device.

Type	Model
3287	0
3812	1
4214	2
4234	2
4245	T12 or T20
5219	D1 or D2
5224	1 or 2
5225	1, 2, 3, or 4
5256	1, 2, or 3
5262	1
5553	B01
5583	200
6252	T08
*IPDS	0

For ASCII devices only:

***ASCII:** This value is used if the model number cannot be easily determined for the printer. The system assigns a model number (if the device has a model number). The model number assigned by the system may not be the actual model number for the printer. It is recommended that the user determine the model number of the printer if at all possible; and enter that number.

Note: *ASCII printers are described by 1 to 6 characters.

The possible choices for model numbers associated with the device type are as follows:

Type	Model
4019	1(*ASCII) IBM 4019 LaserPrinter
4201	2(*ASCII) IBM Proprinter II
	3(*ASCII) IBM Proprinter III
4202	1(*ASCII) IBM Proprinter XL
	2(*ASCII) IBM Proprinter II XL
	3(*ASCII) IBM Proprinter III XL
4207	1(*ASCII) IBM Proprinter X24
	2(*ASCII) IBM Proprinter X24E
4208	1(*ASCII) IBM Proprinter XL24
	2(*ASCII) IBM Proprinter XL24E
4216	10(*ASCII) IBM Personal Pageprinter
4224	301(*ASCII), 302, 3E3
4234	13(*ASCII)
5204	1(*ASCII) IBM Quickwriter
6252	(*ASCII) AS8

Optional Parameters

EMLDEV

Specifies the twinaxial printer that the ASCII printer emulates. This is a required parameter for ASCII device types.

3812: The 3812 printer is emulated. This value is valid for the 4019, 4216, and 5204 printers.

5219: The 5219 printer is emulated. This value is valid for all printers except 4201, 4202, and 4234.

5224: The 5224 printer is emulated. This value is valid for all printers.

5256: The 5256 printer is emulated. This value is valid for all printers.

AFP

Specifies whether the printer is capable of advanced function printing (AFP). This parameter is used only if *LCL or *RMT is specified on the DEVCLS parameter, and *IPDS is specified on the TYPE parameter.

Most IPDS printers can be configured with AFP(*YES) or AFP(*NO). Most non-IPDS printers can be configured only with AFP(*NO).

***NO:** The printer is not used for advanced function printing.

***YES:** The printer is used for advanced function printing.

AFPATTACH

Specifies the type of attachment used for AFP printers. This parameter is used only if AFP(*YES) is specified.

***WSC:** 3812 or 3816 printers are attached to a local or remote work station controller.

***APPC:** 3820, 3827, 3825, or 3835 printers are attached to a synchronous data link control (SDLC) or token-ring network using advanced program-to-program communications (APPC) or advanced peer-to-peer networking (APPN).

PORT

Specifies the port number for the device. Valid values range from 0 through 17.

For ASCII devices, valid values range from 0 through 17 and indicate which port of the ASCII work station controller the printer is attached to. Without the 12-port expansion feature, only ports 0 through 5 are valid. With the 12-port expansion feature, ports 6 through 17 are added.

SWTSET
Specifies, for local twinaxial printers, the switch setting or device address for the device. Valid values range from 0 through 6.

LOCADR
Specifies the local location address. Specify a 2-character hexadecimal value ranging from 00 through FE.

AUXPRT
Specifies, for ASCII printers, whether the printer description is for an auxiliary printer. An auxiliary printer is attached to an ASCII display station, instead of being physically attached to the ASCII work station controller. All data sent to the printer passes through the display station. The user must have the display station power on before the printer can be used. This is a required parameter for ASCII printers.

***YES:** The printer is attached to the auxiliary port of an ASCII display station.

***NO:** The printer is physically attached to the ASCII work station controller.

Note: If AUXPRT(*YES) is specified, the following parameters must match those specified for the ASCII display station that the printer is attached to:

ATTACH
LINESPEED
WORDLEN
PARITY
STOPBITS

EMLASCII
Specifies, for ASCII devices, whether the device being configured is emulating a supported ASCII device type (TYPE parameter). When an ASCII device is configured, choices for the following parameters are restricted to the valid range for that device:

ATTACH
LINESPEED
WORDLEN
PARITY
STOPBITS

These restrictions may not apply to ASCII devices emulating supported ASCII device types because the emulating device may have a wider range of choices for these parameters than the emulated ASCII device.

When EMLASCII(*YES) is specified, these parameter restrictions are not enforced. The user is allowed to enter the full range of values for these parameters, but is also responsible for verifying that the values selected are valid for the device being configured.

***NO:** The device being configured is one of the supported ASCII device types (identified by the TYPE parameter). Therefore, the device-specific restrictions for the physical attachment, line speed, word length, and stop bits parameters are enforced.

***YES:** The device being configured is emulating one of the supported ASCII device types (identified by the TYPE parameter), and the device-specific restrictions for the physical attachment, line speed, word length, and stop bits parameters are not enforced.

ATTACH
Specifies, for ASCII printers, the physical attachment of the printer to the ASCII work station controller:

- If AUXPRT(*YES) is specified, then the printer is an auxiliary printer, and the physical attachment specified is the attachment between the ASCII display station and the ASCII work station controller.
- If AUXPRT(*NO) is specified, then the physical attachment specified is the attachment between the printer and the ASCII work station controller.

This is a required parameter for ASCII device types.

***EIA422:** The EIA-422 attachment is used. This value is valid only for auxiliary printers and the following printers, which support *EIA422 for both auxiliary and non-auxiliary attachments:

4201-3
4202-3
4207-2
4208-2
4224
4234

***DIRECT:** The EIA-232 direct attachment is used.

***MODEM:** The EIA-232 modem attachment is used, and is valid only for auxiliary printers.

***PTT:** Post Telephone and Telegraph (PTT) attachment is used, and is valid only for auxiliary printers.

***WIRE3:** The EIA-232 three-wire attachment is used.

***WIRE4:** The EIA-232 four-wire attachment is used.

ONLINE
Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL
Specifies the name of the controller description to which this object is attached.

LNGTYPE
Specifies the language keyboard country identifier for an ASCII printer.

***SYSVAL:** The default printer language is *SYSVAL, which instructs the system to use the QKBDTYPE system value.

language-type: Specify one of the following language keyboard country identifiers from the Keyboard Mapping table which are supported for all printers:

Table 32. Keyboard Mapping Table for Printers

Language/Country	Identifier	ASCII Device Group
Arabic X/Basic	CLB	D
Austria/Germany	AGB	A, B
Austria/Germany Multinational	AGI	A, B
Belgium Multinational	BLI	B
Brazilian Portuguese	BRB	
Canadian French	CAB	A, B
Canadian French Multinational	CAI	A, B
Cyrillic	CYB	
Denmark	DMB	B
Denmark Multinational	DMI	B
Finland/Sweden	FNB	B
Finland/Sweden Multinational	FNI	B
France (Azerty)	FAB	A, B
France (Azerty) Multinational	FAI	A, B
France (Qwerty)	FQB	
France (Qwerty) Multinational	FQI	
Greece	GKB	
Greece	GNB	
Hebrew	NCB	D
Iceland	ICB	
Iceland Multinational	ICI	
International	INB	
International Multinational	INI	
Italy	ITB	A, B
Italy Multinational	ITI	A, B
Japan English	JEB	
Japan English Multinational	JEI	
Japan Kanji (For PS/55* and 5295 display stations)	JKB	
Japan United States Basic	JUB	
Japan Katakana (For 5251, 5291, 5292, and 3180 Katakana display stations)	KAB	
Korea	KOB	
Latin-2/ROECE	ROB	
Netherlands	NEB	
Netherlands Multinational	NEI	
Norway	NWB	B
Norway Multinational	NWI	B
Portugal	PRB	B
Portugal Multinational	PRI	B
Simplified Chinese	RCB	
Spain	SPB	B
Spain Multinational	SPI	B
Spanish Speaking	SSB	B
Spanish Speaking Multinational	SSI	B
Sweden	SWB	B
Sweden Multinational	SWI	B
Switzerland/French Multinational	SFI	B
Switzerland/German Multinational	SGI	B
Thailand	THB	
Traditional Chinese	TAB	
Turkey	TKB	
United Kingdom	UKB	A, B
United Kingdom Multinational	UKI	A, B
United States/Canada	USB	A, B, C
United States/Canada Multinational	USI	A, B, C
Languages of the former Yugoslavia	YGI	

PRTQLTY

Specifies whether the default print quality for ASCII printers should be draft (*DRAFT), standard (*STD), or near-letter quality (*NLQ), from least to best quality. All ASCII printer types (with all emulations) support this parameter. If the printer is emulating a 5219 (EMLDEV parameter), this quality setting is overridden by individual printer files sent to this printer.

***STD:** The output is printed with standard quality.

***DRAFT:** Draft print quality setting is used.

***NLQ:** Near-letter print quality setting is used.

Table 34 at the end of this command description shows the valid values for the 4214, 4224, 4234, and 5219 printers.

FONT

Specifies, for the 3812, 3816, 3820, 3825, 3827, 3829, 3835, 3900, 4224, and 5219 printers (including ASCII printers emulating the 5219 printer), the font identifier and point size to be used with this printer file. A font identifier (up to 10 digits) is used with this printer file. Each font identifier has an implied characters-per-inch (CPI) value. If an identifier or point size is not specified, the system automatically sets one.

More information about the valid font identifiers, the display value, the characters per inch value implied with each font style, a description of each font style, and whether the font is supported on a particular printer is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

Note: Some fonts may be substituted by the printer. Consult the various printer reference guides for details.

Element 1: Font Identifier

identifier: Specify the 3-, 4-, or 5-digit font identifier associated with this printer.

Element 2: Point Size

***NONE:** The point size is supplied by the system and is determined by the specified font identifier.

point-size: Specify a point size ranging from 0.1 through 999.9.

FORMFEED

Specifies, for the 4214, 5219, and 5553 printers (including ASCII printers that are configured as an SCS 4214 or SCS 5219 printer), and for IPDS printers, the mode in which forms are fed into the device.

Note: The FORMFEED parameter is overridden by the value specified on the PPRSRC1 parameter when the host print transform function is enabled.

***TYPE:** Form feed value is determined by the system based on printer type.

***CONT:** Continuous forms are used by the printer, and the tractor feed attachment must be on the device. For ASCII devices only: Continuous forms are used by the printer. This value is valid for all ASCII printers, except the 4216 Printer emulating a 5219 Printer.

***CUT:** Single-cut sheets are manually fed into the printer. For cut sheets, the forms alignment message is not sent. This value is valid for all ASCII printers, except the 4216 Printer, that are emulating a SCS 5219 Printer.

***AUTOCUT:** The sheet-feed attachment must be on the printer. Single-cut sheets are automatically fed into the printer. The forms alignment message is not sent for cut sheets. For ASCII devices only: Single-cut sheets are automatically fed into the printer; valid for 4207, 4208, 4216, 4224, and 5204 Printers that are emulating a 5219 Printer.

SEPDRAWER

Specifies which drawer is selected for printing separators.

***FILE:** The separator pages are printed on paper from the same drawer as the rest of the spooled file.

1: The separator pages are printed from drawer 1.

2: The separator pages are printed from drawer 2.

3: The separator pages are printed from drawer 3.

Note: SEPDRAWER(3) implies that two paper drawers and an envelope drawer are used.

SEPPGM

Specifies a style of separator page by allowing you to call a user exit program while printing job and file separators.

***NONE:** The separator pages are not changed.

The name of the exit program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

exit-program-name: Specify an exit program name.

NBRDRAWER

Specifies, for ASCII devices only, and only if FORMFEED(*AUTOCUT) is also specified, the number of drawers supported by the printer.

1: One drawer is supported.

2: Two drawers are supported.

3: Three drawers are supported.

For the printers that support FORMFEED(*AUTOCUT), the following number of drawers are supported:

- For 4207-1 and 4208 printers, only NBRDRAWER(1) is valid.
- For a 4224 printer, only NBRDRAWER(3) is valid.
- For 4207-2, 4216, and 5204 printers, NBRDRAWER values of 1, 2, and 3 are supported.

Note: NBRDRAWER(3) implies that two paper drawers and an envelope drawer are used.

PRTERRMSG

Specifies whether the device has inquiry messages or informational messages sent for recoverable errors. This parameter is not valid for printers attached to ASCII work station controllers.

***INQ:** Inquiry messages are sent for recoverable errors.

***INFO:** Informational messages are sent for recoverable errors whenever possible.

MSGQ

Specifies the qualified name of the message queue to which messages are sent.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QSYSOPR: Messages are sent to the QSYSOPR message queue.

message-queue-name: Specify the name of the message queue to which messages are sent.

MAXLENRU

Specifies the maximum request unit (RU) length (in bytes) allowed.

***CALC:** The system determines the value to use.

maximum-length-request-unit: Specify a value to be used as the maximum length for incoming request units. Values 241 and 247 are valid only for X.25 devices.

PACING

Specifies the rate at which the receiver accepts data (measured in request units) from the SNA data flow. This parameter cannot be used for 3287 remote printers.

7: A value of 7 is used as the RU pacing value.

request-units: Specify the number of request units that can be accepted by the receiver. Valid values range from 0 through 7.

APPTYPE

Specifies the application type being used.

***NONE:** The device is not used for any application.

***NRF:** The device is used for the network routing facility application.

***DEVINIT:** The device starts (initiates) the session.

***APPINIT:** The application program starts (initiates) the session.

LINESPEED

Specifies the line speed (in bits per second) that ASCII printers use to communicate over the interface (attachment) between the ASCII work station and the printer. The value of *TYPE specifies 19200 bits per second for all printers.

***TYPE:** The system uses the suggested setting (in bits per second) for the device type. *TYPE specifies 19200 for all printers.

Valid values are 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 19200, and 38400.

The valid line speeds for each printer are as follows:

Printer	Line Speed
4201-2	150, 300, 600, 1200, 2400, 4800, 9600, 19200
4201-3	150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
4202-1	150, 300, 600, 1200, 2400, 4800, 9600, 19200
4202-2	150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
4202-3	150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
4207-1	150, 300, 600, 1200, 2400, 4800, 9600, 19200
4207-2	150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
4208-1	150, 300, 600, 1200, 2400, 4800, 9600, 19200
4208-2	150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
4216	600, 1200, 2400, 4800, 9600, 19200
4224	300, 600, 1200, 2400, 4800, 9600, 19200
4234	150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400
5204	300, 600, 1200, 2400, 4800, 9600, 19200
6252	300, 600, 1200, 2400, 9600, 19200, 38400

WORDLEN

Specifies the word length (in bits per character) that ASCII printers use to communicate with the printer.

All ASCII printers support both 7-bit and 8-bit word lengths except for the 4224 printer and the 6252, which support only 8 bits. *TYPE indicates 8-bit word lengths for all printers.

***TYPE:** The system uses the suggested setting for this printer.

7: 7-bit word lengths are used.

8: 8-bit word lengths are used.

PARITY

Specifies the type of parity ASCII printers use to communicate with the printer. All printers support *NONE, *ODD, and *EVEN. *TYPE specifies *EVEN parity for all printers. Even parity is valid for all printers.

***TYPE:** The system uses the suggested setting for this printer.

***EVEN:** Even parity is used.

***NONE:** No parity bit is used.

***ODD:** Odd parity is used.

***MARK:** Mark parity is used.

***SPACE:** Space parity is used.

STOPBITS

Specifies the number of stop bits ASCII printers use to communicate with the printer. All ASCII printers support both one and two stop bits, except for the 4224, which supports only one stop bit. *TYPE specifies one stop bit for all printers.

***TYPE:** The system uses the suggested setting for this printer.

1: One stop bit is used.

2: Two stop bits are used.

MAXPNDRQS

Specifies the maximum number of print requests that can be queued for AFP printers. This parameter is valid only when AFP(*YES) is specified.

6: Up to six print requests can be queued.

maximum-pending-requests: Specify a value ranging from 1 through 31, indicating the maximum number of print requests that can be queued.

PRTCVT

Specifies whether a file using AFP must be completely converted to IPDS before printing can begin.

***YES:** Printing begins prior to complete IPDS conversion.

***NO:** Printing does not begin prior to complete IPDS conversion.

PRTRQSTMR

Specifies the number of seconds to wait after a print request has been sent to a continuous forms printer configured for advanced function printing before the last printed output is forced into the stacker. This parameter

is used only if AFP(*YES) and FORMFEED(*CONT) are specified.

***NOMAX:** The timer is disabled.

print-request-timer: Specify a value ranging from 1 through 3600, indicating the number of seconds to wait after a print request has been sent.

FORMDF

Specifies the library and form definition to be used for print requests that do not specify a form definition. This parameter is used only if AFP(*YES) is specified.

The name of the form definition can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***F1C10110:** Form definition F1C10110 is used.

form-definition-name: Specify the name of the form definition to be used.

CHRID

Specifies the default character identifier (graphic character set and code page). This parameter is valid only if AFP(*YES) and AFPATTACH(*APPC) are specified, or if TRANSFORM(*YES) is specified.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values that match the attributes of the printer. Valid values range from 1 through 32767.

Element 2: Code Page

code-page: Specify the code page set values that match the attributes of the printer. Valid values range from 1 through 32767.

WSCST

Specifies the qualified name of a work station customizing object to be created.

***NONE:** No work station customizing object is specified.

work-station-customizing-object: Specify the work station customizing object.

The name of the work station customizing object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

Note: If a work station customizing object is specified for the WSCST parameter, all language keyboard country identifiers are valid for ASCII devices except for the following: FQB, FQI, INB, INI, JEB, JEI, JKB, JUB, KAB, KOB, RCB, and TAB. See Table 14 for a list of the language keyboard country identifiers.

RMTLOCNAME

Specifies the name of the remote system. This parameter is valid if AFP(*YES) and AFPATTACH(*APPC) are specified. This parameter is required when APPTYPE(*APPINIT) is specified. The remote location name for an APPTYPE(*APPINIT) device is the Virtual Telecommunications Access Method/IBM Network Control Program (VTAM/NCP) name of the physical device.

LCLLOCNAME

Specifies the local location name. This parameter is valid only when AFP(*YES) and AFPATTACH(*APPC) are specified, or when APPTYPE(*APPINIT) is specified. The local location name for an APPTYPE(*APPINIT) device is the name of the independent logical unit (LU) in the IBM Network Control Program (NCP).

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the local location name.

RMTNETID

Specifies the identifier (ID) of the remote network. This parameter is required when AFP(*YES) and AFPATTACH(*APPC) are specified, or when APPTYPE(*APPINIT) is specified.

***NETATR:** The RMTNETID value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

remote-network-ID: Specify the ID of the remote network.

MODE

Specifies the name of the mode used to define the session limits and session characteristics for the device. This parameter is valid only if AFP(*YES) and AFPATTACH(*APPC) are specified.

QSPWTR: QSPWTR is the mode used.

***NETATR:** The mode name specified in the network attributes is used.

mode-name: Specify the mode name.

IGCFEAT

Specifies which double-byte character set (DBCS) table is used, by the language and device. This parameter is valid for DBCS-capable devices only.

Table 35 at the end of this command description specifies the device features and last code point.

INACTTMR

Specifies an inactivity timer (time-out) value. This parameter also specifies what happens when the time-out value is exceeded, dependent on other attributes of the device:

- For devices connected using SNA pass-through (SNPT) support, the user is informed by a message to QSYSOPR and the session is ended when the amount of time that the device is not bound to a host application exceeds the time-out value. The user must reestablish the connection and session.
- For devices with an application type value of *APPINIT, *DEVINIT, or *NRF, the session is ended when the device is inactive (the file opened against the device is closed and no additional requests to open files are received for the device) for a period of time that exceeds the time-out value.

Note: This timer is not used by devices allocated to a subsystem (normal interactive use) because the subsystem always has a file open for the device. The timer is used by batch jobs that open and close files for the device.

For all other attachments, valid values range from 1 through 30 minutes.

***ATTACH:** This value varies by the value on the physical attachment (ATTACH parameter) and certain values on the device class (DEVCLS) and application type (APPTYPE) parameters. For DEVCLS(*SNPT) or APPTYPE(*DEVINIT) support, *ATTACH maps to *NOMAX. For APPTYPE(*NRF) and APPTYPE(*APPINIT) support, *ATTACH maps to 1 minute.

***NOMAX:** Maximum inactivity time is not tracked.

***SEC15:** A 15-second time-out period is used.

***SEC30:** A 30-second time-out period is used.

inactivity-timer: Specify a time-out value.

ACTTMR

Specifies, for switched lines, the amount of time (in seconds) that the SNA pass-through support waits for the device to respond to the activation request from the host AS/400 system. If the device does not respond within this time, it is considered not available.

170: Specifies the activate time of 170 seconds.

activation-timer: Specify a number ranging from 1 through 2550 indicating the number of seconds before the device is considered not available.

SNPTDEV

Specifies the name of the associated SNA pass-through device that is attached to a host or advanced program-to-program communications (APPC) controller.

***NONE:** No name is specified.

associated-device-name: Specify the name of a device that is attached to a host or an APPC controller that is associated with this device.

SNPTGRP

Specifies the name configured for a group of host devices in a configuration list. This indicates that this device is associated with any one of the devices in that group which is available.

***NONE:** No name is specified.

group-name: Specify the name configured for a group of host devices that must be associated with this device.

LOGON

Specifies the sign-on (logon) text. This parameter is allowed when DEVCLS(*SNPT) or APPTYPE(*NRF) is specified. APPTYPE(*NRF) specifies the logon string that is sent to the host system when a request is made to establish a session. DEVCLS(*SNPT) specifies the sign-on text that is sent to the host system after starting SNA pass-through support.

Specifies the logon string that is sent to the system services control point (SSCP) on the host network when the file is opened.

***NONE:** No text is sent to the host system.

host-logon-command: Specify text that is sent to the host system. The text must be enclosed in apostrophes if it contains blanks or other special characters. All apostrophes within the text must be represented by two apostrophes. A maximum of 256 characters can be specified.

TRANSFORM

Specifies whether the printer uses the host print transform function.

***NO:** The printer does not use the host print transform function

***YES:** The printer uses the host print transform function.

MFRTYPMDL

Specifies the manufacturer, type, and model for a printer using the host print transform function. The following list shows the values to be specified for the device you are using:

Note: If *WSCST is specified for MFRTYPMDL, a workstation customizing object must be specified.

Table 33 (Page 1 of 2). Manufacturer Type and Model (MFRTYPMDL) Parameter

*IBM2380	IBM 2380 Personal Printer Series II
*IBM2381	IBM 2381 Personal Printer Series II
*IBM2390	IBM 2390 Personal Printer Series II
*IBM2391	IBM 2391 Personal Printer Series II

Table 33 (Page 2 of 2). Manufacturer Type and Model (MFRTYPMDL) Parameter

*IBM3812	IBM 3812 Pageprinter
*IBM3816	IBM 3816 Pageprinter
*IBM4019	IBM 4019 LaserPrinter
	IBM 4019E LaserPrinter E
*IBM4019HP	IBM 4019 LaserPrinter (HP** Mode)
	IBM 4019E LaserPrinter E (HP Mode)
*IBM4029	IBM 4029-010 LaserPrinter 5E
	IBM 4029-020 LaserPrinter 6
	IBM 4029-030 LaserPrinter 10
	IBM 4029-040 LaserPrinter 10L
*IBM4029HP	IBM 4029-010 LaserPrinter 5E (HP Mode)
	IBM 4029-020 LaserPrinter 6 (HP Mode)
	IBM 4029-030 LaserPrinter 10 (HP Mode)
	IBM 4029-040 LaserPrinter 10L (HP Mode)
*IBM4039HP	IBM 4039 LaserPrinter 10R (HP Mode)
	IBM 4039 LaserPrinter 12R (HP Mode)
	IBM 4039 LaserPrinter 12L (HP Mode)
	IBM 4039 LaserPrinter 16L (HP Mode)
*IBM4070	IBM 4070 IJ
*IBM4070EP	IBM 4070 IJ (Epson** Mode)
*IBM4072	IBM 4072 ExecJet*
*IBM42011	IBM 4201-1 Proprinter
*IBM42012	IBM 4201-2 Proprinter II
*IBM42013	IBM 4201-3 Proprinter III
*IBM42021	IBM 4202-1 Proprinter XL
*IBM42022	IBM 4202-2 Proprinter II XL
*IBM42023	IBM 4202-3 Proprinter III XL
*IBM42071	IBM 4207-1 Proprinter X24
*IBM42072	IBM 4207-2 Proprinter X24E
*IBM42081	IBM 4208-1 Proprinter XL24
*IBM42082	IBM 4208-2 Proprinter XL24E
*IBM4212	IBM 4212 Proprinter 24P
*IBM4216	IBM 4216-10 Personal Pageprinter
*IBM4226	IBM 4226-302 Printer
*IBM47121	IBM 4712-1 Transaction Printer
*IBM47122	IBM 4712-2 Transaction Printer
*IBM47221	IBM 4722-1 Document Printer
*IBM47222	IBM 4722-2 Document Printer
*IBM5152	IBM 5152 Graphics Printer
*IBM5201	IBM 5201-2 Quietwriter*
*IBM5202	IBM 5202-1 Quietwriter III
*IBM5204	IBM 5204-1 Quickwriter
*IBM5216	IBM 5216 Wheelprinter
*CPQPM15	COMPAQ** PageMark 15 (HP Mode)
*CPQPM20	COMPAQ PageMark 20 (HP Mode)
*HPII	HP LaserJet** Series II
*HPIID	HP LaserJet IID
*HPIIP	HP LaserJet IIP
*HPIII	HP LaserJet III
*HPIIID	HP LaserJet IIID
*HPIIIP	HP LaserJet IIIP
*HPIIISI	HP LaserJet IIISi
*HPIV	HP LaserJet 4
*HP500	HP DeskJet 500
*HP550C	HP DeskJet 550C
*HPPAINT	HP PaintJet
	HP PaintJet XL
	HP PaintJet XL300
*EPAP2250	Epson ActionPrinter 2250
*EPAP3250	Epson ActionPrinter 3250
*EPAP5000	Epson ActionPrinter 5000
*EPAP5500	Epson ActionPrinter 5500
*EPDFX5000	Epson DFX-5000
*EPDFX8000	Epson DFX-8000
*EPFX850	Epson FX-850

Table 33 (Page 2 of 2). Manufacturer Type and Model (MFRTYPMDL) Parameter

*EPFX870	Epson FX-870
*EPFX1170	Epson FX-1170
*EPLQ570	Epson LQ-570
*EPLQ860	Epson LQ-860
*EPLQ870	Epson LQ-870
*EPLQ1070	Epson LQ-1070
*EPLQ1170	Epson LQ-1170
*EPLX810	Epson LX-810
*EPLQ510	Epson LQ-510
*EPLQ2550	Epson LQ-2550
*EPSQ870	Epson SQ-870
*EPSQ1170	Epson SQ-1170
*EPEPL7000	Epson EPL-7000
*EPEPL8000	Epson EPL-8000
*NECP2	NEC** P2 Pinwriter
*NECP2200	NEC P2200 Pinwriter
*NECP2200XE	NEC P2200 XE Pinwriter
*NECP5200	NEC P5200 Pinwriter
*NECP5300	NEC P5300 Pinwriter
*NECP6200	NEC P6200 Pinwriter
*NECP6300	NEC P6300 Pinwriter
*OKI184IBM	Okidata** Microline 184 Turbo (IBM Mode)
*OKI320IBM	Okidata Microline 320 (IBM Mode)
*OKI321IBM	Okidata Microline 321 (IBM Mode)
*OKI390IBM	Okidata Microline 390 Plus (IBM Mode)
*OKI391IBM	Okidata Microline 391 Plus (IBM Mode)
*OKI393IBM	Okidata Microline 393 Plus (IBM Mode)
*OKI590IBM	Okidata Microline 590 (IBM Mode)
*OKI591IBM	Okidata Microline 591 (IBM Mode)
*OKI400	Okidata OL400 LED Page Printer
*OKI800	Okidata OL800 LED Page Printer
*OKI810	Okidata OL810 LED Page Printer
*OKI820	Okidata OL820 LED Page Printer
*OKI3410	Okidata Pacemark 3410
*PAN1123EP	Panasonic** KX-P1123 (Epson Mode)
*PAN1124EP	Panasonic KX-P1124 (Epson Mode)
*PAN1124IEP	Panasonic KX-P1124i (Epson Mode)
*PAN1180EP	Panasonic KX-P1180 (Epson Mode)
*PAN1180IEP	Panasonic KX-P1180i (Epson Mode)
*PAN1191EP	Panasonic KX-P1191 (Epson Mode)
*PAN1624EP	Panasonic KX-P1624 (Epson Mode)
*PAN1654EP	Panasonic KX-P1654 (Epson Mode)
*PAN1695EP	Panasonic KX-P1695 (Epson Mode)
*PAN2123EP	Panasonic KX-P2123 (Epson Mode)
*PAN2124EP	Panasonic KX-P2124 (Epson Mode)
*PAN2180EP	Panasonic KX-P2180 (Epson Mode)
*PAN2624EP	Panasonic KX-P2624 (Epson Mode)
*PAN4410HP	Panasonic KX-P4410 (HP Mode)
*PAN4420HP	Panasonic KX-P4420 (HP Mode)
*PAN4430HP	Panasonic KX-P4430 (HP Mode)
*PAN4450IHP	Panasonic KX-P4450i (HP Mode)
*PAN4451HP	Panasonic KX-P4451 (HP Mode)
*WSCST	Printer is not listed

PPRSRC1

Specifies the type of paper used in paper source one.

***MFRTYPMDL:** The system uses the suggested setting for this printer.

***LETTER:** The paper for this source is letter-sized (8.5 x 11 inches).

***LEGAL:** The paper for this source is legal-sized (8.5 x 14 inches).

***EXECUTIVE:** The paper for this source is executive-sized (7.25 x 10.5 inches).

***A4:** The paper for this source is A4-sized (210mm x 297mm).

***A5:** The paper for this source is A5-sized (148 x 210mm).

***B5:** The paper for this source is B5-sized (182 x 257mm).

***CONT80:** The paper for this source is continuous form (8.0 inches).

***CONT132:** The paper for this source is continuous form (13.2 inches).

***NONE:** No paper source number one is specified.

PPRSRC2

Specifies the type of paper used in paper source two.

***MFRTYPMDL:** The system uses the suggested setting for this printer.

***LETTER:** The paper for this source is letter-sized (8.5 x 11 inches).

***LEGAL:** The paper for this source is legal-sized (8.5 x 14 inches).

***EXECUTIVE:** The paper for this source is executive-sized (7.25 x 10.5 inches).

***A4:** The paper for this source is A4-sized (210mm x 297mm).

***A5:** The paper for this source is A5-sized (148 x 210mm).

***B5:** The paper for this source is B5-sized (182 x 257mm).

***NONE:** No paper source number two is specified.

ENVELOPE

Specifies the type of envelopes used in the third paper source.

***MFRTYPMDL:** The system uses the suggested setting for this printer.

***MONARCH:** The envelopes for this source are monarch-sized (3.875 x 7.5 inches).

***NUMBER9:** The envelopes for this source are number 9-sized (3.875 x 8.875 inches).

***NUMBER10:** The envelopes for this source are number 10-sized (4.125 x 9.5 inches).

***B5:** The envelope for this source is B5-sized (176 x 250mm).

***C5:** The envelopes for this source are C5-sized (162mm x 229mm).

***DL:** The envelopes for this source are DL-sized (110mm x 220mm).

***NONE:** No envelope source is specified.

ASCII899

Specifies whether the printer has ASCII code page 899 installed.

***NO:** The printer does not have ASCII code page 899 installed.

***YES:** The printer has ASCII code page 899 installed.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Table 34. Print Quality (PRTQLTY Parameter)

Printer	*NLQ	*STD	*DRAFT
4214	50 CPS print speed	50 CPS print speed	200 CPS print speed
4224 Model 1	50 CPS print speed	100 CPS print speed	200 CPS print speed
4224 Model 2	100 CPS print speed	200 CPS print speed	400 CPS print speed
4234	120 LPM print speed	300 LPM print speed	410 LPM print speed
5219	Ribbon advances in normal mode	Ribbon advances in normal mode	Ribbon advances one-third as fast as standard mode

Table 35. DBCS-Capable Device Features (IGCFEAT Parameter)

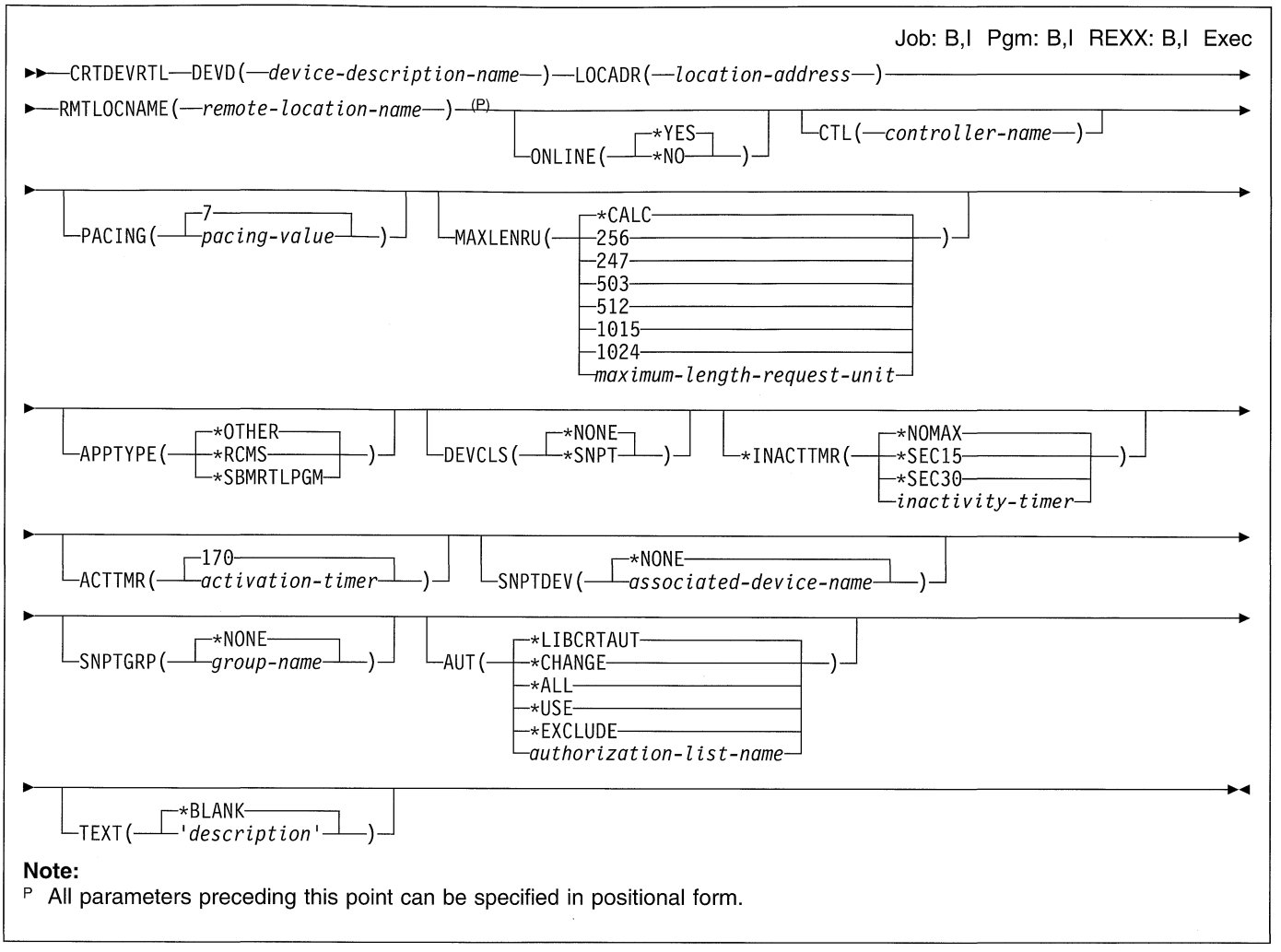
Language/ Device	Type of Physical DBCS-Capable Device	Type-Model	Last Code Point
Japanese 24x24 Printers	Attached to 5295-001 Display Stations	5553-B01	2424J1 55FE
	Attached to 5295-002 Display Stations	5553-B01	2424J1 68FE
	Attached to Personal System/55	5553-B01	2424J1 68FE
	5227-001 Printer	5553-B01	2424J2 55FE
	5327-001 Printer	5553-B01	2424J2 68FE
Japanese 32x32 Printers	5337-001 Printer	5553-B01	3232J0
	5383-200 Printer	5583-200	3232J0
Korean 24x24 Printers	Attached to 5295 Display Stations	5553-B01	2424K0
	Attached to Personal System/55	5553-B01	2424K0
	5227-002 Printer	5553-B01	2424K2 52FE
Traditional Chinese 24x24 Printers	Attached to 5295 Display Stations	5553-B01	2424C0
	Attached to Personal System/55	5553-B01	2424C0
	5227-003 Printer	5553-B01	2424C2 5CFE
Simplified Chinese 24x24 Printers	Attached to Personal System/55	5553-B01	2424S0
	5227-005 Printer	5553-B01	2424S2 6FFE

Example

```
CRTDEVPRT   DEVD(PRT1)  TYPE(5219)  DEVCLS(*LCL)
  MODEL(D1)  PORT(0)  SWTSET(1)  CTL(CTL01)
  FONT(011)
```

This command creates a device description for a local printer named PRT1. It is a 5219 Model D1 Printer attached to Port 0 of CTL01. It has an address of 1 and uses the Courier font.

CRTDEVRTL (Create Device Description (Retail)) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Retail) (CRTDEVRTL) command creates a device description for a retail device. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

LOCADR

Specifies the local location address. Specify a 2-character hexadecimal value ranging from 01 through FE.

RMTLOCNAME

Specifies the remote location name of the system with which this object communicates.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

PACING

Specify a value ranging from 0 through 7 to indicate the pacing count used when communicating with this device.

7: A value of 7 is used as the RU pacing value.

pacing-value: Specify a value ranging from 0 through 7 to indicate the pacing count used for this device.

MAXLENRU

Specifies the default maximum request unit (RU) length that can be sent or received by the local systems if the maximum size is not specified in the bind.

Note: If the LOCADR parameter has a value of 01, then the only valid lengths for the maximum request unit is *CALC or 256 bytes.

***CALC:** The system determines the value to use.

maximum-length-request-unit: Specify either 247, 256, 503, 512, 1015, or 1024 bytes for the maximum length of incoming request units.

APPTYPE

Specifies the application type being used.

***OTHER:** This device is used to communicate with either the Host Command Processor (HCP) (if LOCADDR = 01) or an application on the controller.

***RCMS:** This device communicates with the remote change management server (RCMS).

***SBMRTLPGM:** This device is used with the SBMRTLPGM command to start a program on the retail controller using the Advanced Data Communication for Stores (ADCS) SUP (Start User Program) support. This value is meaningful only when the Retail Point-of-Sale Communications Facility Licensed Program is being used on the controller.

DEVCLS

Specifies the device class for this device.

***NONE:** This device description will not be using SNA Passthrough support.

***SNPT:** This device description is for a display station connected to an SNA pass-through advanced program-to-program communications (APPC) controller. SNA pass-through support allows the user to connect this display station with host logical unit (LU) types 0 through 3 applications.

INACTTMR

Specifies an inactivity timer (time-out) value. This parameter also specifies what happens when the time-out value is exceeded, dependent on other attributes of the device:

- For devices connected using SNA pass-through (SNPT) support, the user is informed by a message to QSYSOPR and the session is ended when the amount of time that the device is not bound to a host application exceeds the time-out value. The user must reestablish the connection and session.
- For all other attachments, valid values range from 1 through 30 minutes.

Note: This timer is not used by devices allocated to a subsystem (normal interactive use) because the subsystem always has a file open for the device. The timer is used by batch jobs that open and close files for the device.

***NOMAX:** Maximum inactivity time is not tracked.

***SEC15:** A 15-second time-out period is used.

***SEC30:** A 30-second time-out period is used.

inactivity-timer: Specify a time-out value.

ACTTMR

Specifies, for switched lines, the amount of time (in seconds) that the SNA pass-through support waits for the device to respond to the activation request from the host AS/400 system. If the device does not respond within this time, it is considered not available.

170: Specifies the activate time of 170 seconds.

activation-timer: Specify a number ranging from 1 through 2550 indicating the number of seconds before the device is considered not available.

SNPTDEV

Specifies the name of the associated SNA pass-through device that is attached to a host or advanced program-to-program communications (APPC) controller.

***NONE:** No name is specified.

associated-device-name: Specify the name of a device that is attached to a host or an APPC controller that is associated with this device.

SNPTGRP

Specifies the name configured for a group of host devices in a configuration list. This indicates that this device is associated with any one of the devices in that group which is available.

***NONE:** No name is specified.

group-name: Specify the name configured for a group of host devices that must be associated with this device.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object.

The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

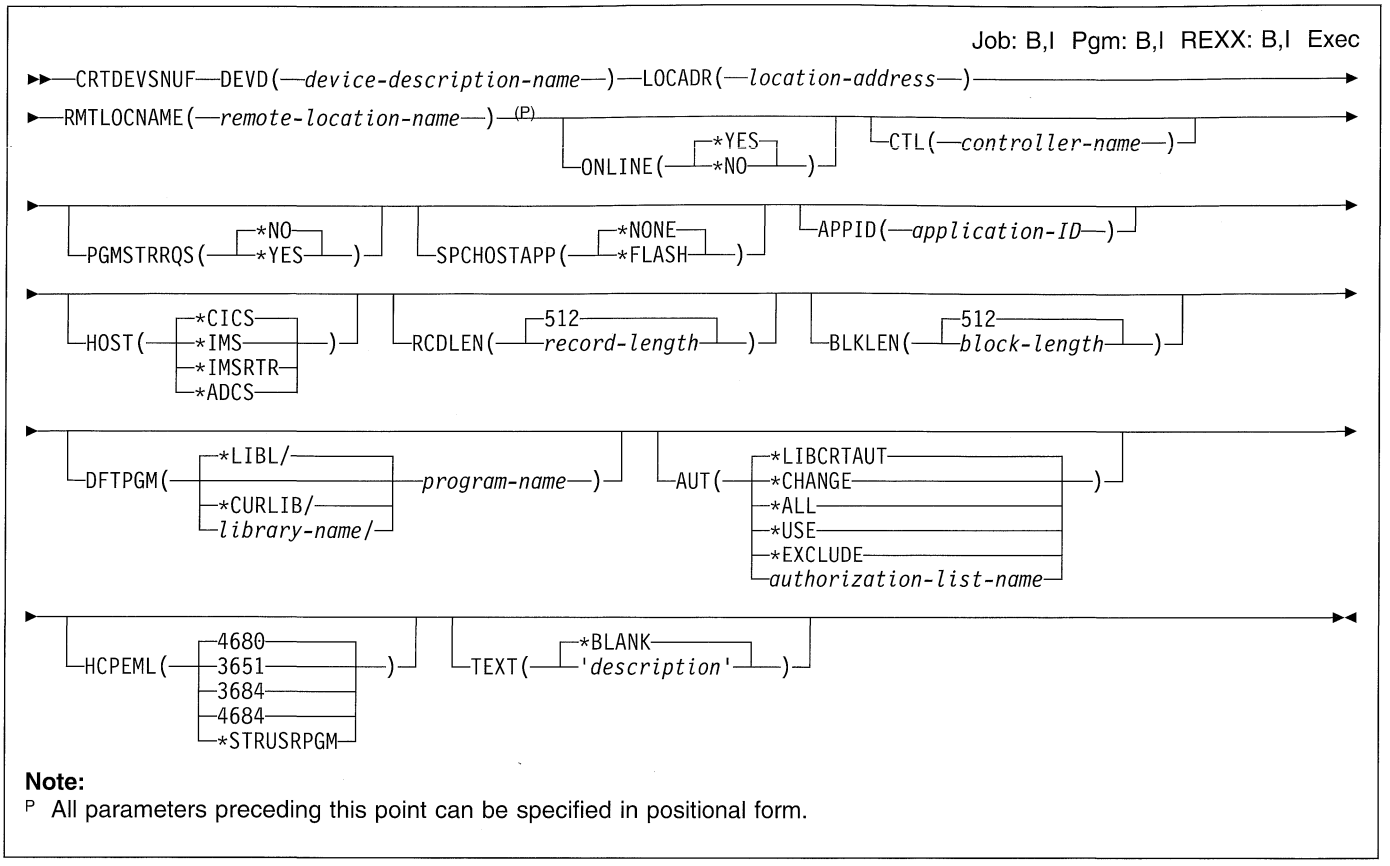
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVRTL   DEVD(RTL1)   LOCADR(05)
  RMTLOCNAME(DETROIT)  CTL(CTL4)
```

This command creates a retail device description named RTL1. The location address of the retail device is X'05'. The remote-location name is Detroit, and the device is attached to the retail controller CTL4.

CRTDEVSNUF (Create Device Description (SNUF)) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (SNUF) (CRTDEVSNUF) command creates a device description for a Systems Network Architecture Upline Facility (SNUF) device. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

DEVD

Specifies the name of the device description being created.

LOCADR

Specifies the local location address. Specify a 2-character hexadecimal value ranging from 01 through FE.

RMTLOCNAME

Specifies the remote location name of the system with which this object communicates.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

PGMSTRRQS

Specifies whether this device is reserved for host system call through a Program Start Request.

***NO:** This device is not reserved for a start program request.

***YES:** This device is reserved for a start program request.

SPCHOSTAPP

Specifies whether SNUF customizes support for special host applications outside the Customer Information Control System for Virtual Storage (CICS/VS) or Information Management System for Virtual Storage (IMS/VS) application layer.

***NONE:** SNUF does not customize support for special host applications.

***FLASH:** SNUF customizes support for the Federal Link Access for Secondary Half-sessions (*FLASH) protocol application.

APPID

Specifies the Virtual Telecommunications Access Method (VTAM) Application Identifier sent with the sign-on message.

HOST

Specifies the type of host system with which the device communicates.

***CICS:** The host system type is CICS/VS.

***IMS:** The host system type is IMS/VS.

***IMSRTR:** Information Management System Ready to Receive (IMSRTR) is the type of the host system.

***ADCS:** Advanced Data Communication for Stores (ADCS) is the type of host system.

RCDLEN

Specifies the maximum record length allowed when communicating with this device. The maximum value for this parameter is 32767. The value must not exceed the block length value for this device.

512: The record length is 512.

record-length: Specify the maximum record length (in bytes) to use with this device file. The value must be at least the size of the largest record sent. If a record is longer than the specified record length, a run time error occurs when the record is sent or received. Valid values range from 1 through 32767 bytes for SNUF communications. For BSCEL communications, the maximum record length is 8192 bytes.

BLKLEN

Specifies the maximum block length allowed when communicating with this device. The maximum value for this parameter is 32767. This value must be greater than or equal to the record length.

512: The block length is 512.

block-length: Specify the maximum block length (in bytes) of records sent. The value must be at least the size of the largest record sent. Valid values range from 1 through 32767.

DFTPGM

Specifies the qualified name of the program called if a program start request is received from the host system not using *EXEC/*EXEX/*TXTC/*TXTX format. This parameter is required if the value specified for the program start request parameter is *YES.

The name of the program name can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the default program.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

HCPEML

Specifies the Host Command Processor (HCP) emulation to be done by the AS/400 Retail Licensed Program in support of the Advanced Data Communications for Stores (*ADCS) value on the HOST parameter. HCPEML can only be specified when *ADCS is the value for the HOST parameter.

4680: The host is running ADCS and uses this Device Description for a 4680 HCP emulated session.

3651: The host is running ADCS and uses this Device Description for a 3651 HCP emulated session.

3684: The host is running ADCS and uses this Device Description for a 3684 HCP emulated session.

4684: The host is running ADCS and uses this Device Description for a 4684 HCP emulated session.

***STRUSRPGM:** The host is running ADCS and uses this Device Description for the Start User Program (SUP) emulated session.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

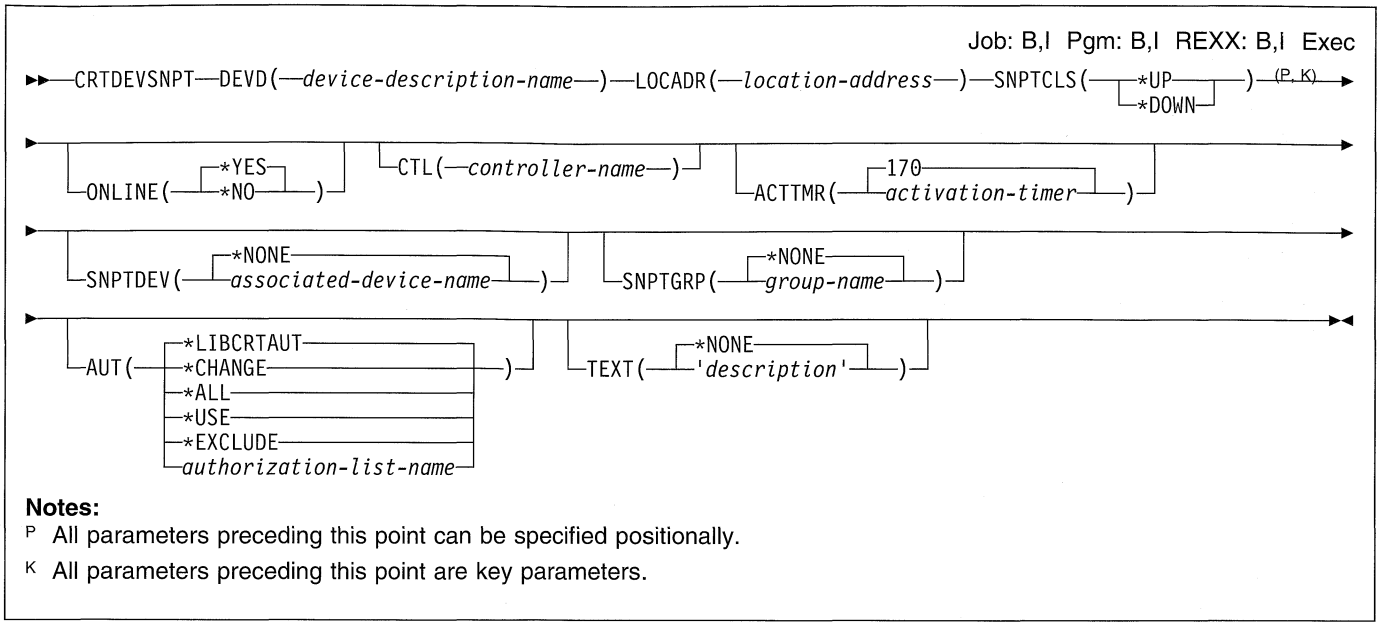
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVSNUF   DEVD(SNUFDEV02)  LOCADR(FE)
  RMTLOCNAME(DETROIT)  CTL(SNUFCTL01)
  PGMSTRRQS(*YES)  DFTPGM(PAYROLL)
```

This command creates a device description for a SNUF communications device named SNUFDEV02. The device is attached to the communications controller SNUFCTL01 in Detroit, and is at location address X'FE'. The device will be program start request-capable and the default program is PAYROLL.

CRTDEVSNPT (Create Device Description (SNA Pass-Through)) Command



Notes:
P All parameters preceding this point can be specified positionally.
K All parameters preceding this point are key parameters.

Purpose

The Create Device Description (SNA Pass-Through) (CRTDEVSNPT) command creates a device description for an SNA pass-through device.

Required Parameters

DEVD
Specifies the name of the device description being created.

LOCADR
Specifies the local location address for this SNA pass-through device. Specify two hexadecimal characters with values ranging from 01 through FF.

SNPTCLS
Specifies whether this device is attached to a SNA pass-through host controller or to a SNA pass-through downstream (APPC) controller.

***UP:** Specifies the device to be attached to a SNA pass-through host controller.

***DOWN:** Specifies the device to be attached to a SNA pass-through downstream (APPC) controller.

Optional Parameters

ONLINE
Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The device is automatically varied on at IPL.

***NO:** This device is not automatically varied on at IPL.

CTL
Specifies the name of the controller description to which this object is attached.

ACTTMR
Specifies, for switched lines, the amount of time (in seconds) that the SNA pass-through support waits for the device to respond to the activation request from the host AS/400 system. If the device does not respond within this time, it is considered not available.

170: Specifies the activate time of 170 seconds.

activation-timer: Specify a number ranging from 1 through 2550 indicating the number of seconds before the device is considered not available.

SNPTDEV
Specifies the name of the associated SNA pass-through device that is attached to a host or advanced program-to-program communications (APPC) controller.

***NONE:** No name is specified.

associated-device-name: Specify the name of a device that is attached to a host or an APPC controller that is associated with this device.

SNPTGRP
Specifies the name configured for a group of host devices in a configuration list. This indicates that this device is associated with any one of the devices in that group which is available.

***NONE:** No name is specified.

group-name: Specify the name configured for a group of host devices that must be associated with this device.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of an existing authorization list. Users included in the authorization list are granted authority as specified by the list.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NONE:** No text is specified.

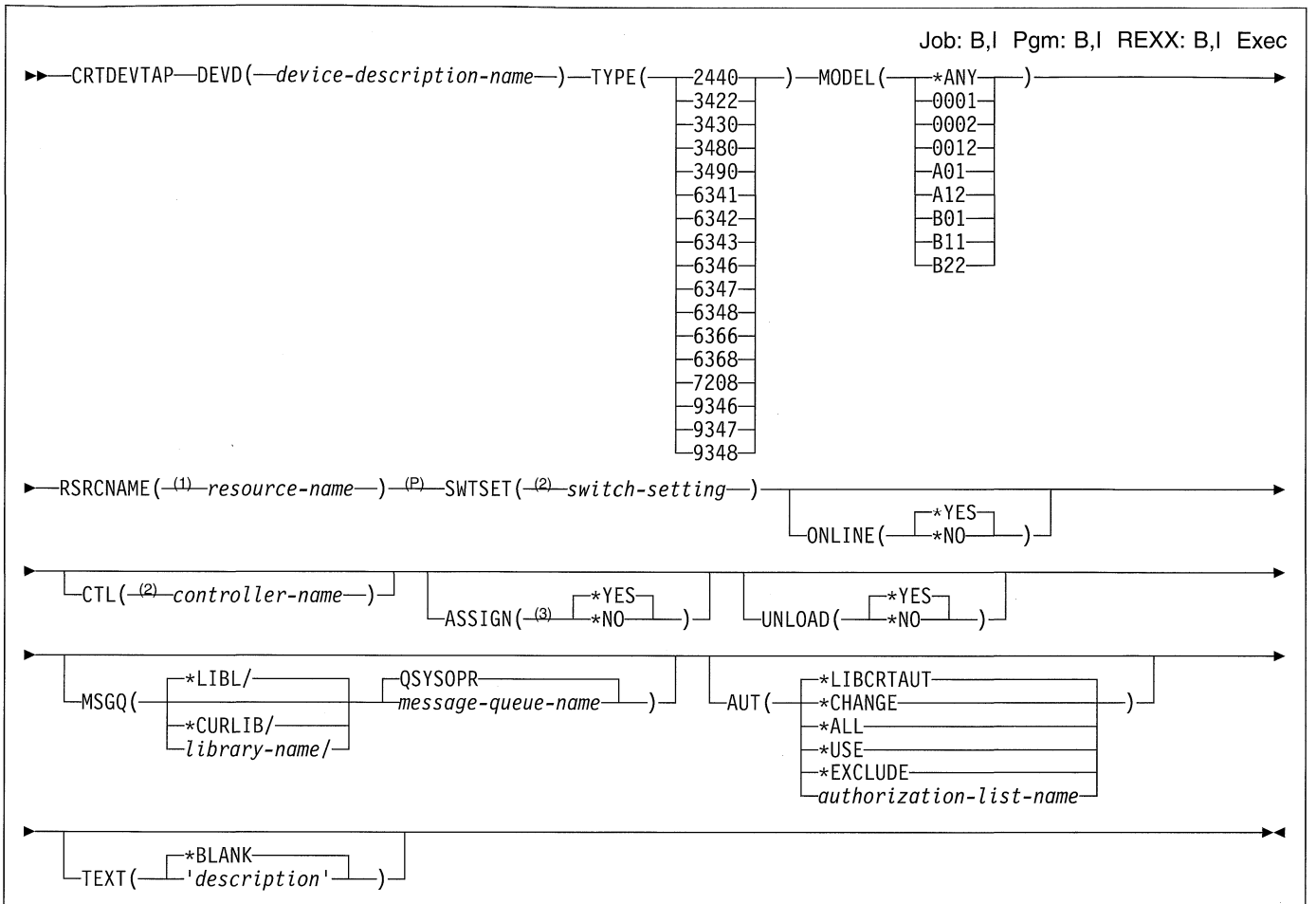
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVSNPT   DEVD(SNPTDEV1)  LOCADR(05) SNPTCLS(*UP)
  CTL(MYCTL)  SNPTDEV(DOWNDEV1)
```

This command creates a SNA pass-through device description named SNPTDEV1. The location address of the device is X'05'. The device is attached to controller MYCTL. The SNA pass-through class is *UP and the device name associated with this device is DOWNDEV1.

CRTDEVTAP (Create Device Description (Tape)) Command



Notes:

1 Types 2440, 6346, 6347, 6366, 9346, 9347, and 9348 only

2 Types 3422, 3430, 3480, and 3490 only

3 Types 3480 and 3490 only

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Device Description (Tape) (CRTDEVTAP) command creates a device description for a tape device. More information about using this command is in the *Device Configuration Guide*.

Required Parameters

DEVD

Specifies the name of the device description being created.

TYPE

Specifies the type of device this description represents. Any of the following types, listed in numeric order, are valid:

2440	6341	6348	9347
3422	6342	6366	9348
3430	6343	6368	
3480	6346	7208	
3490	6347	9346	

MODEL

Specifies the model number of the device type for this description.

***ANY:** This value is for 3490 device types.

0001: This model number is for a 6341, 6342, 6343, 6346, 6347, 6348, 6366, 6368, 9346, 9347, or 9348 device type.

0002: This model number is for a 7208, 9346, or 9348 device type.

0012: This model number is for a 7208 device type.

A01: The model number is for a 3422 or 3430 device type.

A12: The model number is for a 2440 device type.

B01: The model number is for a 3422 or 3430 device type.

B11: This model number is for a 3480 device type.

B22: This model number is for a 3480 device type.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *STG specified on the TYPE parameter to determine the resource name.

SWTSET

Specifies a 1-character switch setting field with a value ranging from 0 through F.

Note: This parameter is required for a device type of 3422, 3430, 3480, or 3490, but is not allowed for other device types. Valid values for a 3422, 3480 or 3490 device type ranges from 0 through F. Valid values for a 3430 device type range from 0 through 3.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The object is automatically varied on at IPL.

***NO:** This object is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

Note: This parameter is valid only for 3422, 3430, 3480 and 3490 device types.

ASSIGN

Specifies whether the tape drive is assigned to the system when it is varied on.

***YES:** The tape drive is assigned when the device is varied on.

***NO:** The tape drive is not assigned when the device is varied on.

UNLOAD

Specifies whether the tape drive is unloaded when the device is varied off.

***YES:** The tape drive is unloaded when the device is varied off.

***NO:** The tape drive is not unloaded when the device is varied off. The tape is rewound, but not past the beginning-of-tape marker.

MSGQ

Specifies the qualified name of the message queue to which messages are sent.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QSYSOPR: Messages are sent to the QSYSOPR message queue.

message-queue-name: Specify the name of the message queue to which operational messages are sent.

AUT

Specifies the authority given to users who do not have specific authority to the device description, who are not on an authorization list, and whose user group has no specific authority to the device description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the device description.

***USE:** The user can perform basic operations on the device description, such as running a program or reading a file. The user cannot change the device description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the device description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

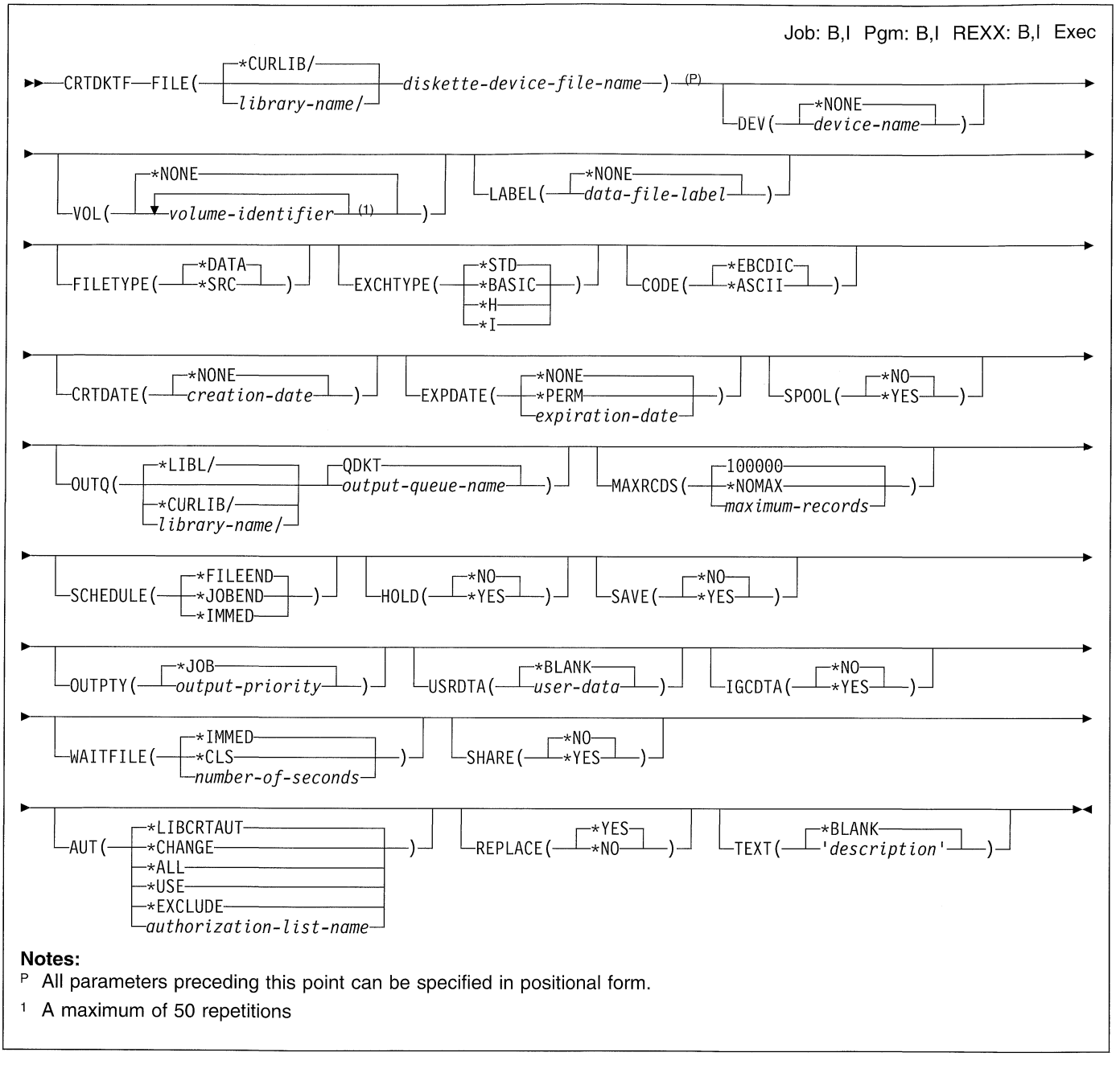
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDEVTAP   DEVD(TAP01)  TYPE(9347)  MODEL(0001)
  RSRCNAME(TAP01)
```

This command creates a device description for a tape drive named TAP01. It is a 9347 Model 0001 tape drive with a resource name of TAP01.

CRTDKTF (Create Diskette File) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 50 repetitions

Purpose

The Create Diskette File (CRTDKTF) command creates a diskette device file. The device file contains the file description, which identifies the device to be used and specifies the input and output data spooling requirements; the device file does not contain data. The diskette device file is used to read and write records on diskettes that are in the diskette device and that have been initialized in the basic, H, or I exchange format. The same device file is used for both input and output operations.

Note: This command is not used to create device files for use in save/restore operations. User-created device files are not needed for save/restore operations.

All the information in the diskette file description is contained in the command that creates it; there is no data description specifications (DDS) for diskette device files. The diskette file has only one record format for input/output operations. The record format consists of one character field containing the input data retrieved from the device or the output data written to the device. The program using the device file must describe the fields in the record format so that the program can arrange the data either received from or sent to the

device in the manner specified by the diskette file description.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the diskette file can be qualified by one of the following library values:

***CURLIB:** The diskette file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the diskette file is created.

diskette-device-file-name: Specify the name of the diskette device file being created.

Optional Parameters

DEV

Specifies the name of the diskette device used with this diskette device file to perform input/output data operations. The device name of the IBM-supplied diskette device description is QDKT. This parameter is ignored if SPOOL(*YES) is specified for the file when it is opened.

***NONE:** No device name is specified. The name of the diskette device can be specified before the device file is opened, in a Change Diskette File (CHGDKTF) command or Override with Diskette File (OVRDKTF) command, or in the high-level language program that opens the file.

device-name: Specify the name of the device that is used with this diskette device file. The device name must already exist on the system as a device description before this device file is created.

VOL

Specifies one or more volume identifiers used by the file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NONE:** The diskette volume identifiers are not specified for this file in this command. They can be specified later before the device file is opened, either in a Override with Diskette File (OVRDKTF) command or a Change Diskette File (CHGDKTF) command, or in the high-level language program. Otherwise, no volume identifier checking is done.

volume-identifier: Specify the identifiers of one or more volumes in the order in which they are put on the device and used. Each volume identifier contains up to 6 alphanumeric characters. A blank is used as a separator character when listing multiple identifiers.

LABEL

Specifies the data file label of the data file on diskette that is used with this diskette device file. For input files (diskette input to system), this label specifies the identifier of the file that exists on the diskette. For output files (system output to diskette), the label specifies the identifier of the file that is created on the diskette. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NONE:** The data file label is not specified here. It can be specified before the device file is opened, in a CHGDKTF or OVRDKTF command, or in the high-level language program that opens the file.

data-file-label: Specify up to 8 characters for the identifier of the data file used with this diskette device file.

FILETYPE

Specifies whether the diskette device file being created describes data records or describes source records (statements) for a program or another file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***DATA:** The diskette file describes data records.

***SRC:** The diskette file describes source records.

EXCHTYPE

Specifies, for diskette output files only, the exchange type used by the device file when the system is writing diskette data. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***STD:** The basic exchange format is used for a type 1 or a type 2 diskette. The H exchange type is used for a type 2D diskette.

***BASIC:** The basic exchange type is used.

***H:** The H exchange type is used.

***I:** The I exchange type is used.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

CRTDATE

Specifies the date when the diskette data file was created on the diskette.

Note: The creation date parameter is valid only for diskette input data files. If the creation date written on the diskette containing the data file does not match the date specified for the device file when it is opened, an error message is sent to the user program.

***NONE:** The creation date is not specified. It is not checked unless it is supplied before the device file is opened, either in a OVRTAPF command or CHGTAPF command, or in the high-level language program.

creation-date: Specify the creation date of the data file used by this device file. The date must be specified in the format defined by the job attributes DATFMT and, if separators are used, DATSEP. However, the specified date is put in the diskette label in the format yymmdd.

EXPDATE

Specifies the expiration date. The files cannot be overwritten until the expiration date. The expiration date must be later than or equal to the current date.

***NONE:** No expiration date for the data file is specified; the file is protected for 1 day. Its protection ends the day after it is created.

***PERM:** The data file is permanently protected. An expiration date of 999999 is assigned.

expiration-date: Specify the expiration date of the data file. The date must be specified in the format defined by the job attributes DATFMT and, if separators are used, DATSEP. However, the specified date is put in the diskette label as yymmdd.

SPOOL

Specifies whether the input or output data for the diskette device file is spooled.

***NO:** The data is not spooled. If this file is opened for input, the data is read directly from the diskette. If this is an output file, the data is written directly to the diskette as it is processed by the program.

Note: If SPOOL(*NO) is specified, the following parameters in this command are ignored: OUTQ, MAXRCDS, SCHEDULE, HOLD, SAVE, OUTPTY, and USRDTA.

***YES:** The data is spooled. If this file is opened for input, an inline data file having the specified name is processed; otherwise, the next unnamed inline spooled file is processed. More information on named and unnamed inline files is in the *Guide to Programming for Tape and Diskette*. If this is an output file, the data is spooled for processing by a diskette or print writer.

OUTQ

Specifies the qualified name of the output queue.

The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDKT: The spooled output data is sent to the IBM-supplied QDKT output queue. If no library qualifier is specified, *LIBL is used to find the output queue.

output-queue-name: Specify the name of the output queue to which the output data is spooled. The IBM-supplied output queue that is used by the diskette file is the QDKT output queue, stored in the QGPL library.

MAXRCDS

Specifies, for spooled files only, the maximum number of records in the spooled file for spooled jobs using this diskette device file.

100000: Up to 100,000 records are in the spooled file for the diskette data file that is produced by this device file.

***NOMAX:** The system maximum is used.

maximum-records: Specify the maximum number of diskette records that are in the spooled file. Valid values range from 1 through 500000.

SCHEDULE

Specifies, for spooled output only, when the spooled file is available to a writer.

***FILEEND:** The spooled file is made available to the writer as soon as the file is closed in the program.

***JOBEND:** The spooled file is made available to the writer only after the entire job is completed.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

HOLD

Specifies, for spooled output only, whether the spooled file is held. The spooled file can be released by using the Release Spooled File (RLSSPLF) command.

***NO:** The spooled printer file is not held by the output queue. The spooled output is available to a writer based on the SCHEDULE parameter value.

***YES:** The spooled file is held until released by the Release Spool File (RLSSPLF) command.

SAVE

Specifies, for spooled output only, whether the spooled file is saved (left on the output queue) after the output has been produced.

***NO:** The spooled file data is not saved on the output queue after it has been produced.

***YES:** The spooled file data is saved on the output queue until the file is deleted.

OUTPTY

Specifies the output priority for spooled output files that are produced by this job. The highest priority is 1 and the lowest priority is 9. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***JOB:** The output priority associated with the job that created the spooled file is used.

output-priority: Specify the output priority. Valid values range from 1 (high priority) through 9 (low priority).

USRDTA
Specifies, for spooled output only, the user-specified data that identifies the file.

***BLANK:** Ten blanks are used as the user data.

user-data: Specify up to 10 characters of text.

IGCDTA
Specifies whether the file processes double-byte character set (DBCS) data.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

WAITFILE
Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds that the program waits for the file resources to be allocated to the diskette file when the file is opened, or the wait time for the device allocated when an acquire operation is performed to the file. Valid values range from 1 through 32767 seconds.

SHARE
Specifies whether the open data path (ODP) for the diskette file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

AUT
Specifies the authority given to users who do not have specific authority to the diskette file, who are not on an authorization list, and whose user group has no specific authority to the diskette file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the diskette file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the diskette file). The public authority is determined when the diskette file is created. If the CRTAUT value for the library changes after the diskette file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the diskette file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the diskette file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the diskette file.

***USE:** The user can perform basic operations on the diskette file, such as running a program or reading a file. The user cannot change the diskette file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the diskette file.

authorization-list-name: Specify the name of the authorization list used.

REPLACE
Specifies whether an existing file is replaced by the new diskette file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** The existing diskette file is replaced by the one being created.

***NO:** The existing file, if any, is not replaced by the diskette file.

TEXT
Specifies text that briefly describes the diskette device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating Diskette Device File

```
CRTDKTF   FILE(DSPHST)
```

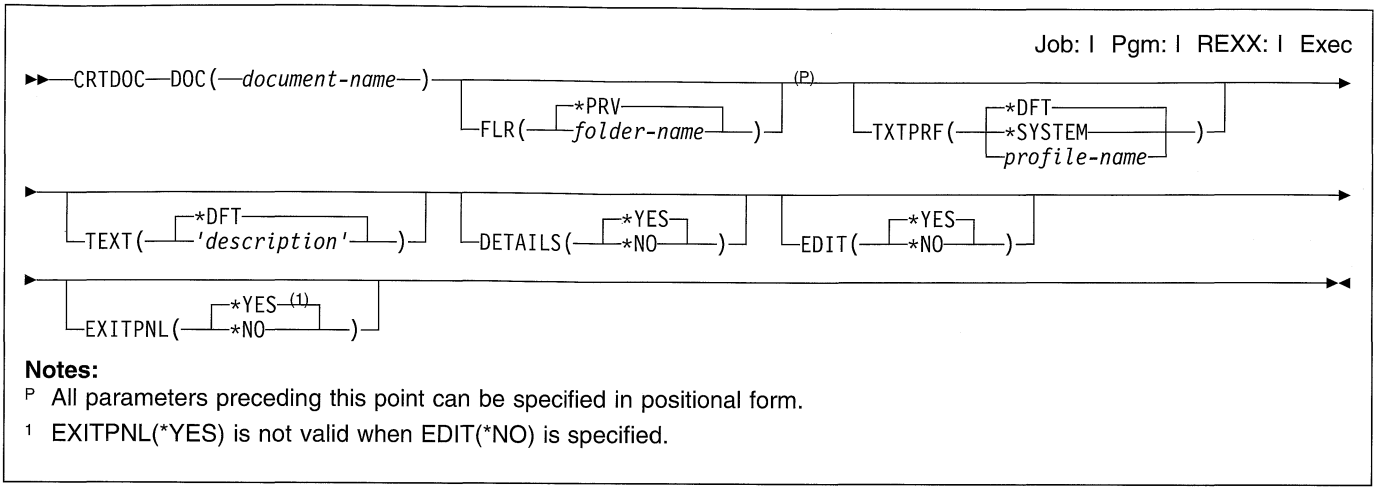
This command creates a diskette device file named DSPHST. The defaults for all the other parameters are assumed. The device name, diskette volume, file label, and the creation date of the data file on diskette must be specified in another CL command or in each program that uses the device file. The device file describes diskette data files that are in EBCDIC code and that are spooled for both input and output. Output goes to the QDKT output queue and then onto diskette as soon as the file is closed by the program. When output is produced from the output queue, only one copy is produced.

Example 2: Specifying DBCS Data Processing

```
CRTDKTF   FILE(IGCLIB/IGCDKT) IGCDTA(*YES)
```

This command creates a diskette device file named IGCDKT, which is stored in the library IGCLIB, and it can process double-byte character set (DBCS) data.

CRTDOC (Create Document) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 EXITPNL(*YES) is not valid when EDIT(*NO) is specified.

Purpose

The Create Document (CRTDOC) command allows the user to create a new document when using the word processing function of OfficeVision/400.

First the Create Document Details display is shown. Then, if the Enter key is pressed on this display, the Edit display is shown.

More information on creating documents is in *Using OfficeVision/400* Word Processing*.

Required Parameter

DOC

Specifies the name of the document being created. Specify the name of the document. Up to 12 characters can be specified in the required format (document.ext).

Optional Parameters

FLR

Specifies the name of the folder to contain the document being created.

***PRV:** The name used in the previous session is used.

folder-name: Specify the name of the folder to contain the document being created.

TXTPRF

Specifies the text profile used as the base for the document.

***DFT:** The default text profile is used.

***SYSTEM:** The system text profile is used.

profile-name: Specify the name of the text profile to use. A maximum of 12 characters can be specified.

TEXT

Specifies text that briefly describes the document description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***DFT:** A default description is specified for the document.

description: Specify a maximum of 44 characters, enclosed in apostrophes.

DETAILS

Specifies whether to request or bypass the Document Details display.

***YES:** The Document Details display is shown.

***NO:** The Document Details display is not shown.

EDIT

Specifies whether document editing is bypassed.

***YES:** The document is edited after being created.

***NO:** The document is not edited after being created.

EXITPNL

Specifies whether the Exit Document display is shown when F3(Exit) or F12(Cancel) is pressed to end the editing.

***YES:** The Exit Document display is shown when F3(Exit) or F12(Cancel) is pressed to end the editing.

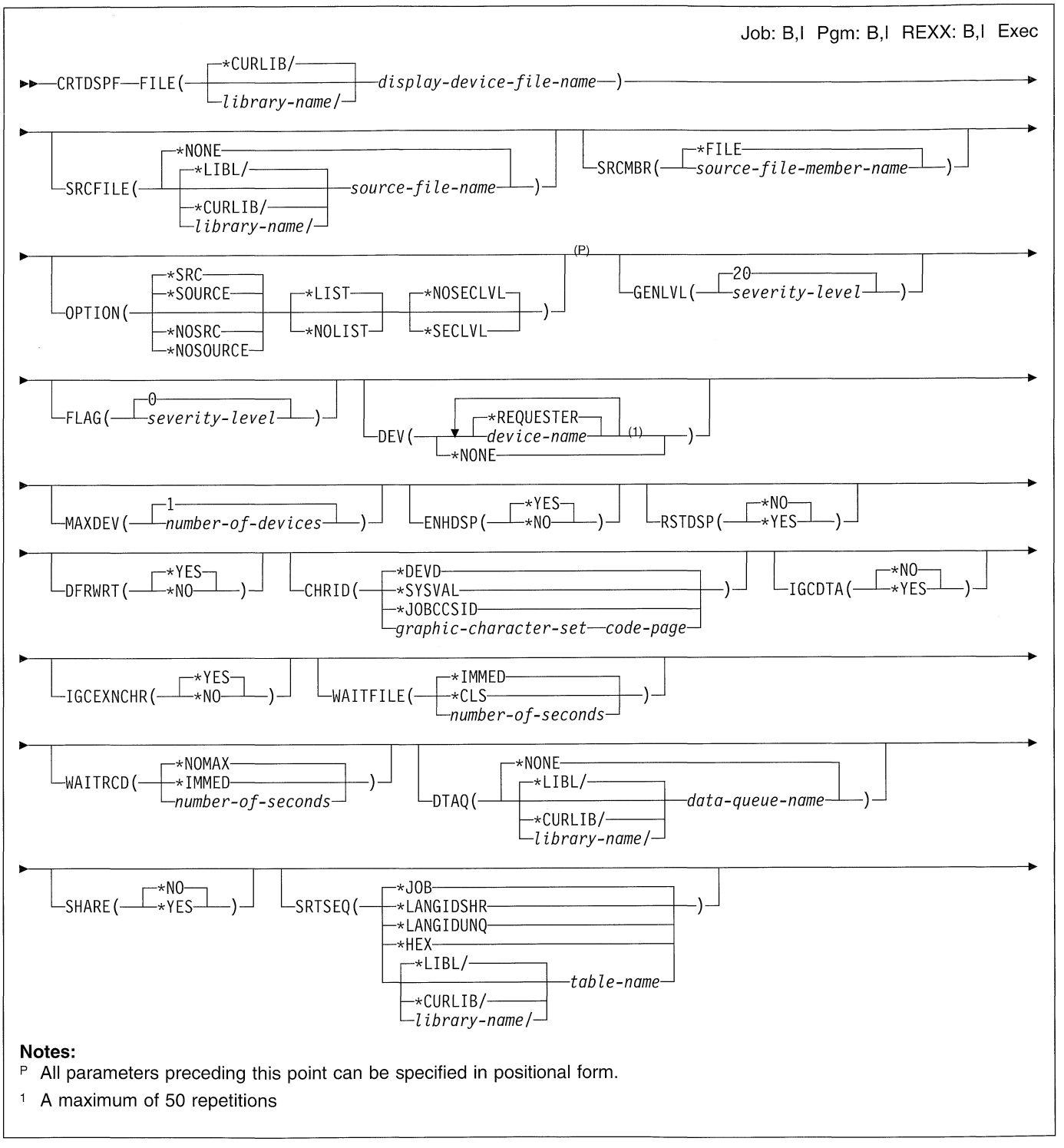
***NO:** The Exit Document display is not shown when F3(Exit) or F12(Cancel) is pressed to end the editing.

Example

```
CRTDOC   DOC(NEWDOC)  FLR(MYFLR)
```

This command creates a new document called NEWDOC in folder MYFLR.

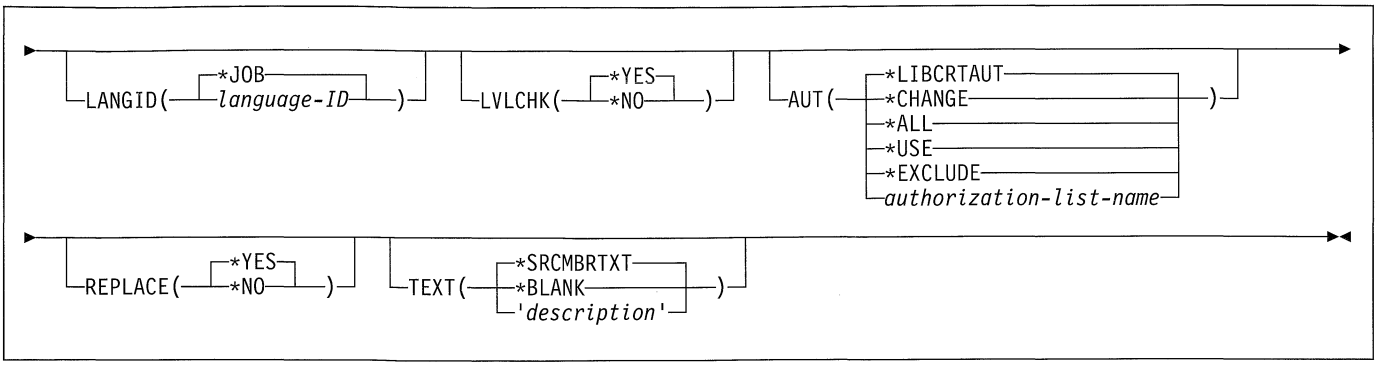
CRTDSPF (Create Display File) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 50 repetitions



Purpose

The Create Display File (CRTDSPF) command creates a display device file. The device file contains the file description, which identifies the device used and, optionally, the record formats used by the device (if specified in data description specifications (DDS)); the device file does not contain data. The display device file sends records to one or more display devices associated with the device file, and to receive records from the display devices.

The display file description contains of information that is specified in two places: (1) in the source file that contains the DDS (if used); and (2) in the CRTDSPF command. The DDS contains the specifications for each record format in the device file and for the fields in each record format.

The Change Display File (CHGDSPF) or Override Display File (OVRDSPF) command is used in a program to change or override the parameter values specified in the display file description; the override command must be run before the display file is opened by the program. Overridden values are changed only for the running of the program; once the program ends, the original parameter values specified for the display file are used.

Note: If an application program attempts to acquire a work station on a switched line and the line connection has been lost or has never been established, the application program waits indefinitely until the connection is established.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the file can be qualified by one of the following library values:

***CURLIB:** The file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the file is created.

display-device-file-name: Specify the name of the display device file.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file (if specified) that contains the DDS for the records in the display device file. More information on the specifications stated in DDS is in the *Guide to Programming Displays* and the *DDS Reference*.

***NONE:** There is no DDS source file for this display device file; either the display device file has only one record format with no fields, or the program that uses the file must describe the record formats and their fields.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

source-file-name: Specify the name of the source file that contains the DDS for this display device file.

SRCMBR

Specifies the name of the member in the source file that contains the DDS for this display device file. The SRCMBR parameter is valid only if SRCFILE is specified.

***FILE:** The source file member name is the same as the device file name specified in the FILE parameter.

source-file-member-name: Specify the name of the member in the source file specified by SRCFILE that is used to create the display device file.

OPTION

Specifies the type of printout produced when the file is created. Up to three of the following values can be

specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value is used.

Note: The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

Program Creation Options

***SRC** or ***SOURCE:** A printout of the source statements used to create the file and errors that occur is created.

***NOSRC** or ***NOSOURCE:** No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the record format containing the error.

Source Listing Options

***LIST:** An expanded source printout is created, showing a detailed list of the file specifications that result from the source statements and references to other file descriptions.

***NOLIST:** An expanded source printout is not created.

***NOSECLVL:** The messages section of the data description specifications (DDS) listing does not contain the second-level message text for the errors found during DDS processing.

***SECLVL:** Second-level message text is included in the source listing.

GENLVL

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

Note: This parameter applies only to messages created while processing DDS source statements. Messages created elsewhere in the file creation process are not affected by this parameter.

20: If errors occur in the DDS source with a severity level greater than or equal to 20, the file is not created.

severity-level: Specify a severity level ranging from 0 through 30. The file is not created if the severity level specified for this parameter equals 0 or is less than the severity level that occurs in the data description specifications (DDS) source. This value must be greater than or equal to value specified on the FLAG parameter.

FLAG

Specifies the minimum severity level of messages to be listed in the DDS source listing.

0: All messages are listed.

severity-level: Specify the minimum severity level of messages to be listed. Valid values range from 0 through 30. The severity level specified must be less than or equal to the severity level specified on the GENLVL parameter.

DEV

Specifies the names of one or more display devices used with this display device file to pass data records between the users of the display devices and their jobs. The device name specified in the display device file supplied by IBM is *REQUESTER.

***REQUESTER:** The device from which the program is called is assigned to the file when the file is opened.

device-name: Specify the names of one or more display devices used with this device file to pass data records between the users of the devices and the system. Each device name must already be known on the system by a device description before this device file is created. *REQUESTER can be specified as one of the names. Up to 50 names can be specified in this command, but the total number cannot exceed the number specified on the MAXDEV parameter.

***NONE:** No device name is specified. The name of the display device must be specified later in a CHGDSPF or OVRDSPF command, or in the HLL program that opens the file.

MAXDEV

Specifies the maximum number of display devices that are connected to the display device file at the same time, while the file is open. However, if a CL program is written to get access to more than one work station through the same file (through a single running of the program), this parameter must specify a value greater than 1.

The names of the devices are specified in the DEV parameter of this command, in a later CHGDSPF or OVRDSPF command, or in the HLL program that opens the file.

1: Only one device name, or *REQUESTER, can be specified for this display device file.

number-of-devices: Specify the maximum number of devices that are connected to this display file at the same time. Valid values range from 1 through 256.

ENHDSP

Specifies whether the data being shown at a display station by this display file is using the enhanced capabilities available on the display station.

***YES:** The data for the display file is shown using any enhanced capabilities available on the display station. These capabilities can include mnemonics, selection cursor, and graphical window borders.

***NO:** The data for this display file is shown as it would be on a 5250 display station. No enhanced capabilities that are available on the display, such as mnemonics, selection cursor, or graphical window borders, are used. This value is normally used to preserve character-based interaction across all display stations.

RSTDSP

Specifies whether data being shown on a display by this display file is saved at the time the file is suspended

(made temporarily inactive) so that another display file can show different data on the same device. If the data for this file is saved, it is restored to the display of the device when the file is used again.

***NO:** The data being shown by this file is not saved when the file is suspended. When control is returned to the programs using this file, the data is not restored.

***YES:** The data being shown when the file is suspended is saved so it can be shown on the display when the file is used again.

DFRWRT

Specifies that the writing of data is deferred (delayed) until it is written out with other data when a read request is made. Control is returned to the program immediately after the data is received. This may result in improved performance.

***YES:** When the program issues a write request, control is returned after the buffer is processed. The data may not be shown immediately; the actual display of the data may take place later when a read or combined write/read operation is performed. The buffer is then available to be prepared for the next read or combined write/read operation.

***NO:** After a write operation, the user program does not regain control until the input/output operation is completed (with the data displayed and the input/output feedback information available).

CHRID

Specifies the character identifier (graphic character set and code page) that a work station display device supports. When a display file that was created with the CHRID DDS keyword is used with the device, the system converts data sent to and received from the device to ensure that the correct characters are shown and that the correct hexadecimal byte values are returned to the application program. More information about display file CHRID processing and the translation tables that are used to convert data sent to and received from the display are in the *Guide to Programming Displays*.

***DEVD:** The value specified on the CHRID parameter in the device description of the work station on which the application is running, is used. If no CHRID value is specified, the QCHRID system value for the system on which the application is running, is used. No conversion is necessary because the file has the same character identifier as the work station. For a list of valid values, see Table 30 in the CRTDEVDSP command description.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***JOBCCSID:** The character data is changed from the device CHRID to the CCSID (coded character set identifier) of the job on display file input, and from the CCSID of the job to the device CHRID on display file output.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values that match the attributes of the display device. Valid values range from 1 through 32767.

Element 2: Code Page

code-page: Specify the code page set values that match the attributes of the display device. Valid values range from 1 through 32767.

IGCDTA

Specifies, for program-described original files, whether the file processes double-byte character set (DBCS) data. For externally described printer files, this parameter specifies DBCS attributes of the file.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

IGCEXNCHR

Specifies whether the system processes double-byte character set (DBCS) extension characters.

***YES:** The system processes DBCS extension characters.

***NO:** The system does not process DBCS extension characters; it displays extension characters as the undefined character.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds that the program waits for the file resources to be allocated to the display device file when the file is opened, or the wait time for the device allocated when an acquire operation is performed to the file. Valid values range from 1 through 32767 seconds.

WAITRCD

Specifies the number of seconds the program waits for the completion of a read-from-invited-device operation to a multiple device file in a high-level language program. Refer to the appropriate high-level language reference manual to determine when a file is treated as a multiple

device file. The program performing the read operation waits for input from all invited devices currently accessing the file. If a record is not returned from an invited device in the specified amount of time, a notify message is sent to the program. This parameter has no effect on an input operation directed to a specific device.

Note: This parameter is also used to specify the time (seconds) that a CL program waits to complete a WAIT command. If a record is not returned from any of the devices that should return a record, an escape message is sent to the CL program. More information on the WAITRCD parameter is in the Receive File (RCVF), Send File (SNDF), Send/Receive File (SNDRCVF), and WAIT (Wait) command descriptions.

***NOMAX:** There is no limit on the time the system waits for the completion of the operation.

***IMMED:** The program does not wait for the read-from-invited-device operation for the completion of the file. A record must be available from an invited program device when the read-from-invited-program-device operation is performed. If a record is not already available when the read-from-invited-program-device operation is performed, a notify message is sent to the program.

number-of-seconds: Specify the number of seconds that the program waits for the completion of the read-from-invited-device operation. Valid values range from 1 through 32767.

DTAQ

Specifies the name of the data queue that receives an entry from the system when a data-available event is signaled from an invited display device. The data queue need not exist when the display file is created since the name specified on this parameter is not evaluated until the file is used. More information on the data queue function is in the *CL Programmer's Guide*.

***NONE:** A data queue does not receive an entry from the system.

The name of the data queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

data-queue-name: Specify the name of the data queue that is to receive an entry from the system when the data-available event is signaled.

SHARE

Specifies whether the open data path (ODP) for the display file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

SRTSEQ

Specifies the sort sequence used for this user profile. The sort sequence is used in conjunction with the LANGID parameter to determine which sort sequence table is used.

***JOB:** The SRTSEQ value specified on the job attribute is used.

***LANGIDSHR:** The sort sequence table can contain the same weight for multiple characters, and is the shared-weight table associated with the language specified in the LANGID parameter.

***LANGIDUNQ:** The sort sequence table must contain a unique weight for each character in the code page.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.

The name of the sort sequence table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify a table name.

LANGID

Specifies the language identifier used when *LANGIDSHR or *LANGIDUNQ is specified on the SRTSEQ parameter. The language identifier is used with the SRTSEQ parameter to determine which sort sequence table the file uses.

***JOB:** The language ID specified in the job description is used.

language-id: Specify a language identifier to be used by the file.

LVLCHK

Specifies whether the record format level identifiers in the program are checked against those in the device file when the file is opened. If so, the record format identifiers in the program must match those in the device file. Because the same record format name can exist in more than one file, each record format is given an internal system identifier when it is created.

***YES:** The level identifiers of the record formats are checked when the file is opened. If the level identifiers do not match, an error message is sent to the program that requested the open, and the file is not opened.

***NO:** The level identifiers are not checked when the file is opened.

AUT

Specifies the authority given to users who do not have specific authority to the display file, who are not on an authorization list, and whose user group has no specific authority to the display file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the display file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the display file). The public authority is determined when the display file is created. If the CRTAUT value for the library changes after the display file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the display file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the display file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the display file.

***USE:** The user can perform basic operations on the display file, such as running a program or reading a file. The user cannot change the display file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the display file.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether an existing file is replaced by the new display file More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** file'. The existing display is replaced by the one being created.

***NO:** The existing file, if any, is not replaced by the display.

TEXT

Specifies text that briefly describes the display device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member used to create the printer file. If the source file is a database file, the text is taken from the source member. If the source file is an inline file or a device file, the text is blank.

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Specifying Default Optional Parameters

```
CRTDSPF   FILE(DSPHIST)  SRCFILE(PRSNNL/JOBHIST)
```

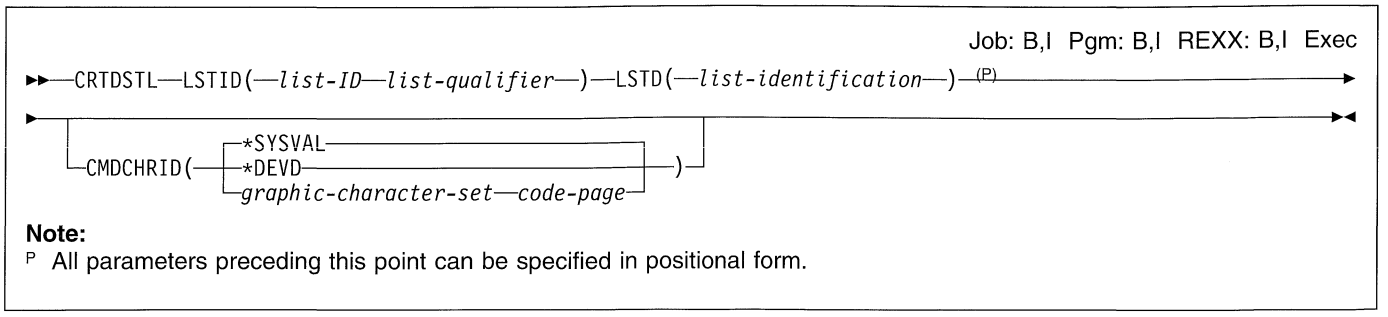
This command creates a display device file named DSPHIST which is stored in the current library using the source file named JOBHIST that is stored in the PRSNNL library. The defaults for all the other parameters are assumed. Only the device requesting the program that uses this device file (that is, *REQUESTER) is assigned to the device file. The level identifiers of the record formats are checked when the file is opened. The public has only object operational authority for the device file.

Example 2: Specifying DBCS Data Processing

```
CRTDSPF  FILE(IGCDSP) SRCFILE(IGCLIB/IGCSRC)
  IGCDTA(*YES)
```

This command creates the display file IGCDSP from the source file IGCSRC in the library IGCLIB. The file processes double-byte character set (DBCS) data.

CRTDSTL (Create Distribution List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Distribution List (CRTDSTL) command creates a new distribution list. A distribution list is a list of entries from the distribution directory. It can include entries for local, remote, indirect, and independent work station users. It can also include remote distribution lists, but not local distribution lists. More information about distribution lists is in the *Communications: Distribution Services Network Guide*.

The CRTDSTL command creates the distribution list with no entries. The Add Distribution List Entry (ADDDSTLE) command is used to add entries to the distribution list.

Restriction: The list identifier (ID) must be unique to all local user IDs, as well as to other list IDs in the directory. If a list ID is not unique, the list is not created and an error message is returned.

Required Parameters

LSTID
Specifies the two-part list identifier of the distribution list that is being created.

Element 1: List Identifier

list-identifier: Specify the list identifier (ID) of a distribution list.

Element 2: List Qualifier

list-qualifier: Specify the list qualifier of the distribution list.

Note: The distribution list identifier/list qualifier is in two parts, separated by at least one space. If any lowercase characters are specified, the system changes them to uppercase.

The naming rules for the two-part list ID are identical to the rules for the user ID and address. A complete description of these rules is in the *Distribution Services Network Guide*.

LSTD
Specifies the description of the distribution list. The description further identifies the distribution list.

Optional Parameter

CMDCHRID
Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the command parameters. Valid values range from 1 through 9999.

Element 2: Code Page

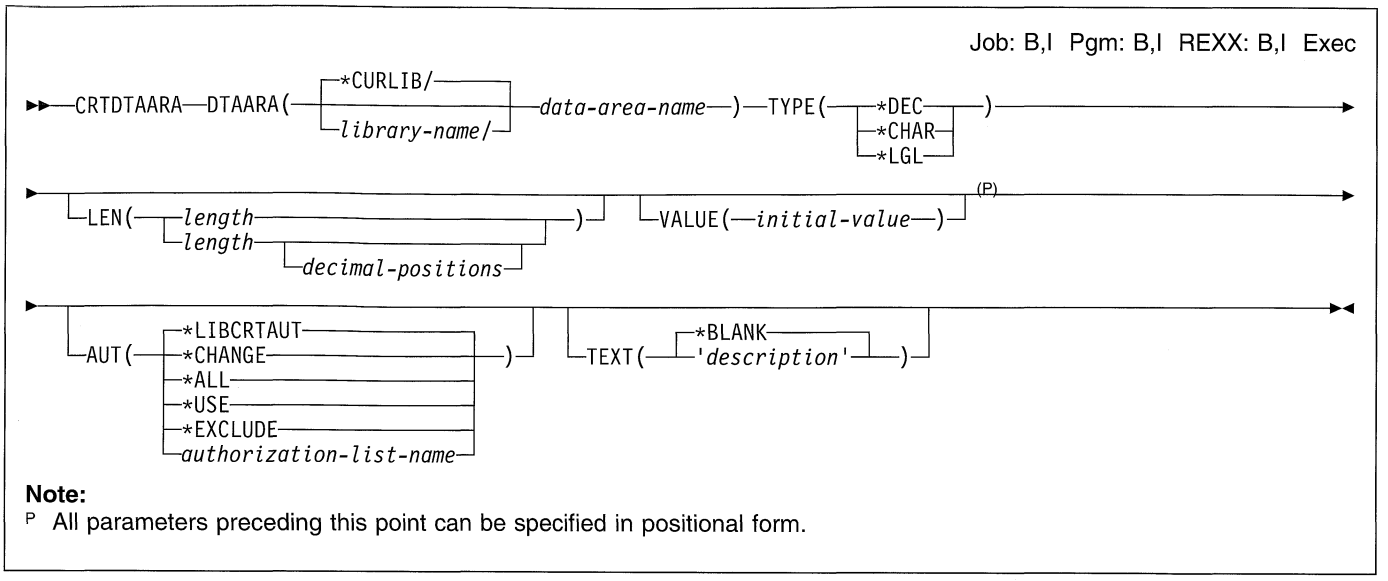
code-page: Specify the code page. Valid values range from 1 through 9999.

Example

```
CRTDSTL   LSTID(DEPT48K DLIST)
  LSTD('Department 48K Distribution List')
```

This command creates a distribution list that contains the members of Department 48K. If this list ID is unique, the distribution list is created.

CRTDTAARA (Create Data Area) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Data Area (CRTDTAARA) command creates a data area and stores it in a specified library. It also specifies the attributes of the data. The data area can optionally be initialized to a specific value.

Data areas (which are a type of operating system object) are used to communicate and store data used by several programs either within a job or between jobs. A program can use the value of a data area by using the Retrieve Data Area (RTVDTAARA) command.

If a data area is not used by more than one job at a time, it can be explicitly allocated to the appropriate job. If a data area is used by two or more jobs at the same time, it is protected from simultaneous changes occurring from different jobs. A data area is changed by using the Change Data Area (CHGDTAARA) command. The system does not allow two CHGDTAARA commands to change the same data area at the same time.

A data area is updated in auxiliary storage any time the data area is changed. This ensures that the changes are not lost in the event of a program or system failure.

Restriction: To use this command, the user must have object operational and add authority for the library in which the data area is placed.

Required Parameters

DTAARA

Specifies the qualified name of the data area being created.

The name of the data area can be qualified by one of the following library values:

***CURLIB:** The data area is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the data area is created.

data-area-name: Specify the name of the data area being created.

TYPE

Specifies the type of value contained in the data area being created. The data area can contain a character value, a decimal value, or a logical one (1) or zero (0).

***DEC:** The data area contains a decimal value.

***CHAR:** The data area contains a character string value.

***LGL:** The data area contains a logical value of either one (1) or zero (0) that can be used to represent two opposing conditions such as on/off, true/false, or yes/no.

Optional Parameters

LEN

Specifies the length of the data area being created. If it is a decimal data area, the number of decimal digits to the right of the decimal point can be optionally specified. The type of data area (specified by the TYPE parameter) determines the maximum length that its value can have and the default length that is assumed if LEN is not specified. The maximum lengths and the defaults for each of the three types are described in Table 36, at the end of this description.

Note: For character types, the default length is the same as the length of the initial value, if one is specified in the VALUE parameter.

Element 1: Value Length

length: Specify the maximum length of the value in this data area. Valid values range from 1 through 2000.

Element 2: Number of Decimal Positions

decimal-positions: Specify the number of decimal positions that the value in this data area can have. This option is valid only for decimal data areas. The length of the value in the data area includes the number of decimal positions in the value. The maximum length of the decimal value is 24 digits, of which no more than 9 can be to the right of the decimal point. The total length of the data area in CL programs cannot exceed 15 digits. If TYPE(*DEC) is specified and the number of decimal positions is not specified, a value of 0 is assumed. Valid values range from 0 through 9.

VALUE

Specifies the beginning value that is assigned to the data area when it is created. The beginning value must be of the type specified by the TYPE parameter. If no beginning value is specified, a character data area is initialized to blanks, a decimal data area is initialized to a value of 0, and a logical data area is initialized to '0'.

AUT

Specifies the authority given to users who do not have specific authority to the data area, who are not on an authorization list, and whose user group has no specific authority to the data area. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the data area is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the data area). The public authority is determined when the data area is created. If the CRTAUT value for the library changes after the data area is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the data area.

***USE:** The user can perform basic operations on the data area, such as running a program or reading a file. The user cannot change the data area. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the data area.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the data area. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Table 36. Data Area Lengths

Type	Maximum Length	Default Length
Decimal	24 digits, 9 decimal positions	15 digits, 5 decimal positions
Character	2000 characters	32 characters
Logical	1 character	1 character

Examples

Example 1: Creating a Data Area with a Value of Zero

```
CRTDTAARA   DTAARA(TOTSALES)  TYPE(*DEC)  LEN(15 2)
  VALUE(0)  TEXT('Total sales accumulator')
```

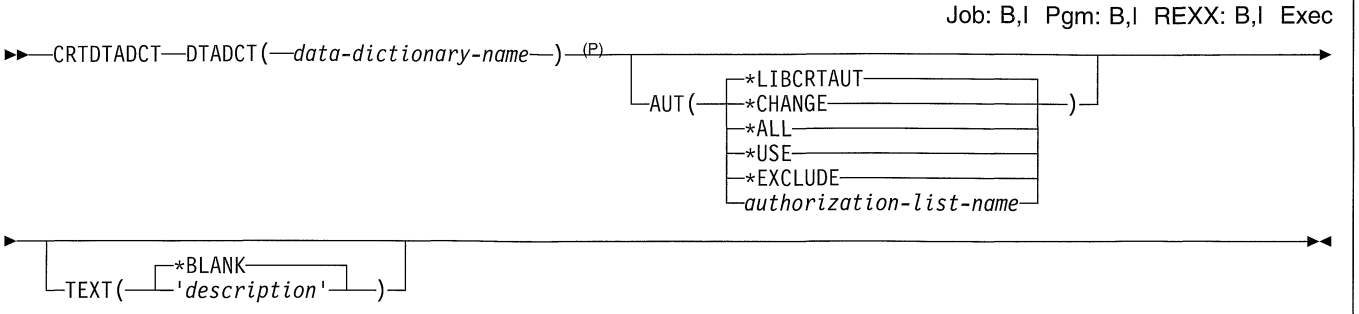
This command creates a data area named TOTSALES and stores it in the current library specified for the job. TOTSALES has the following data attributes: it is a 15-position numeric data area with two decimal positions and with an initial value of 0.

Example 2: Creating a Data Area Initialized to Blanks

```
CRTDTAARA   DTAARA(CUSTOMER)  TYPE(*CHAR)  LEN(148)
  TEXT('Customer name area')
```

This command creates the data area named CUSTOMER. It can contain as many as 148 characters in the character string. Because no initial value is specified, the data area is initialized to blanks.

CRTDTADCT (Create a Data Dictionary) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create a Data Dictionary (CRTDTADCT) command is used to create a data dictionary. A data dictionary must be created before the user can use the Interactive Data Definition Utility (IDDU) to describe and create database files. The user can create a data dictionary with IDDU or with the CRTDTADCT command. More information is in the *IDDU User's Guide*.

Required Parameter

DTADCT

Specifies the name of the library where the data dictionary is created and the name assigned to the data dictionary when it is created. The library must already exist. The dictionary name must be the same as that of the library where it is created.

Optional Parameters

AUT

Specifies the authority given to users who do not have specific authority to the data dictionary, who are not on an authorization list, and whose user group has no specific authority to the data dictionary.

***LIBCRTAUT:** The public authority for the data dictionary is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the data dictionary). The public authority is determined when the data dictionary is created. If the CRTAUT value for the library changes after the data dictionary is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the data dictionary.

***USE:** The user can perform basic operations on the dictionary, such as displaying the data dictionary or linking a dictionary definition to a file (see the Link Data Definition (LNKDTADFN) command). The user cannot change the dictionary or any definition.

***EXCLUDE:** The user cannot access the data dictionary.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the data dictionary. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a Data Dictionary and Granting USE Authority

```
CRTDTADCT   DTADCT(DEPT547)  AUT(*USE)
  TEXT('dept547 dictionary')
```

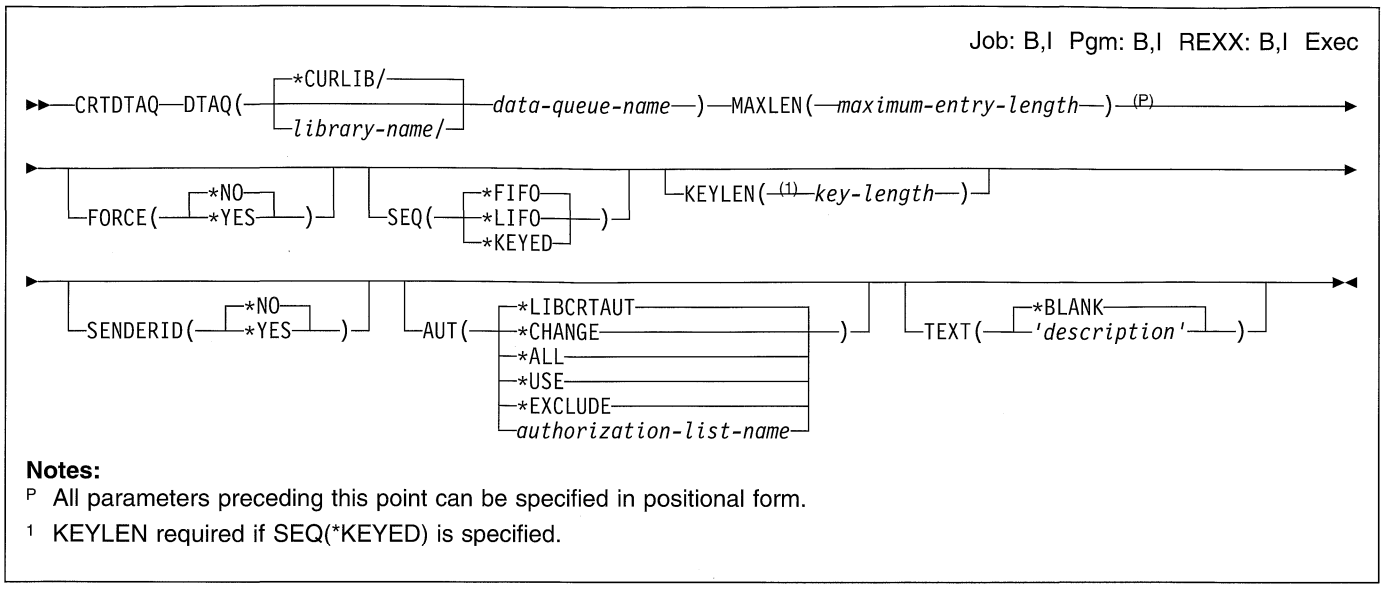
This command creates a data dictionary named DEPT547 in library DEPT547. The authority given for the dictionary to other users is *USE.

Example 2: Creating a Restricted Data Dictionary

```
CRTDTADCT   DTADCT(DEPT245)  AUT(MYLIST)
  TEXT('restricted dictionary')
```

This command creates a restricted data dictionary named DEPT245 in library DEPT245. The authority given for the dictionary to the users comes from the authorization list named MYLIST.

CRTDTAQ (Create Data Queue) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 KEYLEN required if SEQ(*KEYED) is specified.

Purpose

The Create Data Queue (CRTDTAQ) command creates a data queue and stores it in a specified library.

Data queues, which are a type of OS/400 system object, are used to communicate and store data used by several programs either within a job or between jobs. Multiple jobs can send or receive data from a single queue.

Restriction: Users of this command must have add authority for the library where the data queue is located, object operational and add authority to the sending data queue, and use authority to the receiving data queue.

Required Parameters

DTAQ

Specifies the qualified name of the data queue being created.

The name of the data queue can be qualified by one of the following library values:

***CURLIB:** The data queue is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the data queue is created.

data-queue-name: Specify the name of the data queue being created.

MAXLEN

Specifies the maximum length of the entry that is sent to the data queue. Valid values range from 1 through 64512.

Optional Parameters

FORCE

Specifies whether the data queue entries sent to or received by this data queue are forced to auxiliary storage.

***NO:** Send and receive operations are not immediately forced to auxiliary storage.

***YES:** Send and receive operations are immediately forced to auxiliary storage. This ensures that the changes are not lost if a system failure occurs. This requires additional system overhead.

SEQ

Specifies the sequence in which entries are received from the data queue.

***FIFO:** Data queue entries are received in a first-in first-out sequence.

***LIFO:** Data queues entries are received in a last-in first-out sequence.

***KEYED:** Data queues entries are received by key. A key is a prefix added to an entry by its sender.

KEYLEN

Specifies the number of characters in the key. Valid values range from 1 through 256. This parameter is valid only if SEQ(*KEYED) is specified.

SENDERID

Specifies whether a sender ID is attached to each message sent to the data queue. The ID contains the qualified job name and the sender's current user profile.

***NO:** Messages sent do not include the sender ID.

***YES:** Messages sent include the sender ID.

AUT

Specifies the authority given to users who do not have specific authority to the data queue, who are not on an authorization list, and whose user group has no specific authority to the data queue.

***LIBCRTAUT:** The public authority for the data queue is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the data queue). The public authority is determined when the data queue is created. If the CRTAUT value for the library changes after the data queue is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove user ids.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the data queue.

***USE:** The user can perform basic operations on the data queue, such as running a program or reading a file. The user cannot change the data queue. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the data queue.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the data queue. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

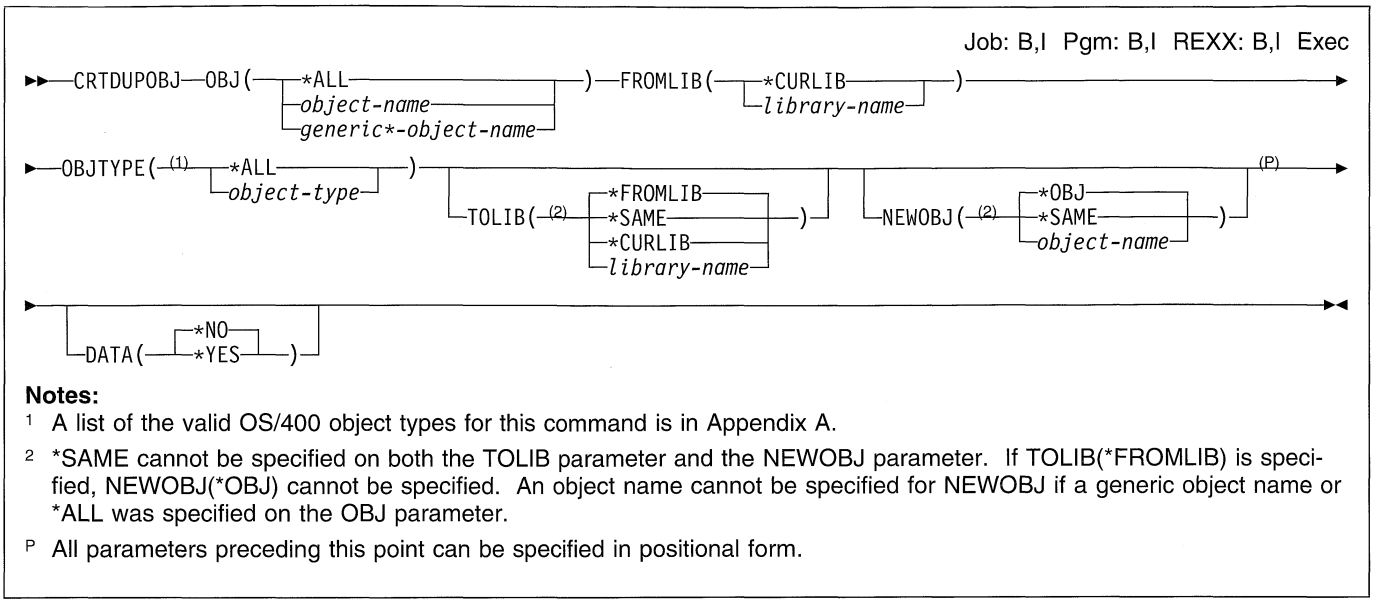
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTDTAQ   DTAQ(DEPTADTA)  MAXLEN(100)  AUT(*EXCLUDE)
  TEXT('Special data + files for DEPTA')
```

This command creates a data queue named DEPTADTA and puts it in the current library; the maximum length entry is 100. Because AUT(*EXCLUDE) is specified, the data queue can be used and controlled only by the user who created the queue and by users who have been given specific authority. Users in Department A can be given authority to use this data queue by using the Grant Object Authority (GTROBJAUT) command.

CRTDUPOBJ (Create Duplicate Object) Command



Notes:

1 A list of the valid OS/400 object types for this command is in Appendix A.

2 *SAME cannot be specified on both the TOLIB parameter and the NEWOBJ parameter. If TOLIB(*FROMLIB) is specified, NEWOBJ(*OBJ) cannot be specified. An object name cannot be specified for NEWOBJ if a generic object name or *ALL was specified on the OBJ parameter.

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Duplicate Object (CRTDUPOBJ) command copies a single object or a group of objects. It does not create an exact duplicate of files. The newly created object must be renamed if it is stored in the same library as the original object. If it is stored in a library other than the one that contains the original object, it can retain the name of the original object. Objects with the same name cannot exist in the same library. The user can specify a generic object name to copy a group of related objects, and the library in which the newly created objects are stored. The user can also specify whether data in physical files is copied. The duplicate object is authorized the same as the original object. The user who issues the command owns the duplicate object.

For files, the duplicate object shares the format of the original file. When a logical file is copied into another library, two cases determine the basing for the file. First, if both the logical file and its based-on physical file are originally in the same library, a duplicate of the physical file must be created in the new library before a duplicate of the logical file is created. After these two duplicates are created, the new logical file is based on the new physical file.

Second, if the logical file and its based-on physical file are originally in different libraries, it is not necessary to duplicate the physical file before duplicating the logical file. In this case, the duplicated logical file is based on the same physical file as was the original logical file. Unlike the first case, even if the physical file is copied into the new library before the logical file is copied, the duplicated logical file is based on the original physical file, not on the duplicated physical file.

Restrictions:

1. The user must have object operational and read authorities (*USE) and object management authority to the existing object.
2. The user must have add authority to the library for the new object.
3. The user must have authorization list management authority if the object is an authorization list.
4. The user must have object operational and object management authorities based on the physical file when duplicating a logical file.
5. The user must have object operational and read authorities to the Create Save File (CRTSAVF) command to create a duplicate save file. The contents of the save file are not duplicated.
6. When an object is being duplicated, it is created on the same auxiliary storage pool (ASP) as its duplicate object.
7. If DATA(*YES) is specified when the CRTDUPOBJ command is used to create a copy of a file, the new duplicate file object is seized (similar to an *EXCL lock with no timeout) for the duration of the data copy making access impossible. An attempt to use a function that refers to the new duplicate file object while the data copy is in progress results in a lock up for that work station until the data copy is completed. The following are examples of functions that should not be used on the new duplicate file object until the data copy is completed:
   - WRKACTJOB (Select 11-Locks; Select 8-WRKOBJLCK)
   - DSPDBR
   - DSPFD
   - DSPFFD
   - DSPJOB (Option 12-Locks; F10-Job record locks; Option 14-Open files)

- DSPLIB (The library containing the new duplicate file)
- DSPOBJD
- WRKOBJLCK
- DSPRCDLCK
- Any other function which refers to the new duplicate file

8. When creating a duplicate object of type *GSS, *FNTRSC, *FORMDF, *OVL, *CSI, *PAGDFN, or *PAGSEG, the name of the new object cannot exceed 8 characters in length.
9. The user space (*USRSPC) and user index (*USRIDX) user domain objects can be copied only into libraries that are permitted in the system value QALWUSRDMN (allow user objects in library). However, if the user object was created in the system domain, it is not restricted.

Required Parameters

OBJ

Specifies the name of one or more objects that are being duplicated.

***ALL:** All the objects with the value specified for OBJTYPE in the specified library are duplicated.

object-name: Specify the name of the object that is being duplicated.

generic-object-name:* Specify the generic name of the object to be duplicated. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be duplicated only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

FROMLIB

Specifies the name of the library that contains the object to be duplicated.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

OBJTYPE

Specifies the types of OS/400 system objects to be duplicated. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ALL:** All valid objects in the library are to be duplicated.

object-type: Specify the value for the type of object that is to be duplicated.

Optional Parameters

TOLIB

Specifies the name of the library in which the duplicate object is created.

Note: If the library is in a user auxiliary storage pool (ASP), the object to be duplicated must be a valid object type that can reside in a user ASP. If this object type is not valid, an error message is displayed.

***FROMLIB:** The new object will be in the same library as the original object.

***SAME:** The new object is in the same library as the original object. A name different from the original object must be assigned to the new object with the NEWOBJ parameter.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used. The new object will be in the current library for this job.

library-name: Specify the name of the library to be searched.

NEWOBJ

Specifies the name of the single duplicated object. A name must be specified if TOLIB(*FROMLIB) or TOLIB(*SAME) is specified. The names of members in a database file that is being copied remain the same in the new file.

***OBJ:** The new object has the same name as the original, but it must reside in a different library.

***SAME:** The new object has the same name as the original, but it must reside in a different library.

object-name: Specify the name of the object that is duplicated.

DATA

Specifies whether the data records of physical files that are duplicated are copied to the new object. Members of physical files are copied whether or not the data contained in them is copied.

***NO:** The data records in the members of physical files that are duplicated are not copied to the new object.

***YES:** The data records in the members of physical files that are duplicated are copied to the new object.

Notes:

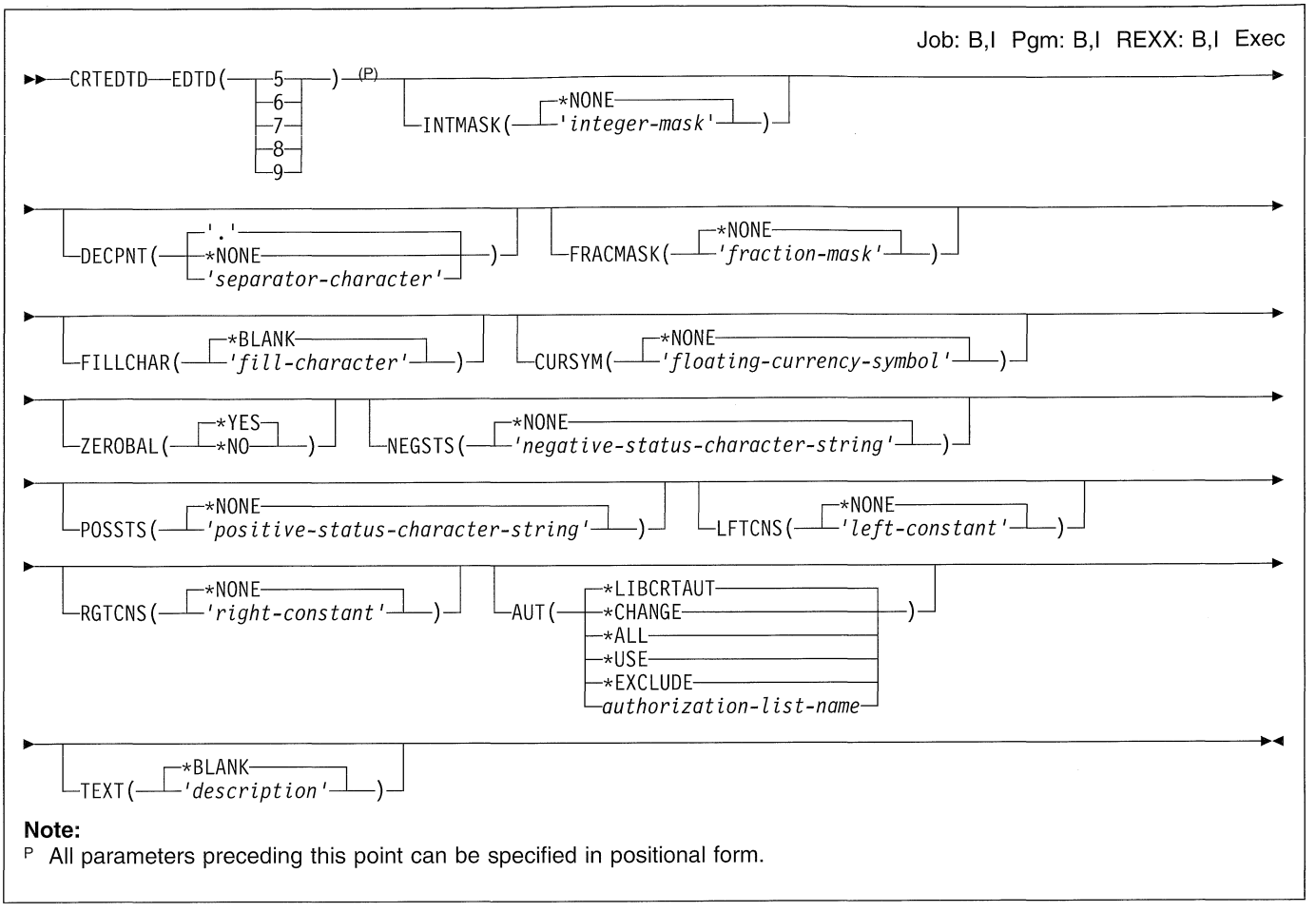
1. A physical file cannot be duplicated while it is in use for update by another job.
2. The relative record numbers in the new file are the same as those in the original file.

Example

```
CRTDUPOBJ   OBJ(FILEA)  FROMLIB(LIB1)
  OBJTYPE(*FILE)  TOLIB(LIB2)  DATA(*YES)
```

The file named FILEA in library LIB1 is duplicated and stored in library LIB2. Authorities granted for FILEA are granted to the new FILEA in LIB2, and data associated with FILEA is copied to the newly created FILEA in LIB2.

CRTEDTD (Create Edit Description) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Edit Description (CRTEDTD) command defines an edit mask for the specified edit description and stores it in the QSYS library. As many as five edit descriptions can be defined by the user. They must be identified as edit descriptions 5, 6, 7, 8, or 9. The actual object names for the edit descriptions are *QEDITn*, where *n* is the single-digit identifying code. The OS/400 system supplies a version of each of these edit descriptions in the QSYS library. More information on the IBM-supplied versions is in the *Guide to Programming Displays*. To create a new version, the IBM-supplied version must first be deleted by the Delete Edit Description (DLTEDTD) command.

Edit descriptions can be used in data description specifications and high-level language programs to edit numeric fields.

Required Parameter

EDTD

Specifies a single-digit code (5, 6, 7, 8, or 9) that identifies the user-defined edit description being created. This digit is used in Data Description Specifications (DDS) to refer to the edit mask that is created by this CRTEDTD command. The actual name of the created object (which is stored in the QSYS library) is *QEDITn*, where *n* is the single digit edit code specified in this parameter.

Optional Parameters

INTMASK

Specifies a character string (mask) that describes the editing of the integer portion of a decimal field. Characters other than a blank, a zero, or an ampersand (&) are handled as constants in the editing process. Blank, zero, and ampersand have the following meanings:

- Blank: Each blank is replaced with a fill character or with a digit from the number being edited once zero suppression ends (by a significant digit or by the farthest left zero in the mask).
- Zero (0): The farthest left zero is a digit replacement character and also ends zero suppression. All other zeros in the integer mask are handled as constants.
- Ampersand (&): Blank substitution.

Note: You cannot specify both INTMASK(*NONE) and FRACMASK(*NONE) on the CRTEDTD command. Instead, specify blanks for INTMASK and FRACMASK, and specify GENLVL(30) on the Create Printer File (CRTPRTF) or Create Display File (CRTDSPF) command, which allows the file to create, but ignores the edit code keyword.

***NONE:** No editing mask is used on the integer portion of decimal fields.

'integer-mask': Specify the character string that is used as the editing mask for the integer portion of a decimal field. Up to 31 characters, enclosed in apostrophes, can be used in the integer mask.

DECPNT

Specifies, for decimal fields, a single character used as a decimal point to separate the integer (INTMASK) and fraction (FRACMASK) portions of the edited result. If the field has no decimal places, this character is not used and is not considered in the width of the edited results.

Note: If the separator character specified for DECPNT is also used in the INTMASK parameter, it has no special meaning in the integer mask; it is handled only as a constant or as a digit replacement character in the integer mask.

'.': The period (or decimal point) is the separator character. It must be enclosed in apostrophes.

***NONE:** No separator character is specified; a decimal point is not needed in the edited result.

'separator-character': Specify the separator character, such as the comma (,), that is used as a decimal point. Any alphanumeric or special character can be used, but a special character must be enclosed in apostrophes.

FRACMASK

Specifies a character string (mask) that describes the editing of the fraction portion of a decimal field (to the right of the decimal point). The characters have the same meaning as described for the INTMASK parameter except that all zeros are handled as constants and blanks are not replaced with a fill character.

***NONE:** No editing mask is used on the fraction portion of decimal fields.

'fraction-mask': Specify the character string that is used as the editing mask for the fraction portion of a decimal field. Up to 31 characters, enclosed in apostrophes, can be used in the fraction mask.

FILLCHAR

Specifies the character that is used in each position of a result that is zero suppressed. The specified character replaces all leading zeros that are to the left of the first significant digit in the integer mask (or a forced zero).

***BLANK:** The fill character is a blank.

'fill-character': Specify the character that is used as the fill character. Any alphanumeric or special character can be used, but a special character must be enclosed in apostrophes.

CURSYM

Specifies the character string that is used as the floating currency symbol. If CURSYM is specified, the character string appears immediately to the left of the first significant digit (or constant). If the first significant digit is a zero, occurring in the position that ended zero suppression, the CURSYM character string ends in the position occupied by that zero.

***NONE:** No floating currency symbol is specified; none is needed in the edited result.

'floating-currency-symbol': Specify the character string that is used as the floating currency symbol for monetary amount fields. Up to 15 alphanumeric and special characters, enclosed in apostrophes, can be specified.

ZEROBAL

Specifies the editing action for zero values.

***YES:** The normal editing rules are followed. For information on editing rules, refer to "Editing Rules," following the description of the CRTEDTD command parameters.

***NO:** If the field being edited has a value of zero, the entire field (integer, decimal point, or fraction) is replaced by the fill character, including constants in the edit mask.

NEGSTS

Specifies the character string that immediately follows the body of the edited result if the field status is negative. If the field is positive, blanks are substituted for the length of the string, unless a value for POSSTS is also specified.

***NONE:** No character string is specified; blanks are used to the right of the field in the edited result.

'negative-status-character-string': Specify the character string that immediately follows the edited field when the field is negative in value. Up to 31 characters, enclosed in apostrophes, can be specified as the negative status character string.

POSSTS

Specifies the character string that immediately follows the body of the edited result if the field is positive or zero. If the field is negative, blanks are substituted for the length of the string, unless a value for NEGSTS is also specified.

***NONE:** No character string is specified; blanks are used to the right of the field in the edited result.

'positive-status-character-string': Specify the character string that immediately follows the edited field when the field is positive in value. Up to 31 characters, enclosed in apostrophes, can be specified as the positive status character string.

LFTCNS

Specifies the character string constant that always appears as the farthest left portion of the edited result.

***NONE:** No constant appears on the left side of edited fields.

'left-constant': Specify the character string that always appears on the left side of an edited field. Up to 31 characters, enclosed in apostrophes, can be specified.

RGTCNS

Specifies the character string constant that always appears as the farthest right portion of the edited result.

***NONE:** No constant appears on the right side of edited fields.

'right-constant': Specify the character string that always appears on the right side of an edited field. Up to 31 characters, enclosed in apostrophes, can be specified.

AUT

Specifies the authority given to users who do not have specific authority to the edit description, who are not on an authorization list, and whose user group has no specific authority to the edit description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the edit description.

***USE:** The user can perform basic operations on the edit description, such as running a program or reading a file. The user cannot change the edit description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the edit description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the edit description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** No text is specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

The examples assume the following:

FIELDA Six digits (four integer and two decimal positions) with a value of 001234

FIELDB Same as FIELDA but with a negative value (-001234)

FIELDC Same as FIELDA but with a zero value (000000)

DATE Six digits (0 decimal positions) with a value of 091878

The character ␣ is used to represent blank spaces.

Example 1

```
CRTEDTD   EDTD(5)
  INTMASK('␣␣␣,␣␣␣,␣␣0')
  FRACMASK('␣␣␣␣')
  NEGSTS('DB␣␣␣␣')  POSSTS('CREDIT')
  LFTCNS('$')  RGTCNS('␣**')
```

FIELDA Logical mask is '$␣,␣␣0.␣␣ DB␣␣␣␣ ␣**' for a negative value or '$␣,␣␣0.␣␣ CREDIT ␣**' for a positive value

Edited result is $␣␣␣12.34CREDIT␣**

FIELDB Same logical mask

Edited result is $␣␣␣12.34DB␣␣␣␣**

FIELDC Same logical mask

Edited result is $␣␣␣␣.00CREDIT␣** or, if ZEROBAL(*NO) had been specified, $␣␣␣␣␣␣␣CREDIT␣**

Example 2

```
CRTEDTD   EDTD(6)
  INTMASK('␣␣␣.␣0␣')  DECPNT(',')
  FRACMASK('␣␣␣')  CURSYM('DM')  NEGSTS('-␣**')
```

FIELDA Logical mask is '␣␣␣.␣0␣,␣␣-␣**' with floating DM

Edited result is ␣␣␣DM12,34␣␣␣␣

FIELDB Same logical mask

Edited result is ␣␣␣DM12,34-␣**

FIELDC Same logical mask

Edited result is ␣␣␣␣DM0,00␣␣␣␣ or, if ZEROBAL(*NO) had been specified, ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣

Example 3

```
CRTEDTD   EDTD(7)
  INTMASK('0␣MONTH␣␣DAY&␣␣YEAR')
  LFTCNS('DATE␣IS␣')
```

DATE Logical mask is equal to the INTMASK parameter value

Edited result is DATEbISbb9MONTH18DAYb78YEAR'

Example 4

```
CRTEDTD   EDTD(9)  +
  INTMASK('bb,bb0')  DECPNT('.')
  FRACMASK('bb')  FILLCHAR('*')  NEGSTS('bERRORb**')
```

FIELDA Logical mask is 'b,bb0.bbbbbb bbbbb' or 'b,bb0.bbbERRORb**' (Both use the * as the fill character)

Edited result is ***12.34bbbbbbbbb

FIELDB Same logical mask

Edited result is ***12.34bERRORb**

FIELDC Same logical mask

Edited result is *****.00bbbbbbbbb or, if ZEROBAL(*NO) had been specified, ********bbbbbbbbb

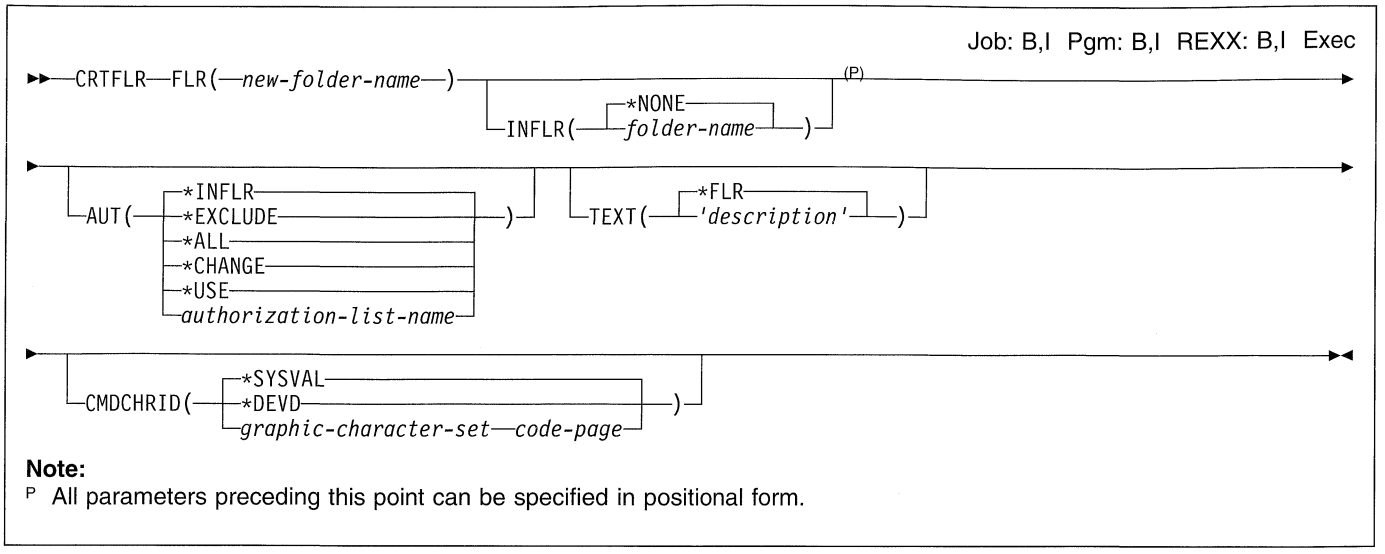
Additional Considerations

Editing Rules

- The field to be edited is aligned with respect to the two portions of the edit mask (integer and fraction).
- As a result of the number of integer digits in the field to be edited, the integer mask INTMASK is truncated on the left side immediately before the farthest left digit replace character that can be used. If a leading zero occurs in the truncated portion of the integer mask, no zero suppression occurs.
- The separator character used as the decimal point (DECPNT) immediately follows the integer mask.
- The fraction mask (FRACMASK) immediately follows the separator character (or the integer mask if DECPNT(*NONE) is specified). As a result of the number of decimal positions in the field to be edited, the fraction mask is truncated on the right side immediately following the farthest right digit replace character that could be used.
- The width of the edited result can be calculated as follows:

  (length of LFTCNS) + (length of CURSYM) +
  (length of truncated INTMASK) +
  (1 (or 0 if DECPNT equals *NONE)) +
  (length of truncated FRACMASK) +
  (length of NEGSTS or POSSTS) +
  (length of RGTCNS) = width of edited result
- If either the integer mask or the fraction mask does not contain enough digit replacement characters to include the digits that can be contained in the respective portions of the field, editing of the field is diagnosed and ignored, and an error message is sent to the user (or program).
- Changing the edit description has no effect on previously created file formats. These file formats must be recreated if the new edit mask is desired.

CRTFLR (Create Folder) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Folder (CRTFLR) command creates folders. Folders are used to organize documents and other folders.

Restriction: If the folder is being created and put into an existing folder, *CHANGE authority for the existing folder is required.

Required Parameter

FLR

Specifies the name (ranging from 1 to 12 characters, including an optional extension) of the folder being created. If no extension is included, a folder name can have up to 8 characters. If an extension is included, the extension must start with a period and can have 1, 2, or 3 additional characters. An extension in the folder name allows the user to identify the folder by using specific information that can help the user do a selective listing of folders on the system.

The following characters are not allowed in folder names: /, *, and ?.

Optional Parameters

INFLR

Specifies the folder that contains the folder being created.

***NONE:** The folder is not created into another folder. This folder is considered a first-level folder.

folder-name: Specify the name (ranging from 1 through 63 characters) of the folder in which the newly created folder is contained.

Note: Because folders can reside within other folders, and because any given folder name is unique only within its containing folder, it may be necessary to link several folder names together to identify a folder. This is commonly called the path to an object within a folder. When more than one folder name is used to indicate the path to a folder:

- Each folder name is separated by a forward slash (/).
- The complete folder name cannot exceed 63 characters in length.

The syntax of a folder name using multiple names is as follows:

```
folder-1.ext/folder-2.ext/folder-3.ext
```

The first folder name (`folder-1`) is known as a first-level folder because it contains the other folders that follow. The second folder name is the second-level folder, and so on.

Examples of valid folder paths:

```
FOLDER1        (one folder without
                an extension)
F1.EXT/F2.EXT (a folder within a folder)
```

Examples of invalid folder paths:

```
F1///F2             ('null' folders
                     not allowed)
/F1/F2              (leading slash
                     not allowed)
F1/NAMETOOBIG/F3    (invalid folder
                     name in path)
```

AUT

Specifies the authority given to users who do not have specific authority to the folder, who are not on an authorization list, and whose user group has no specific authority to the folder. More information on this param-

eter is in Appendix A, "Expanded Parameter Descriptions."

***INFLR:** The created folder is given the same authority as the folder in which it is included. If INFLR(*NONE) is specified, the authority is *EXCLUDE.

***EXCLUDE:** The user cannot access the folder.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the folder.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the folder, such as running a program or reading a file. The user cannot change the folder. *USE authority provides object operational authority and read authority.

authorization-list-name: Specify the name of the authorization list.

TEXT

Specifies text that briefly describes the folder object. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***FLR:** The text is the same as the folder name specified in the FLR parameter.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

CMDCHRID

Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set of the value supplied in the TEXT parameter. Valid values range from 1 through 999.

Element 2: Code Page

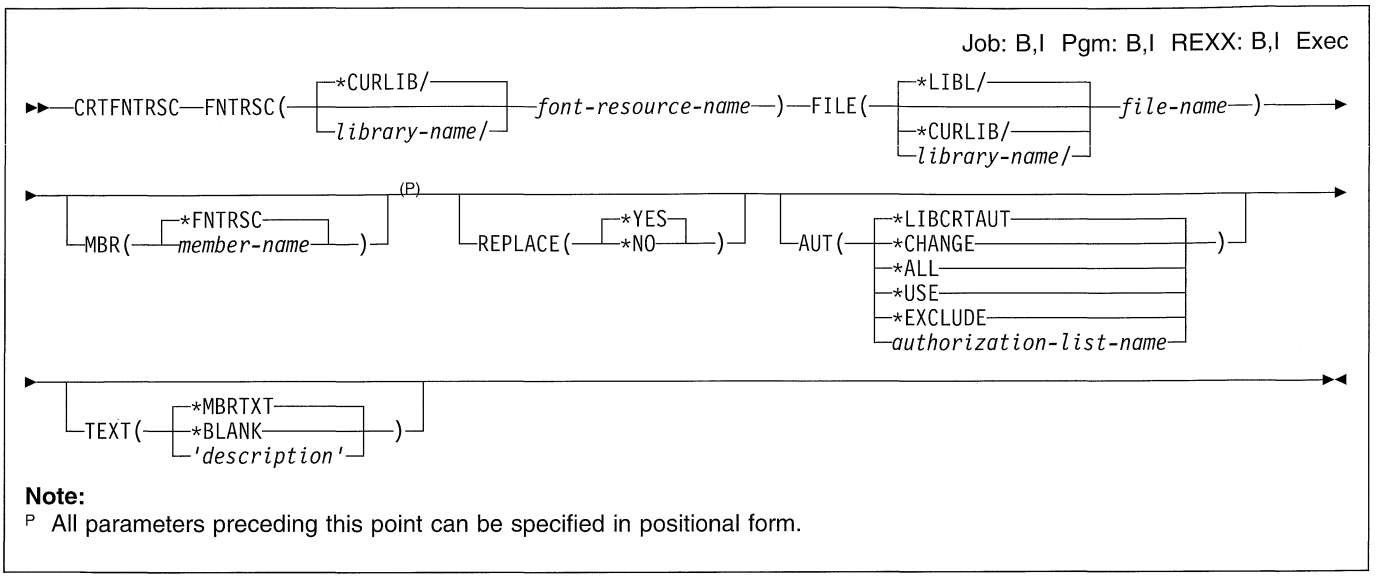
code-page: Specify the code page value used to create the command parameters. Valid values range from 1 through 999.

Example

```
CRTFLR   FLR(QTR1)  INFLR('PAYROLL/1987')
  AUT(*CHANGE)  TEXT('first quarter payroll')
```

This command creates the folder QTR1 in folder PAYROLL/1987. The public is granted *CHANGE authority to the folder, which allows adding a document to the folder, changing the folder description, or showing the contents of the folder. Folder 1987 is in the PAYROLL folder, which is a first-level folder.

CRTFNTRSC (Create Font Resources) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Create Font Resources (CRTFNTRSC) command creates a font resource object from a physical file. The physical file contains the font resource information. The font resource information, can, for example, come from an S/370 host system and be in the Systems Application Architecture* (SAA) format. Depending on the type of information processed by the CRTFNTRSC command, the results are either a font character set, a code page, or a coded font.

Required Parameters

FNTRSC

Specifies the qualified name of the font resource being created.

| The name of the font resource can be qualified by one of the following library values:

***CURLIB:** The font resource is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the font resource is created.

font-resource-name: Specify the name of the font resource being created.

FILE

Specifies the qualified name of the database file in which the font resource data resides.

| The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file to create the database file.

Optional Parameters

MBR

Specifies the name of the file member used to create the font resource.

***FNTRSC:** The name of the file member that contains the input data is the same as the font resource being created.

member-name: Specify the name of the file member that contains the font resource.

REPLACE

Specifies whether the page definition is replaced.

***YES:** The page definition is replaced.

***NO:** No replacement occurs.

AUT

Specifies the authority given to users who do not have | specific authority to the font resources, who are not on | an authorization list, and whose user group has no spe- | cific authority to the font resources.

| ***LIBCRTAUT:** The public authority for the font | resources is taken from the value on the CRTAUT | parameter of the target library (the library that is to | contain the font resources). The public authority is | determined when the font resources is created. If the | CRTAUT value for the library changes after the font

resources is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the font resources.

***USE:** The user can perform basic operations on the font resources, such as running a program or reading a file. The user cannot change the font resources. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the font resources.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MBRTXT:** The text is taken from the file member used to create the font resource. Text can be added or changed for a database source member by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** No text is specified.

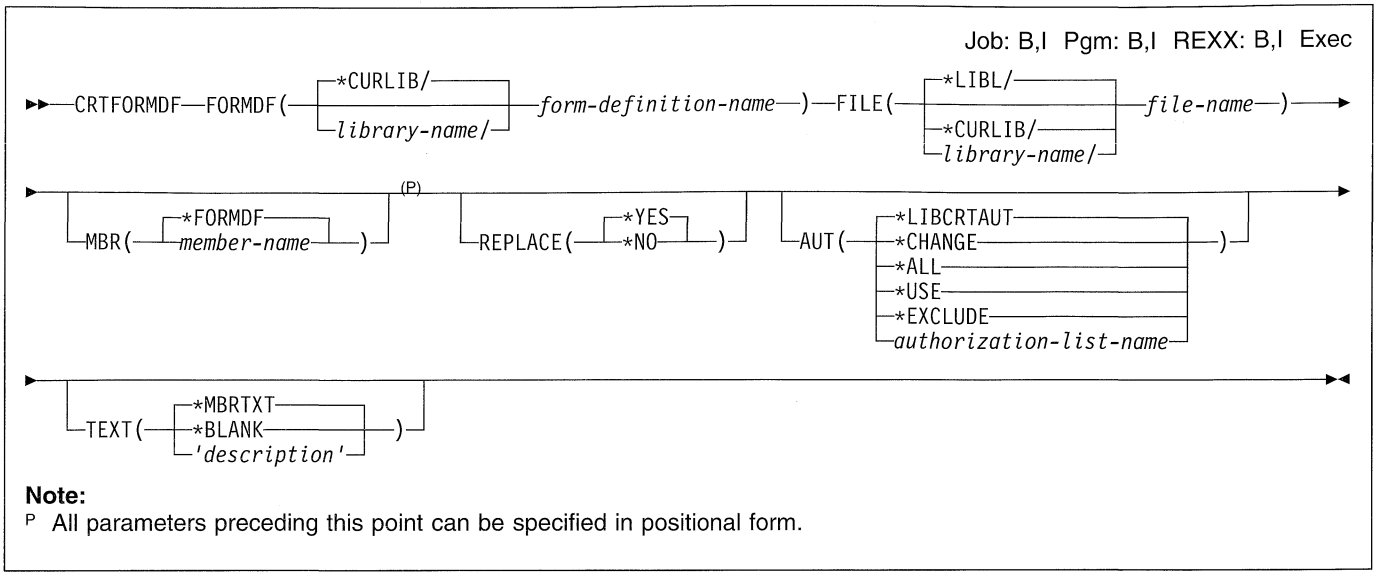
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTFNTRSC  FNTRSC(MYLIB/GOTHIC12)  FILE(*LIBL/FONTRSCS)
  MBR(*FNTRSC)  AUT(*USE)  TEXT('Gothic Font 12 Pitch')
```

This command creates font resource GOTHIC12 in MYLIB. Source file FONTRSCS, in the user's library list, with member GOTHIC12 is used as input. Specifying *USE for the AUT parameter allows other users to access GOTHIC12, but not to change it. The text describes the font resource.

CRTFORMDF (Create Form Definition) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Form Definition (CRTFORMDF) command creates a form definition from a physical file. The physical file contains the form definition information. The form definition information, can, for example, come from a S/370* host system and be in the Systems Application Architecture (SAA) format.

Restriction: If networking spooled files to a System/370* system, the first two characters of the form definition name must be 'F1'.

Required Parameters

FORMDF

Specifies the qualified name of the form definition being created.

The name of the form definition can be qualified by one of the following library values:

***CURLIB:** The form definition is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the form definition is created.

form-definition-name: Specify the name of the form definition being created.

FILE

Specifies the qualified name of the file being used to create the database file.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file being used to create the database file.

Optional Parameters

MBR

Specifies the name of the file member used to create the database file.

***FORMDF:** The name of the file member that contains the input data is the same as the database file member being created.

member-name: Specify the name of the file member that contains the database file.

REPLACE

Specifies whether the page definition is replaced.

***YES:** The page definition is replaced.

***NO:** No replacement occurs.

AUT

Specifies the authority given to users who do not have specific authority to the form definition, who are not on an authorization list, and whose user group has no specific authority to the form definition.

***LIBCRTAUT:** The public authority for the form definition is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the form

definition). The public authority is determined when the form definition is created. If the CRTAUT value for the library changes after the form definition is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the form definition.

***USE:** The user can perform basic operations on the form definition, such as running a program or reading a file. The user cannot change the form definition. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the form definition.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MBRTXT:** The text is taken from the source file member being used to create the RPG program. You can add or change text for a database source member by using either the Add Physical File Member (ADDPFM) command or Change Physical File Member (CHGPFM) command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** No text is specified.

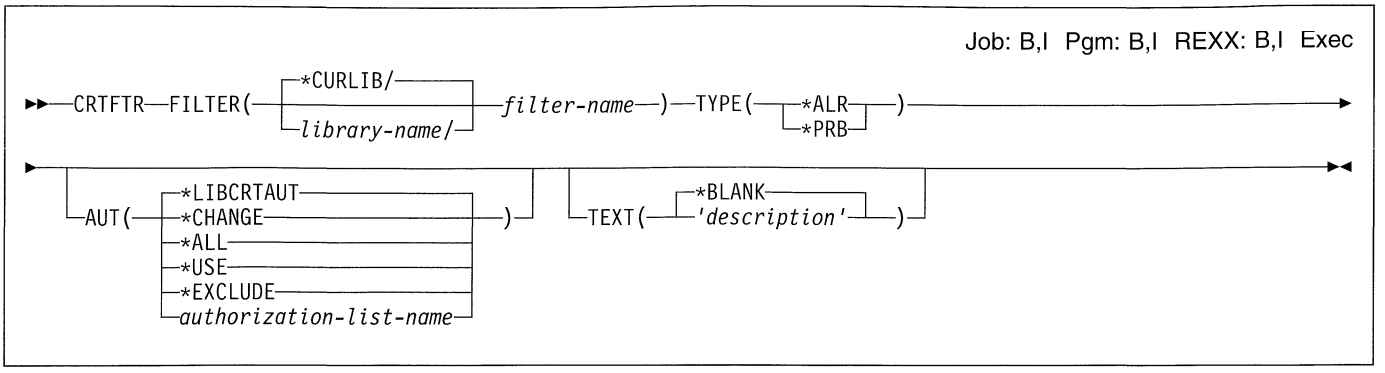
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTFORMDF  FORMDF(*CURLIB/FORMDF1)  FILE(*CURLIB/FORMDF)
  MBR(F1A01238)  AUT(*EXCLUDE)
  TEXT('Default form definition for AFP printers')
```

This command creates form definition FORMDF1 in the current library, or in the QGPL library if there is no current library. Input is taken from source file FORMDF with member F1A01238, in the current library. Specifying *EXCLUDE for authority restricts the usage of the object to the owner. The text describes what the form definition represents.

CRTFTR (Create Filter) Command



Purpose

The Create Filter (CRTFTR) command creates a filter object of the specified type. Filters contain selection entries and action entries. A filter allows the user to categorize data into groups and specify special actions to be applied to each group. The typical user is a system programmer or operator responsible for system management.

Required Parameters

FILTER

Specifies the qualified name of the filter that is created.

The possible library values are:

***CURLIB:** The current library is used to locate the filter. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the filter is created.

filter-name: Specify the name of the filter that is to be created.

TYPE

Specifies the type of filter being created. The type of filter determines which applications can use the filter and the type of entries that can be placed in the filter.

***ALR:** The filter is an alert filter. The OS/400 Alert Manager can use the filter on alerts that it receives or generates.

***PRB:** The filter is a problem filter. The OS/400 Problem Manager uses the filter on problem entries that are created, changed, or deleted.

Optional Parameters

AUT

Specifies the authority given to users who do not have specific authority to the filter, who are not on an authorization list, and whose user group has no specific authority to the filter. More information on this parameter is in "Appendix A. Expanded Parameter Descriptions" in the *CL Reference*.

***LIBCRTAUT:** The public authority for the filter is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the filter). The public authority is determined when the filter is created. If the CRTAUT value for the library changes after the filter is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the filter.

***USE:** The user can perform basic operations on the filter, such as running a program or reading a file. The user cannot change the filter. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the filter.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the filter. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

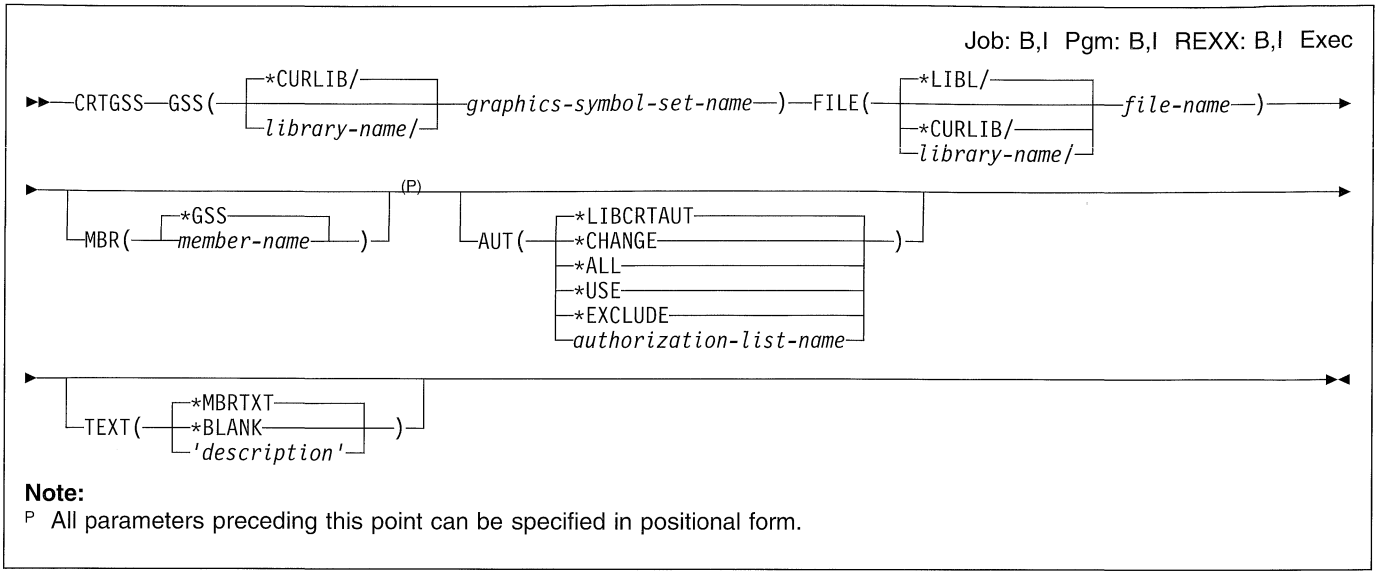
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTFTR  FILTER(MYLIB/MYFILTER)  TYPE(*ALR)
  AUT(*CHANGE)  TEXT('My filter')
```

This command creates an alert filter called MYFILTER in the library MYLIB. The type is *ALR and the public has *CHANGE authority to the filter, described as 'My filter'.

CRTGSS (Create Graphics Symbol Set) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Graphics Symbol Set (CRTGSS) command creates a graphics symbol set object from a physical file that contains symbol set data. Depending upon the contents of the file, the CRTGSS command creates either a vector symbol set (mode 3 graphics characters) or an image symbol set (mode 2 graphics characters).

The symbol set object can be used in a graphical data display manager (GDDM*) or presentation graphics routines (PGR) graphics application program or in a Business Graphics Utility chart as an alternative to an IBM-supplied graphics symbol set. More information on the Business Graphics Utility is in the *BGU User's Guide and Reference*.

Restriction: A physical file used with this command must contain records with no less than 80 bytes and no more than 400 bytes, and a source file must contain no less than 92 bytes and no more than 412 bytes. The contents of the file must be in symbol set format.

Required Parameters

GSS

Specifies the qualified name of the graphics symbol set being created. Graphics symbol set names cannot exceed eight characters.

The name of the graphics symbol set can be qualified by one of the following library values:

***CURLIB:** The graphics symbol set is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the graphics symbol set is created.

graphics-symbol-set-name: Specify the name of the graphics symbol set being created.

FILE

Specifies the qualified name of the file being used to create the symbol set.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file being used to create the graphics symbol set.

Optional Parameters

MBR

Specifies the name of the file member being used to create the symbol set.

***GSS:** The name of the file member that contains the input data is the same as the symbol set being created.

member-name: Specify the name of the file member that contains the symbol set input data.

AUT

Specifies the authority given to users who do not have specific authority to the graphics symbol set, who are not on an authorization list, and whose user group has no specific authority to the graphics symbol set. More infor-

mation on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the graphics symbol set is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the graphics symbol set). The public authority is determined when the graphics symbol set is created. If the CRTAUT value for the library changes after the graphics symbol set is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the graphics symbol set.

***USE:** The user can perform basic operations on the graphics symbol set, such as running a program or reading a file. The user cannot change the graphics symbol set. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the graphics symbol set.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the symbol set. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MBRTXT:** The text is taken from the file member being used to create the symbol set.

***BLANK:** No text is specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating Set With Same Name as Source File

```
CRTGSS   GSS(GSSLIB/ADMVARP)  FILE(GSSLIB/ADMVARP)
```

This command creates a symbol set of the same name and in the same library as the source file.

Example 2: Creating Set From Different Library

```
CRTGSS   GSS(*CURLIB/VECTOR1)  FILE(GSSLIB/QDATASRC)
  MBR(SCHEM)  AUT(*ALL)  TEXT('Schematic vector symbols')
```

This command creates a symbol set named VECTOR1 in the QGPL library from member SCHEM in file QDATASRC in library GSSLIB. The public has complete authority over the symbol set. Despite the fact that the symbol set data is stored in source physical file QDATASRC, it cannot be edited or shown by the source entry utility (SEU) because some of the contents of the symbol set data cannot be shown.

Additional Considerations

To use the System/370 symbol sets in AS/400 OS/400 system Graphics, you must transport the source data from the System/370 to the AS/400 system, and then process the CRTGSS command to convert the symbol set source data into an AS/400 system *GSS object type.

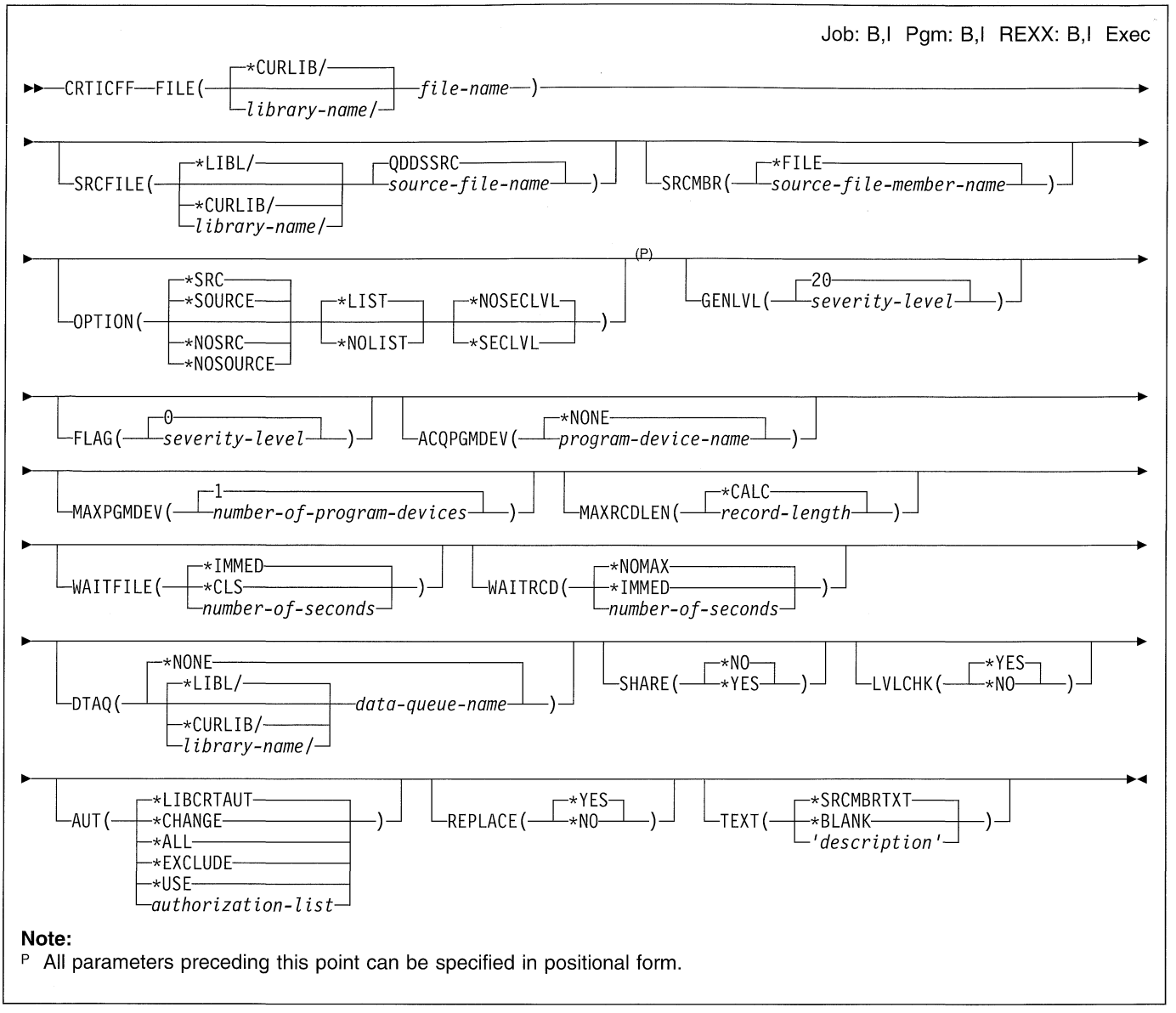
Because System/370 symbol set source data usually has a length of 400, some communications links between the AS/400 system and System/370 require that the source data be deblocked into 80-byte records before being sent to the AS/400 system. Other links allow the source data to remain in 400-byte records. However, the CRTGSS command accepts source data with record lengths in the range of 80 through 400 bytes. Symbol set source data with a record length of 400 can be sent back to the System/370 for modifications, if necessary, while 80-byte data requires blocking again before the symbol editors will accept it.

Symbol set source data can be placed into a physical file of either type *DATA or *SRC. Type *SRC data cannot be edited or browsed, and the record length of the file must be 12 bytes longer than the source data placed into it, to allow for the 12-byte sequence number fields associated with *SRC files. Therefore, the allowable record length range is 92 through 412 bytes for type *SRC files.

Once the symbol set source data is contained as a member in an AS/400 system database file, it can be used to create a symbol set object (object type *GSS) in the CRTGSS command.

More information on the use of the CRTGSS command and the formats for vector and image symbol set database source files is in the *GDDM Programming Guide*.

CRTICFF (Create Intersystem Communications Function File) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Create Intersystem Communications Function File (CRTICFF) command creates an intersystem communications function (ICF) device file used to communicate with program-to-program devices.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the ICF file can be qualified by one of the following library values:

***CURLIB:** The ICF file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the ICF file is created.

file-name: Specify the name of the file.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file that contains the data description specifications (DDS) used to

create the ICF file. The source file contains the specifications that describe the record formats and their fields.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The source file, QDDSSRC, contains the DDS used to create the ICF file.

source-file-name: Specify the name of the source file that contains the DDS used to create the ICF file.

SRCMBR

Specifies the name of the source file member that contains the DDS for this ICF file; the member is in the source file specified in the SRCFILE parameter (or its default, QDDSSRC).

***FILE:** The member name of the source file is the same as the ICF file name specified in the FILE parameter.

source-file-member-name: Specify the name of the member in the source file that contains the DDS used to create the ICF file.

OPTION

Specifies the type of printout produced when the file is created. Up to three of the following values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value will be used.

Note: The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

Program Creation Options

***SRC** or ***SOURCE:** A printout is created of the source statements used to create the file and any errors that occur.

***NOSRC or *NOSOURCE:** No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the record format containing the error.

Source Listing Options

***LIST:** An expanded source printout is created, showing a detailed list of the file specifications that result from the source statements and references to other file descriptions.

***NOLIST:** An expanded source printout is not created.

***NOSECLVL:** The messages section of the DDS printout does not contain the second-level message text for errors found during DDS processing.

***SECLVL:** Second-level message text is included in the source listing.

GENLVL

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

20: The file is not created if a DDS error message with a severity level of 20 or greater occurs.

severity-level: Specify a severity level ranging from 0 through 30. The file is not created if the severity level specified for this parameter equals 0 or is less than the severity level that occurs in the data description specifications (DDS) source. This value must be greater than or equal to value specified on the FLAG parameter.

FLAG

Specifies the minimum severity level of messages to be listed in the DDS source listing.

0: All messages are listed.

severity-level: Specify the minimum severity level of messages to be listed. Valid values range from 0 through 30. The severity level specified must be less than or equal to the severity level specified on the GENLVL parameter.

ACQPGMDEV

Specifies which program device is acquired when the file is opened.

***NONE:** The file is opened without any program device acquired. All program devices used with this file are explicitly acquired before input/output operations can be used with them.

program-device-name: Specify the name of the first program device acquired when the file is opened. The program device is defined to the file when the file is opened. The name must be specified on the PGMDEV parameter on an ADDICFDEVE, CHGICFDEVE, or OVRICFDEVE command before the file is opened.

MAXPGMDEV

Specifies the maximum number of program devices used with this ICF file. The program devices are defined using the Add Intersystem Communications Function (ICF) Device Entry (ADDICFDEVE), Change ICF Device Entry (CHGICFDEVE), or Override ICF Device Entry (OVRICFDEVE) command. This number also determines the maximum number of program devices that can be added to the ICF file by the ADDICFDEVE command.

1: Only one program device is used with this ICF file.

number-of-program-devices: Specify the maximum number of program devices used with this ICF file. Valid values range from 1 through 256.

MAXRCDLEN

Specifies the maximum record length used when the file is opened.

***CALC:** The length calculated for the largest record in the file is used when the file is opened.

record-length: Specify the maximum record length (in bytes) used when the file is opened. Valid values range from 1 through 32767.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the maximum number of seconds (ranging from 1 through 32767) the program waits for the resources being allocated.

WAITRCD

Specifies the number of seconds the program waits for the completion of a read-from-invited-device operation to a multiple device file in a high-level language program. Refer to the appropriate high-level language reference manual to determine when a file is treated as a multiple device file. The program performing the read operation waits for input from all invited devices currently accessing the file. If a record is not returned from an invited device in the specified amount of time, a notify message is sent to the program. This parameter has no effect on an input operation directed to a specific device.

Note: This parameter is also used to specify the time (seconds) that a CL program waits to complete a WAIT command. If a record is not returned from any of the devices that should return a record, an escape message is sent to the CL program. More information on the WAITRCD parameter is in the Receive File (RCVF), Send File (SNDF), Send/Receive File (SNDRCVF), and WAIT (Wait) command descriptions.

***NOMAX:** There is no limit on the time the system waits for the completion of the operation.

***IMMED:** The program does not wait for the read-from-invited-device operation for the completion of the file. A record must be available from an invited program device when the read-from-invited-program-device operation is performed. If a record is not already available when the read-from-invited-program-device operation is performed, a notify message is sent to the program.

number-of-seconds: Specify the maximum number of seconds, ranging from 1 through 32767, that the program waits for the completion of the read-from-invited-program-devices operation.

DTAQ

Specifies the name of the data queue that receives an entry from the system when a data-available event is signaled from an invited display device. The data queue need not exist when the display file is created since the name specified on this parameter is not evaluated until the file is used. More information on the data queue function is in the *CL Programmer's Guide*.

***NONE:** A data queue does not receive an entry from the system.

The name of the data queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

data-queue-name: Specify the name of the data queue that is to receive an entry from the system when the data-available event is signaled.

SHARE

Specifies whether the open data path (ODP) for the ICF file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

LVLCHK

Specifies whether the record format level identifiers in the program are checked against those in the device file when the file is opened. If so, the record format identifiers in the program must match those in the device file.

Because the same record format name can exist in more than one file, each record format is given an internal system identifier when it is created.

***YES:** The level identifiers of the record formats are checked when the file is opened. If the level identifiers are not all the same or have not been specified in the program, an open error message is sent to the program that tried to open the file.

***NO:** The level identifiers are not checked when the file is opened.

AUT

Specifies the authority given to users who do not have specific authority to the ICF file, who are not on an authorization list, and whose user group has no specific authority to the ICF file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the ICF file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the ICF file). The public authority is determined when the ICF file is created. If the CRTAUT value for the library changes after the ICF file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the ICF file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the ICF file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the ICF file.

***EXCLUDE:** The user cannot access the ICF file.

***USE:** The user can perform basic operations on the ICF file, such as running a program or reading a file. The user cannot change the ICF file. *USE authority provides object operational authority and read authority.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether an existing file is replaced by the new ICF file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** An existing file is replaced by the ICF file being created.

***NO:** No replacement occurs.

TEXT

Specifies text that briefly describes the ICF file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source member used to create the ICF file.

***BLANK:** Text is not specified.

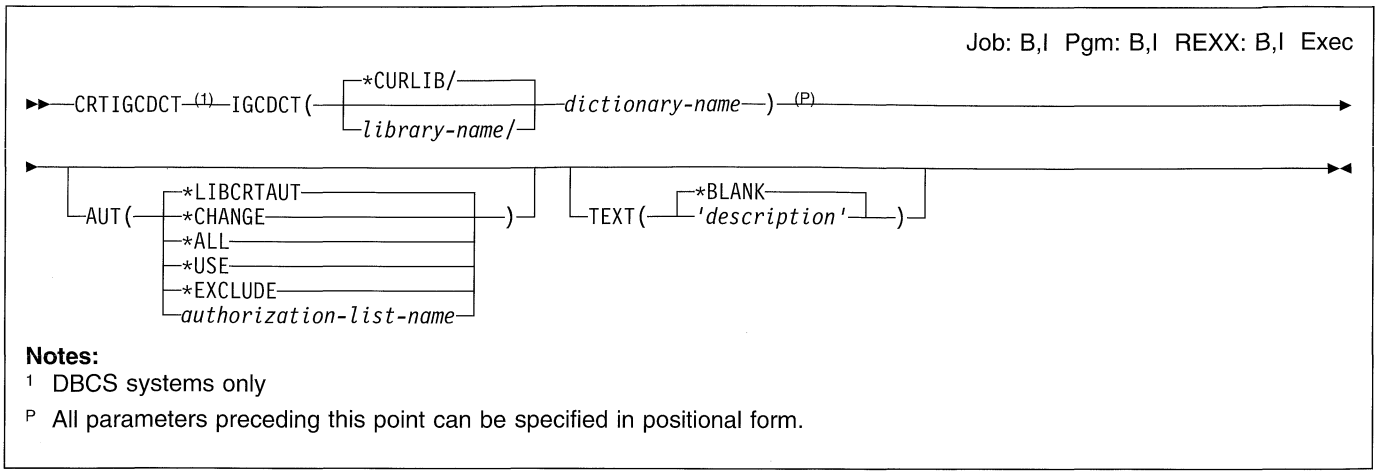
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTICFF   FILE(QGPL/ICFTEST)  SRCFILE(QGPL/QDDSSRC)
  MAXPGMDEV(5)  ACQPGMDEV(DENVER)
```

This command creates the file ICFTEST in the QGPL library. The DDS source used to create the file is in member ICFTEST from file QDDSSRC in the QGPL library. Up to five program devices can be used with the file. The program device DENVER is acquired when the file is opened.

CRTIGCDCT (Create DBCS Conversion Dictionary) Command



Notes:
1 DBCS systems only
P All parameters preceding this point can be specified in positional form.

Purpose

The Create DBCS Conversion Dictionary (CRTIGCDCT) command creates the specified double-byte character set (DBCS) conversion dictionary and stores it in the specified library. The dictionary contains alphameric entries and their related words. The system refers to DBCS conversion dictionaries when doing DBCS conversion.

Required Parameter

IGCDCT

Specifies the qualified name of the DBCS conversion dictionary being created and the library in which it is stored.

The name of the DBCS conversion dictionary can be qualified by one of the following library values:

***CURLIB:** The DBCS conversion dictionary is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the DBCS conversion dictionary is created.

dictionary-name: Specify the name of the dictionary being created.

Note: A DBCS conversion dictionary created by the user can have any name, but is used for doing DBCS conversion only if it is named QUSRIGCDCT.

Optional Parameters

AUT

Specifies the authority given to users who do not have specific authority to the DBCS conversion dictionary, who are not on an authorization list, and whose user group has no specific authority to the DBCS conversion dictionary. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the DBCS conversion dictionary is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the DBCS conversion dictionary). The public authority is determined when the DBCS conversion dictionary is created. If the CRTAUT value for the library changes after the DBCS conversion dictionary is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the DBCS conversion dictionary.

***USE:** The user can perform basic operations on the DBCS conversion dictionary, such as running a program or reading a file. The user cannot change the DBCS conversion dictionary. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the DBCS conversion dictionary.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the dictionary. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** No text is specified.

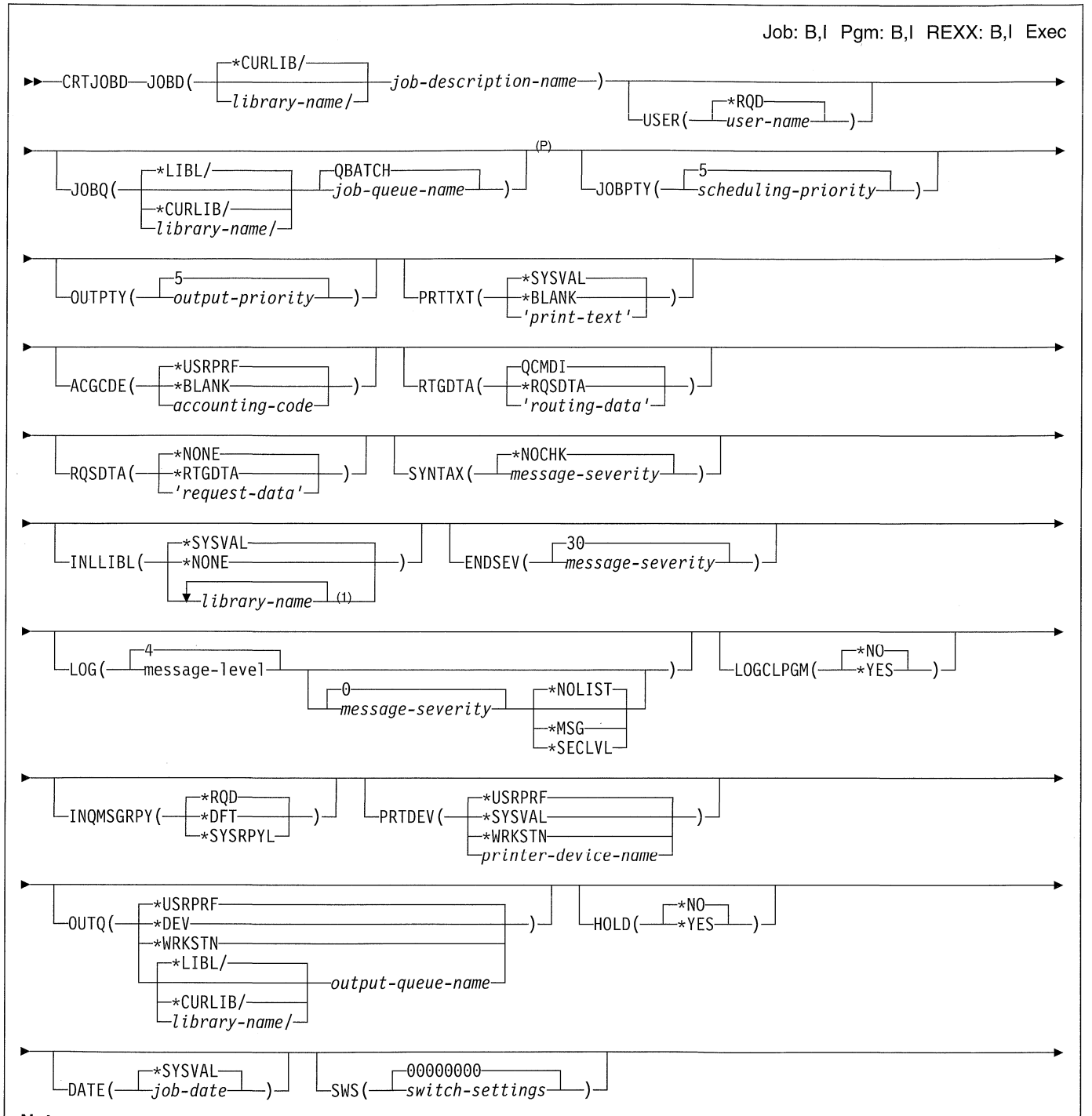
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTIGCDCT   IGCDCT(DBCSLIB/QUSRIGCDCT)
```

This command creates a DBCS conversion dictionary named QUSRIGCDCT, which is stored in the library DBCSLIB.

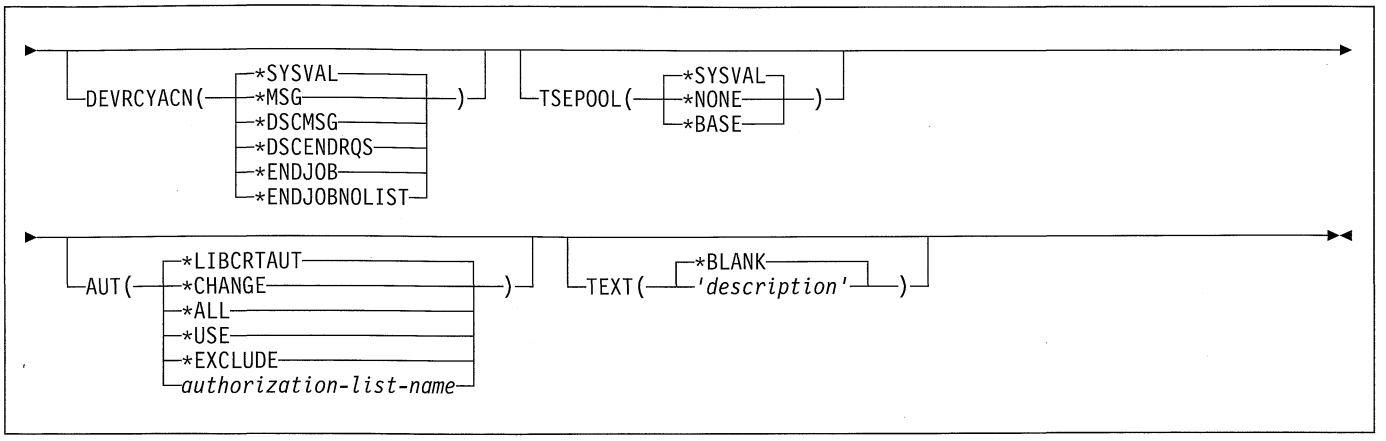
CRTJOBD (Create Job Description) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 25 repetitions



Purpose

The Create Job Description (CRTJOBD) command creates a job description object that contains a specific set of job-related attributes that can be used by one or more jobs. The attributes determine how each job is run on the system. The same job description can be used by multiple jobs. The values in the job description are usually used as the default values of the corresponding parameters in the Batch Job (BCHJOB) and Submit Job (SBMJOB) commands when their parameters are not specified.

The values in the job description can be overridden by the values specified in the BCHJOB and SBMJOB commands.

Restrictions:

1. To use this command, the user must have object operational authority for the user profile specified on the USER parameter (if any); that is, the user must have the authority to start a job on behalf of that user.
2. The user must also have *ADD authority for the library where the job description is placed. To create a job description with an accounting code (ACGCDE) parameter other than *USRPRF, the user must have *USE authority to the Change Accounting Code (CHGACGCDE) command.

Required Parameter

JOBD

Specifies the qualified name of the job description being created.

| The name of the job description can be qualified by one of the following library values:

***CURLIB:** The job description is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the job description is created.

job-description-name: Specify the name of the job description being created.

Optional Parameters

USER

Specifies the name of the user profile associated with this job description. The names QSECOFR, QSPL, QDOC, QDBSHR, QRJE, QTSTRQS, and QDFTOWN are not valid entries for this parameter.

***RQD:** A user name is required to use the job description. For work station entries, you must enter your password when you sign on at the work station; the associated user name becomes the name used for the job. *RQD is not valid for job descriptions specified for job entries that start automatically, or for those used by the BCHJOB command. It is valid on the SBMJOB command only if USER(*CURRENT) is specified.

user-name: Specify the user name that identifies the user profile associated with batch jobs that use this job description. For interactive jobs, this is the default user name used when signing on without entering a password.

JOBQ

Specifies the qualified name of the job queue on which this job is placed.

| The name of the job queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QBATCH: The QBATCH job queue is the queue on which the jobs are placed.

job-queue-name: Specify the qualified name of the job queue that is associated with this job description.

If the job queue does not exist when the job description is created, a library qualifier must be specified because

the name of the qualified job queue is retained in the job description.

JOBPTY
Specifies the scheduling priority for each job that uses this job description. The highest priority is 1 and the lowest priority is 9. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

5: The scheduling priority that any job using this job description is 5.

scheduling-priority: Specify the scheduling priority for any job using this job description. Valid values range from 1 through 9.

OUTPTY
Specifies the output priority for spooled files that are produced by jobs using this job description. The highest priority is 1 and the lowest priority is 9. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

5: The output priority for spooled files produced using this job description is 5.

output-priority: Specify the priority of spooled files produced using this job description. Valid values range from 1 through 9.

PRTTXT
Specifies up to 30 characters of text to be printed at the bottom of each page of output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SYSVAL:** The system value, QPRTTXT, is used.

***BLANK:** Text is not specified.

'print-text': Specify the character string that is printed at the bottom of each page. Up to 30 characters can be entered, enclosed in apostrophes.

ACGCDE
Specifies the accounting code used when logging system resource use for a job having this job description. To specify an accounting code other than *USRPRF, the user must have *USE authority for the Change Accounting Code (CHGACGCDE) command.

Note: If the job is submitted by the SBMJOB command, the submitter's accounting code is used.

***USRPRF:** The accounting code for jobs using this job description is obtained from the job's user profile.

***BLANK:** An accounting code of 15 blanks is assigned to jobs that use this job description.

accounting-code: Specify the character string that is used as the accounting code for jobs that use this job description and have accounting statistics logged in the system accounting journal QSYS/QACGJRN. If less than 15 characters are entered, the string is padded on the right with blanks. Note that the character string '*USRPRF' is considered a special value, and cannot be used as an accounting code on this command.

RTGDTA
Specifies the routing data that is used with this job description to start jobs. The routing data is used to determine the routing entry in the subsystem description that identifies the program where the job runs.

QCMDI: The default routing data QCMDI is used by the IBM-supplied interactive subsystem to route the job to the IBM-supplied control language processor QCMD in the QSYS library.

***RQSDTA:** Up to the first 80 characters of the request data specified in the RQSDTA parameter are used as the routing data for the job.

'routing-data': Specify the character string, enclosed in apostrophes, that is used as the routing data for jobs that use this job description. For example, the value QCMDI is the routing data used by the IBM-supplied interactive subsystem (QINTER) to route interactive jobs to the IBM-supplied control language processor QCMD. Up to 80 characters can be entered.

RQSDTA
Specifies the request data that is placed as the last entry in the job's message queue for jobs using this job description. For example, when a CL command is supplied as request data on a Submit Job (SBMJOB) command, it becomes a message that can be read by the control language processor, QCMD (if the submitted job is routed to QCMD).

***NONE:** No request data is placed in the job's message queue.

***RTGDTA:** The routing data specified in the RTGDTA parameter is placed as the last entry in the job's message queue.

'request-data': Specify the character string that is placed as the last entry in the job's message queue as a single request. Up to 256 characters can be entered, enclosed in apostrophes. When a CL command is entered, it must be enclosed in single apostrophes, and where apostrophes would normally be used *inside* the character string, double apostrophes must be used instead.

SYNTAX
Specifies whether requests placed on the job's message queue are checked for syntax as CL commands. When syntax checking is specified, the commands are checked for syntax as they are submitted rather than when the job is run, thus providing an earlier diagnosis of syntax errors. If checking is specified, the message severity that causes a syntax error to end processing of a job is also specified. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NOCHK:** The request data is not checked for syntax as CL commands. If the message severity is specified, it is used only when the job description is used by a job command that also has RQSDTA(*) specified and the requests are CL commands.

message-severity: Specify a value, ranging from 00 through 99, that specifies the lowest message severity that can cause running of a job to end. The request data is checked for CL command syntax; if a syntax error occurs that has severity equal to or greater than the error message severity specified here, the running of the job is suppressed.

INLLIBL

Specifies the first user portion of the library list that is used for jobs using this job description.

Note: Duplicating libraries in the library list is not allowed. If specifying a list of libraries for INLLIBL (or if specifying *JOBD and the job description used specifies a list of libraries), ensure that there are no duplicates contained in the list being used. To do this, compare the libraries in the list to the libraries contained in system value QSYSLIBL. Duplicates must be removed from the INLLIBL parameter.

More information on the use of library lists is in the *CL Programmer's Guide*.

***SYSVAL:** The system default user library list is used for jobs that use this job description. The default user library list contains the library names that were specified in the system value QUSRLIBL at the time that a job using this job description is started.

***NONE:** The user portion of the first library list is empty; only the system portion is used.

library-name: Specify the names of one or more libraries that are in the user portion of the library list for jobs that use this job description. No more than 25 names can be specified; the libraries are searched in the same order as they are listed here.

ENDSEV

Specifies the message severity level of escape messages that can cause a batch job to end. The batch job is ended when a request in the batch input stream sends an escape message whose severity code is greater than or equal to that specified here to the request processing program. This parameter value is compared with the severity of any unmonitored escape message that occurs as a result of running a noncompiled CL command in a batch job. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

30: An escape message resulting from a request in the batch input stream whose severity is equal to or greater than 30 causes the job to end.

message-severity: Specify a value, ranging from 00 through 50, for the message severity of an escape message that results from a request in the batch input stream and that causes the job using this job description to end. Because escape messages typically go up to a severity level of 50, a value of 50 or lower must be specified for a job to end as a result of an escape message. An unhandled escape message whose severity is equal to or greater than the value specified causes the job to end. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

LOG

Specifies the message logging values used to determine the amount and type of information sent to the job log by this job. This parameter has three elements: the message (or logging) level, the message severity, and the level of message text. If no values are specified on this parameter, the values specified in the job description associated with this job are used.

Note: If no values are specified for the LOG parameter, the values 4, 0, and *NOLIST are assumed by the system.

Element 1: Message Level

The first of the three values in the LOG parameter specifies one of five logging levels ranging from 0 through 4.

Notes:

1. A **high-level** message is sent to the program message queue of the program that received the request or commands being logged from a CL program.
2. Several system displays provide a retrieve function (the F9 key) which gets a logged command into a command line. Commands are not necessarily logged unless the logging level is set to 3 or 4. If the logging level is set to 0, 1, or 2, each command that is completed cannot be retrieved into a command line because the logging level causes the command to be removed from the job log. When each command is completed, it cannot be retrieved into a command line if the message logging level for the job causes the command to be removed from the job log.

The logging levels are described below:

0 No data is logged.

1 The following information logged: All messages sent to the job's external message queue with a severity greater than or equal to the message severity specified (this includes the indications of job start, job end, and job completion status).

2 The following information is logged:

- Logging level 1 information
- All requests entered on a command line and commands being logged from a CL program for which messages are issued with a severity code greater than or equal to the severity specified
- All messages that are associated with a request entered on a command line or a command being logged from a CL program that results in a high-level message with a severity greater than or equal to the severity specified

3 The following information is logged:

- Logging level 1 information
- All requests entered on a command line and commands being logged from a CL program
- All messages that are associated with a request entered on a command line or a command being logged from a CL program that results in a high-level message with a severity greater than or equal to the severity specified

4 The following information is logged:

- All requests entered on a command line and commands being logged from a CL program
- All messages with a severity code greater than or equal to the severity specified, including trace messages

4: A message logging level of 4 is used for job messages created when this job description is used.

message-level: Specify the message logging level used for job messages produced when this job description is used. Valid values range from 0 through 4.

Element 2: Message Severity

The second of the three values in the LOG parameter specifies the minimum severity level that causes error messages to be logged in the job logs of jobs that use this job description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

0: The message severity level is set at 0. Errors that have a severity code of 0 or greater cause an error message to be logged in the job log in which the error occurred.

message-severity: Specify the lowest severity level that causes an error message to be logged in the job's log. Valid values range from 00 through 99.

Element 3: Message Text Level

The third of the three values in the LOG parameter specifies the level of message text that is written in the job log or displayed to the user when an error message is created by the first two values.

***NOLIST:** No job log is produced for jobs that use this job description and end normally. If the job ends abnormally (if the end of job code is 20 or higher), a job log is produced. The messages appearing in the job's log contain both message and help text.

***MSG:** Only first-level message text is written to the job log.

***SECLVL:** Both the message and second-level message text of the error message is written to the job log.

For an interactive job, the LOG parameter value on the SIGNOFF command takes precedence over the LOG parameter value specified for the job.

LOGCLPGM

Specifies whether the loggable commands that are run in a control language program are logged to the job log by way of the CL program's message queue. This parameter sets the status of the job's logging flag. If *NO is specified, the logging flag status is off and CL commands are not logged. If *YES is specified and the LOG (*JOB) value is specified in the Create CL Program (CRTCLPGM) command, all commands in the CL program that can be logged are logged to the job log.

For more information on request logging, refer to the LOG parameter in the CRTCLPGM command description.

***NO:** Commands in a CL program are not logged to the job log.

***YES:** Commands in a CL program are logged to the job log.

INQMSGRPY

Specifies the way that inquiry messages are answered for jobs that use this job description. The user can specify that inquiry messages are to be answered in the usual manner, that a default reply be issued, or that if certain conditions are met, an answer is issued to the inquiry based on those conditions. The conditions met are listed in the system reply list entries; refer to the Add Reply List Entry (ADDRPYLE) command description for more information.

***RQD:** An answer is required for any inquiry message that occurs while a job is running under this job description.

***DFT:** The default reply to the inquiry message is sent. If no default reply is specified in the message description of the inquiry message, the system default reply, *N, is used.

***SYSRPYL:** The system reply list is checked to determine whether there is an entry for any inquiry message issued as a result of running a job under this job description. If a match occurs, the reply value in that entry is used. If no entry exists for that message, the inquiry is issued.

PRTDEV

Specifies the qualified name of the default printer device for this job. If OUTQ(*DEV) is specified, the file is placed on an output queue with the same name as the printer.

***USRPRF:** The printer device name for jobs using this description is obtained from the user profile of the job at the time the job is started.

***SYSVAL:** The value specified in the system value QPRTDEV is used.

***WRKSTN:** The output queue assigned to the user's work station is used.

printer-device-name: Specify the name of the printer device that is used with this job description.

OUTQ

Specifies the qualified name of the output queue.

***USRPRF:** The output queue name for jobs using this job description is obtained from the user profile of the job at the time the job is started.

***DEV:** The output queue specified on the PRTDEV parameter is used.

***WRKSTN:** The output queue assigned to the user's work station is used.

The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

output-queue-name: Specify the qualified name of the default output queue that is used with this job description.

If the output queue does not exist when the job description is created, a library qualifier must be specified because the output queue name is retained in the job description.

HOLD

Specifies whether jobs using this job description are placed on the job queue in the hold condition. A job placed on the job queue in the hold condition is held until it is either released by the Release Job (RLSJOB) command or canceled by the End Job (ENDJOB) or Clear Job Queue (CLRJOBQ) command. If the job is not run before the next power-down of the system, the job queue can be cleared (and the job ended) when the next initial program load (IPL) is done.

***NO:** Jobs using this job description are not held when they are put on the job queue.

***YES:** The spooled file is held until released by the Release Spool File (RLSSPLF) command.

DATE

Specifies the date that is assigned to the job that uses this job description when the job is started.

***SYSVAL:** The value in the QDATE system value at the time the job is started is used as the job date.

job-date: Specify the value that is used as the job date for the job being started. The format that is currently specified for the system value QDATFMT must be used. The QDATFMT system value is in the *Work Management Guide*

SWS

Specifies the initial settings for a group of eight job switches used by jobs that use this job description. Only zeros (off) and ones (on) can be used. These switches can be set or tested in a CL program and used to control the flow of the program. For example, if a certain switch is on, another program could be called. The job switches may also be valid in other high-level language (HLL) programs.

00000000: The first setting for the job switches is all zeros for jobs that use this job description.

switch-settings: Specify any combination of eight zeros and ones that is used as the first switch setting for jobs using this job description.

DEVRCYACN

Specifies the recovery action to take for the job when an input/output error is encountered on the *REQUESTER device for interactive jobs that use this job description.

Note: This attribute is ignored for non-interactive jobs.

***SYSVAL:** The system value, QDEVRCYACN, is used as the device recovery action for this job description.

***MSG:** The application program requesting the input/output operation receives an error message that indicates the input/output operation has failed.

***DSCMSG:** The job is automatically disconnected. After the job is reconnected, it receives an error message indicating that an input/output error has occurred, and that the device has been recovered. Even though the device has been recovered, the contents of the screen prior to the error must be redisplayed.

***DSCENDRQS:** The job is automatically disconnected. After the job is reconnected, the ENDRQS command is issued specifying the processor that made the previous request. If no request processor is available, the ENDRQS command fails, and the message issued during the DSCMSG case is signaled.

***ENDJOB:** The job is ended with the *IMMED option. A job log is produced for the job.

***ENDJOBNOLIST:** The job is ended with the *IMMED option. No job log is produced for the job.

TSEPOOL

Specifies whether interactive jobs are moved to another main storage pool when they reach the end of the time slice.

***SYSVAL:** The value in QTSEPOOL (system value) at the time the job is started is used as the time slice end pool action for this job description.

***NONE:** The job is not moved when the end of the time slice is reached.

***BASE:** The job is moved to the base pool when the end of the time slice is reached.

AUT

Specifies the authority given to users who do not have specific authority to the job description, who are not on an authorization list, and whose user group has no specific authority to the job description. More information on

this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the job description is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the job description). The public authority is determined when the job description is created. If the CRTAUT value for the library changes after the job description is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the job description.

***USE:** The user can perform basic operations on the job description, such as running a program or reading a file. The user cannot change the job description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the job description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the job description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a Job Description for Interactive Jobs

```
CRTJOBD  JOBD(INT4)  USER(*RQD)
  RTGDTA(QCMDI)  INQMSGRPY(*SYSRPYL)
  TEXT('Interactive #4 job description
  for Department 127')
```

This command creates a job description named INT4 in the user's current library. This job description is for interactive jobs and is used by Department 127. When you sign on, you must type your password. The characters QCMDI are used as routing data that is compared with the routing table of the subsystem where the job is run. All inquiry messages are compared to the entries in the system reply list to determine whether a reply is issued automatically.

Example 2: Creating a Job Description for Jobs on a Specified Queue

```
CRTJOBD  JOBD(BATCH3)  USER(*RQD)
  JOBQ(NIGHTQ)  JOBPTY(4)  OUTPTY(4)
  ACGCDE(NIGHTQ012345)
  RTGDTA(QCMDB)  TEXT('Batch #3 job
  description for high pty night work')
```

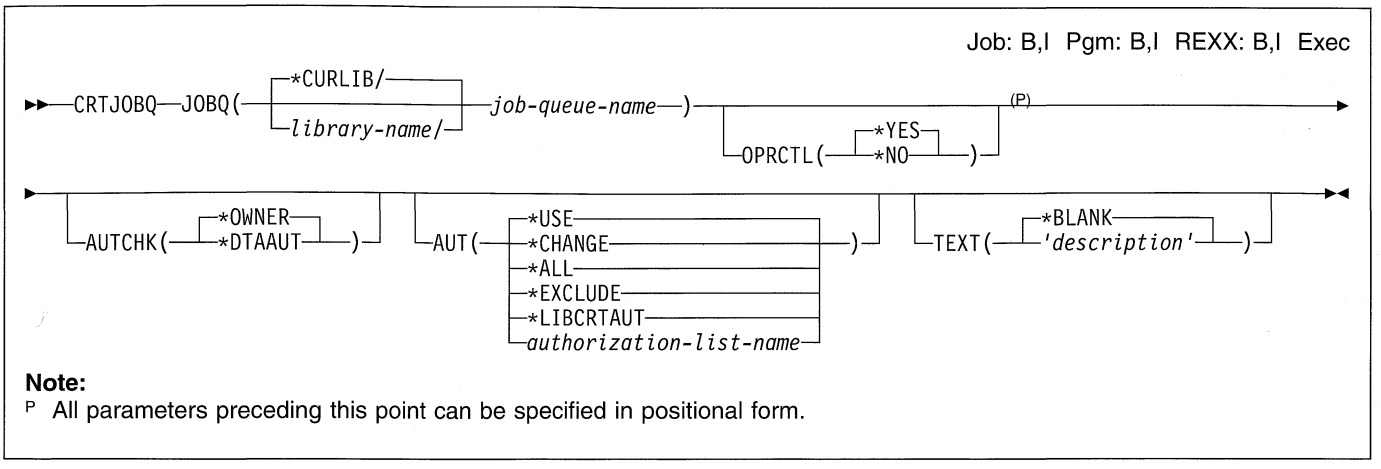
This command creates a job description named BATCH3 in the user's current library. The jobs using this description are placed on the job queue NIGHTQ. The priority for jobs using this description and their spooled output is 4. QCMDB is the routing data that is compared with entries in the routing table of the subsystem where the job runs. The accounting code of NIGHTQ012345 is used when recording accounting statistics for jobs that use this job description.

Example 3: Specifying Request Data

```
CRTJOBD  JOBD(PAYWK)  USER(QPGMR)   RTGDTA(QCMDB)
  RQSDTA('CALL PAY025 PARM(WEEKLY UNION)')
```

This command creates a job description named PAYWK in the user's current library. Jobs using this job description run under the IBM-supplied user profile for the programmer, QPGMR, and use the accounting code found in that user profile. If the job is started via the SBMJOB command, the accounting code of the person submitting the command is automatically used. The routing data QCMDB is compared with entries in the routing table of the subsystem where the job is run. The request data passed to the command processing program is a CALL command that names the application program that is run and passes a parameter to it.

CRTJOBQ (Create Job Queue) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Job Queue (CRTJOBQ) command creates a new job queue. A job queue contains entries for jobs submitted by using the following commands that are being processed by the system:

- Start Database Reader (STRDBRDR)
- Start Diskette Reader (STRDKTRDR)
- Start Printer Writer (STRPRTWTR)
- Start Diskette Writer (STRDSKWTR)
- Submit Job (SBMJOB)
- Submit Database Jobs (SBMDBJOB)
- Submit Diskette Jobs (SBMDKTJOB)
- Transfer Job (TFRJOB)

To add an entry for this job queue to a subsystem description, use the Add Job Queue Entry (ADDJOBQE) command.

Required Parameter

JOBQ

Specifies the qualified name of the job queue being created.

The name of the job queue can be qualified by one of the following library values:

***CURLIB:** The job queue is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the job queue is created.

Note: The temporary library QTEMP is not a valid library name. Job queues must be in permanent libraries.

job-queue-name: Specify the name of the job queue being created

Optional Parameters

OPRCTL

Specifies whether a user who has job control authority is allowed to control this job queue and manage the jobs on the queue. Users have job control authority if SPCAUT(*JOBCTL) is specified in their user profile.

***YES:** A user with job control authority can control the queue and manage the jobs on the queue.

***NO:** This queue and its jobs cannot be controlled by users with job control authority unless they have some other special authority.

AUTCHK

Specifies whether the commands that check the requester's authority to the job queue also check for ownership authority or data authority.

***OWNER:** The requester must have ownership authority to the job queue to pass the job queue authorization test. The requester can have ownership authority by being the owner of the job queue, sharing a group profile with the job queue owner, or running a program that adopts the job queue owner's authority.

***DTAAUT:** The requester who does not have the appropriate data authority to the job queue (*READ, *ADD, and *DELETE) cannot pass the job queue authorization test.

AUT

Specifies the authority given to users who do not have specific authority to the job queue, who are not on an authorization list, and whose user group has no specific authority to the job queue.

***USE:** The user can perform basic operations on the job queue, such as running a program or reading a file. The user cannot change the job queue. *USE authority provides object operational authority and read authority.

Note: The user can control jobs submitted by other users if AUTCHK(*DTAAUT) is specified.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the job queue.

***EXCLUDE:** The user cannot access the job queue.

***LIBCRTAUT:** The public authority for the job queue is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the job queue). The public authority is determined when the job queue is created. If the CRTAUT value for the library changes after the job queue is created, the new value does not affect any existing objects.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the job queue. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

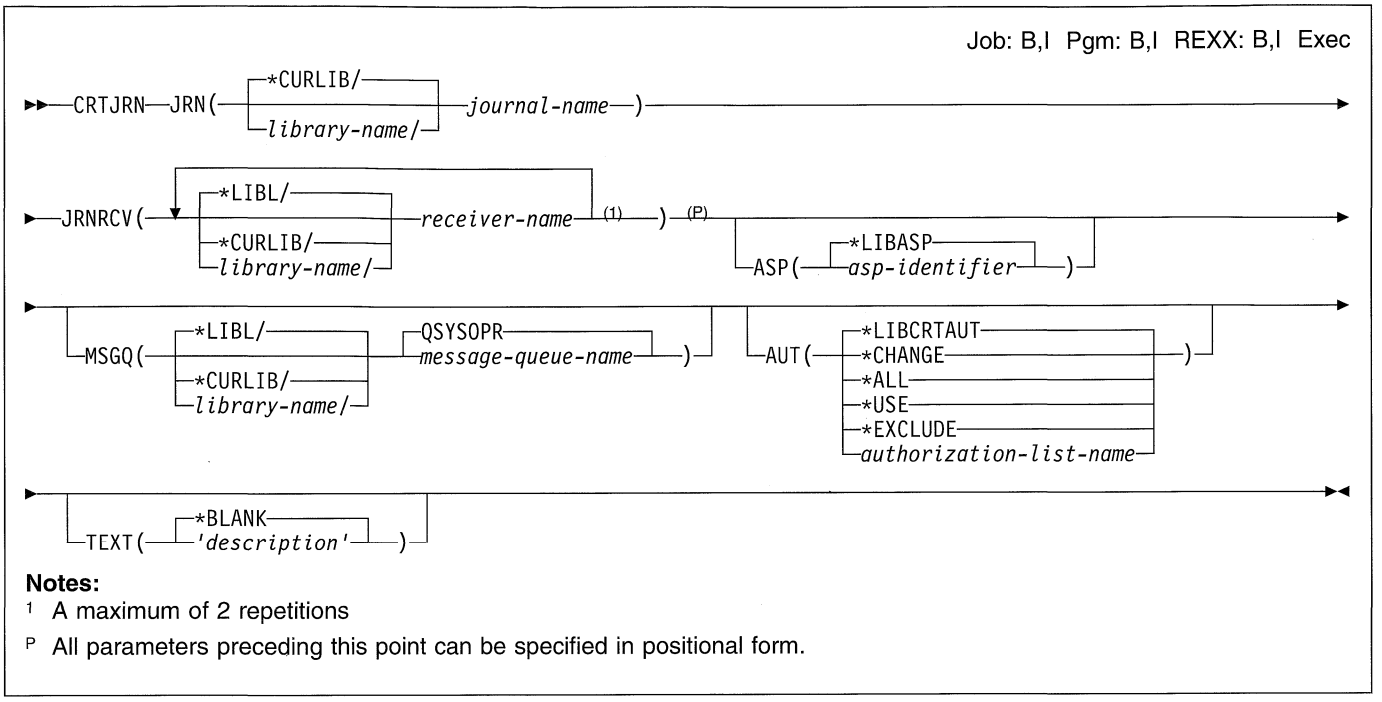
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTJOBQ   JOBQ(DEPTA)  AUT(*EXCLUDE)
  TEXT('Special queue for Dept A jobs')
```

This command creates a job queue named DEPTA and puts it in the current library. Because AUT(*EXCLUDE) is specified and OPRCTL(*YES) is assumed, the job queue is used and controlled only by the user who created the queue and by users with job control authority (*JOBCTL). Also, users with spool control authority (*SPLCTL) can control the queue.

CRTJRN (Create Journal) Command



Notes:

1 A maximum of 2 repetitions

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Journal (CRTJRN) command creates a journal with the specified attributes, and attaches the specified journal receivers to the journal. Once a journal is created, file changes can be journaled to it or user entries can be sent to it.

Restrictions:

1. A journal cannot be created in library QTEMP.
2. A message queue that is in library QTEMP cannot be specified for the MSGQ parameter.
3. The receivers specified must be created before issuing this command and they must be empty (that is, the receivers must not have been previously attached to any journal).

Required Parameters

JRN

Specifies the qualified name of the journal to be created.

The name of the journal can be qualified by one of the following library values:

***CURLIB:** The journal is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the journal is created.

journal-name: Specify the name of the journal to be created.

JRNRCV

Specifies the qualified names of the journal receivers to be attached to the specified journal.

The name of the journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

receiver-name: Specify the name of the journal receiver.

The journal receivers must not have been previously attached to any journal. Up to 2 journal receivers can be specified.

Optional Parameters

ASP

Specifies the auxiliary storage pool (ASP) from which the system allocates storage for the journal.

***LIBASP:** The storage space for the journal is allocated from the same ASP as the storage space of the journal's library.

asp-identifier: Specify a value ranging from 1 through 16 to specify the identifier of the ASP from which to have the storage space of the journal allocated. Valid values depend on how ASPs are defined on the system.

Note: The value of 1 is the system ASP.

MSGQ

Specifies the qualified name of the message queue associated with this journal. The message sent when a journal receiver's storage limit (threshold) is surpassed is sent to this message queue.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QSYSOPR: The message is sent to the QSYSOPR message queue.

message-queue-name: Specify the name of the message queue to which the message is sent. If this message queue is not available when the journal receiver threshold value is exceeded, the message is sent to the QSYSOPR message queue. To set the threshold value, refer to the Create Journal Receiver (CRTJRNRCV) command description.

AUT

Specifies the authority given to users who do not have specific authority to the journal, who are not on an authorization list, and whose user group has no specific authority to the journal.

***LIBCRTAUT:** The public authority for the journal is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the journal). The public authority is determined when the journal is created. If the CRTAUT value for the library changes after the journal is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the journal.

***USE:** The user can perform basic operations on the journal, such as running a program or reading a file. The user cannot change the journal. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the journal.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the journal. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

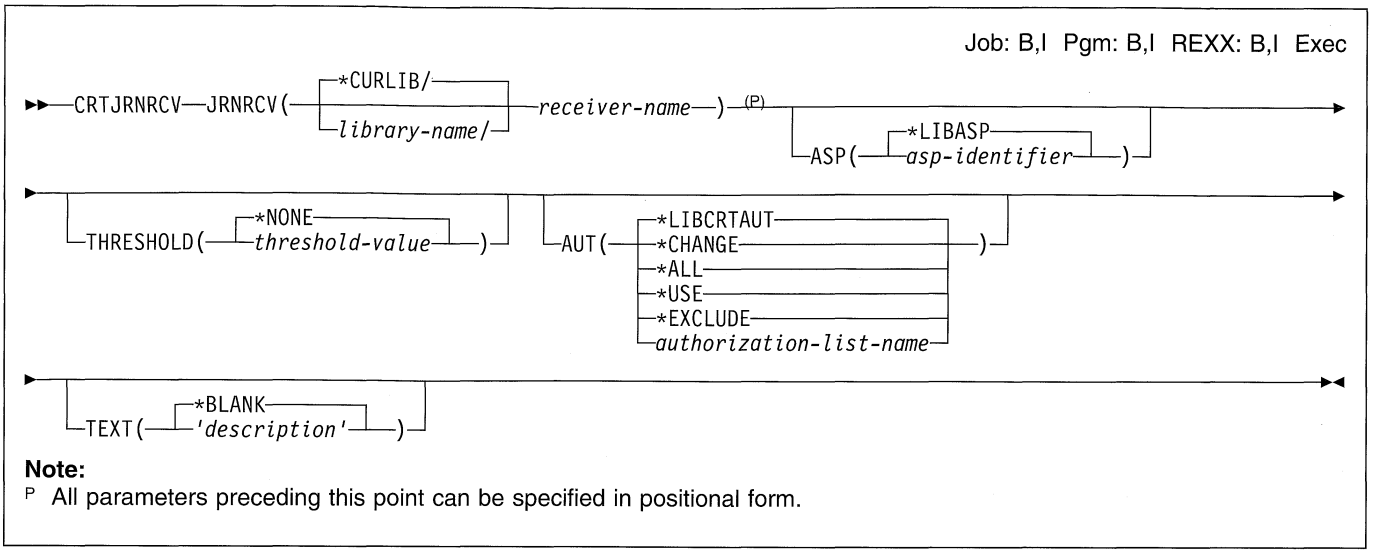
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTJRN   JRN(MYLIB/JRNLA)  JRNRCV(MYLIB/RCV01
  MYLIB/RCV01A)  ASP(3)
```

This command creates a journal named JRNLA in library MYLIB. Storage space for the journal is allocated from user auxiliary storage pool (ASP) 3. Two journal receivers (named RCV01 and RCV01A in library MYLIB) are attached to journal JRNLA. The public authority for the journal is taken from the CRTAUT parameter for library MYLIB.

CRTJRNRCV (Create Journal Receiver) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Journal Receiver (CRTJRNRCV) command creates a journal receiver. Once a journal receiver is attached to a journal (with the Create Journal (CRTJRN) or Change Journal (CHGJRN) command), journal entries may be placed in it. A preferred auxiliary storage pool (ASP), and a storage space threshold value can be specified for the journal receiver. When the size of the journal receiver is larger than the size specified, a message is sent to the message queue associated with the journal.

Restriction: A journal receiver cannot be created in library QTEMP.

Required Parameter

JRNRCV

Specifies the qualified name of the journal receiver to be created.

The possible library values are:

***CURLIB:** The journal receiver is created in the current library. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the journal receiver is to be created.

receiver-name: Specify the name of the journal receiver to be created.

Optional Parameters

ASP

Specifies the ASP from which the system allocates storage for the journal receiver.

***LIBASP:** The storage space for the journal receiver is allocated from the same ASP as the storage space of the journal receiver's library.

asp-identifier: Specify a value ranging from 1 through 16 to specify the identifier of the ASP from which to have the storage space of the journal receiver allocated. Valid values depend on how ASPs are defined on the system.

Note: The value of 1 is the system ASP.

THRESHOLD

Specifies a storage space threshold value (in K-bytes) for the journal receiver. If the threshold value is exceeded during journaling, a message (CPF7099) is sent to the message queue specified on the Create Journal (CRTJRN) or the Change Journal (CHGJRN) command. No action is taken by the system other than sending the message; you may want to take some action, however, such as issuing a CHGJRN command or saving the receivers.

***NONE:** No threshold value is specified (no message is sent).

threshold-value: Specify a value ranging from 1 through 1920000 that indicates K-bytes of storage. For example, an entry of 1000 specifies 1,024,000 bytes of storage space. When the size of the space for the journal receiver is larger than the size specified by this value, a message is sent to the identified message queue and journaling continues.

Note: It is recommended that the threshold value be no less than 250. Otherwise, you may see the threshold exceeded message too frequently. Also, if the threshold value is too small, the threshold exceeded message may occur when the journal receiver is attached to a journal either

with the Create Journal (CRTJRN) command or the Change Journal (CHGJRN) command.

AUT

Specifies the authority given to users who do not have specific authority to the journal receiver, who are not on an authorization list, and whose user group has no specific authority to the journal receiver.

***LIBCRTAUT:** The public authority for the journal receiver is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the journal receiver). The public authority is determined when the journal receiver is created. If the CRTAUT value for the library changes after the journal receiver is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the journal receiver.

***USE:** The user can perform basic operations on the journal receiver, such as running a program or reading a file. The user cannot change the journal receiver. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the journal receiver.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the journal receiver. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

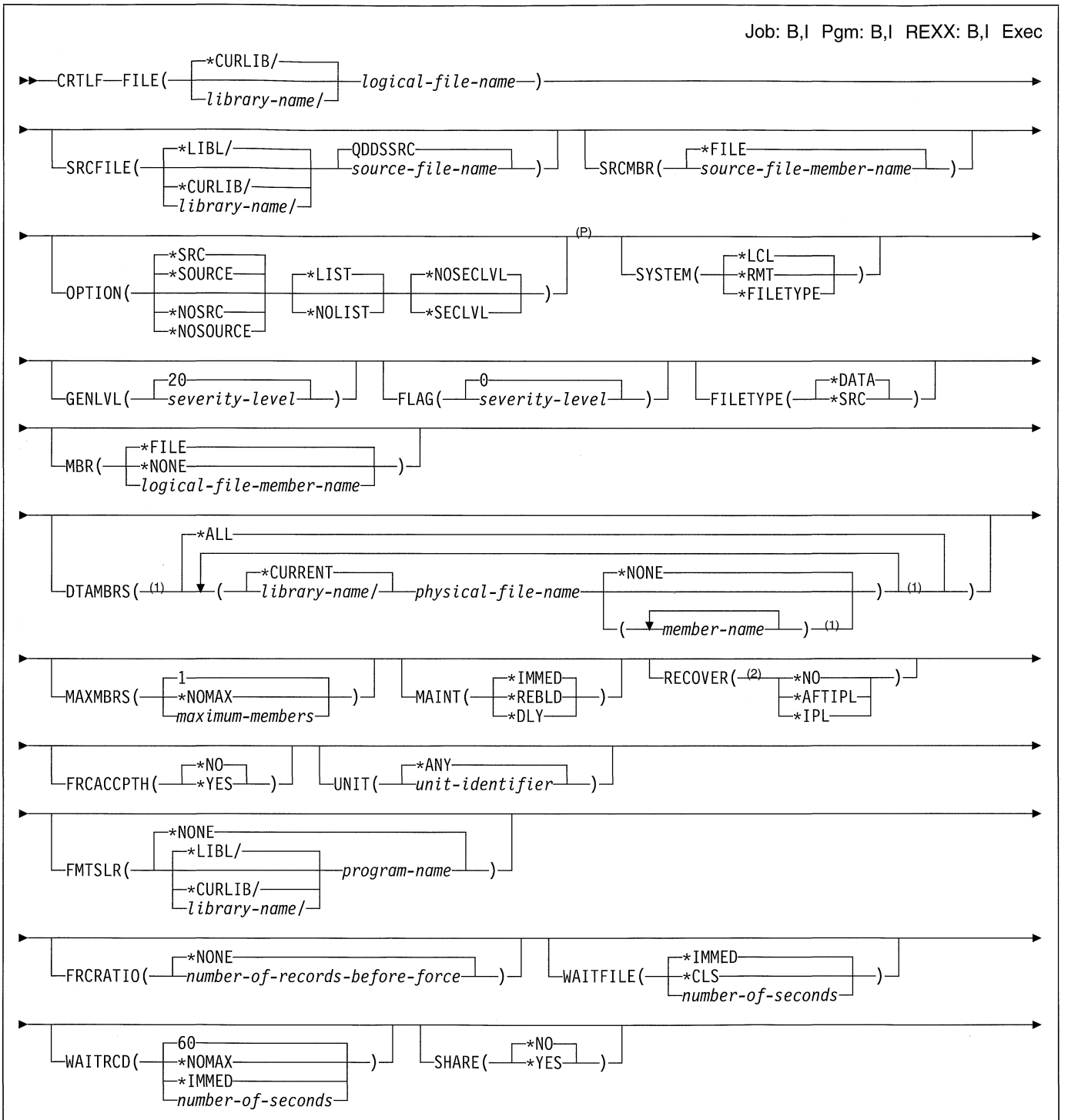
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTJRNRCV   JRNRCV(MYLIB/JRNRCLA)  ASP(3) THRESHOLD(1000)
  AUT(*ALL) TEXT('RECEIVER FOR WEEK 37')
```

This command creates a journal receiver named JRNRCLA in library MYLIB. Storage space for the journal receiver is allocated from user auxiliary storage pool (ASP) 3. When the size of JRNRCLA is larger than 1000 K (1,024,000) bytes, a message is sent to the message queue associated with the journal. The public authority to the journal receiver is *ALL.

CRTLF (Create Logical File) Command

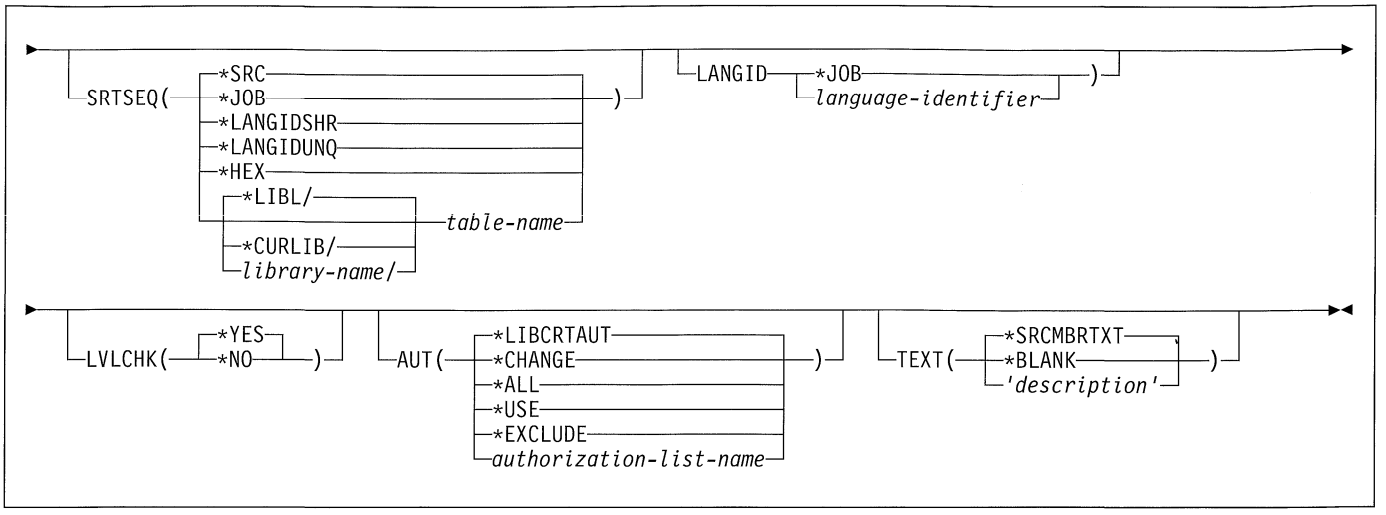


Notes:

1 The sum of **all** member names specified for **all** specified files cannot exceed 32.

P All parameters preceding this point can be specified in positional form.

2 Refer to the parameter description for the default action taken for this parameter.



Purpose

The Create Logical File (CRTLF) command creates a logical file in the database. The logical file is created from the file description parameters in this CRTLF command and from the previously entered data description specifications (DDS) that contain the source description of the logical file. A logical file is a database file that describes how data records contained in one or more physical files are presented to a program. The logical file does not contain data records. The data records are contained in the physical files associated with the logical file. To override the attributes of the logical file after it has been created, use the Override Database File (OVRDBF) command before the file is opened. To change the attributes of the logical file after it has been created, use the Change Logical File (CHGLF) command.

Restriction: To create a keyed logical file over one or more physical files, the user must have object management authority and object operational authority for each of the files specified on the PFILE or JFILE keywords in DDS. To create a non-keyed logical file, only object operational authority is required.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the logical file can be qualified by one of the following library values:

***CURLIB:** The logical file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the logical file is created.

Note: If a logical file and the physical file on which it is based are in different libraries, and the logical or physical file does not exist when it is to be restored (such as during disaster recovery or when the files are deleted), the access path is not restored. It is rebuilt. To make it possible for access paths to be restored and not rebuilt, the logical files and the based-on physical files must be in the same library. More information on the restoring of saved access paths is in the *Basic Backup and Recovery Guide*.

logical-file-name: Specify the name of the file that is to be created.

Note: If a DDM file is specified, the logical file (specified in the RMTFILE parameter on the Create Distributed Data Management File (CRTDDMF) command) is created on a remote system (specified in the RMTLOCNAME parameter of the CRTDDMF command). See the SYSTEM parameter of this command.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file that contains the DDS used to create the logical file. The source file contains the specifications that describe the record formats and their fields, and the access paths for the file and its members. The specifications that can be made in DDS are discussed in the *Database Guide* and the *DDS Reference*.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The source file, QDDSSRC, contains the DDS used to create the logical file.

source-file-name: Specify the name of the source file that contains the DDS used to create the logical file.

SRCMBR

Specifies the name of the source file member that contains the DDS for the logical file being created; the member is in the source file specified in the SRCFILE parameter (or its default, QDDSSRC).

***FILE:** The source file member name is the same as that of the logical file being created.

source-file-member-name: Specify the name of the member in the source file specified by the SRCFILE parameter that is used to create the logical file.

OPTION

Specifies the type of printout produced when the file is created. Up to three of the following values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value will be used.

Note: The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

Program Creation Options

***SRC** or ***SOURCE:** A printout of the source statements used to create the file, and of errors that occur, is created.

***NOSRC** or ***NOSOURCE:** No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the record format that contains the error.

Source Listing Options

***LIST:** An expanded source printout is created, showing a detailed list of the file specifications that result from the source statements and references to other file descriptions.

***NOLIST:** An expanded source printout is not created.

***NOSECLVL:** The messages section of the DDS printout does not contain the second-level message for the errors found during DDS processing.

***SECLVL:** Second-level message text is included in the source listing.

SYSTEM

Specifies whether the logical file is created on the local system or the remote system.

***LCL:** The logical file is created on the local system.

***RMT:** The logical file is created on a remote system using DDM. The logical file name specified on the FILE parameter must be the name of the DDM file, which is created using the Create Distributed Data Management File (CRTDDMF) command. The DDM file contains the name of the logical file being created (RMTFILE parameter on the CRTDDMF command) and the name of the remote system (RMTLOCNAME parameter on the CRTDDMF command) on which the file is to be created.

***FILETYPE:** If the name specified on the FILE parameter is a DDM file, the logical file is created on the remote system specified by CRTDDMF(RMTLOCNAME) for that DDM file. Otherwise, the name on the FILE parameter cannot be the name of an existing file since a logical file of that name is created on the local system.

GENLVL

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

Note: This parameter applies only to messages created while processing DDS source. Messages created anywhere else in the file creation process are not affected by this parameter.

20: If errors occur in the DDS source with a severity level greater than or equal to 20, the file is not created.

severity-level: Specify a severity level ranging from 0 through 30. The file is not created if the severity level specified for this parameter equals 0 or is less than the severity level that occurs in the data description specifications (DDS) source. This value must be greater than or equal to value specified on the FLAG parameter.

FLAG

Specifies the minimum severity level of messages to be listed in the DDS source listing.

0: All messages are listed.

severity-level: Specify the minimum severity level of messages to be listed. Valid values range from 0 through 30. The severity level specified must be less than or equal to the severity level specified on the GENLVL parameter.

FILETYPE

Specifies whether each member of the logical file being created contains data records, or contains source records (statements) for a program or another file. The file could contain, for example, RPG source statements for an RPG program or DDS source statements for another physical, logical, or device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***DATA:** The logical file contains data records.

***SRC:** The logical file contains source records. This value cannot be specified for join logical files.

MBR

Specifies the name of the logical file member that is added when the logical file is created. Other members

can be added to the file after it is created by using the Add Logical File Member (ADDLFM) command.

***FILE:** The member being added has the same name as that of the logical file that contains the member. This name is specified on the FILE parameter.

***NONE:** No member is added when the file is created.

logical-file-member-name: Specify the name of the member that is added when the logical file is created.

DTAMBRS

Specifies the names of the physical files and members that contain the data associated with the logical file member being added by this command. A logical file member can be based on all of the physical files and members on which the logical file itself is based, specified by DTAMBRS(*ALL), or the member can be based on a subset of the total members and files, specified by DTAMBRS(qualified-file-names (member-names)).

For DDM file:

- The file names specified in the DTAMBRS parameter must be the names of the DDM files that represent the remote based-on physical files. If a member name was specified as part of the remote file name in the DDM file, only that member name can be specified on the DTAMBRS parameter. The member names must be the actual remote file member names.
- The based-on physical files must be at the same system location as the logical file to which the member is being added.
- When no member name is specified for the remote file name in the DDM file, all members are accessible. When only one member name is specified, only that member is accessible through that DDM file.

When the FILE parameter specifies a join logical file or an arrival sequence logical file, only one data member must be specified on the DTAMBRS parameter for each physical file that was specified on the PFILE or JFILE keyword in the DDS. *ALL is valid only if each based-on physical file has only one member. If any of the physical files has more than one member, the specific physical file member must be specified on the DTAMBRS parameter.

The same physical file name can be specified more than once on the JFILE keyword. In this case, each occurrence of the file name is treated as a different based-on physical file, and must be specified on the DTAMBRS parameter.

***ALL:** The logical file member being added is based on all the physical files and members (that exist at the time this CRTLF command is entered) used by the logical file. At least one member must exist in at least one of the physical files. The physical file names are specified on the PFILE or JFILE parameter in the DDS.

The physical file names must match a name on the PFILE or JFILE keywords in the DDS and cannot be specified more often on the DTAMBRS parameter than on the PFILE or JFILE keywords in the DDS. For join logical files, all physical files specified on the JFILE keyword must be specified on the DTAMBRS parameter and each physical file must contain only one member. If a physical file name is not specified for a physical file that is on a PFILE or JFILE keyword in the DDS, the logical file member is not based on any member of that physical file.

Up to 32 qualified physical file names and physical file member names can be specified. Also, the total number of member names cannot exceed 32. For example, one file can specify 32 members, two files can each have 16 members, or 32 files can each have one member specified.

When a logical file is created, the physical files specified on the PFILE or JFILE DDS keyword are used to create the logical file. If no library name is specified for the physical files on the PFILE or JFILE keyword, the library list (*LIBL) at file creation time is used to find the physical files; the physical files from the library list are used to create the logical file. The qualified physical files from the PFILE or JFILE keyword (regardless of whether a library name was specified or if the library list was used to find the files) are the physical files associated with the logical file. The names of the physical files associated with the logical file are saved in the description of the logical file. When a member is added to the logical file, the DTAMBRS parameter is used to specify the physical file members associated with the logical file member. Each physical file name specified on the DTAMBRS parameter must be the name of a physical file that is associated with the logical file (saved in the description of the logical file).

If a library name is not specified, the current library name (*CURRENT) from the logical file description is used. If the library name is specified, the physical file must be a physical file associated with the logical file. If the logical file is associated with more than one physical file of the same name, the library name must be specified.

The following examples show the syntax for specifying single and multiple members for single and multiple physical files. In the examples, the abbreviation PF represents a physical file name, LIB represents a library qualifier, and M represents a member name.

Single physical file and member:
```
DTAMBRS((PFA M1))
```
Single file with multiple members:
```
DTAMBRS(PFA (M1 M2 M3))
```
Multiple files with single members and no members:
```
DTAMBRS((PFA M1) (PFB M4) (PFE *NONE))
```
Multiple files with multiple members:
```
DTAMBRS((PFA (M1 M3 M4)) (PFB (M1 M2 M4)))
```
Multiple files with the same name in different libraries:
```
DTAMBRS((LIBX/PFA M1) (LIBY/PFA (M1 M2)))
```

Multiple files with the same name in the same library:
DTAMBRS((LIBX/PFA M1) (LIBX/PFA M1))

When more than one physical file member is specified for a physical file, the member names are specified in the order in which records are retrieved when duplicate key values occur across those members.

Refer to "Additional Considerations" at the end of this command description for more details.

Element 1: Names of Physical Files

physical-file-name: Specify the names of the physical files that contain the data being accessed by the logical file member being added.

Element 2: Names of Members

***NONE:** A member name is not specified.

member-name: Specify the names of the members that contain the data being accessed by the logical file member being added.

MAXMBRS

Specifies the maximum number of members that the logical file being created can contain.

1: Only one member can be contained in the file.

***NOMAX:** The system maximum is used.

maximum-members: Specify the maximum number of members that the logical file can contain. Valid values range from 1 through 32767.

MAINT

Specifies, for join logical files or files with keyed sequence access paths, the type of access path maintenance being used for every member of the logical file.

Note: For a join logical file, this parameter value applies to all join secondary files even if the join file is not a keyed file.

***IMMED:** The access path is maintained for each physical file member whether the source physical file is opened or closed. The access path is changed whenever a record is updated, added to, or deleted from a member of this file or a logical file member based on a member of this file.

***REBLD:** The access path is completely rebuilt when a logical file member is opened. The access path is continuously maintained until the member is closed; the access path maintenance is then ended. *REBLD is not valid for access paths that require unique key values.

***DLY:** The maintenance of the access path is delayed until the logical file member is opened. Then the access path is changed only for records that have been added, deleted, or changed since the file was last opened. While the file is open, all changes made to based-on file members are immediately reflected in the access paths of the opened file's own members, no matter what is specified for this parameter. To prevent a long rebuilding time when the file is opened, *DLY should be specified only when the number of changes to the access path between successive open operations are small; that is, when the file is opened frequently or when the key fields in records for this access path change infrequently. *DLY is not valid for access paths that require unique key values.

If the number of changes between a close operation and the next open operation reaches approximately 10 percent of the access path size, the system stops saving changes and the access path is completely rebuilt the next time the file is opened.

RECOVER

Specifies, for files having immediate or delayed maintenance on their access paths, when recovery processing of the file is performed after a system failure occurs while the access path is being changed. This parameter is valid only for join logical files or files with a keyed access path.

If *IMMED or *DLY is specified for the MAINT parameter, the access path can be rebuilt during initial program load (IPL) (before any user can run a job), after IPL has ended (during concurrent job running), or when the file is next opened. While the access path is being rebuilt, the file cannot be used by any job.

During the IPL, an Override Access Path Recovery display lists those paths that must be recovered and what the RECOVER parameter value is for each path. The user can override the RECOVER parameter value on this display. More information is in the *Advanced Backup and Recovery Guide*.

If *REBLD is specified for the MAINT parameter, the access path is rebuilt the next time its file is opened.

***NO:** The access path of the file is not rebuilt during or after the IPL. The file's access path is rebuilt when the file is next opened. *NO is the default for all files that do not require unique keys.

***AFTIPL:** The file has its access path rebuilt after the IPL has been completed. This option allows other jobs not using this file to start processing immediately after the completion of IPL. If a job tries to allocate the file while its access path is being rebuilt, a file open exception occurs. *AFTIPL is the default for files that require unique keys.

***IPL:** The file has its access path rebuilt during the IPL. This ensures that the file's access path is rebuilt before the first user program tries to use it; however, no jobs can start running until after all files that specify RECOVER(*IPL) have their access paths rebuilt.

FRCACCPTH

Specifies, for join logical files or files with keyed access paths, whether access path changes are forced to auxiliary storage along with the associated records in the file whenever the access path is changed. FRCACCPTH(*YES) minimizes (but does not remove) the possibility that an abnormal job end could cause damage to the access path that would require it to be rebuilt.

Note: For a join logical file, this parameter value applies to all join secondary files even if the join file is not a keyed file.

***NO:** The access path and changed records are not forced to auxiliary storage whenever the access path is changed.

***YES:** The access path and changed records are forced to auxiliary storage whenever the access path is changed. If this value is specified, MAINT(*REBLD) cannot be specified.

FRCACCPTH(*YES) slows the response time of the system if the access path is changed in an interactive job. If the access path is changed frequently, the overall performance of the system is affected somewhat.

UNIT

Specifies the unit identifier of the auxiliary storage unit on which the storage space for the file, the file members, and their access paths is allocated. The system attempts to allocate the storage space for the file and for all its members and their keyed sequence access paths on the specified unit. This includes the initial allocation for each member and any additional extensions needed later. If the system cannot allocate the storage space on the specified unit, it allocates the space on any available unit and sends a message to the job log. In any case, the file is entirely usable.

The unit identifier is a number ranging from 1 through 255, assigned when a new disk device is configured. The configured disk device can be shown and changed by using the Work with Disk Devices display from the Start System Service Tool (STRSST) command. More information on the Start System Tools (SST) is in the *Advanced Backup and Recovery Guide*.

Note: For a join logical file, this parameter value applies to all join secondary files even if the join file is not a keyed file.

***ANY:** The storage space for the file and its members is allocated on any available auxiliary storage unit.

unit-identifier: Specify the storage unit on which the system attempts to allocate the storage space for the file's access paths.

If the unit specified is part of any user auxiliary storage pool (ASP) (ASP 2 through 16), the system allocates space from ASP 1, the system auxiliary storage pool.

FMTSLR

Specifies the qualified name of a record format selector program that is called when the logical file member contains more than one record format. The user-written selector program is called when a record is inserted into the database file and a record format name is not included in the high-level language program. The selector program receives the record as input, determines the record format used, and returns it to the database. This program must perform this function for every member in the logical file that has more than one record format, unless the high-level language program itself specifies the record format name. More information about the use of format selector programs is in the *Database Guide*.

This parameter is not valid if the logical file has only one record format.

***NONE:** There is no selector program for this logical file. If the file has more than one logical record format, the high-level language program must specify the record format name.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the format selector program to be called when a record is inserted into a member having more than one format. The selector program name can be optionally qualified by the name of the library in which the program is stored. If no library qualifier is given, *LIBL is used to find the program.

A program specified as the format selector program cannot be created with USRPRF(*OWNER) specified in its create program command.

FRCRATIO

Specifies the number of inserted, updated, or deleted records that are processed before they are forced to auxiliary (permanent) storage. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

The force write ratio specified for a logical file cannot be less than or equal to the smallest force write ratio of its based-on files. If a larger force write ratio is specified, it is ignored and a message is sent informing the user of the action.

For example, if the force ratios of three physical files are 2, 6, and 8, the logical file force ratio that is based on these three physical files must be as restrictive as the least of them; that is 2 in this case. Two would be used even if the FRCRATIO parameter is not specified. Thus, each time a program inserts, updates, or deletes two records in the logical file (regardless of which based-on physical files are affected), those records are forced to permanent storage.

If a physical file associated with this logical file is being journaled, a large force write ratio or *NONE is specified. More information on journal management is in the *Advanced Backup and Recovery Guide*.

***NONE:** There is no specified force ratio; the system determines when the records are written to auxiliary storage.

number-of-records-before-force: Specify the number of inserted, updated, or deleted records that are processed before they are explicitly forced to auxiliary storage.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds that a program waits for the file resources to be allocated to the job. Valid values range from 1 through 32767 seconds.

WAITRCD

Specifies the number of seconds that a program waits for a record to be updated or deleted, or for a record read in the commitment control environment with LCKLVL(*ALL) specified. More information on record locking is in the *Database Guide*. If the record is not allocated in the specified wait time, an error message is sent to the program.

60: A program waits for 60 seconds.

***NOMAX:** The system maximum is used.

***IMMED:** The program does not wait; when a record is locked, an immediate allocation of the record is required.

number-of-seconds: Specify the number of seconds that a program waits for the file resources to be allocated to the job. Valid values range from 1 through 32767 seconds.

SHARE

Specifies whether the open data path (ODP) for the logical file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

SRTSEQ

Specifies the sort sequence used for this file. The sort sequence is used with the LANGID parameter to determine which sort sequence table is used.

***SRC:** The table specified in the DDS on the ALTSEQ parameter is used. If ALTSEQ is not used in the DDS, use the value specified for *JOB on this parameter.

***JOB:** The sort sequence must be resolved when the object is created.

***LANGIDSHR:** The sort sequence table can contain the same weight for multiple characters, and is the shared-weight table associated with the language specified in the LANGID parameter.

***LANGIDUNQ:** The sort sequence table must contain a unique weight for each character in the code page.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.

The name of the table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify a table name.

LANGID

Specifies the language identifier to be used when SRTSEQ(*LANGIDUNQ) or SRTSEQ(*LANGIDSHR) is specified. The language identifier is used with the SRTSEQ parameter to determine which sort sequence table is used.

***JOB:** The language identifier specified for the job is used.

language-identifier: Specify a language identifier.

LVLCHK

Specifies whether the record format level identifiers in the program are checked against those in the logical file when the file is opened. If so, the record format identi-

fiers in the program must match those in the logical file. This value can be overridden by the Override with Database File (OVRDBF) command at run time.

***YES:** The level identifiers of the record formats are checked when the file is opened. If the level identifiers do not match, an error message is sent to the program requesting the open, and the file is not opened.

***NO:** The level identifiers are not checked when the file is opened.

AUT

Specifies the authority given to users who do not have specific authority to the logical file, who are not on an authorization list, and whose user group has no specific authority to the logical file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the logical file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the logical file). The public authority is determined when the logical file is created. If the CRTAUT value for the library changes after the logical file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the logical file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the logical file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the logical file.

***USE:** The user can perform basic operations on the logical file, such as running a program or reading a file. The user cannot change the logical file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the logical file.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the logical file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member being used to create the logical file. If the source file is a database file, the text is taken from the source member. Text can be added or changed for a database source member by using the Source Entry Utility, or by using either the Add Logical File Member (ADDLFM) or Change Logical File Member (CHGLFM) command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a Logical File Without Members

```
CRTLF   FILE(INVEN/STOCKCTL)  SRCFILE(SRCLIB/STKLFSRC)
  MBR(*NONE)
```

This command creates a logical file named STOCKCTL, in the INVEN library. The source descriptions in the source file STKLFSRC in the SRCLIB library are used to create the logical file. The file is created without any members (*NONE was specified), and only one member can be added later (because one member is the default for the MAXMBRS parameters). The logical file accesses the data contained in the physical files specified in the DDS source file used to create this logical file. For successful completion of the CRTLF command, the user must have object operational authority for all the physical files specified in the DDS. If the logical file is keyed, object management authority is also required.

Example 2: Creating a Logical File With Members

```
CRTLF  FILE(PAYLIB/PAYCODESEQ)
  SRCFILE(PAYLIB/PAYTXSRC)
  DTAMBRS(PAYTRANS FIRSTQTR)
  UNIT(02)  AUT(*EXCLUDE)
  TEXT('Pay taxes in code sequence')
```

This command creates a logical file and logical file member, both named PAYCODESEQ in the PAYLIB library. The file and its member are created from the PAYTXSRC source file that is in the same library. The logical file is stored on auxiliary storage unit 02. The logical file member accesses the data contained in the FIRSTQTR member of the physical file PAYTRANS. The logical file is secured for the private use of the owner. The owner must have object operational authority for the PAYTRANS file to create the member. If the logical file is keyed, object management authority is also required.

Additional Considerations

This section supplies additional information for coding the DTAMBRS parameter when physical file names and member names are specified.

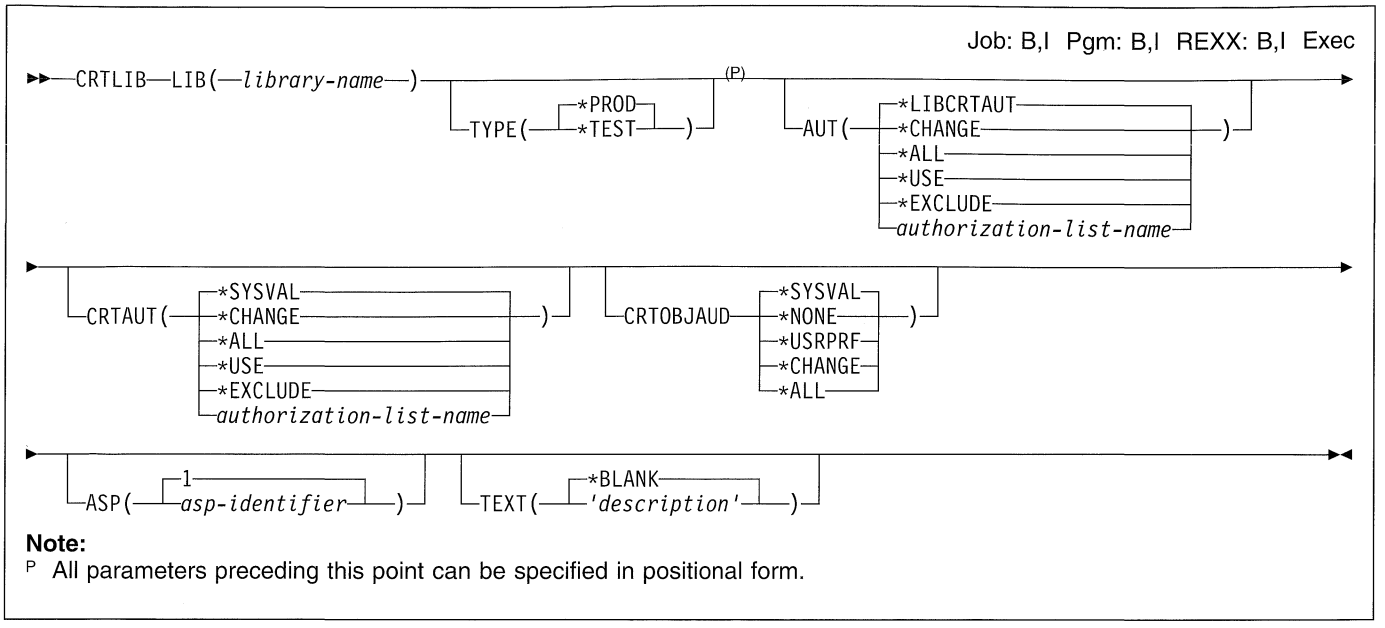
- Each physical file specified on the DTAMBRS parameter must be a physical file associated with the logical file. If a library name is not specified on the DTAMBRS parameter for a physical file, the current library name (*CURRENT) from the logical file description is used. If the library name is specified, the physical file must be associated with the logical file. The library name must be specified if the logical file is associated with more than one physical file of the same name (for example,

physical file PF in library LIB1 and physical file PF in library LIB2).

- When more than one physical file member is specified for a physical file, the member names are specified in the order in which records are presented when a duplicate key value occurs across those members. If multiple members from one physical file are specified, add operations are not possible for that record format from high-level language programs.
- The logical file description contains a list of the physical files associated with the logical file. The list contains the content and order of the physical files. This list can be displayed by using the DSPFD (Display File Description) command if TYPE(*ACCPTH) is specified.

More information about data members is in the *Database Guide*.

CRTLIB (Create Library) Command



Purpose

The Create Library (CRTLIB) command adds a new library to the system. The library must have been created before any objects can be placed into it. When the library is created, it is actually stored as part of the internal system. However, although it is a separate library, it appears as though it exists in the QSYS (system) library.

Restriction: You must have *AUDIT special authority to specify a value other than *SYSVAL on the CRTOBJAUD parameter.

Required Parameter

LIB

Specifies the name of the library being created.

Note: Do not use a name that begins with the character Q. The system assumes that libraries with those names are system libraries.

Optional Parameters

TYPE

Specifies the type of library being created.

***PROD:** This is a production library. Database files in production libraries cannot be opened for updating if a user is in debug mode and requests that production libraries be protected. A user can protect all database files in production libraries by specifying UPDPROD(*NO) on the Start Debug (STRDBG) command to begin testing. However, this protection does not prevent the program from deleting database files or from changing other objects (such as data areas) in the library.

***TEST:** This is a test library. All objects in a test library can be updated during testing, even if special protection is requested for production libraries.

AUT

Specifies the authority given to users who do not have specific authority to the library, who are not on an authorization list, and whose user group has no specific authority to the library.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove user ids.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the library.

***USE:** The user can perform basic operations on the library, such as running a program or reading a file. The

user cannot change the library. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the library.

authorization-list-name: Specify the name of the authorization list used.

CRTAUT
Specifies, when an object is created in this library, the authority given to users who do not have specific authority to an object (public authority), who are not on the authorization list, and whose user group has no specific authority to an object.

***SYSVAL:** The QCRTAUT system value is used.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations on the object except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence, specify the security for the object, change the object, and perform basic functions on the object. The user cannot transfer ownership of the object. If the object is an authorization list, the user cannot add, change, or remove user IDs.

***USE:** The user can perform basic operations on the library, such as running a program or reading a file. The user cannot change the library. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the library.

authorization-list-name: Specify the name of the authorization list whose authority is used for the object.

CRTOBJAUD
Specifies the auditing value for objects created in this library.

***SYSVAL:** The value specified in the system value QCRTOBJAUD is used.

***NONE:** Using or changing this object will not cause an audit entry to be sent to the security journal.

***USRPRF:** The user profile of the user accessing this object is used to determine if an audit record will be sent for this access. The OBJAUD keyword of the CHGUSRAUD command is used to turn on auditing for a specific user.

***CHANGE:** All change accesses to this object by all users are logged.

***ALL:** All change or read accesses to this object by all users are logged.

ASP
Specifies the auxiliary storage pool (ASP) from which the system allocates storage for the library. For libraries created in a user ASP, all objects in the library must be in the same ASP as the library.

1: The storage space for the library is allocated from the system auxiliary storage pool.

asp-identifier: Specify a value ranging from 2 through 16 for the ASP identifier. These values depend on how ASPs are defined on the system.

TEXT
Specifies text that briefly describes the library. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Adding a Production Library

```
CRTLIB   LIB(MYLIB)  TEXT('My Production Library')
```

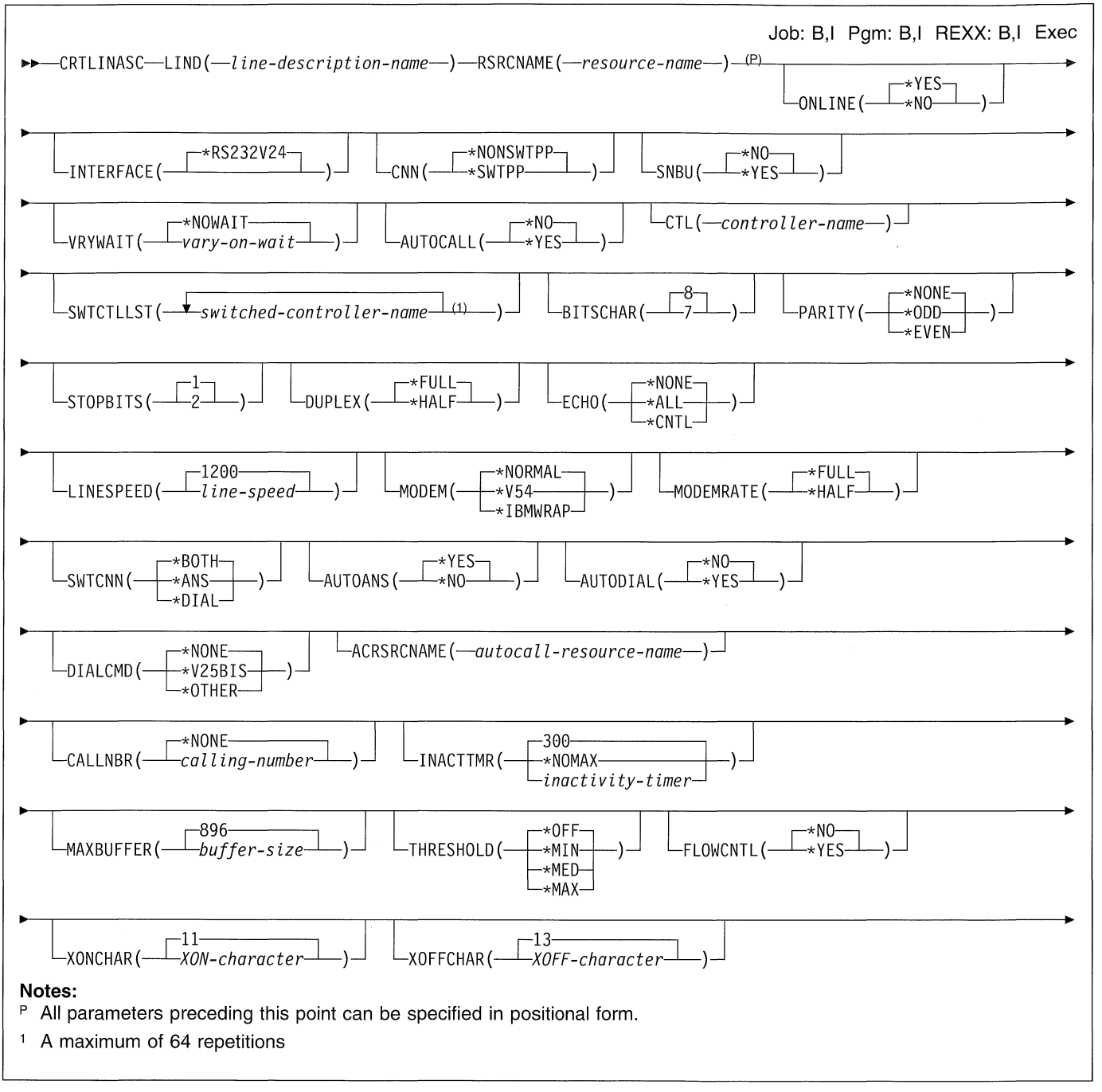
The library MYLIB is added to the system. The library is a production library; only the owner has object existence and object management authority for it. The authority for other users to library MYLIB is determined by the create authority of library QSYS, since *LIBCRTAUT was assumed for the AUT parameter. The text, 'My Production Library', is displayed whenever the library description for MYLIB is displayed.

Example 2: Adding a Test Library

```
CRTLIB   LIB(Z)  TYPE(*TEST)  AUT(*EXCLUDE)
  TEXT('This is a test library')
```

Test library Z is added to the system. Only the owner of Z can use it because no other users have been granted any authority. The specified text ('This is a test library') is displayed whenever the library description for Z is displayed.

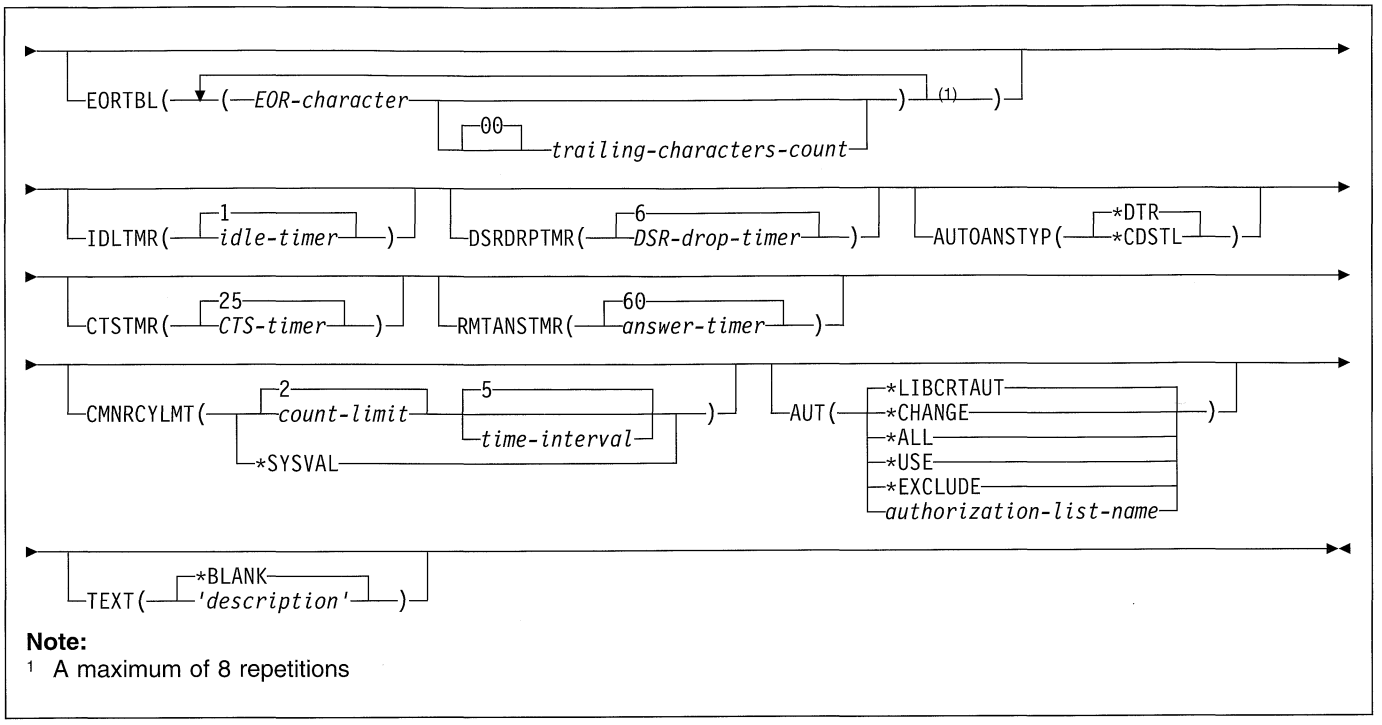
CRTLINASC (Create Line Description (Async)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions



Note:
1 A maximum of 8 repetitions

Purpose

The Create Line Description (Async) (CRTLINASC) command creates a line description for an asynchronous line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND
Specifies the name of the line description being created.

RSRCNAME
Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is two, then the resource name would be LIN012.

Optional Parameters

ONLINE
Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

INTERFACE
Specifies the type of physical interface on the IOA port. RS-232/V.24 is the default.

CNN
Specifies the type of line connection used.

***NONSWTPP:** A nonswitched point-to-point line is used.

***SWTPP:** A switched point-to-point line is used.

SNBU
Specifies, for controllers attached to nonswitched lines only, whether the switched network backup (SNBU) feature is activated or deactivated. This feature lets the user bypass a broken nonswitched connection by establishing a switched connection. This parameter applies only if SWITCHED(*NO) and SNBU(*YES) are specified when the controller description is created.

Note: The Change Line Description (Async) (CHGLINASC) and Change Controller Description (Async) (CHGCTLASC) commands must be used to activate the switched backup feature. Switched network backup (SNBU) is valid only if the local modem and remote modem both support the SNBU feature.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The modem has the SNBU feature.

VRYWAIT
Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

AUTOCALL

Specifies, for SNBU, whether the line has an associated autocall unit which performs automatic calling to the remote system.

***NO:** This switched line does not have an autocall unit.

***YES:** This switched line has an autocall unit.

CTL

Specifies the name of the controller description to which this object is attached.

Note: The controller description must already exist. Do not use this parameter when following the normal procedure of creating descriptions for lines first, controllers second, and devices last. Use this parameter only when the associated controller descriptions have been created before this line description.

SWTCTLLST

Specifies the names of up to 64 controllers that can establish a connection with the switched line. The controller descriptions must already exist.

Note: Do not use this parameter when following the normal procedure of creating descriptions for lines first, controllers second, and devices last. Use this parameter only when the associated controller descriptions have been created before this line description.

BITSCHAR

Specifies the number of data bits per character (excluding the parity bit, if any).

8: The default value is eight data bits per character.

7: Seven data bits per character are used.

PARITY

Specifies the type of parity used for error checking. A parity bit is a binary digit inserted in each byte of data to make the arithmetic sum of all the digits, including the parity bit, always odd or always even.

Note: The remote system must use the type of parity specified by the PARITY parameter.

***NONE:** No parity bit is inserted in the data byte.

***ODD:** The arithmetic sum of all the digits, including the parity bit, is odd.

***EVEN:** The arithmetic sum of all the digits, including the parity bit, is even.

STOPBITS

Specifies the number of bits added to the end of each character. These bits are used to keep the local and remote ends of the line synchronized. The remote system must use the same number of stop bits as specified by the STOPBITS parameter for the local system.

Note: At line speeds of 300 bps or lower, two stop bits are recommended.

1: The default value adds one stop bit to each character.

2: Two stop bits are used.

DUPLEX

Specifies whether request-to-send (RTS) is permanently turned on (for full-duplex modems) or turned on only when data transmission is required (for half-duplex modems).

***FULL:** Request-to-send (RTS) is permanently turned on (for full-duplex modems).

***HALF:** RTS is turned on only when transmission is required (for half-duplex modems).

ECHO

Specifies whether the system sends back (echoes) all characters it receives to the remote system, or echoes all characters except end-of-record (EOR) characters, or if echo is inhibited.

***NONE:** No characters received are echoed to the remote system.

***ALL:** All characters received are echoed to the remote system.

***CNTL:** All characters received prior to EOR characters are echoed to the remote system.

Note: Specify *ALL or *CNTL when communicating with a remote system that requires echo. If *ALL or *CNTL are specified for the echo prompt, *FULL must also be specified for the duplex prompt.

LINESPEED

Specifies the line speed in bits per second (bps).

1200: The default value is 1200 bps.

line-speed: Specify the line speed (bps). Valid lines speeds are: 50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 7200, 9600, and 19200 bps.

MODEM

Specifies the type of modem supported on the communications line. Refer to the modem documentation to determine the appropriate value.

***NORMAL:** No attempt is made to run diagnostic tests on the modem.

***V54:** A certain type of diagnostic test (as defined by the CCITT recommendations) is run to the modem. The AS/400 system supports CCITT V.54 loop 3 (local loop back) and loop 2 (remote loop back).

***IBMWRAP:** An IBM modem with wrap test capabilities is used on the communications line.

MODEMRATE

Specifies the speed at which the line operates if the modem has the data rate select feature.

Note: The user is responsible for ensuring that the line speed corresponds to the actual modem rate.

***FULL:** The line operates at the full rate of the modem.

***HALF:** The line operates at one-half the full rate, or at the alternate rate, of the modem.

SWTCNN

Specifies, for the switched line and the SNBU line, whether the line is used for incoming calls, outgoing calls, or both incoming and outgoing calls.

***BOTH:** The line is used for both incoming and outgoing calls.

***ANS:** The line is used for incoming calls only.

***DIAL:** The local system starts the call.

AUTOANS

Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically answers a call from a remote system to establish the connection or whether the user must manually answer the call and place the modem in data mode.

***YES:** The system automatically answers incoming calls.

***NO:** The system operator must manually answer incoming calls.

Note: *YES is valid only if the modem has the automatic answer feature.

AUTODIAL

Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically calls a remote system to establish a connection or whether the system operator must manually place the call.

***NO:** The AS/400 system does not automatically call a remote system.

***YES:** The AS/400 system automatically calls a remote system.

Note: *YES is valid only if the system is using an autocall unit or if the modem being used is capable of calling though a command interface.

DIALCMD

Specifies the type of dial command used to establish a switched connection with a remote system.

***NONE:** No dial command type is specified. An automatic call unit establishes the connection.

***V25BIS:** The use of one physical interface for call establishment and data transmission is allowed. It is sometimes referred to as a serial automatic call interface because the digits are presented serially on the link from the system (DTE) to the modem (DCE).

***OTHER:** The IBM command set is one example of another command type that is used by asynchronous protocols.

ACRSRCNAME

Specifies the resource name that describes the automatic call unit port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN02 and the port is 1, the resource name is LIN021.

CALLNBR

Specifies the local telephone number of the line used for the V.25 bis call request with identification (CRI) dial command. This parameter is used when the CRI function is needed for V.25 bis. When V.25 bis CRI dialing is used, the system takes the called (connection) number from the CNNNBR parameter of the controller description, adds a separator character (;), and concatenates the calling number at the end. Specify the calling number only if the modem and the network both support the CRI dial command.

***NONE:** The Call Request Normal (CRN) dial command is used by the V.25 bis line.

calling-number: Specify up to 32 characters that represent the local telephone number for V.25 bis CRI autodialing.

INACTTMR

Specifies the time (in tenths of a second) that the system waits for activity on a switched line before disconnecting.

300: The default value is 30 seconds.

***NOMAX:** There is no disconnect limit.

inactivity-timer: Specify a value ranging from 150 through 4200 in 0.1-second intervals.

MAXBUFFER

Specifies the maximum size (number of characters) of inbound and outbound data buffers.

896: The default value is 896 characters.

buffer-size: Specify a value ranging from 128 through 4096 characters.

THRESHOLD

Specifies the temporary error threshold level being monitored by the system. A permanent error is reported only if the errors occurred consecutively and exceeded the retry limit.

Note: Specifying the THRESHOLD parameter affects all threshold errors. They cannot be specified individually.

***OFF:** No monitoring of errors occurs.

***MIN:** The error threshold is set at a minimum monitoring level.

***MED:** Error thresholding is set to a medium monitoring level.

***MAX:** The error threshold is set at a maximum monitoring level.

FLOWCNTL

Specifies whether the hardware automatically generates or responds to XON/XOFF flow control characters.

***NO:** Prevents the hardware from generating or recognizing flow control characters.

***YES:** The system uses the flow control capabilities of the asynchronous protocol. If *YES is specified, the hardware recognizes flow control characters. This means that upon receipt of an XOFF character, the hardware stops transmission until an XON character is received. It also means that the hardware sends an XOFF character to the remote location when it is incapable of receiving characters. When the hardware is again able to receive characters, it sends an XON character to the remote system.

Note: If *YES is specified to flow control then *FULL must be specified for the duplex parameter.

XONCHAR

Specifies the hexadecimal value of the flow control character XON. If the system received an XOFF character while sending data, it automatically stops sending, and it starts sending data again only after receiving an XON character.

11: The default value is hexadecimal 11.

XON-character: Specify the XON-character, which can be any value ranging from hexadecimal 01 through FF; however, choose a character that does not appear in the normal data stream, such as hexadecimal 20 (ASCII blank).

XOFFCHAR

Specifies the hexadecimal value of the flow control character XOFF. If the system receives an XOFF character while sending data, it automatically stops sending, and starts sending data again only after receiving an XON character.

13: The default value is hexadecimal 13.

XOFF-character: The XOFF-character can be any value ranging from hexadecimal 01 through FF; however, choose a character that does not appear in the normal data stream, such as hexadecimal 20 (ASCII blank).

EORTBL

Specifies the EOR table which allows the system to recognize logical records when receiving data. Define a line feed (LF) as an EOR character in the data stream, and have the hardware return the data when the LF character is detected in the data stream.

The EOR table is specified as a set of pairs. The first element of a pair is the EOR character and the second element specifies the number of characters that follow the EOR character.

Element 1: EOR Character

00: The default is 00.

EOR-character: Valid EOR characters are in the range hexadecimal 01 through 7F (if 7 bits-per-character) or hexadecimal 01 through FF (if 8 bits-per-character). EOR characters are specified as they appear on the line after any translation by the asynchronous communications support.

Element 2: Trailing Characters

0: The default is 0.

trailing-character-count: Specify the number of additional characters received after the EOR character is detected. The number of trailing characters can range from 0 through 4.

IDLTMR

Specifies the time (in 0.1 second intervals) that the system waits between characters before the adapter forwards the receive buffer to the system.

1: The default value is 0.5 seconds.

idle-timer: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no timer.

DSRDRPTMR

Specifies the amount of time the system waits for the modem to exit the Data Set Ready (DSR) state before signaling an error.

6: The system waits 6 seconds in the DSR state before signaling an error.

DSR-drop-timer: Specify a value ranging from 3 through 60 seconds.

AUTOANSTYP

Specifies the method the system uses to answer incoming calls.

***DTR:** The system enters the Data Terminal Ready (DTR) state, signals the modem to answer calls, and waits for the modem to enter the DSR state.

***CDSTL:** The system enters the Connect Data Set to Line (CDSTL) state after monitoring the Ring Indicator to signal the modem to answer the call.

CTSTMR
Specifies the amount of time the system waits for the modem to enter or exit the Clear to Send (CTS) state before signaling an error.

25: The system waits up to 25 seconds for the CTS state to begin or end.

CTS-timer: Specify a value ranging from 10 through 60 seconds.

RMTANSTMR
Specifies the amount of time the system waits for the modem to enter the DSR state after dialing before signaling an error.

60: The system waits 60 seconds before signaling an error.

answer-timer: Specify a value ranging from 30 through 120 seconds.

CMNRCYLMT
Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Values:

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT
Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT
Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

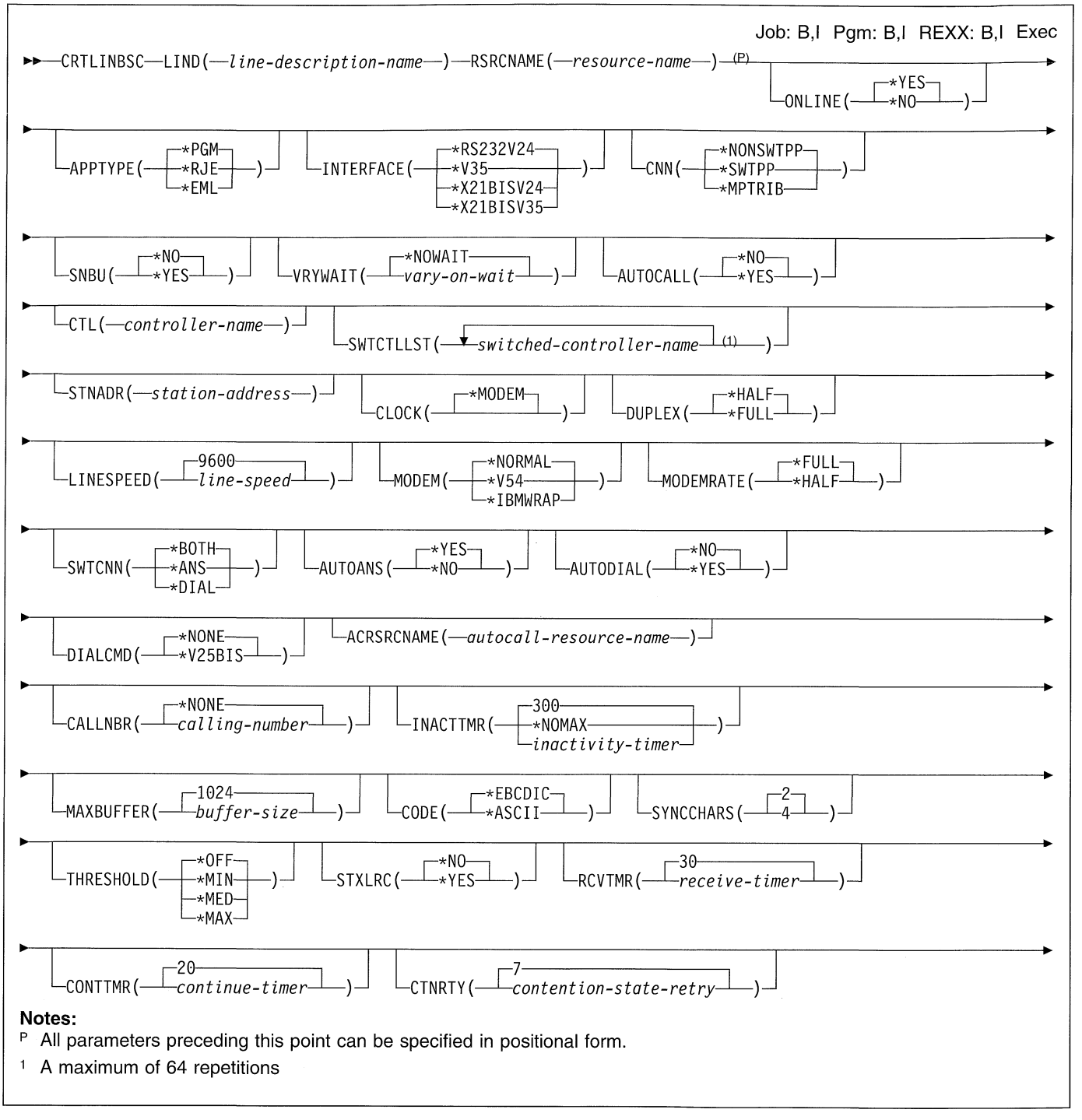
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINASC   LIND(ITF)  RSCRNAME(LIN031)
```

This command creates an asynchronous line description named ITF with a resource name of LIN031.

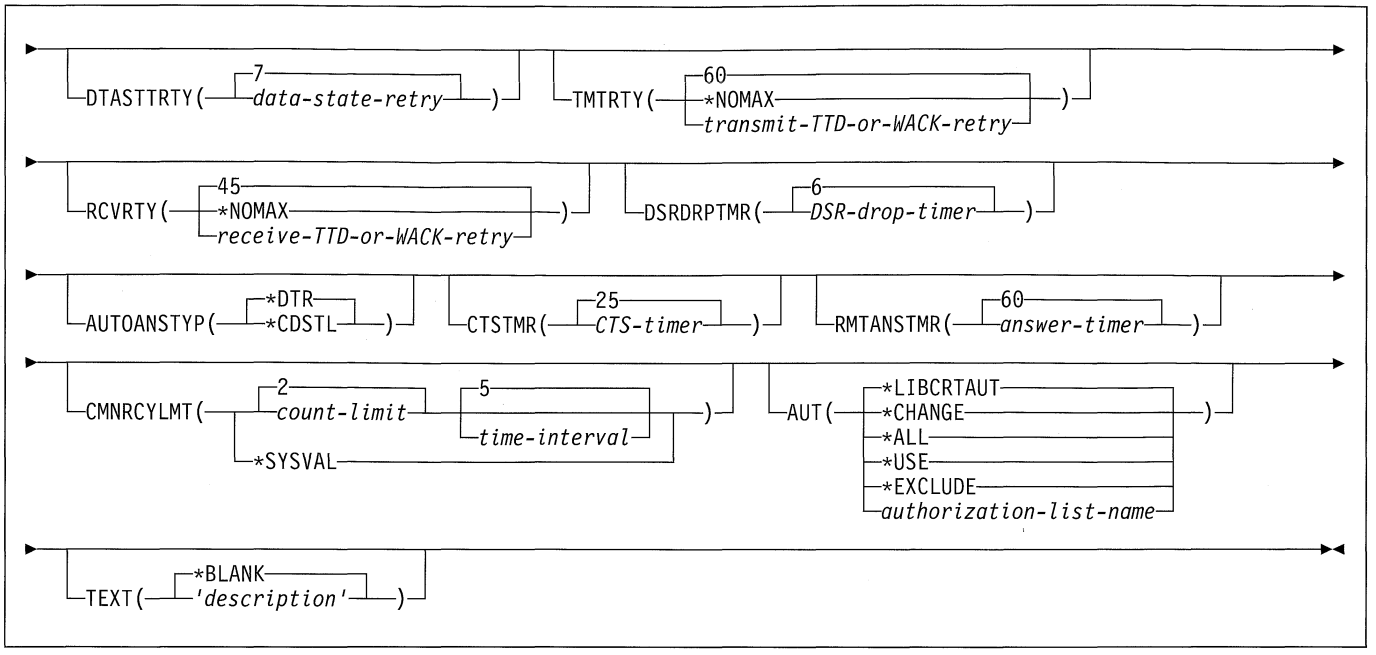
CRTLINBSC (Create Line Description (BSC)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions



Purpose

The Create Line Description (BSC) (CRTLINBSC) command creates a line description for a Binary Synchronous Communications (BSC) line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 2, then the resource name would be LIN012.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

APPTYPE

Specifies the application type being used.

***PGM:** This BSC line is used by a user-written program, not by the Remote Job Entry (RJE) function of the AS/400 Communications Utilities Program or 3270 device emulation.

***RJE:** This BSC line is used by the RJE function.

***EML:** This BSC line is used by the 3270 device emulation.

INTERFACE

Specifies the type of physical interface used on the input/ output adapter (IOA) port.

***RS232V24:** The RS-232/V.24 interface is used.

***V35:** The V.35 interface is used.

***X21BISV24:** The X.21 bis/V.24 interface is used.

***X21BISV35:** The X.21 bis/V.35 interface is used.

CNN

Specifies the type of line connection used.

***NONSWTPP:** A BSC nonswitched point-to-point line is used.

Note: This value cannot be selected if *EML is specified on the APPTYPE parameter.

***SWTPP:** A BSC switched point-to-point line is used.

Note: This value cannot be selected if *EML is specified on the APPTYPE parameter.

***MPTRIB:** A BSC multipoint tributary line is used..

Note: This value cannot be selected if *RJE is specified on the APPTYPE parameter. This value must be specified when APPTYPE(*EML) is specified.

SNBU

Specifies, for controllers attached to nonswitched lines only, whether the switched network backup (SNBU) feature is activated or deactivated. This feature lets the user bypass a broken nonswitched connection by establishing a switched connection. This parameter applies only if SWITCHED(*NO) and SNBU(*YES) are specified when the controller description is created.

Note: The Change Line Description (BSC) (CHGLINBSC) command and Change Controller Description (BSC) (CHGCTLBSC) command must be used to actually activate the SNBU feature. SNBU is valid only if both local and remote modems support this feature.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The modem has the SNBU feature.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

AUTOCALL

Specifies, for switched lines and switched network backup lines, whether the line has an associated autocall unit which performs automatic calling to the remote system.

***NO:** This switched line does not have an autocall unit.

***YES:** This switched line has an autocall unit.

CTL

Specifies the name of the controller description to which this object is attached.

Note: The controller description must already exist. Do not use this parameter when following the normal procedure of creating descriptions for lines first, controllers second, and devices last. Use this parameter only when the associated controller descriptions already exist before this line description is created.

SWTCTLLST

Specifies the names of up to 64 controllers that can establish a connection with the switched line. The controller descriptions must already exist.

Note: Do not use this parameter when following the normal procedure of creating descriptions for lines first, controllers second, and devices last. However, once the controller descriptions are created, specify the list by using the CHGLINBSC command.

STNADR

Specifies, for multipoint tributary lines, the EBCDIC hexadecimal address by which the local system is known to the remote system. The hexadecimal address is the polling address assigned to this AS/400 system.

CLOCK

Specifies that the clocking function for the line is provided by the modem (*MODEM), which is the default on this parameter.

DUPLEX

Specifies whether request-to-send (RTS) is permanently turned on (for full-duplex modems) or turned on only when data transmission is required (for half-duplex modems).

***HALF:** RTS is turned on only when transmission is required (for half-duplex modems).

***FULL:** Request-to-send (RTS) is permanently turned on (for full-duplex modems).

LINESPEED

Specifies the line speed in bits per second (bps).

9600: The default value is 9600 bps.

line-speed: Specify the line speed. Valid line speeds are: 600, 1200, 2400 4800, 7200, 9600, 14400, 19200, 48000, 56000, and 57600 bps.

MODEM

Specifies the type of modem supported on the communications line. Refer to the modem documentation to determine the appropriate value.

***NORMAL:** No attempt is made to run diagnostic tests on the modem.

***V54:** A certain type of diagnostic test (as defined by CCITT recommendations) is run to the modem. The AS/400 system supports CCITT V.54 loop 3 (local loop back) and loop 2 (remote loop back).

Note: Loop 2 diagnostic support is available on nonswitched lines only.

***IBMWRAP:** An IBM modem with wrap test capabilities is used on the communications line.

MODEMRATE
Specifies the speed (bits per second) at which the line operates if the modem has the data rate select feature.

***FULL:** The line operates at the full rate of the modem.

***HALF:** The line operates at one-half the full rate, or at the alternate rate, of the modem.

SWTCNN
Specifies, for the switched and switched network backup, whether the line is used for incoming calls, outgoing calls, or both.

***BOTH:** The line is used for both incoming and outgoing calls.

***ANS:** The line is used for incoming calls only.

***DIAL:** The local system starts the call.

AUTOANS
Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically answers a call from a remote system to establish the connection or whether the user must manually answer the call and place the modem in data mode.

***YES:** The system automatically answers incoming calls.

***NO:** The system operator must manually answer incoming calls.

Note: *YES is valid only if the modem has the automatic answer feature.

AUTODIAL
Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically calls a remote system to establish a connection or whether the system operator must manually place the call.

***NO:** The system operator must manually call a remote system.

***YES:** The AS/400 system automatically calls a remote system.

Note: *YES is valid only if the system is using an autocall unit and AUTOCALL was specified as *YES.

DIALCMD
Specifies the type of dial command used to establish a switched connection with a remote system.

***NONE:** No dial command type is specified. An automatic call unit establishes the connection.

***V25BIS:** The use of one physical interface for call establishment and data transmission is allowed. It is sometimes referred to as a serial automatic call interface because the digits are presented serially on the link from the system data terminal equipment (DTE) to the modem data circuit-terminating equipment (DCE).

*V25BIS is valid only for INTERFACE(RS232V24).

ACRSRCNAME
Specifies the resource name that describes the automatic call unit port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN02 and the port is 1, the resource name is LIN021.

CALLNBR
Specifies the local telephone number of the line used for the V.25 bis call request with identification (CRI) dial command. This parameter is used when the CRI function is needed for V.25 bis. When V.25 bis CRI dialing is used, the system takes the called (connection) number from the CNNNBR parameter of the controller description, adds a separator character (;), and concatenates the calling number at the end. Specify the calling number only if the modem and the network both support the CRI dial command.

***NONE:** Specifies that the CRN dial command is used by the V.25 bis line.

calling-number: Specify up to 32 characters that represent the local telephone number for V.25 bis CRI.

INACTTMR
Specifies the time (in tenths of a second) that the system waits for activity on a switched line before disconnecting.

300: The default value is 30 seconds.

***NOMAX:** There is no disconnect limit.

inactivity-timer: Specify a value ranging from 150 through 4200 in 0.1-second intervals.

MAXBUFFER
Specifies the maximum size (in bytes) of inbound and outbound data buffers.

1024: The default buffer size is 1024 bytes.

buffer-size: Specify a buffer size ranging from 8 through 8192 bytes.

CODE
Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

SYNCCHARS
Specifies the number of BSC SYN (synchronous) control characters to send when transmitting. The SYN control character is used to establish and maintain synchronization and as a time fill in the absence of any data or other control character.

2: The synchronization pattern consists of two consecutive SYN characters. The 9404 system unit always uses two characters.

4: The synchronization pattern consists of four consecutive SYN characters.

THRESHOLD

Specifies the temporary error threshold level being monitored by the system. A permanent error is reported only if the errors occurred consecutively and exceeded the retry limit.

Note: Specifying the THRESHOLD parameter affects all threshold errors. They cannot be specified individually.

***OFF:** No monitoring of errors occurs.

***MIN:** The error threshold is set at a minimum monitoring level.

***MED:** Error thresholding is set to a medium monitoring level.

***MAX:** The error threshold is set at a maximum monitoring level.

STXLRC

Specifies whether the start of text (STX) control character is included in the longitudinal redundancy check (LRC) calculation. This only applies to lines using the ASCII character code.

***NO:** The STX control character is not included in the LRC calculation.

***YES:** The STX control character is included in the LRC calculation. The 9404 system unit always includes STX.

RCVTMR

Specifies the number of seconds the system waits for data from the remote system before a receive timeout occurs.

30: The default value is three seconds.

receive-timer: Specify a value ranging from 30 through 254 in 0.1-second intervals.

CONTTMR

Specifies the number of seconds the system waits before sending a control character that prevents the line from being dropped when the system is not ready to transmit or receive data.

20: The default value is 20 seconds.

continue-timer: Specify a value ranging from 16 through 24 in 0.1-second intervals.

CTNRTY

Specifies the number of contention state retries to attempt before indicating an error and making the line inoperative.

For BSC, contention is the state that exists after the end of transmission (EOT) character is received or sent, and before a starting sequence (ENQ) has been positively acknowledged (ACK0).

7: Seven retries are attempted.

contention-state-retry: Specify a value ranging from 0 through 21 for the number of contention state retries.

DTASTTRTY

Specifies the number of data state retries to attempt before indicating the error and making the line inoperative.

For BSC, the data state is a time during which BSC is sending or receiving text on the communications line.

7: Seven retries are attempted.

data-state-retry: Specify a value ranging from 0 through 255 for the number of data state retries.

TMTRTY

Specifies the number of retries for transmitting temporary-text-delay (TTD) or wait-before-transmitting positive acknowledgement character (WACK) to keep the line operative before indicating the error and ending the session. If the application type is RJE, this parameter cannot be specified.

60: The default value is 60.

***NOMAX:** There is no disconnect limit.

transmit-TTD-or-WACK-retry: Specify a value ranging from 0 through 65534 for the number of retries.

RCVRTY

Specifies the number of retries for receiving TTD or WACK before indicating the error and failing the session. This parameter can only be specified if the application type is *PGM.

45: The default value is 45 retries.

***NOMAX:** There is no disconnect limit.

receive-TTD-or-WACK-retry: Specify a value ranging from 0 through 65534 for the number or retries.

DSRDRPTMR

Specifies the amount of time the system waits for the modem to exit the Data Set Ready (DSR) state before signaling an error.

6: The system waits 6 seconds in the DSR state before signaling an error.

DSR-drop-timer: Specify a value ranging from 3 through 60 seconds.

AUTOANSTYP

Specifies the method the system uses to answer incoming calls.

***DTR:** The system enters the Data Terminal Ready state, signals the modem to answer calls, and waits for the modem to enter the Data Set Ready (DSR) state.

***CDSTL:** The system enters the Connect Data Set to Line (CDSTL) state after monitoring the Ring Indicator to signal the modem to answer the call.

CTSTMR

Specifies the amount of time the system waits for the modem to enter or exit the Clear to Send (CTS) state before signaling an error.

25: The system waits up to 25 seconds for the CTS state to begin or end.

CTS-timer: Specify a value ranging from 10 through 60 seconds.

RMTANSTMR

Specifies the amount of time the system waits for the modem to enter the Data Set Ready (DSR) state after dialing before signaling an error.

60: The system waits 60 seconds before signaling an error.

answer-timer: Specify a value ranging from 30 through 120 seconds.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of second-level recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value:

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

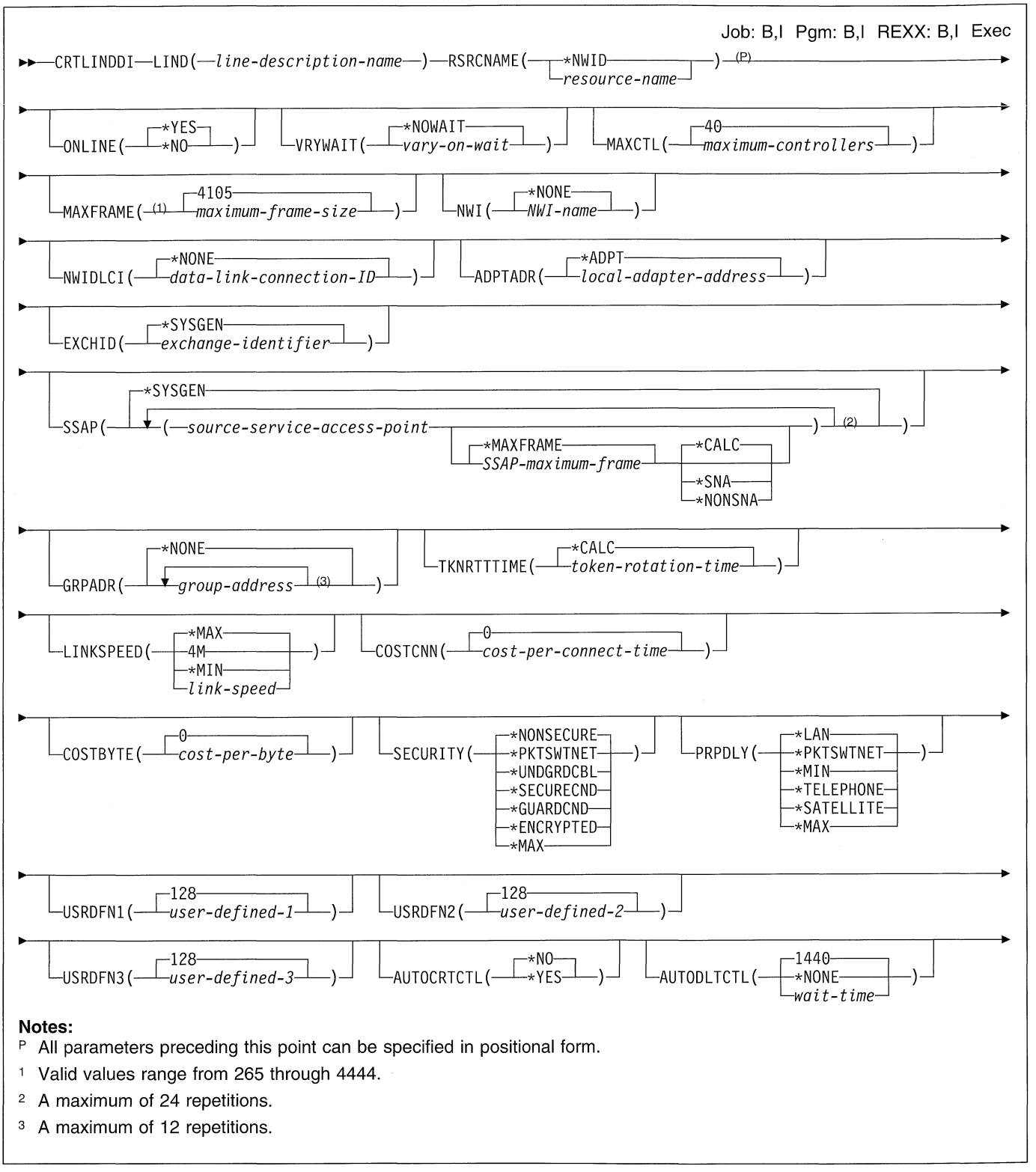
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINBSC   LIND(BRANCHES)  RSRCNAME(LIN021)
  ONLINE(*NO)  CNN(*SWTPP)  AUTOCALL(*YES)
  ACRSRCNAME(LIN032)  SWTCTLLST(BRANCH1 BRANCH2)
```

This command creates a BSC line description for the first port on the second IOA. It is set up to autodial on the second port of the third IOA, or to automatic answer. The controller descriptions in the SWTCTLLST already exist.

CRTLINDDI (Create Line Description (DDI Network)) Command



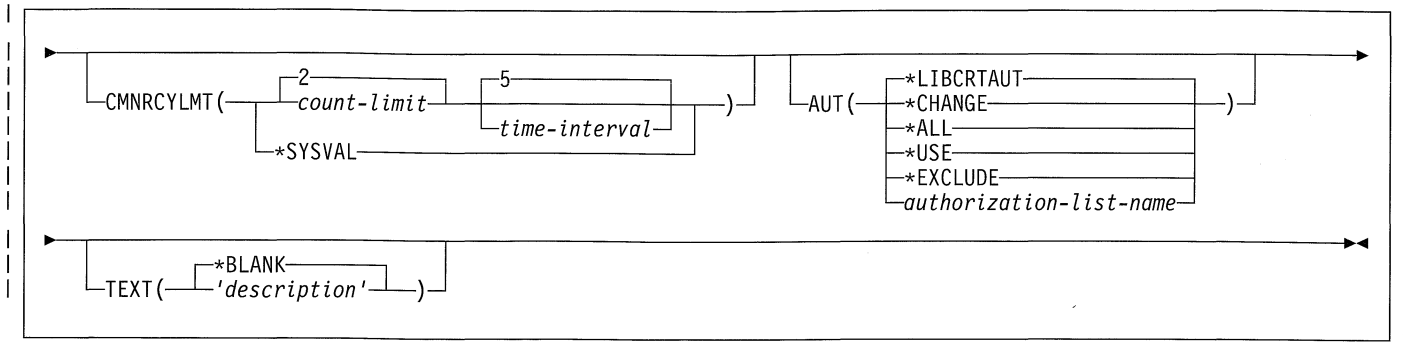
Notes:

P All parameters preceding this point can be specified in positional form.

1 Valid values range from 265 through 4444.

2 A maximum of 24 repetitions.

3 A maximum of 12 repetitions.



Purpose

The Create Line Description (Distributed Data Interface) (CRTLINDDI) command creates a line description for a data-description interface line such as an FDDI (Fiber Distributed Data Interface) local area network. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

RSRCNAME

Specifies the resource name that identifies the hardware that the description represents.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified on the TYPE parameter to help determine the resource name. Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 1, then the resource name is LIN011.

***NWID:** The resource name specified on the attached frame relay network interface description is used.

resource-name: Specify a resource name.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

MAXCTL

Specifies the maximum number of controllers that the line supports.

40: The line supports 40 controllers.

maximum-controllers: Specify a number large enough to account for all controllers currently active to this network, and the controllers that will be attached in the near future. Valid values range from 1 through 256.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

4105: The maximum frame size is 4105 bytes.

maximum-frame-size: Specify the maximum frame size. Valid values range from 265 through 4444 bytes.

NWI

Specifies an attached nonswitched frame relay NWI.

Note: NWI(*NONE) must be specified when

RSRCNAME(*NWID) is not specified. Otherwise, NWI(*NONE) can be specified only when NWIDLCI(*NONE) is also specified.

***NONE:** No network interface is specified.

NWI-name: Specify an attached nonswitched frame relay NWI.

NWIDLCI

Specifies the data link control identifier (DLCI) for the network interface.

Note: NWIDLCI(*NONE) must be specified when RSRCNAME(*NWID) is not specified. Otherwise, NWIDLCI(*NONE) can be specified only when NWI(*NONE) is also specified.

***NONE:** A DLCI is not specified for the network interface.

data-link-connection-ID: Specify the DLCI for the network interface to which this line permanently attaches. Valid values range from 1 through 1018.

ADPTADR

Specifies the 12-character hexadecimal adapter address.

Note: ADPTADR(*ADPT) must not be specified when RSRCNAME(*NWID) is not specified.

***ADPT:** This value gives the user the preset DDI default address for this DDI adapter card. The user may display this by doing a DSPLIND on this line description after it has successfully varied on.

local-adapter-address: Specify an address for this system in the DDI network. Valid values range from hexadecimal 400000000000 through 7FFFFFFFFFFF.

EXCHID

Specifies the hexadecimal exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains three digits for the block number and five digits for the identifier of this system.

***SYSGEN:** The AS/400 system generates the exchange identifier.

exchange-identifier: Specify an exchange identifier composed of eight hexadecimal digits starting with 056.

SSAP

Specifies source service access points (SSAPs). This is the hexadecimal logical address used to route incoming data from the bus to the proper user. A maximum frame size can be specified for each SSAP. Valid SSAP values are 06 and AA (for TCP/IP), and 04 through 9C divisible by 4 (for SNA).

The destination service access point (DSAP), specified by the remote controller, must match one of the SSAPs specified in order for communication to occur. All SSAP values must be unique.

***SYSGEN:** The system automatically creates three SSAPs, hex 04 for SNA, and hex 06 and AA for TCP/IP applications.

Element 1: SSAPs

source-service-access-point: Specify up to 24 SSAPs using valid SSAP values.

Element 2: Frame Size for SSAPs

***MAXFRAME:** The frame size specified on the MAXFRAME parameter is used.

SSAP-maximum-frame: Specify the maximum SSAP frame size (the maximum size of the data field that may be transmitted or received). Valid values for this parameter range from 265 through 4444 bytes, but must not exceed the value of the MAXFRAME parameter.

Element 3: SSAP Type

***CALC:** The system determines the SSAP type based on the following values:

- 04 through 9C, divisible by 4 (for SNA)
- 02 through FE, divisible by 2 (for non-SNA)

***SNA:** The SSAP is used for SNA communications. Valid values range from 04 through 9C and must be divisible by 4.

***NONSNA:** The SSAP is used for non-SNA communications. Valid values range from 02 through FE and must be divisible by 2.

GRPADR

Specifies the distributed data interface group addresses used. Group addresses must each be specified as a 12-digit hexadecimal number. Valid values range from 800000000000 through FFFFFFFFFFFE.

***NONE:** No group addresses are specified.

group-address: Specify the group addresses to be used.

TKNRTTTIME

Specifies the token rotation time requested. This value is used when the station bids on the network. The lowest value of all attached stations on a ring determines the value the ring uses.

Note: TKNRTTTIME(*CALC) must be specified when RSRCNAME(*NWID) is specified.

***CALC:** The system calculates the value based on the type of line that is linked to the controller.

token-rotation-time: Specify a value ranging from 4 through 167 milliseconds.

LINKSPEED

Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

***MAX:** A link speed greater than 16M bps is used.

4M: The link speed is 4M bps.

***MIN:** A link speed of less than 1200 bps is used.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 4M, 10M, and 16M.

COSTCNN
Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

0: The cost per connect time is 0.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE
Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

0: The cost per byte is 0.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY
Specifies the security level of the physical line.

***NONSECURE:** Normal priority is used.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY
Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system. The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***LAN:** The local area network propagation delay is used.

***PKTSWTNET:** The packet switched network propagation delay is used.

***MIN:** The minimum propagation delay is used.

***TELEPHONE:** The telephone propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1
Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2
Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3
Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

AUTOCRTCTL
Specifies whether the system automatically creates controller descriptions when calls are received from adjacent systems on the local area network (LAN).

***NO:** The system does not automatically create a controller description when incoming calls are received.

***YES:** The system automatically creates a controller description when incoming calls are received.

AUTODLTCTL
Specifies the number of minutes an automatically created controller can remain in an idle state (switched from varied on to varied on pending) before the controller description and attached device descriptions are varied off and deleted.

1440: The controller description can be idle for 1440 minutes (24 hours).

***NONE:** The system does not automatically delete or vary off the automatically configured, idle controller descriptions.

wait-time: Specify the number of minutes to wait before deleting the automatically configured, idle controller descriptions for this line. Valid values range from 1 to 10,000 minutes.

CMNRCYLMT
Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must

elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority the user is granting to a user who does not have specific authority to an object, who is not on the authorization list, or whose user group has no specific authority to the object.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can control the object's existence, specify the security for the object, change the object, change the owner for the object, and perform basic functions on the object. All authority allows the user to perform all operations on the object except those limited to the owner, or controlled by authorization list management authority.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

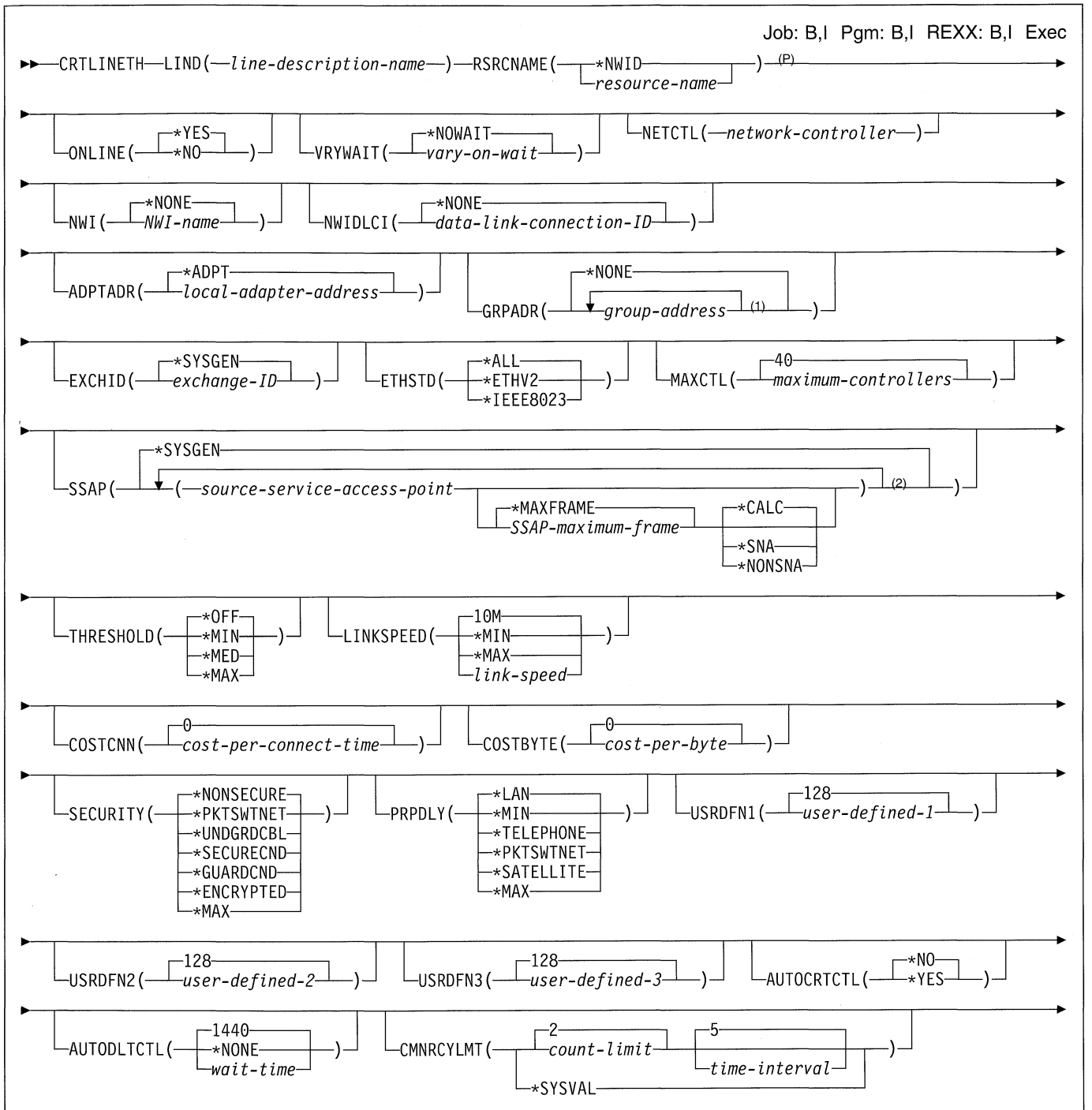
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINDDI   LIND(DDILAN1)  RSRCNAME(LIN011)
  TEXT('Fiber Distributed Data Interface (FDDI) Line')
```

This command creates a DDI line description named DDILAN1 for an FDDI line installed on adapter LIN011 on the system.

CRTLINETH (Create Line Description (Ethernet)) Command

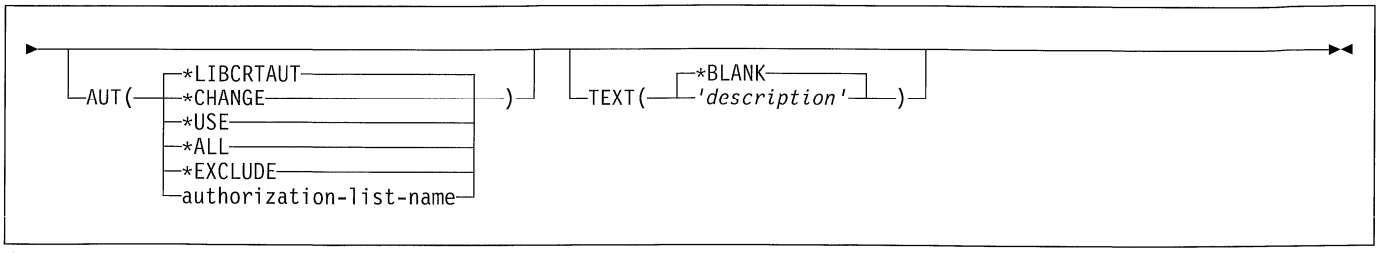


Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 12 repetitions

2 A maximum of 24 repetitions



Purpose

The Create Line Description (Ethernet) (CRTLINETH) command creates a line description for an Ethernet Local Area Network (LAN) line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name.

Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 1, then the resource name would be LIN011.

***NWID:** The resource name is determined by the network interface used.

resource-name: Specify a resource name.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

NETCTL

Specifies the name of an existing network controller. This parameter can be specified for lines attached to a TCP/IP network.

NWI

Specifies for a nonswitched connection, the frame relay network interface description to which this line permanently attaches.

Note: NWI(*NONE) must be specified when RSRCNAME(*NWID) is not specified. Otherwise, NWI(*NONE) can be specified only when NWIDLCI(*NONE) is also specified.

***NONE:** A network interface description is not specified.

NWI-name: Specify the name of the frame relay network interface to which this line permanently attaches.

NWIDLCI

Specifies the data link control identifier (DLCI) for the network interface.

Note: NWIDLCI(*NONE) must be specified when RSRCNAME(*NWID) is not specified. Otherwise, NWIDLCI(*NONE) can be specified only when NWI(*NONE) is also specified.

***NONE:** A DLCI is not specified for the network interface.

data-link-connection-ID: Specify the DLCI for the network interface to which this line permanently attaches. Valid values range from 1 through 1018.

ADPTADR

Specifies the 12-character hexadecimal adapter address.

***ADPT:** The preset Ethernet adapter address is used as the local adapter address. The adapter address can be displayed (DSPLIND) after the line description has been successfully varied on. This value is not valid when RSRCNAME(*NWI) is specified.

local-adapter-address: Specify the local Ethernet adapter address to override the preset local address. The local adapter address must be an individual address (it cannot be a group address). Valid values range from 020000000000 through FEFFFFFFFFFF in hexadecimal format. The second digit (from the left) of the address must be 2, 6, A, or E.

GRPADR

Specifies whether an Ethernet group address is used.

***NONE:** A group address is not used.

group-address: Specify the address to which a subset of units on the Ethernet line responds in addition to their local adapter address. Valid values range from 010000000000 through FDFFFFFFFFFF in hexadecimal format. The second digit (from the left) must be odd. All group addresses must be unique.

EXCHID

Specifies the hexadecimal exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains three digits for the block number and five digits for the identifier of this system.

***SYSGEN:** This value allows the AS/400 system to create the exchange identifier. Use the Display Line Description (DSPLIND) command to see the resulting exchange identifier.

exchange-ID: Specify an 8-character (four hexadecimal bytes) exchange identifier ranging from 05600000 through 056FFFFF.

ETHSTD

Specifies the Ethernet standard frame type that is used on this line.

***ALL:** All Ethernet standards can be used. However, SNA data will be placed in IEEE 802.3 frames.

***ETHV2:** Ethernet Version 2 frames are used for all data.

***IEEE8023:** IEEE 802.3 frames are used for all data.

MAXCTL

Specifies the maximum number of system network architecture (SNA) controllers that the line supports.

40: The default number of controllers is 40.

maximum-controllers: Specify the maximum number of controllers supported by the line. This should be a number large enough to account for all of the controllers that are currently attached to this line, and for those controllers to be attached in the near future. The range is from 1 through 256.

SSAP

Specifies source service access points (SSAPs). This is the hexadecimal logical address used to route incoming data from the Ethernet bus to the proper user. A maximum frame size can be specified for each SSAP. Valid SSAP values are 06 and AA (for TCP/IP), and 04 through 9C divisible by 4 (for SNA).

Note: Ethernet Version 2 (specified as *ETHV2 on the ETHSTD parameter) does not allow the SSAP values of 06 and AA.

The destination service access point (DSAP), specified by the remote controller, must match one of the SSAPs specified in order for communication to occur. All SSAP values must be unique.

***SYSGEN:** For ETHSTD(*ALL or *IEEE8023), the AS/400 system creates three SSAPs: SSAP '04'X for SNA; 'AA'X and '06'X for TCP/IP applications. For ETHSTD(*ETHV2), the system creates SSAP '04'X for SNA.

Element 1: SSAPs

source-service-access-point: Specify up to 24 SSAPs, including hex AA and 06 for TCP/IP, and any hexadecimal number 04 through 9C that is divisible by four for SNA applications.

Element 2: Frame Size for SSAPs

***MAXFRAME:** The system determines the maximum frame size (data field size) that can be transmitted or received. If ETHSTD(*ALL or *IEEE8023) was specified, *CALC produces a frame size of 1496 for TCP/IP and SNA SSAPs. If ETHSTD(*ETHV2) was specified, *CALC produces a frame size of 1493 for SNA SSAPs.

SSAP-maximum-frame: Specify the maximum frame size for each SSAP. Valid values for the maximum frame size range from 265 through 1496.

Note: When RSRCNAME(*NWID) and ETHSTD(*ETHV2) are specified, the valid values for this parameter range from 265 through 1486 bytes. When RSRCNAME(*NWID) and ETHSTD(*ALL) or ETHSTD(*IEEE8023) are specified, the valid values for this parameter range from 265 through 1489 bytes.

Element 3: SSAP Type

***CALC:** The system determines the value to use.

***SNA:** The SSAP is used for SNA communications. Valid values range from 04 through 9C and must be divisible by 4.

***NONSNA:** The SSAP is used for non-SNA communications. Valid values range from 02 through FE and must be divisible by 2.

THRESHOLD

This parameter, and its values *OFF, *MIN, *MED, and *MAX, can be specified but it is not used by the system starting in release V2R3M0. The parameter may be removed in a later release.

LINKSPEED

Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

10M: The link speed is 10M bps.

***MIN:** A link speed of less than 1200 bps is used.

***MAX:** A link speed greater than 16M bps is used.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 4M, 10M, and 16M.

COSTCNN

Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

0: The cost per connect time is 0.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE

Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

0: No rotation occurs. Printing starts at and is parallel to the edge loaded into the printer.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY

Specifies the security level of the line.

***NONSECURE:** Normal priority is used.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY

Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system.

The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***LAN:** The local area network propagation delay is used.

***MIN:** The minimum propagation delay is used.

***TELEPHONE:** The telephone propagation delay is used.

***PKTSWTNET:** The packet switched network propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1

Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2

Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3

Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

AUTOCRTCTL

Specifies whether the system will automatically create controller descriptions when calls are received from adjacent systems on the local area network (LAN).

***NO:** The system will not automatically create a controller description when incoming calls are received.

***YES:** The system will automatically create a controller description when incoming calls are received.

AUTODLTCTL

Specifies the number of minutes an automatically created controller can remain in an idle state (switched from varied on to varied on pending) before the controller description and attached device descriptions are varied off and deleted.

1440: The controller description can be idle for 1440 minutes (24 hours).

***NONE:** The system will not automatically delete or vary off the automatically configured, idle controller descriptions.

wait-time: Specify the number of minutes to wait before deleting the automatically configured, idle controller descriptions for this line. Valid values range from 1 to 10,000 minutes.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

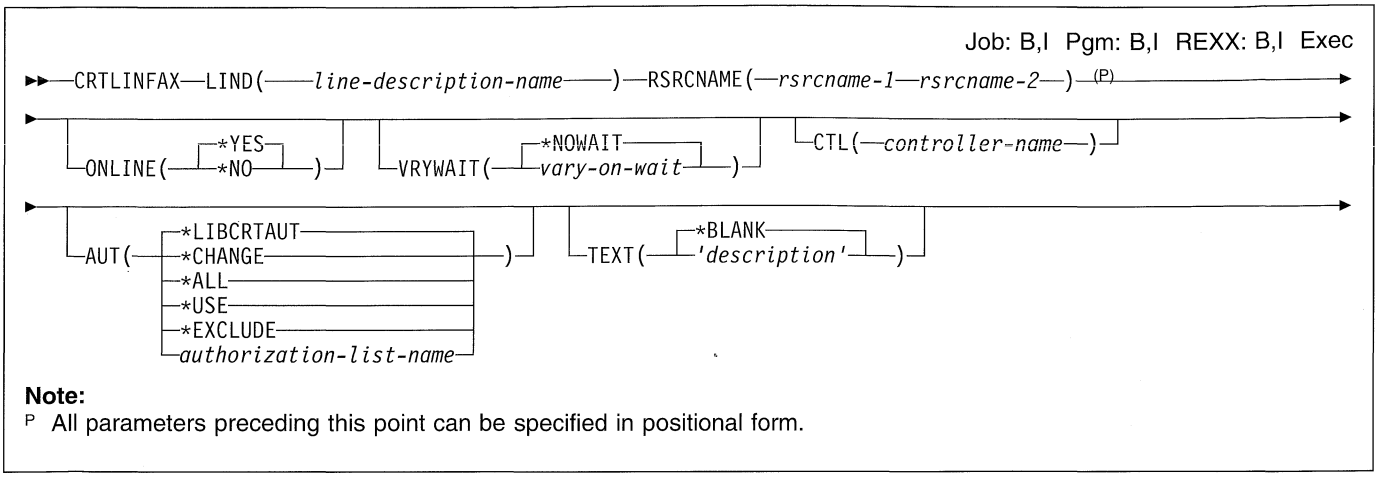
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINETH   LIND(BOSTON)  RSRCNAME(LIN041)
```

This command creates an Ethernet line description named BOSTON with a resource name of LIN041.

CRTLINFAX (Create Line Description (Fax)) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Line Description (Fax) (CRTLINFAX) command creates a line description for a facsimile (fax) line.

Required Parameters

LIND
Specifies the name of the line description being created.

RSRCNAME
Specifies the resource names that describe the fax ports.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name.

The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01, the resource names for ports 1 and 2 are is LIN011 and LIN012.

The resource name for both ports of the fax IOA must be specified. All lines specified must be attached to the same input/output processor.

rsrcname-1: Specify the first resource name to be used to describe the fax ports.

rsrcname-2: Specify the second resource name to be used to describe the fax ports.

Optional Parameters

ONLINE
Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

VRYWAIT
Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

CTL
Specifies the name of the controller description to which this object is attached.

Note: This parameter is valid only when the associated controller description has been created before this line description.

AUT
Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

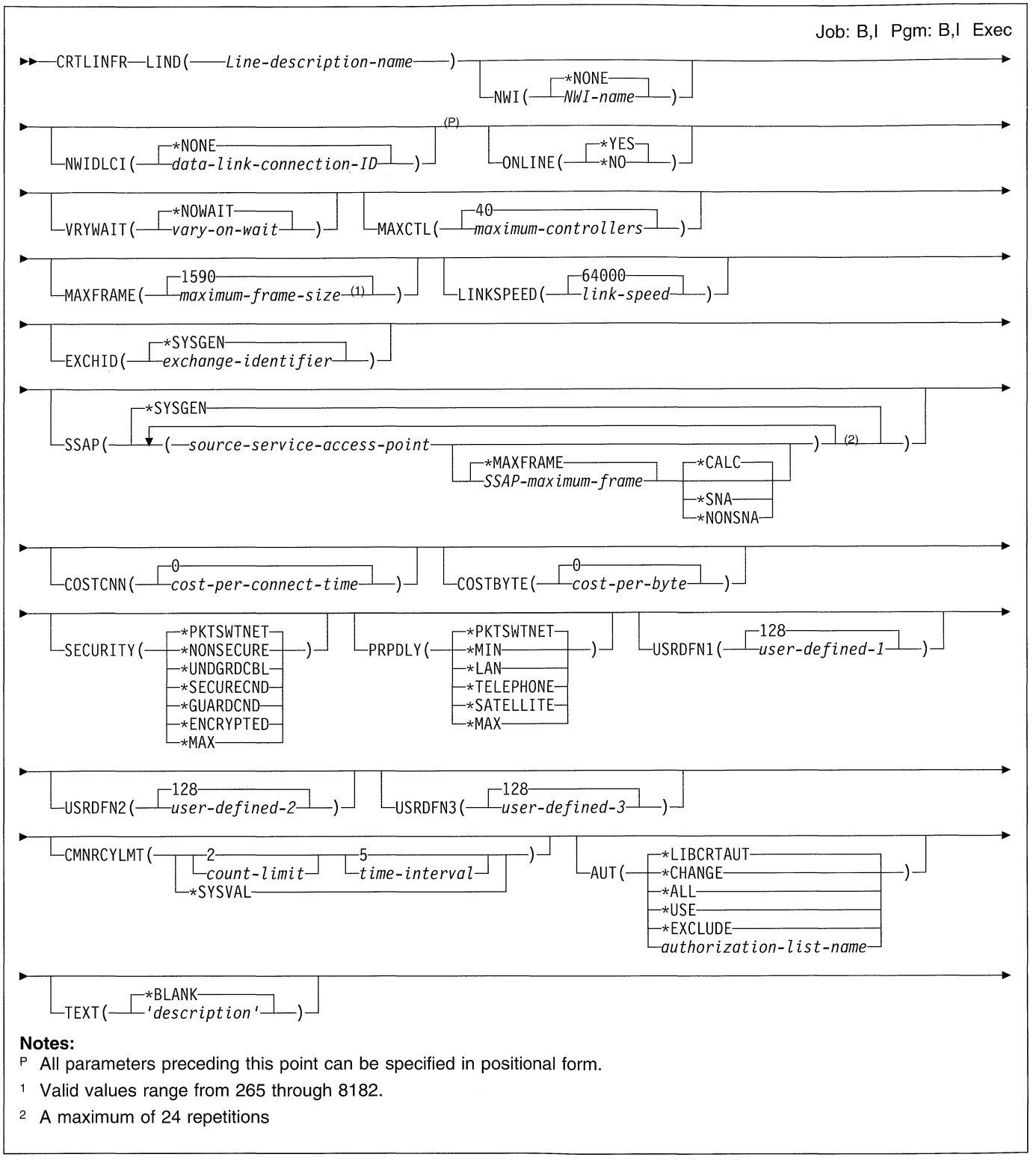
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINFAX   LIND(FAXLINE)  RSRCNAME(LIN041 LIN042)
```

This command creates fax line description named FAXLINE with resource names of LIN041 and LIN042.

CRTLINFR (Create Line Description (Frame Relay Network)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 Valid values range from 265 through 8182.

2 A maximum of 24 repetitions

Purpose

The Create Line Description (Frame-Relay Network) (CRTLINFR) command creates a line description for a frame-relay network (FR) line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameter

LIND

Specifies the name of the line description being created.

line-description-name: Specify the name of a line description.

Optional Parameters

NWI

Specifies, for a nonswitched connection, the frame relay network interface description containing the DLCI to which this line permanently attaches. If a DLCI is not specified for the network interface, a description cannot be specified. If a DLCI is specified for the network interface, a description must be specified.

***NONE:** A network interface description is not specified.

NWI-name: Specify the name of the network interface to which this line permanently attaches.

NWIDLCI

Specifies the data link control identifier (DLCI) number for the network interface, which is supplied by the network provider. Valid DLCI values range from 1 through 1018.

***NONE:** A DLCI is not specified for the network interface.

data-link-connection-ID: Specify the DLCI for the network interface to which this line permanently attaches.

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

MAXCTL

Specifies the maximum number of controllers supported by a line.

40: The line supports 40 controllers.

maximum-controllers: Specify a number large enough to account for all controllers currently active to this network, and the controllers that will be attached in the near future. Valid values range from 1 through 256.

MAXFRAME

Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system. The recommended MAXFRAME values are: 502, 1014, 1590, 2038, 4086, and 8182 bytes.

1590: The maximum frame size is 1590 bytes.

maximum-frame-size: Specify the maximum frame size value to be used. Valid values range from 265 through 8182.

LINKSPEED

Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

64000: The link speed is 64000 bps.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000.

EXCHID

Specifies the hexadecimal exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains three digits for the block number and five digits for the identifier of this system.

***SYSGEN:** The AS/400 system generates the exchange identifier.

exchange-identifier: Specify (if the *SYSGEN value is not specified) an exchange identifier composed of eight hexadecimal digits starting with 056.

SSAP
Specifies the source service access point (SSAP). The most commonly used SNA SSAP is hex 04. All SSAP values must be unique.

***SYSGEN:** The system automatically creates three SSAPs, hex 04 for SNA, and hex AA and 06 for TCP/IP applications.

Element 1: SSAPs

source-service-access-point: Specify up to 24 SSAPs, including hex AA and 06 for TCP/IP, and any hexadecimal number ranging from 04 through 9C that is divisible by four for SNA applications.

Element 2: Frame Size for SSAPs

***MAXFRAME:** The frame size specified on the MAXFRAME parameter is used.

SSAP-maximum-frame: Specify the maximum SSAP frame size (the maximum size of the data field that may be transmitted or received). Valid values for this parameter range from 265 through 8182 bytes. The recommended values are: 502, 1014, 2038, 4086, and 8182 bytes.

Element 3: SSAP Type

***CALC:** The system determines the SSAP type based on the following values:

- 04 through 9C, divisible by 4 (for SNA)
- 02 through FE, divisible by 2 (for non-SNA)

***SNA:** The SSAP is used for SNA communications. Valid values range from 04 through 9C and must be divisible by 4.

***NONSNA:** The SSAP is used for non-SNA communications. Valid values range from 02 through FE and must be divisible by 2.

COSTCNN
Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

0: The cost per connect time is 0.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE
Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

0: The cost per byte is 0.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY
Specifies the security level of the physical line. This parameter is valid only if APPN is used on the system.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***NONSECURE:** Normal priority is used.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY
Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system. The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***PKTSWTNET:** The packet switched network propagation delay is used.

***LAN:** The local area network propagation delay is used.

***MIN:** The minimum propagation delay is used.

***TELEPHONE:** The telephone propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1
Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2
Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3
Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line

that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

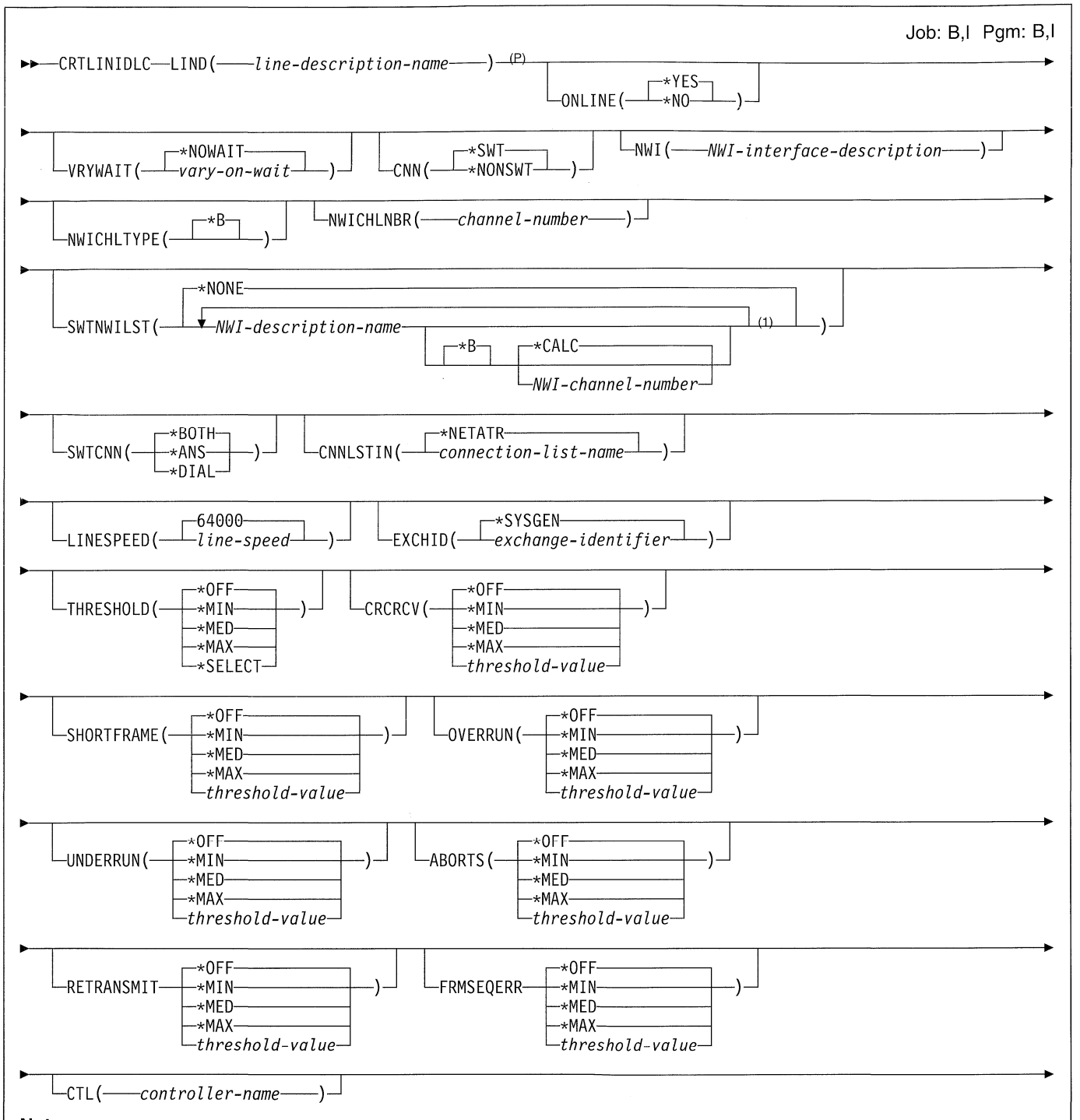
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINFR   LIND(FRLIN)  NWI(NEWONE)  NWIDLCI(1001)
  ONLINE(*YES)  VRYWAIT(*NOWAIT)  MAXFRAME(1600)
  LINKSPEED(2400)
```

This command creates frame relay line FRLIN. FRLIN is attached to a frame relay NWI named NEWONE using DLCI number 1001. FRLIN is automatically varied on at initial program load (IPL). The system does not wait for the vary on to complete; therefore, the line is varied on asynchronously. The maximum frame size for this line is 1600 and the link speed is 2400 bits per second (bps).

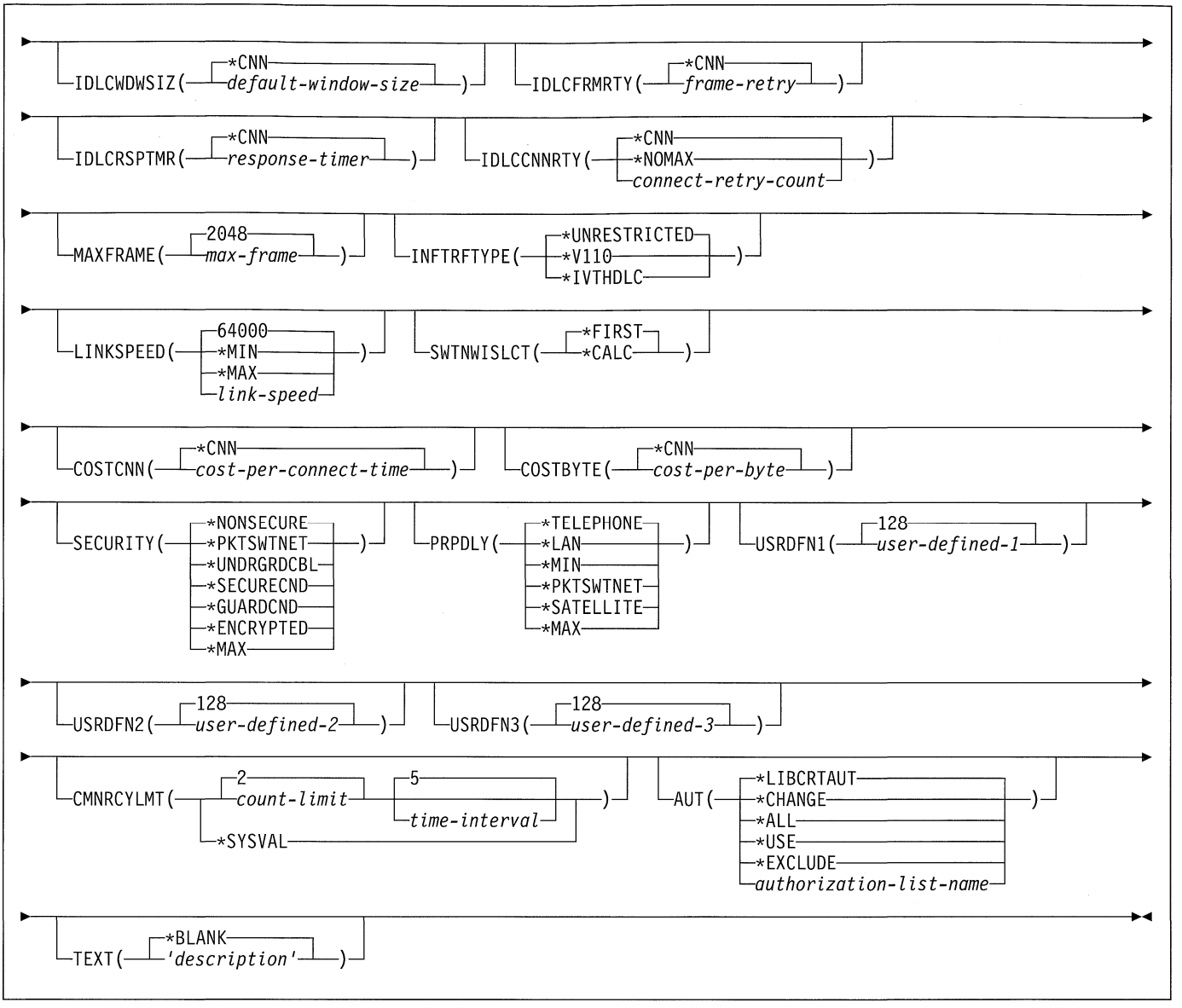
CRTLINIDLC (Create Line Description (IDLC)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 64 repetitions



Purpose

The Create Line Description (IDLC) (CRTLINIDLC) command creates a line description for an integrated services digital network (ISDN) data link control (IDLC) line.

Required Parameter

LIND

Specifies the name of the line description.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

CNN

Specifies whether this line description is switched or nonswitched. If *SWT is specified, the line description is attached to one of the Network Interface descriptions specified on the SWTNWILST parameter at the time an incoming or outgoing call is processed. If *NONSWT is specified, the line description is permanently attached to the Network Interface description specified on the NWI parameter. Once a line is permanently attached, the attachment cannot be changed with a change command; it must be changed by deleting and then recreating the line description or the network interface description.

***SWT:** The line is not permanently attached to a network interface description. The network interface description is determined by the value specified on the SWTNWILST parameter.

***NONSWT:** The line is permanently attached to the network interface description specified on the NWI parameter.

NWI

Specifies, for a nonswitched connection, the ISDN network interface description containing the channel to which this line permanently attaches.

NWICHLTYPE

Specifies, for nonswitched connections, the type of integrated services digital network (ISDN) channels that this line description uses. This parameter is preset to use one ISDN B-channel of the Network Interface description.

***B:** The B channel is used.

NWICHLNBR

Specifies, for a nonswitched connection, the channel number of the network interface description that is used by this line description. Two channels are available for each network interface description, but only one line description can be permanently attached to a channel. The Display Network Interface Description (DSPNWID) command is used to display information about the channel numbers.

SWTNWILST

Specifies, for ISDN switched connections, a list of network interface descriptions to which this line can be attached. The first available line description is chosen from the list at the time an incoming or outgoing call is processed.

***NONE:** No network interface description is specified.

Element 1: Network Interface Description Name

NWI-description-name: Specify, for switched connections, the name of the network interface description to which this line attaches.

Element 2: Network Interface Channel Type

***B:** The B channel is used.

Element 3: Network Interface Channel-Number

***CALC:** The system selects one of the two channel numbers (based on availability) defined for the network interface description when an incoming or outgoing call is processed.

NWI-channel-number: Specify a channel number (1 or 2). Specifying a channel number restricts the line description to that channel.

SWTCNN

Specifies whether the switched line is used for incoming calls, outgoing calls, or both.

***BOTH:** The line is used for both incoming and outgoing calls.

***ANS:** The line is used for incoming calls only.

***DIAL:** The local system starts the call.

CNNLSTIN

Specifies the name of the connection list that is used to retrieve call information (or connection) for identifying authorized incoming calls.

***NETATR:** The connection list used by this line description is taken from the list of system default network attributes that were identified at IPL (Initial Program Load). The Display Network Attributes (DSPNETA) command can be used to see the name of the connection list that is used.

connection-list-name: Specify the name of the connection list used for this line description.

LINESPEED

Specifies the line speed in bits per second (bps).

64000: The line speed is 64,000 bps.

56000: The line speed is 56000 bps.

EXCHID

Specifies the hexadecimal character exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains 3 digits for the block number and 5 digits for the identifier of this system.

***SYSGEN:** The system generates the exchange identifier.

exchange-identifier: Specify an exchange identifier of 8 hexadecimal digits ranging from 05600000 through 056FFFFF.

THRESHOLD

Specifies the threshold for the number of errors beyond which a message is sent informing the user that errors have occurred. All error threshold parameters are set to the value specified, unless *SELECT is specified, in which case each error threshold parameter is set individually. The next five parameters are the threshold parameters. They are: CRCRCV, OVERRUN, UNDERRUN, ABORTS, and SHORTFRAME.

***OFF:** No monitoring of errors occurs.

***MIN:** The error threshold is set at a minimum monitoring level.

***MED:** The system performs a medium amount of error threshold monitoring for all types of errors.

***MAX:** The error threshold is set at a maximum monitoring level.

***SELECT:** The threshold for the number of errors beyond which a message is sent informing the user that errors have occurred is set individually for each parameter.

CRCRCV

Specifies the level of error threshold monitoring done by the system for Cyclic Redundancy Check (CRC) errors.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs a minimum amount of error threshold monitoring for CRC errors: 6 errors in the first 30 seconds or 180 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for CRC errors: 2 errors in the first 30 seconds or 60 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for CRC errors received.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

SHORTFRAME

Specifies the threshold for the level of errors for short frame errors received.

***OFF:** No monitoring of errors occurs.

***MIN:** A minimum of monitoring is done: 6 X.25 or SDLC frames received in the first 30 seconds or 1 received every second for 14 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** A medium amount of monitoring is done: 3 X.25 or SDLC frames received in the first 30 seconds or 1 received every 3-4 seconds for 10-14 minutes.

***MAX:** A maximum amount of monitoring is done.

threshold-value: Specify a threshold level. Valid values range from 1 through 10000.

OVERRUN

Specifies the level of error threshold monitoring done by the system for buffer overrun errors.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs a minimum amount of error threshold monitoring for buffer overrun errors: 2 errors in the first 90 seconds or 20 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for buffer overrun errors: 2 errors in the first 300 seconds or 6 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for buffer overrun errors.

threshold-value: Specify a value ranging from 1 through 3000.

UNDERRUN

Specifies the level of error threshold monitoring done by the system for buffer underrun errors.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs a minimum amount of error threshold monitoring for buffer overrun errors: 2 errors in the first 90 seconds or 20 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for buffer overrun errors: 2 errors in the first 300 seconds or 6 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MAX:** The system performs the maximum amount of error threshold monitoring for buffer underrun errors.

threshold-value: Specify a value ranging from 1 through 3000.

ABORTS

Specifies the level of error threshold monitoring done by the system for frames aborted.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs a minimum amount of error threshold monitoring for aborted frames received: 6 errors in the first 30 seconds or 180 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for aborted frames received: 2

errors in the first 30 seconds or 60 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for aborted frames received.

threshold-value: Specify a value ranging from 1 through 5000.

RETRANSMIT

Specifies the threshold for the number of frame retransmissions beyond which a message is sent informing the user that errors have occurred.

***OFF:** Error threshold monitoring is turned off for retransmitted frames.

***MIN:** The system performs the minimum amount of error threshold monitoring for retransmitted frames: 5 errors in the first 30 seconds or 150 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for retransmitted frames: 2 errors in the first 60 seconds or 30 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for retransmitted frames.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

FRMSEQERR

Specifies the threshold for the number of frame sequence errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** Error threshold monitoring is turned off for frame sequence errors.

***MIN:** The system performs the minimum amount of error threshold monitoring for frame sequence errors: 2 errors in the first 90 seconds or 20 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for frame sequence errors: 2 errors in the first 300 seconds or 6 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for frame sequence errors.

threshold-value: Specify a number, ranging from 1 through 3000, that corresponds to the number of errors allowed in a 15-minute interval.

CTL

Specifies, for nonswitched lines, the name of the controller to which this line is attached. The controller description must already exist. However, this parameter is automatically updated if a controller description is created after this line description is created and if it is attached to this line description.

IDLCWDWSIZ

Specifies the default window size.

***CNN:** The value is determined according to the value specified on the CNN parameter.

default-window-size: Specify a value ranging from 1 through 31.

IDLCFRMRTY

Specifies the maximum number of retransmissions before reporting an error.

***CNN:** The value is determined according to the value specified on the CNN parameter.

frame-retry: Specify a value ranging from 0 through 100.

IDLCRSPTMR

Specifies the amount of time to wait before retransmitting a frame if an acknowledgement has not been received.

***CNN:** The value is determined according to the value specified on the CNN parameter.

response-timer: Specify a value ranging from 10 through 100 tenths of seconds. For example, 100 tenths of seconds equals 10 seconds.

IDLCCNNRTY

Specifies the number of times a transmission can be retried at connection time.

***CNN:** The value is determined according to the value specified on the CNN parameter.

***NOMAX:** There is no disconnect limit.

connect-retry-count: Specify a value ranging from 1 through 100.

MAXFRAME

Specifies the maximum frame size that can be transmitted and received on this line description.

2048: The default size is 2048.

max-frame: Specify a value ranging from 265 through 8196.

INFTRFTYPE

Specifies the information transfer type. The information transfer type determines the layer 1 protocol.

***UNRESTRICTED:** The data-channel traffic appears as digital information; no physical transformation is required and each B-channel operates at capacity (64k bps).

***V110:** The transfer type is V-series Recommendation 110. Each B-channel operates at 56k bps.

***IVTHDLC:** The transfer type is Inverted HDLC. Each B-channel operates at capacity, 64k bps.

LINKSPEED

Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

64000: The link speed is 64,000 bps.

***MIN:** A link speed of less than 1200 bps is used.

***MAX:** A link speed greater than 16M bps is used.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 2048000, 4M, 10M, and 16M.

SWTNWISLCT

Specifies the method used to select network interfaces from the switched network interface list.

***FIRST:** Selection begins with the first network interface specified in the switched network interface list.

***CALC:** The system calculates which network interface is selected.

COSTCNN

Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

***CNN:** The cost per connection time is 0 for non-switched connections, and 128 for switched connections.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE

Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

***CNN:** The cost per connection time is 0 for non-switched connections, and 128 for switched connections.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY

Specifies the security level of the physical line. This parameter is valid only if APPN is used on the system.

***NONSECURE:** Normal priority is used.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY

Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system.

The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***TELEPHONE:** The telephone propagation delay is used.

***LAN:** The local area network propagation delay is used.

***MIN:** The minimum propagation delay is used.

***PKTSWTNET:** The packet switched network propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1

Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2

Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3

Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority the user is granting to a user who does not have specific authority to an object, who is not on the authorization list, or whose user group has no specific authority to the object.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can control the object's existence, specify the security for the object, change the object, change the owner for the object, and perform basic functions on the object. All authority allows the user to perform all operations on the object except those limited to the owner, or controlled by authorization list management authority.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

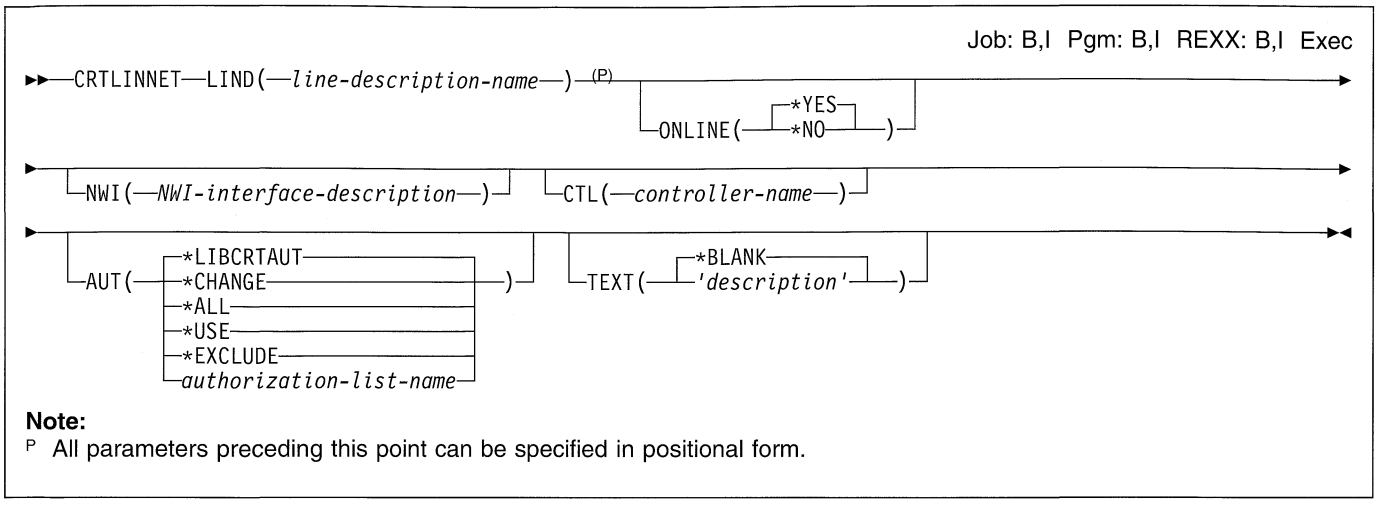
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINIDLC   LIND(IDLCLINE)
```

This command creates an IDLC line description named IDLCLINE

CRTLINNET (Create Line Description (Network)) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Line Description (Network) (CRTLINNET) command creates a line description for a network line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameter

LIND

Specifies the name of the line description being created.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

NWI

Specifies the name of the ISDN network interface to which this line is attached. The network interface description must already exist. The line description attaches to the D channel of the NWI.

CTL

Specifies the name of the controller that is attached to this line. The controller description must already exist.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** Other users are prevented from accessing the object.

authorization-list-name: Specify the name of the authorization list whose authority is used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** No text is specified.

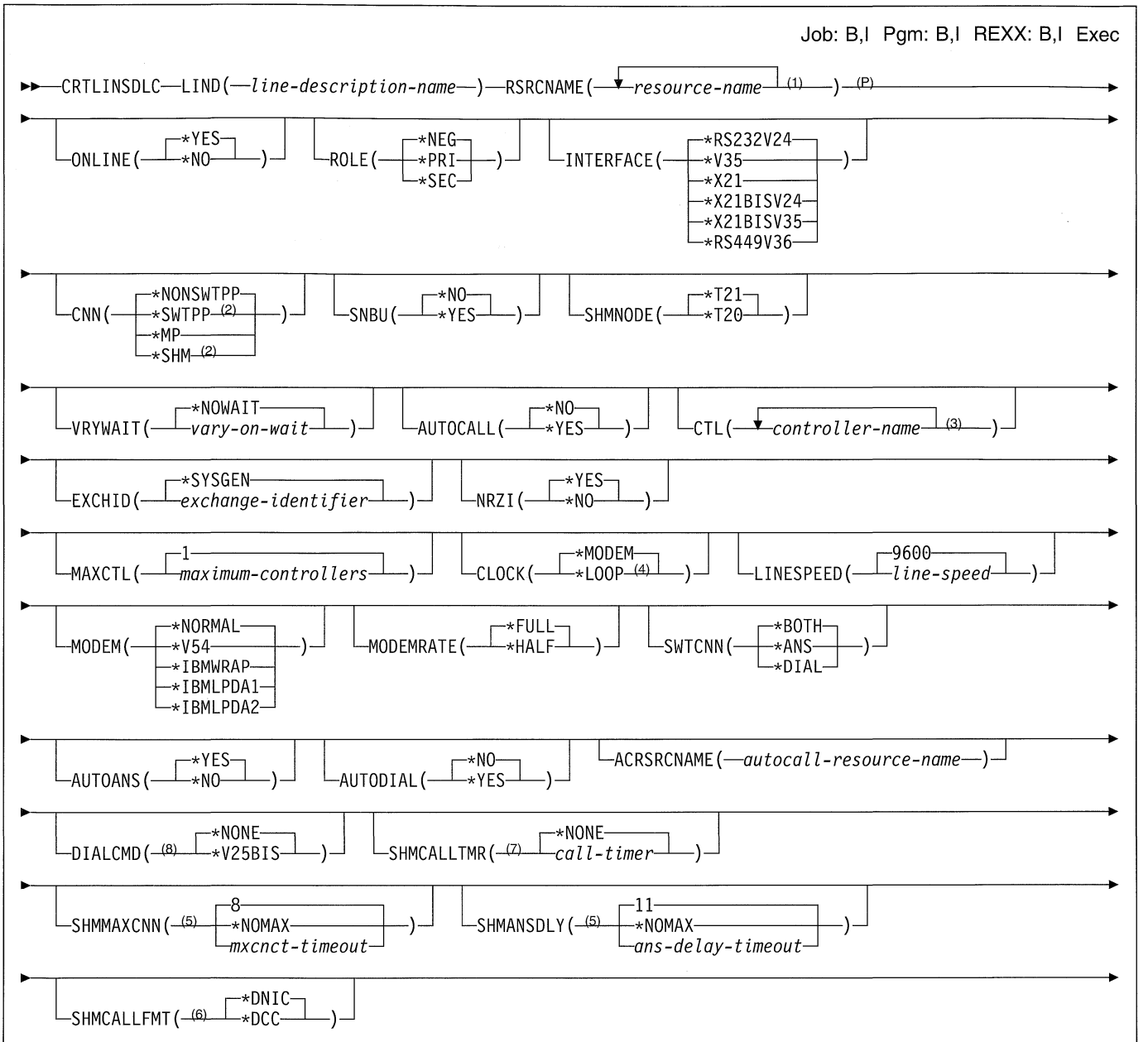
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINNET   LIND(LIN0A)  NWI(NWI01)
```

This command creates a network line description named LIN0A. An existing network interface, NWI01, is specified as the network interface for the line.

CRTLINSDLC (Create Line Description (SDLC)) Command



Notes:

1 A maximum of 6 repetitions

2 This value is not valid when *RS449V36 is specified on the INTERFACE parameter.

3 A maximum of 254 repetitions

4 This value is valid only if *RS449V36 is specified on the INTERFACE parameter.

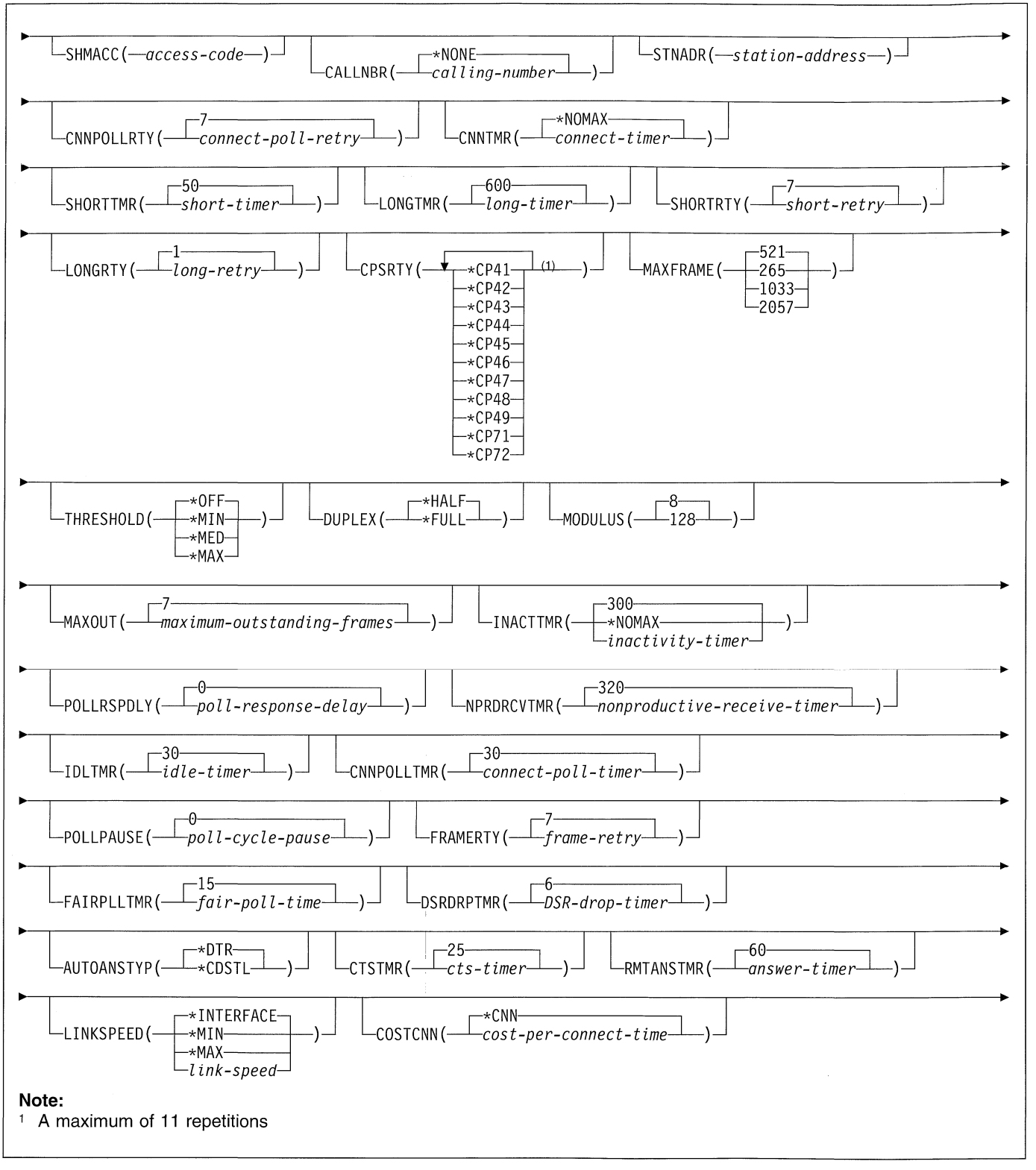
5 This parameter is valid only if *PRI or *NEG is specified on the ROLE parameter and *SHM is specified on the CNN parameter.

6 This parameter is valid only if *SHM is specified on the CNN parameter and *T21 is specified on the SHMNODE parameter.

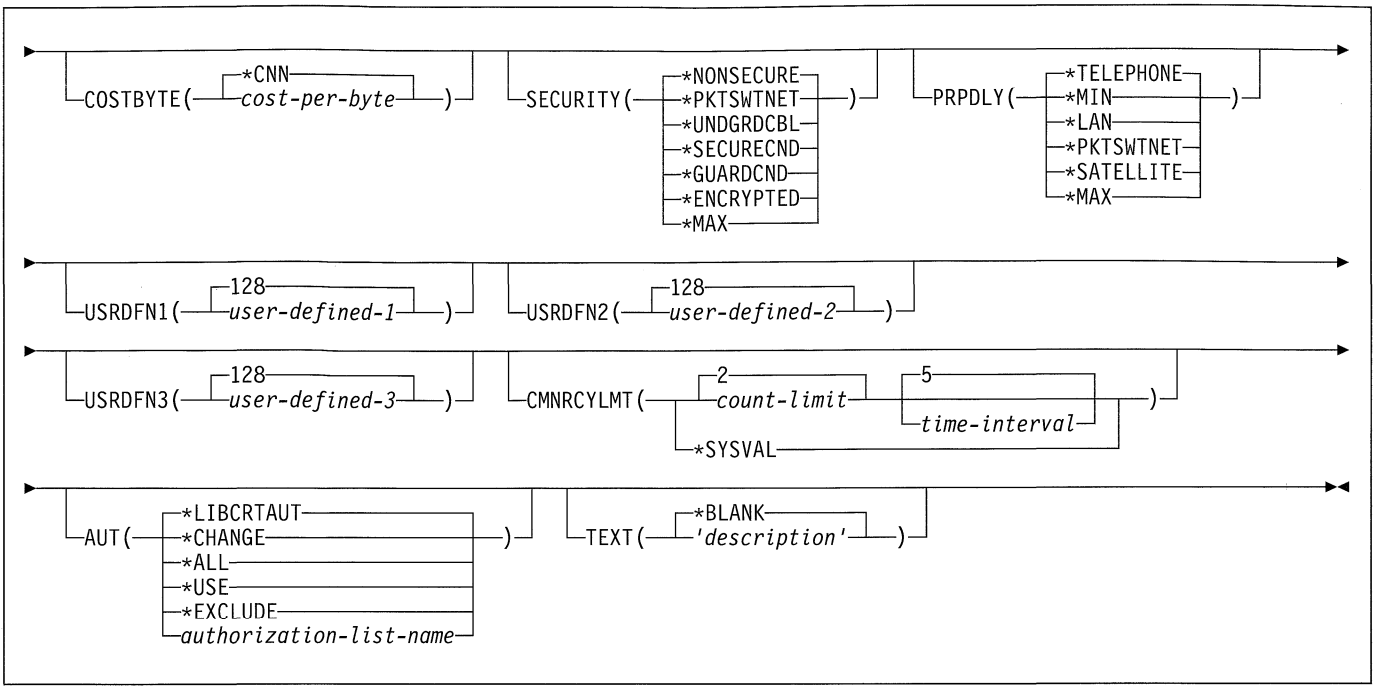
7 This parameter is valid only if *SHM is specified on the CNN parameter.

8 This parameter applies only if *YES is specified on the AUTODIAL parameter.

P All parameters preceding this point can be specified in positional form.



Note:
[1] A maximum of 11 repetitions



Purpose

The Create Line Description (SDLC) (CRTLINSDLC) command creates a line description for a synchronous data link control (SDLC) line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name.

Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is two, then the resource name would be LIN012.

Up to six resource names can be specified if CNN(*SHM) is specified (X.21 short-hold mode). All lines specified must be attached to the same input/output processor. If ROLE(*SEC) and SHMNODE(*T20) are specified, only one resource name can be used.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

ROLE

Specifies whether the system is the primary station, the secondary station, or whether the two systems dynamically negotiate the primary and secondary roles.

The primary station is the controlling station and the secondary station is the responding station. The primary station controls the communications link by sending commands to the secondary station, and the secondary station responds to the commands.

***NEG:** Allows the AS/400 system and the remote system to negotiate which station is primary.

***PRI:** The AS/400 system is the primary station on this communications line.

***SEC:** The AS/400 system is a secondary station on this communications line.

INTERFACE

Specifies the type of physical interface on the Input/Output Adapter (IOA) port.

***RS232V24:** The RS-232/V.24 interface is used.

***V35:** The V.35 interface is used.

***X21:** The X.21 interface is used.

***X21BISV24:** The X.21bis/V.24 interface is used.

***X21BISV35:** The X.21bis/V.35 interface is used.

***RS449V36:** The RS-449/V.36 interface is used.

CNN

Specifies the type of line connection used.

***NONSWTPP:** A nonswitched point-to-point line is used.

***SWTPP:** A switched point-to-point line is used.

***MP:** A nonswitched multipoint line is used.

***SHM:** An X.21 short hold mode line is used.

SNBU

Specifies, for controllers attached to nonswitched lines only, whether the switched network backup (SNBU) feature is activated or deactivated. This feature lets the user bypass a broken nonswitched connection by establishing a switched connection. This parameter applies only if SWITCHED(*NO) and SNBU(*YES) are specified when the controller description is created.

Note: The Change Line Description (SDLC)(CHGLINSDLC) and appropriate Change Controller commands must be used to activate the SNBU feature. Switched network backup is valid only if the local modem and remote modem both support this feature.

***NO:** The remote modem does not have the switched network backup (SNBU) feature.

***YES:** The modem has the SNBU feature.

SHMNODE

Specifies, for X.21 short-hold mode lines only, the physical unit type of the controllers using the line. This parameter is valid only if CNN(*SHM) is specified.

***T21:** Physical unit type 2.1 controllers are used. ROLE(*NEG) must also be specified.

***T20:** Physical unit type 2.0 controllers are used. ROLE must be specified as *PRI or *SEC.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

AUTOCALL

Specifies, for both a switched line and a switched network backup (SNBU) line, whether the line has an associated automatic call unit that performs automatic calling to the remote system.

***NO:** This switched line does not have an automatic unit.

***YES:** This switched line has an automatic call unit.

CTL

Specifies the name of the controller description to which this object is attached.

Note: The controller descriptions must already exist. Do not use this parameter when following the normal procedure of creating descriptions for lines first, controllers second, and devices last. Use this parameter only when the associated controller descriptions have already been created before this line description.

EXCHID

Specifies the hexadecimal exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains three digits for the block number and five digits for the identifier of this system.

***SYSGEN:** The AS/400 system creates the exchange identifier. Use the Display Line Description (DSPLIND) command to see the resulting exchange identifier.

exchange-identifier: Specify an exchange identifier. If the *SYSGEN value is not used, specify an exchange identifier containing eight hexadecimal digits starting with 056.

NRZI

Specifies whether non-return-to-zero-inverted (NRZI) data encoding is used for modems that are sensitive to certain bit patterns in the data stream. This ensures that the signal does not remain the same for an extended period of time.

Note: All data communications equipment on the line must use the same data transmission coding method.

***YES:** NRZI data encoding is used.

***NO:** NRZI data encoding is not used.

MAXCTL

Specifies the maximum number of controllers that the line supports.

1: One controller is supported.

maximum-controllers: Specify the maximum number of controllers supported by the line, a number large enough to account for all of the controllers that are currently attached to this line, and for those controllers to be attached in the near future. Valid values range from 1 through 254.

CLOCK

Specifies how the clocking function for the line is provided.

***MODEM:** The modem supplies the clocking function.

***LOOP:** The receiving clock provided by the modem data circuit-terminating equipment (DCE) is looped back to the modem DCE on the system data terminal equipment (DTE) transmitting clock. This option can be used to improve high speed data transmission when the modem DCE supports such an option.

LINESPEED

Specifies the line speed in bits per second (bps).

9600: 9600 bps is used.

line-speed: Specify the line speeds. Valid lines speeds are: 600, 1200, 2400 4800, 7200, 9600, 14400, 19200, 48000, 56000, 57600, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 392000, 448000, 504000, 512000, 560000, 576000, 616000, 640000, 672000, 704000, 728000, 768000, 784000, 832000, 840000, 896000, 952000, 960000, 1008000, 1024000, 1064000, 1088000, 1120000, 1152000, 1176000, 1216000, 1232000, 1280000, 1288000, 1344000, 1400000, 1408000, 1456000, 1472000, 1512000, 1536000, 1568000, 1600000, 1624000, 1664000, 1680000, 1728000, 1736000, 1792000, 1856000, 1920000, 1984000, or 2048000 bits per second.

MODEM

Specifies the type of modem supported on the communications line.

The modem manual has information to determine the appropriate value for the modem.

***NORMAL:** No attempt is made to run diagnostic tests on the modem.

***V54:** Certain types of diagnostic tests (as defined by the CCITT recommendations) are run to the modem. The AS/400 system supports CCITT V.54 loop three (local loop back) and loop two (a remote loop back).

***IBMWRAP:** An IBM modem with wrap test capabilities is used on the communications line.

***IBMLPDA1:** An IBM modem with Link Problem Determination Aid-1 (LPDA-1) is used on the line.

***IBMLPDA2:** An IBM modem with Link Problem Determination Aid-2 (LPDA-2) is used on the line.

MODEMRATE

Specifies the speed at which the line operates if the modem has the data rate select feature.

***FULL:** The line operates at the full rate of the modem.

***HALF:** The line operates at one-half the full rate, or at the alternate rate, of the modem.

SWTCNN

Specifies, for the switched line and the switched network backup (SNBU) line, whether the line is used for incoming calls, outgoing calls, or both.

***BOTH:** The line is used for both incoming and outgoing calls.

***ANS:** The line is used for incoming calls only.

***DIAL:** The local system starts the call.

AUTOANS

Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically answers a call from a remote system to establish the connection or whether the user must manually answer the call and place the modem in data mode.

***YES:** The system automatically answers incoming calls.

Note: *YES is valid only if the modem has the automatic answer feature or an X.21 circuit switched interface is used.

***NO:** The system operator must manually answer incoming calls.

AUTODIAL

Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically calls a remote system to establish a connection or whether the system operator must manually place the call.

Note: For an X.21 circuit switched interface, automatic dialing must be specified.

***NO:** The system operator must manually call a remote system.

***YES:** The system automatically calls a remote system.

Note: *YES is valid only if the system is using an automatic call unit or an X.21 circuit switched interface, or if the modem can call through a command interface.

ACRSRCNAME

Specifies the resource name that describes the automatic call unit port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN02 and the port is 1, the resource name is LIN021.

DIALCMD

Specifies the type of dial command used to establish a connection with a remote system.

***NONE:** No dial command type (automatic call unit or X.21 circuit switched interface) is specified.

***V25BIS:** The use of one physical interface for call establishment and data transmission is allowed. It is sometimes referred to as a serial automatic call interface because the digits are presented serially on the link from the system data terminal equipment (DTE) to the modem data circuit-terminating equipment (DCE).

SHMCALLTMR

Specifies the interval at which a connection is reestablished on an X.21 short-hold mode line to verify the state of the remote system even if no normal data traffic has occurred in the specified interval.

Note: This parameter is valid only if CNN(*SHM) is specified.

***NONE:** No call is made to verify the connection.

call-timer: Specify an interval, ranging from 1 through 60 minutes, after which a call is made to verify the connection.

SHMMAXCNN

Specifies the amount of time the system allows a connection to continue if there are more controllers than there are available ports. The system clears the connection after the specified amount of time, delay further calls for the amount of time specified on the SHMANSDLY parameter, and then make any calls that had been waiting before re-calling the controller that was interrupted.

Note: This parameter is valid only if *PRI or *NEG is specified on the ROLE parameter and *SHM is specified on the CNN parameter.

8: The system waits 8 seconds before checking for other controllers.

***NOMAX:** The timer is disabled.

mxcnct-timeout: Specify a value ranging from 1 through 254 seconds.

SHMANSDLY

Specifies the amount of time the system waits for controllers to call in before making outgoing calls.

Note: This parameter is valid only if *PRI or *NEG is specified on the ROLE parameter and *SHM is specified on the CNN parameter.

11: The system waits 1.1 seconds before making outgoing calls.

***NOMAX:** The timer is disabled.

ans-delay-timeout: Specify a value ranging from 1 through 254 tenths of seconds. For example, 10 seconds equals 100 tenths of a second.

SHMCALLFMT

Specifies the format of the network identifier used in the local system's connection number (CNNNBR parameter).

Note: This parameter is valid only if *SHM is specified on the CNN parameter and *T21 is specified on the SHMNODE parameter.

***DNIC:** A four-digit Data Network Identification Code is used. The AS/400 system assumes the first four digits of the CALLNBR parameter are the Data Network Identification Code of the user's network. If a call is placed to a controller with the same DNIC (on the controller description CNNNBR parameter), a local call is assumed and the system discards the first four digits of the CNNNBR parameter.

***DCC:** A three-digit Data Country Code is used. The AS/400 system assumes the first three digits of the CALLNBR parameter are the Data Country Code. If a call is placed to a controller with the same DCC (on the controller description CNNNBR parameter), the system discards the first three digits of the CNNNBR parameter.

SHMACC

Specifies the access code used by an X.21 short-hold mode line when calling a system on another network. The access code can be three or four digits and is valid only for X.21 short-hold mode lines with SHMNODE(*T21) specified.

CALLNBR

Specifies the local telephone number of the line used for the V.25 bis call request with identification (CRI) dial command. This parameter is used when the CRI function is needed for V.25 bis. When V.25 bis CRI dialing is used, the system takes the called (connection) number from the CNNNBR parameter of the controller description, adds a separator character (;), and concatenates the calling number at the end. Specify the calling number only if the modem and the network both support the CRI dial command.

Note: This parameter is required for all X.21 short-hold mode lines. The calling number is passed to the remote system at the initial connection. This number is dialed by the remote system to reestablish the short-hold mode connection.

***NONE:** The CRN (Call Request Normal) dial command is used by the V.25 bis line, or the line does not use X.21 short-hold mode.

calling-number: Specify up to 32 characters that represent the local telephone number for V.25 bis CRI (Call Request with Identification) automatic dialing.

For X.21 short-hold mode lines:

- If SHMNODE(*T21) is specified, up to 14 characters can be specified for the calling number. The Data Network Identification Code (DNIC) or Data Country Code (DCC) must be included in the first three or four digits of the calling number.
- If SHMNODE(*T20) is specified, up to 18 characters can be specified for the calling number. Include the DNIC or DCC in the first three or four digits of the calling number only if the remote device always calls from a different DNIC or DCC location.

STNADR
Specifies, for a switched secondary line or negotiable line, the hexadecimal station address to which the local system responds when polled by the remote system if it answers a call. Specify the station address in hexadecimal characters 01 through FE.

CNNPOLLRTY
Specifies, for a switched primary line or negotiable line, the number of connect poll retries before indicating the error and making the station inoperative.

7: Seven retries are attempted.

connect-poll-retry: Specify a value ranging from 0 through 64 for the number of retries.

CNNTMR
Specifies, for X.21 circuit-switched lines, the number of seconds an automatic answer connect request waits for an incoming call to be accepted.

***NOMAX:** There is no disconnect limit.

connect-timer: Specify a value ranging from 1 through 32767 in 0.1-second intervals.

SHORTTMR
Specifies, for X.21 circuit-switched lines, the number of seconds the system waits between connection attempts. This timer is used during bursts of retry operations.

50: The system waits five seconds.

short-timer: Specify a value ranging from 10 through 600 in 0.1-second intervals.

LONGTMR
Specifies, for X.21 circuit-switched lines, the number of seconds the timer waits between bursts of retry operations. After a burst of retry attempts, the system waits for this timeout period before the next attempt.

600: The system waits 60 seconds.

long-timer: Specify a value ranging from 100 through 6000 in 0.1-second intervals.

SHORTRTY
Specifies, for X.21 circuit-switched lines, the number of retry attempts made during a burst of retries.

7: Seven retries are attempted.

short-retry: Specify a value ranging from 0 through 254 for the number of retries.

LONGRTY
Specifies, for X.21 circuit-switched lines, the number of retry burst attempts made when processing a connect request.

1: One retry is attempted.

long-retry: Specify a value ranging from 0 through 254 for the number of retries.

CPSRTY
Specifies whether call progress signals are retried for X.21 circuit-switched lines or X.21 short-hold mode lines.

Up to 11 values can be specified; duplicate values are ignored.

Possible values are:

*CPS41	*CPS44	*CPS47	*CPS71
*CPS42	*CPS45	*CPS48	*CPS72
*CPS43	*CPS46	*CPS49	

This parameter can be specified only if INTERFACE(*X21) and either *SHM or *SWTPP are specified on the CNN parameter.

MAXFRAME
Specifies the maximum frame (path information unit (PIU)) size that the controller can send or receive. This value is used to calculate request unit (RU) sizes. Since the maximum PIU size that the controller can send or receive is negotiated at exchange identifier time, the maximum PIU size used at run time may be different. This value matches the corresponding value on the host system.

Note: The possible frame sizes are 265, 521 (default), 1033, and 2057.

THRESHOLD
Specifies the temporary error threshold level being monitored by the system. A permanent error is reported only if the errors occurred consecutively and exceeded the retry limit.

Note: Specifying the THRESHOLD parameter affects all threshold errors. They cannot be specified individually.

***OFF:** No monitoring of errors occurs.

***MIN:** The error threshold is set at a minimum monitoring level.

***MED:** Error thresholding is set to a medium monitoring level.

***MAX:** The error threshold is set at a maximum monitoring level.

DUPLEX
Specifies whether request-to-send (RTS) is permanently turned on (for full-duplex modems) or turned on only when data transmission is required (for half-duplex modems).

Note: For X.21 circuit-switched interface lines, full duplex permanent RTS must be specified.

***HALF:** RTS is turned on only when transmission is required (for half-duplex modems).

***FULL:** Request-to-send (RTS) is permanently turned on (for full-duplex modems).

MODULUS
Specifies whether extended sequence numbers are used.

8: Extended sequence numbers are not used (Modulus 8).

128: Extended sequence numbers are used (modulus 128).

MAXOUT

Specifies the maximum number of frames that are sent sequentially to a remote system before the remote system must respond. For modulus 8, the maximum number of frames range from 1 through 7. For modulus 128, the maximum number of frames range from 8 through 28.

7: The frame is sent seven times.

maximum-outstanding-frames: Specify a value ranging from 1 through 28 for the number of frames.

INACTTMR

Specifies, for a secondary or negotiable line, the time (in tenths of a second) that the system waits for a valid frame to flow before reporting the error and disconnecting the line. This timer is started at connection time, restarted upon transmission of any frame, and reset upon receipt of a frame with a valid frame check sequence.

300: The system waits 30.0 seconds.

***NOMAX:** There is no disconnect limit.

inactivity-timer: Specify a value ranging from 150 through 4200 in 0.1-second intervals.

POLLRSPDLY

Specifies, for a secondary line or negotiable line, the minimum number of seconds the system must wait before it responds to a data poll if there is no frame to transmit.

0: There is no delay.

poll-response-delay: Specify a value ranging from 1 through 2048 in 0.0001-second intervals.

NPRDRCVTMR

Specifies, for a primary line or a negotiable line, the number of seconds the system waits for either a final frame or an idle signal while the secondary station is continuously sending. If this time ends, the nonproductive receive condition is reported.

320: The system waits 32.0-seconds.

nonproductive-receive-timer: Specify a value ranging from 160 through 4200 in 0.1-second intervals.

IDLTMR

Specifies, for a primary or negotiable line, the time (in 0.1 second intervals) that the system waits before sampling the line for an idle signal. If an idle signal is found, error recovery procedures are started.

30: The system waits three seconds.

idle-timer: Specify a value ranging from 5 through 300 in 0.1-second intervals.

CNNPOLLTMR

Specifies, for a primary line or negotiable line, the number of seconds the system waits for the response to a connect poll before sending the poll again.

30: The system waits three seconds.

connect-poll-timer: Specify a value ranging from 2 through 300 in 0.1-second intervals.

POLLPAUSE

Specifies, for a primary or negotiable line, the number of seconds the system pauses after the last remote system in the poll list is polled.

0: There is no pause.

poll-cycle-pause: Specify a value ranging from 1 through 2048 in 0.0001-second intervals.

FRAMERTY

Specifies the number of retries for an unanswered command frame or unacknowledged information frame before indicating the error and disconnecting the station.

7: Seven retries are attempted.

frame-retry: Specify a value from 0 through 64 for the number of retries.

FAIRPLLTMR

Specifies the maximum amount of time for which the system sends data to one or more work stations before requesting input from the work stations.

15: The system waits 15 seconds (during output data transfers to work stations) before polling for input.

fair-poll-time: Specify the amount of time (in seconds) the system waits (during output transfers) before polling for input. Valid values for this parameter range from 5 through 60 seconds.

DSRDRPTMR

Specifies the amount of time the system waits for the modem to exit the Data Set Ready (DSR) state before signaling an error.

6: The system waits 6 seconds in the DSR state before signaling an error.

DSR-drop-timer: Specify a value ranging from 3 through 60 seconds.

AUTOANSTYP

Specifies the method the system uses to answer incoming calls.

***DTR:** The system enters the Data Terminal Ready state, signals the modem to answer calls, and waits for the modem to enter the Data Set Ready (DSR) state.

***CDSTL:** The system enters the Connect Data Set to Line (CDSTL) state after monitoring the Ring Indicator to signal the modem to answer the call.

CTSTMR
Specifies the amount of time the system waits for the modem to enter or exit the Clear to Send (CTS) state before signaling an error.

25: The system waits up to 25 seconds for the CTS state to begin or end.

CTS-timer: Specify a value ranging from 10 through 60 seconds.

RMTANSTMR
Specifies the amount of time the system waits for the modem to enter the Data Set Ready (DSR) state after dialing before signaling an error.

60: The system waits 60 seconds before signaling an error.

answer-timer: Specify a value ranging from 30 through 120 seconds.

LINKSPEED
Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

***INTERFACE:** The following link speeds, based on the physical interface type, are used: 9600 bps for RS-232/V.24 and X.21bis/V.24, 48000 bps for V.35 and X.21bis/V.35, and 64000 bps for X.21 and RS-449/V.36.

***MIN:** A link speed of less than 1200 bps is used.

***MAX:** A link speed greater than 16M bps is used.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 2048000, 4M, 10M, and 16M.

COSTCNN
Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

***CNN:** The cost per connection time is 0 for non-switched connections, and 128 for switched connections.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE
Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

***CNN:** The cost per byte is 0 for nonswitched connections and 128 for switched connections.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY
Specifies the security level of the physical line. This parameter is used only if APPN is used on the system.

***NONSECURE:** Normal priority is used.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY
Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system. The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***TELEPHONE:** The telephone propagation delay is used.

***MIN:** The minimum propagation delay is used.

***LAN:** The local area network propagation delay is used.

***PKTSWTNET:** The packet switched network propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1
Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2
Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3
Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

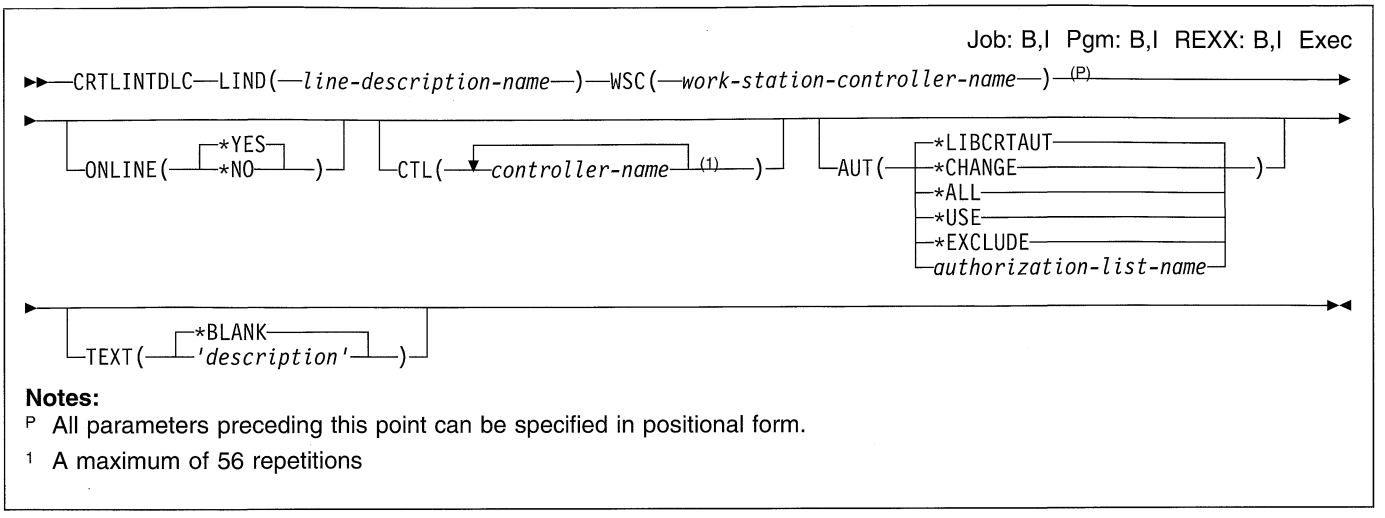
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINSDLC   LIND(BOSTON)  RSRCNAME(LIN041)
```

This command creates an SDLC line description named BOSTON with a resource name of LIN041.

CRTLINTDLC (Create Line Description (TDLC)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 56 repetitions

Purpose

The Create Line Description (TDLC) (CRTLINTDLC) command creates a line description for a twinaxial data link control (TDLC) line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

WSC

Specifies the name of the work station controller to which the PC is attached.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

CTL

Specifies the name of the controller description to which this object is attached.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

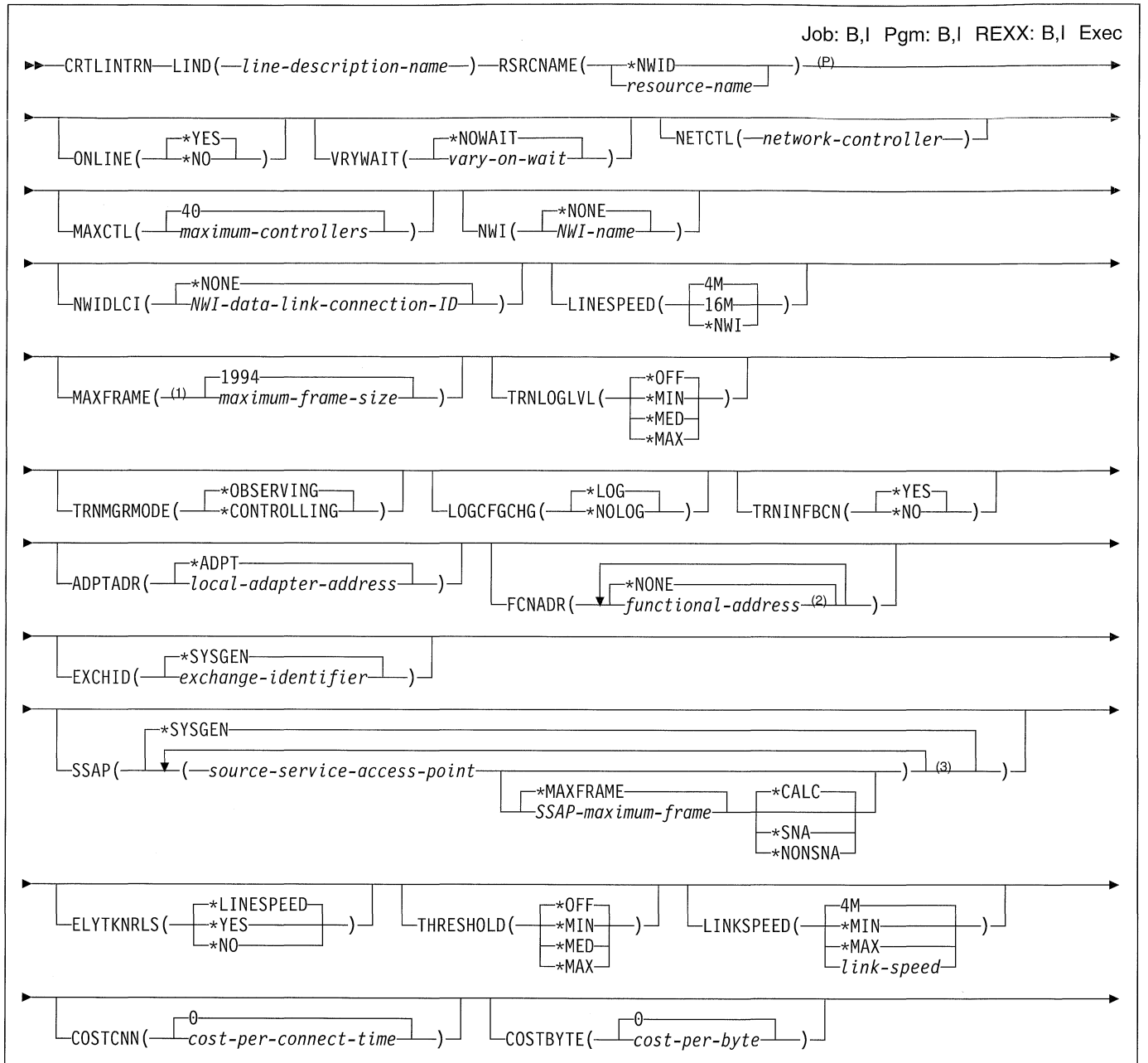
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINTDLC   LIND(WSFLINE)  WSC(CTL01)
```

This command creates a TDLC line description named WSFLINE that is attached to work station controller CTL01.

CRTLINTRN (Create Line Description (Token-Ring Network)) Command



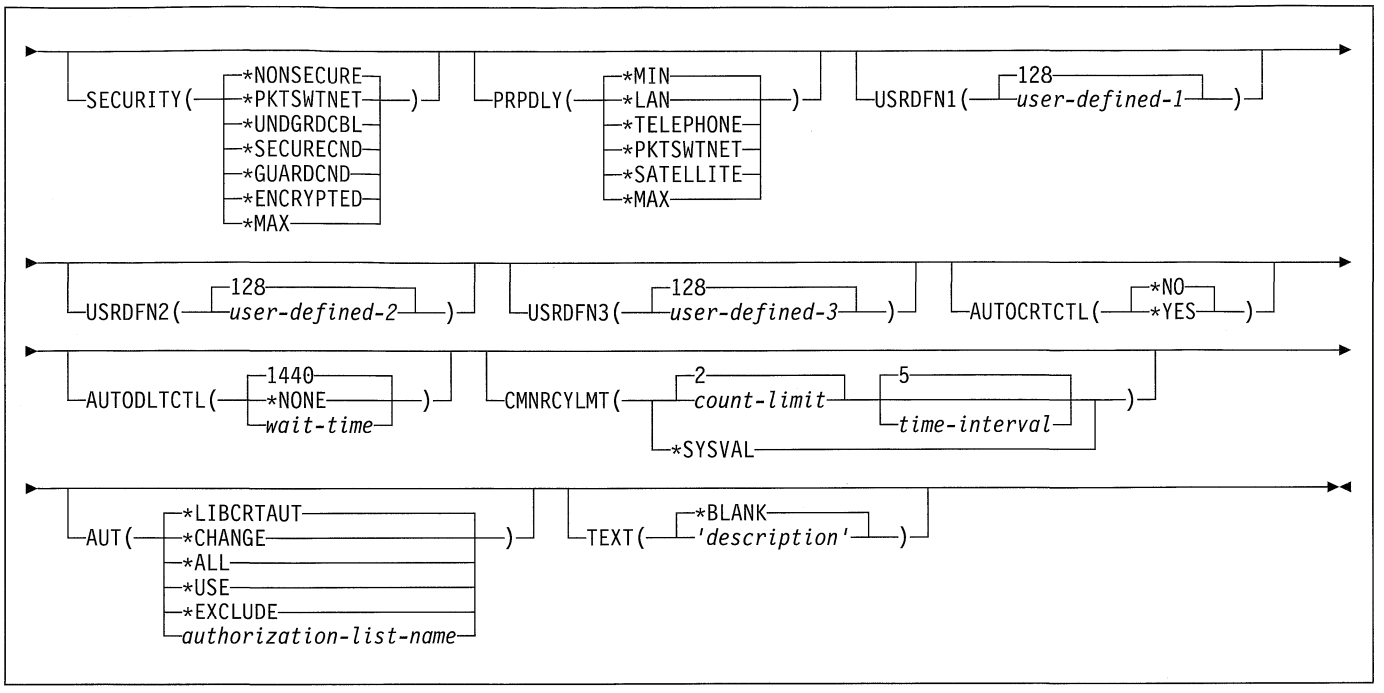
Notes:

P All parameters preceding this point can be specified in positional form.

1 Any value ranging from 265 through 8151 can be specified when RSRCNAME(*NWID) is specified. Otherwise, the valid values range from 265 to 16393.

2 A maximum of 31 repetitions

3 A maximum of 24 repetitions



Purpose

The Create Line Description (Token-Ring Network) (CRTLINTRN) command creates a line description for a token-ring network (TRN) line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

RSRCNAME

Specifies the resource name that describes the automatic call unit port.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 1, then the resource name is LIN011.

| ***NWID:** The resource name is determined by the
| network interface used.

| *resource-name:* Specify a resource name.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

VRYWAIT

Specifies whether the line is varied on asynchronously or
| synchronously. For synchronous vary on, specifies how
| long the system waits for the vary on to complete.

| ***NOWAIT:** The system does not wait for the vary on to
| complete. The line is varied on asynchronously.

| *vary-on-wait:* Specify the time (in seconds) to wait.
| Valid values range from 15 through 180. The system
| waits until the line is varied on, or until the specified time
| passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

NETCTL

Specifies the name of an existing network controller. This parameter can be specified for lines attached to a Transmission Control Protocol/Internet Protocol (TCP/IP) network.

MAXCTL

Specifies the maximum number of controllers that the line supports.

40: The line supports 40 controllers.

maximum-controllers: Specify a number large enough to account for all controllers currently active to this network, and the controllers that will be attached in the near future. Valid values range from 1 through 256.

NWI

Specifies the frame relay network interface to be used.

Note: NWI(*NONE) must be specified when RSRCNAME(*NWID) is not specified. Otherwise, NWI(*NONE) can be specified only when NWIDLCI(*NONE) is also specified.

***NONE:** No network interface is specified.

NWI-name: Specify the name of the frame relay network interface to be used.

NWIDLCI

Specifies the network interface data link connection identifier to be used.

Note: NWIDLCI(*NONE) must be specified when RSRCNAME(*NWID) is not specified. Otherwise, NWIDLCI(*NONE) can be specified only when NWI(*NONE) is also specified.

***NONE:** No network interface data link connection identifier is specified.

NWI-data-link-connection-ID: Specify the network interface data link connection identifier to be used. Valid values range from 1 through 1018.

LINESPEED

Specifies the line speed in bits per second (bps).

Note: When RSRCNAME(*NWID) is specified, *NWI must be specified on this parameter.

4m: The line speed is 4M bps.

16m: The line speed is 16M bps.

***NWI:** The line speed used is for a network interface.

Note: LINESPEED(*NWI) is not valid when RSRCNAME(*NWID) is not specified.

MAXFRAME

Specifies the maximum frame size that can be transmitted and received on this line description.

Note: If the LINESPEED is 4M, a value greater than 4060 cannot be specified. If the token-ring adapter supports only a 4M LINESPEED, only values 1994 and lower can be specified. When RSRCNAME(*NWID) is specified, valid values for this parameter range from 265 through 8148 bytes.

1994: The maximum frame size is 1994 bytes.

maximum-frame-size: Specify the maximum frame size value to be used. The valid frame sizes (in bytes) range from 256 through 8148 bytes when the network interface is a frame relay. Otherwise, valid frame sizes (in bytes) range from 256 through 16393.

TRNLOGLVL

Specifies the error logging level used by the Token Ring Local Area Network (TRLAN) Manager.

Note: TRNLOGLVL(*OFF) must be specified when RSRCNAME(*NWID) is specified.

***OFF:** No monitoring of errors occurs.

***MIN:** Starts the minimum reporting level, which reports only conditions that indicate degraded performance and beaconing.

***MED:** Starts the medium reporting level, which reports conditions that indicate potential degraded performance as well as those reported for the *MIN reporting level.

***MAX:** The maximum error logging level, which reports all error conditions, including those reported for the *MIN and *MED reporting levels, is used.

TRNMGRMODE

Specifies whether the token-ring manager is controlling or observing.

Note: TRNMGRMODE(*OBSERVING) must be specified when RSRCNAME(*NWID) is specified.

***OBSERVING:** The token-ring manager is observing.

***CONTROLLING:** The token-ring manager is controlling.

LOGCFGCHG

Specifies whether NAUN (nearest active upstream neighbor) changes are logged.

Note: LOGCFGCHG(*LOG) must be specified when RSRCNAME(*NWID) is specified.

***LOG:** NAUN changes are logged.

***NOLOG:** NAUN changes are not logged.

TRNINFBCN

Specifies whether an informational message is sent to QSYSOPR when beaconing occurs.

Note: TRNINFBCN(*YES) must be specified when RSRCNAME(*NWID) is specified.

***YES:** A message is sent to QSYSOPR when beaconing occurs.

***NO:** A message is not sent to QSYSOPR when beaconing occurs.

ADPTADR

Specifies the 12-character hexadecimal adapter address.

Note: ADPTADR(*ADPT) is not valid when RSRCNAME(*NWID) is specified.

***ADPT:** This value gives the user the preset token-ring default address for this token-ring adapter card. The user may display this by doing a DSPLIND on this line description after it has successfully varied on.

local-adapter-address: Specify an address for this system in the token-ring network. Valid values range from hexadecimal 400000000000 through 7FFFFFFFFFFF.

FCNADR

Specifies whether token ring functional addresses are used.

***NONE:** A functional address is not used.

functional-address: Specify a group of hexadecimal functional addresses that are encoded in bit-significant format. Valid values range from hex C00000000001 through hex C00040000000. The first digit must be C. Functional addresses must be unique.

EXCHID

Specifies the hexadecimal exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains three digits for the block number and five digits for the identifier of this system.

***SYSGEN:** The AS/400 system generates the exchange identifier.

exchange-identifier: Specify (if the *SYSGEN value is not specified) an exchange identifier composed of eight hexadecimal digits starting with 056.

SSAP

Specifies the source service access point (SSAP). The most commonly used SNA SSAP is hex 04. All SSAP values must be unique.

***SYSGEN:** The system automatically creates three SSAPs, hex 04 for SNA, and hex AA and 06 for TCP/IP applications.

Element 1: SSAPs

source-service-access-point: Specify up to 24 SSAPs, including hex AA and 06 for TCP/IP, and any hexadecimal number ranging from 04 through 9C that is divisible by four for SNA applications.

Element 2: Frame Size for SSAPs

***MAXFRAME:** The frame size specified on the MAXFRAME parameter is used.

SSAP-maximum-frame: Specify the maximum SSAP frame size (the maximum size of the data field that may be transmitted or received). When RSRCNAME(*NWID) is specified, valid values for this parameter range from 265 through 8148 bytes. Otherwise, valid values for this parameter range from 265 through 16393 bytes.

Element 3: SSAP Type

***CALC:** The system determines the SSAP type based on the following values:

- 04 through 9C, divisible by 4 (for SNA)
- 02 through FE, divisible by 2 (for non-SNA)

***SNA:** The SSAP is used for SNA communications. Valid values range from 04 through 9C and must be divisible by 4.

***NONSNA:** The SSAP is used for non-SNA communications. Valid values range from 02 through FE and must be divisible by 2.

ELYTKNRLS

Specifies whether the early token release option is used.

Note: ELYTRNRLS(*LINESPEED) must be specified when RSRCNAME(*NWID) is specified.

***LINESPEED:** The value specified on the LINESPEED parameter is used to determine whether the early token release option is used.

***YES:** The early token release option is used. *YES cannot be specified if the LINESPEED parameter contains the value 4M.

***NO:** The early token release option is not used.

THRESHOLD

This parameter, and its values *OFF, *MIN, *MED, and *MAX, can be specified but it is not used by the system starting in release V2R3M0. The parameter may be removed in a later release.

LINKSPEED

Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

4M: The link speed is 4M bps.

***MIN:** A link speed of less than 1200 bps is used.

***MAX:** A link speed greater than 16M bps is used.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 4M, 10M, and 16M.

COSTCNN

Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

0: The cost per connect time is 0.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE

Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

0: No rotation occurs. Printing starts at and is parallel to the edge loaded into the printer.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY

Specifies the security level of the physical line. This parameter is used only if APPN is used on the system.

***NONSECURE:** Normal priority is used.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY

Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system. The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***MIN:** The minimum propagation delay is used.

***LAN:** The local area network propagation delay is used.

***TELEPHONE:** The telephone propagation delay is used.

***PKTSWTNET:** The packet switched network propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1

Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2

Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3

Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

AUTOCRTCTL

Specifies whether the system will automatically create controller descriptions when calls are received from adjacent systems on the local area network (LAN).

***NO:** The system will not automatically create a controller description when incoming calls are received.

***YES:** The system will automatically create a controller description when incoming calls are received.

AUTODLTCTL

Specifies the number of minutes an automatically created controller can remain in an idle state (switched from varied on to varied on pending) before the controller description and attached device descriptions are varied off and deleted.

1440: The controller description can be idle for 1440 minutes (24 hours).

***NONE:** The system will not automatically delete or vary off the automatically configured, idle controller descriptions.

wait-time: Specify the number of minutes to wait before deleting the automatically configured, idle controller descriptions for this line. Valid values range from 1 to 10000 minutes.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

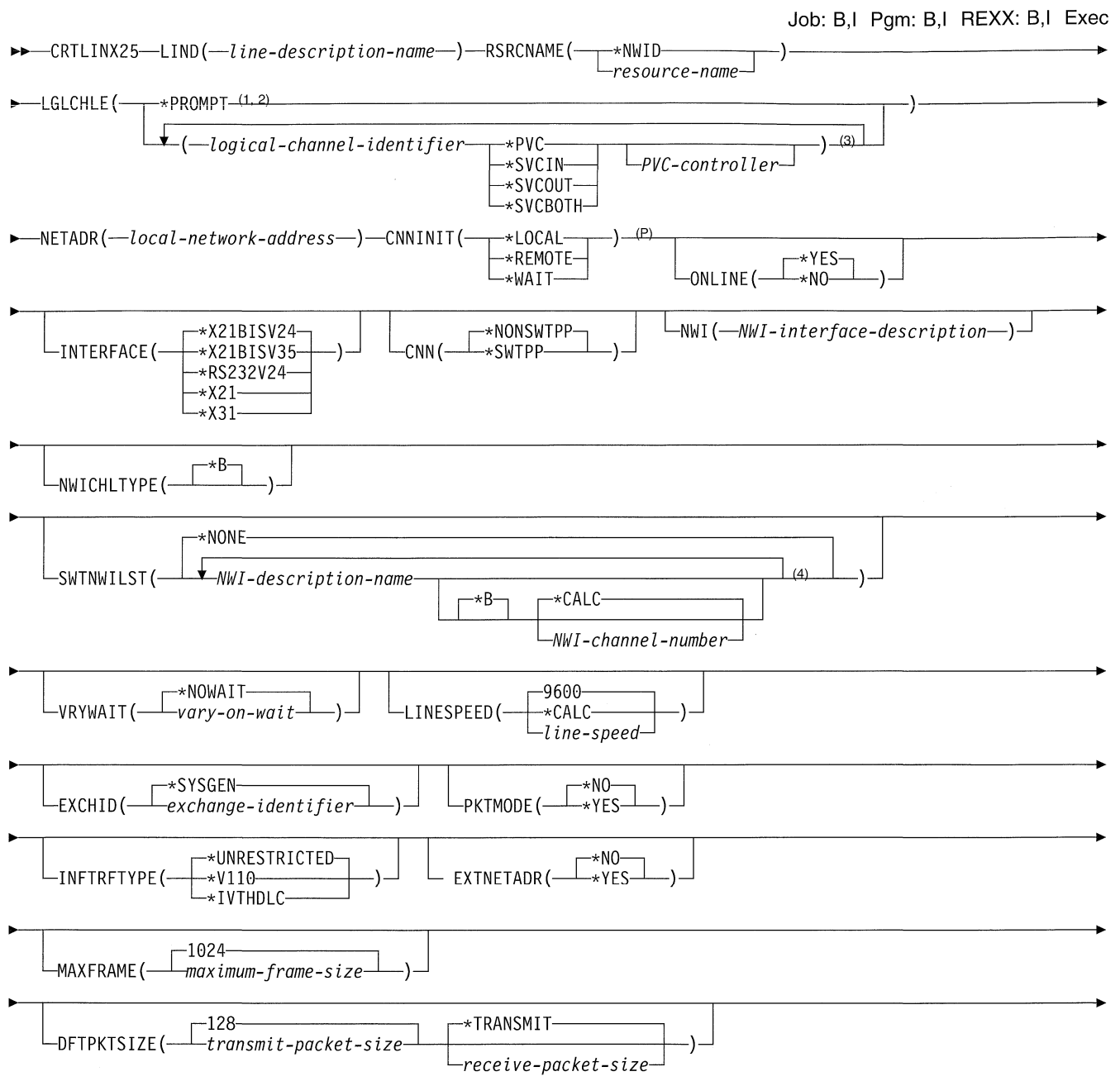
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINTRN   LIND(TRLAN1)  RSRCNAME(LIN011)
  TEXT('Token-Ring Line')
```

This command creates a token-ring line (TRLAN1) with resource name LIN011 and exchange identifier 05612345.

CRTLINX25 (Create Line Description (X.25)) Command



Notes:

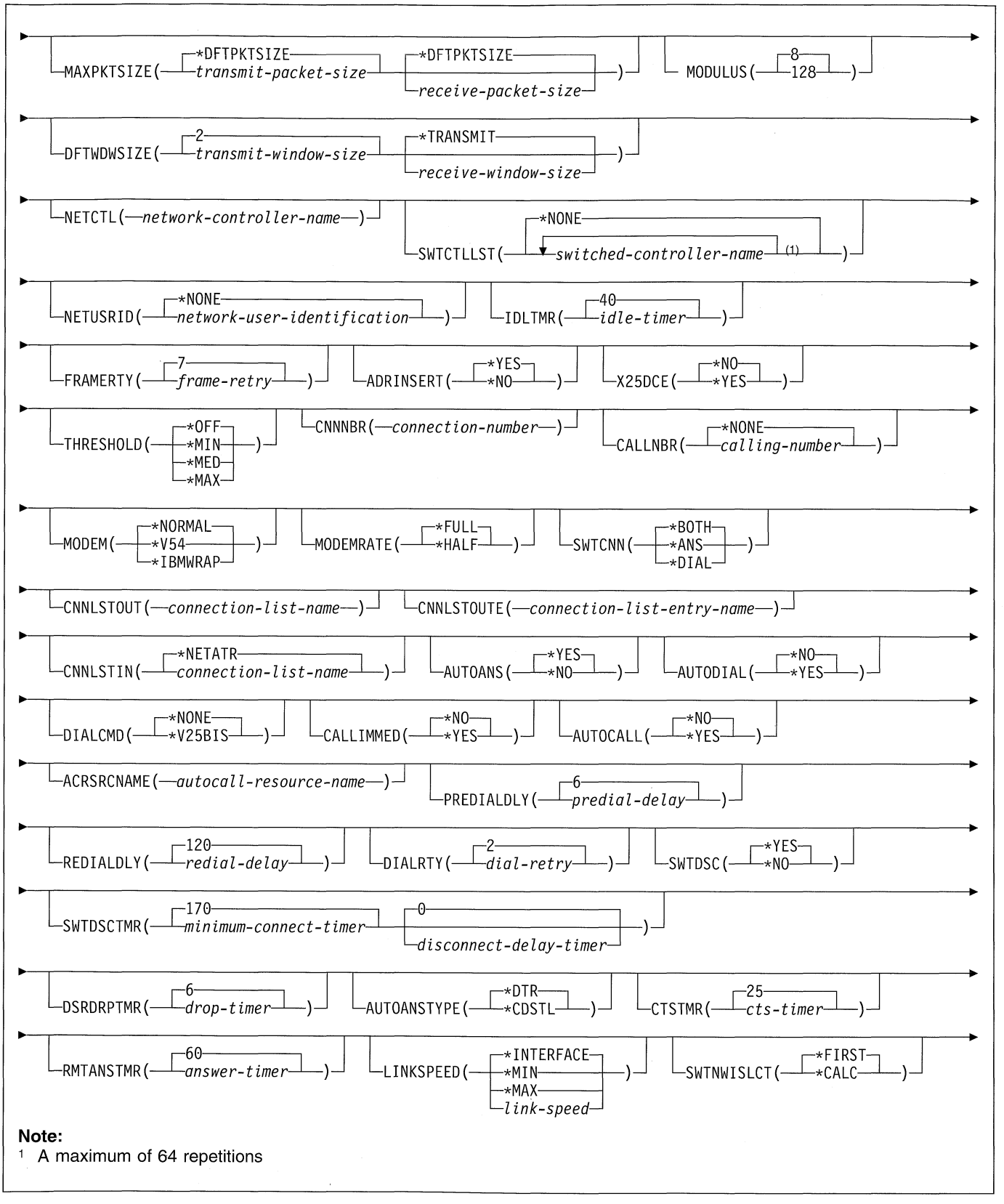
1 *PROMPT cannot be specified in batch mode.

2 Up to 64 logical channel entries can be specified.

3 Does not apply for *SVC.

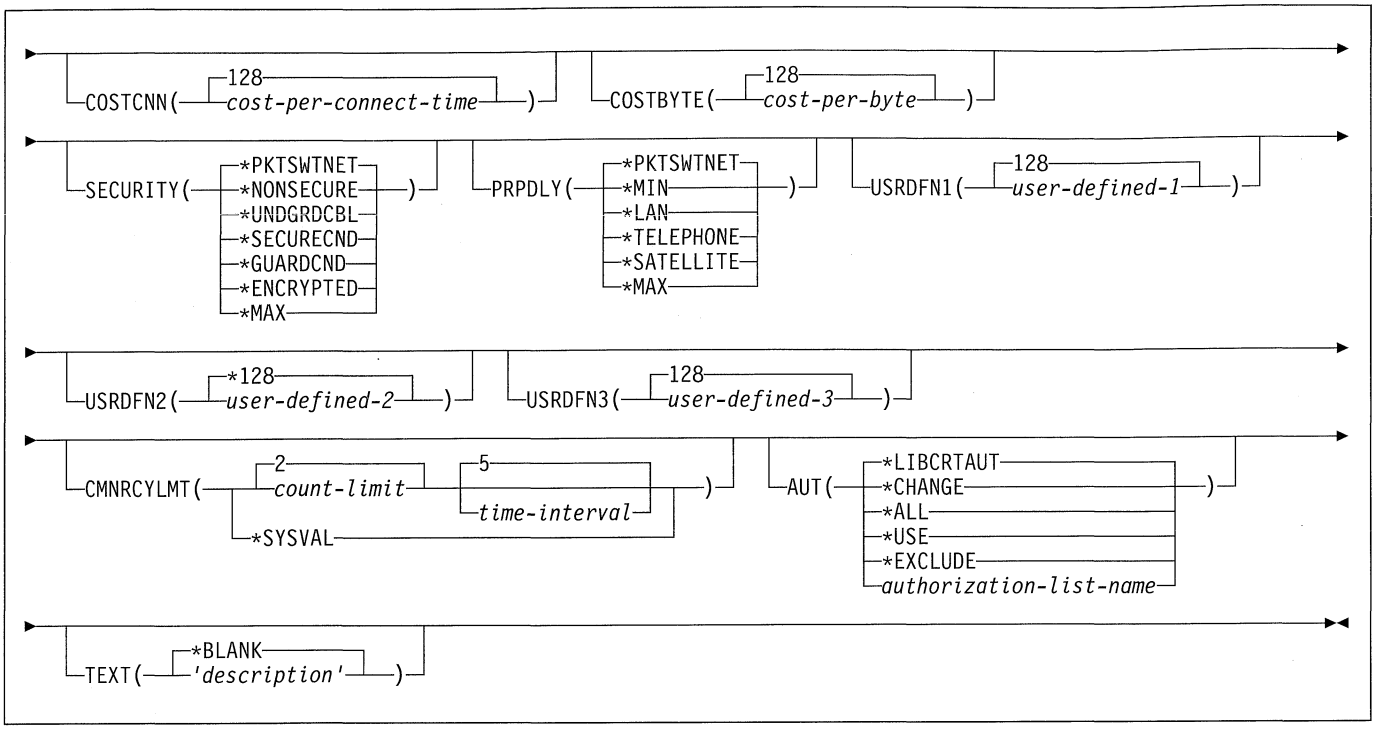
P All parameters preceding this point can be specified in positional form.

4 A maximum of 64 repetitions



Note:

[1] A maximum of 64 repetitions



Purpose

The Create Line Description (X.25) (CRTLINX25) command creates a line description for an X.25 line. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

LIND

Specifies the name of the line description being created.

RSRCNAME

Specifies the resource name that identifies the hardware that the description represents.

***NWID:** The resource name specified on the attached ISDN network interface description is used.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 2, then the resource name would be LIN012.

resource-name: Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 2, the resource name would be LIN012.

LGLCHLE

Specifies, when using the Create Line Description (X.25) CRTLINX25 or CHGLINX25 command, a list of entries to be added, removed, or changed in the logical channel table. A channel entry consists of a channel identifier, a logical channel type, and a PVC controller.

Note: Logical channel entries with attached permanent virtual circuit (PVC) controllers cannot be added, removed, or changed.

***PROMPT:** By using the *PROMPT option, any entry can be added, removed, or altered. Specifying *PROMPT shows the current logical channel entries.

Element 1: Logical Channel Identifier

logical-channel-identifier: Specify a value ranging from 001 to FFF for the logical channel identifier. The first digit (from left to right) is the logical channel group number; the second and third digits make up the logical channel number.

Element 2: Logical Channel Type

***PVC:** The logical channel is a permanent virtual circuit.

***SVCIN:** The logical channel is a switched virtual circuit for input only.

***SVCOUT:** The logical channel is a switched virtual circuit for output only.

***SVCBOTH:** The logical channel is a switched virtual circuit for both input and output.

Element 3: PVC Controller

PVC-controller: Specify (optionally) the name of the permanent virtual circuit (PVC) controller that is assigned to

the logical channel. This field is only valid when the channel type is *PVC.

NETADR

Specifies the local network address for this system. Up to 17 characters can be specified if *YES is specified for the EXTNETADR parameter. Otherwise up to 15 characters can be specified.

CNNINIT

Specifies whether the local system or the remote system initiates the X.25 Data Link connection.

***LOCAL:** The local system initiates the connection (local system sends SABM).

***REMOTE:** The remote system initiates the connection (the local system waits for SABM).

***WAIT:** The local system waits for a disconnect (DISC) or disconnect mode (DM) from the data circuit-terminating equipment (DCE) before attempting to activate the link.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The line is automatically varied on at IPL.

***NO:** This line is not automatically varied on at IPL.

INTERFACE

Specifies the type of physical interface on the input/output adapter (IOA) port. The valid interface types are:

***X21BISV24:** The X.21 bis/V.24 interface is used.

***X21BISV35:** The X.21 bis/V.35 interface is used.

***RS232V24:** The RS232/V.24 interface is used.

***X21:** The X.21 interface is used.

***X31:** The X.31 interface is used. This is the International Telegraph and Telephone Consultative Committee (CCITT) recommendation which defines the support of packet mode terminal equipment by an integrated services digital network (ISDN). Specify this value to run X.25 over an ISDN B channel.

CNN

Specifies the type of line connection used.

***NONSWTPP:** A nonswitched point-to-point line is used.

***SWTPP:** A switched point-to-point line is used.

NWI

Specifies, for a nonswitched connection, the network interface description containing the channel to which this line permanently attaches.

NWICHLTYPE

Specifies, for a nonswitched connection, the type of ISDN channels that this line description uses. This parameter is preset to use one ISDN B-channel of the network interface description.

***B:** The B channel is used.

SWTNWILST

Specifies, for ISDN switched connections, a list of network interface descriptions to which this line can be attached. A network interface description is chosen from the list based on the value specified by the switched NWI selection parameter (SWTNWISLCT) at the time an incoming or outgoing call is processed.

***NONE:** No network interface description is specified.

Element 1: Network Interface Description Name

NWI-description-name: Specify, for switched connections, the name of the network interface description to which this line attaches.

Element 2: Network Interface Channel Type

***B:** The B channel is used.

Element 3: Network Interface Channel-Number

***CALC:** The system selects one of the two channel numbers (based on availability) defined for the network interface description when an incoming or outgoing call is processed.

NWI-channel-number: Specify a channel number (1 or 2) to which the line description is restricted.

VRYWAIT

Specifies whether the line is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The line is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the line is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the line description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the line or reach the wait-time value.
2. The time required to vary on a line is the time it takes to put tasks in place to manage the line, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, line protocol, and other factors.

LINESPEED

Specifies the line speed in bits per second (bps).

9600: The line speed is 9600 bps.

***CALC:** The line speed is calculated by the system.

line-speed: Specify one of the following values (in bps) for the line speed:

```
 600     4800    19200    57600
1200     7200    48000    64000
2400    14400    56000
```

EXCHID

Specifies the hexadecimal exchange identifier that is used to identify the local system to the remote system. The 8-digit hexadecimal exchange identifier contains three digits for the block number and five digits for the identifier of this system.

***SYSGEN:** The AS/400 system generates the exchange identifier.

exchange-identifier: Specify the exchange identifier. If the *SYSGEN value is not used, an exchange identifier consisting of eight hexadecimal digits starting with 056 must be specified.

PKTMODE

Specifies whether to access the ISDN virtual circuit service.

***NO:** The ISDN network is used to provide transparent access to an X.25 packet switched network external to the ISDN (Case A).

***YES:** The ISDN virtual circuit service is accessed (Case B).

INFTRFTYPE

Specifies the information transfer type. The information transfer type determines the layer 1 protocol.

***UNRESTRICTED:** The data-channel traffic appears as digital information; no physical transformation is required and each B-channel operates at capacity (64k bps).

***V110:** The transfer type is V-series Recommendation 110. Each B-channel operates at 56k bps.

***IVTHDLC:** The transfer type is Inverted HDLC. Each B-channel operates at capacity, 64k bps.

EXTNETADR

Specifies whether network addressing is extended to permit the use of 17 characters in an address name.

***NO:** Network addresses can be up to 15 characters.

***YES:** Network addresses can be up to 17 characters.

MAXFRAME

Specifies the maximum frame size that can be transmitted and received on this line description.

1024: The default frame size is 1024.

maximum-frame-size: Specify one of the following values: 1024, 2048, 4096, 8192, or 16384.

DFTPKTSIZE

Specifies the default packet size used by the X.25 network.

Element 1: Transmit Packet Size

128: The default packet size is 128.

transmit-packet-size: Specify a default packet size for transmission to all controllers attached to this line. The controllers can override this default with the DFTPKTSIZE parameter on the controller commands. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Receive Packet Size

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception from all controllers attached to this line. The controllers can override this default with the DFTPKTSIZE parameter on the controller commands. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

MAXPKTSIZE

Specifies the maximum packet size for transmission and reception on an X.25 network. The value specified must not be less than the default packet size specified.

Element 1: Maximum Transmit Packet Size

***DFTPKTSIZE:** The maximum packet size for transmission is the same as that specified as the default packet size for transmission on the DFTPKTSIZE parameter in this command.

transmit-packet-size: Specify a default packet size for transmission to all controllers attached to this line. The valid values for the packet size are 64, 128, 256, 512, 1024, 2048, and 4096.

Element 2: Maximum Receive Packet Size

***DFTPKTSIZE:** The maximum packet size for reception is the same as that specified as the default package size for reception on the DFTPKTSIZE parameter on this command.

***TRANSMIT:** The value specified as the default packet size for transmission is used as the default for reception.

receive-packet-size: Specify a default packet size for reception from all controllers attached to this line. The controllers can override this default with the DFTPKTSIZE parameter on the controller commands.

MODULUS

Specifies the packet numbering used. The valid numbers are listed below.

8: Modulus 8 packet numbering is used.

128: Modulus 128 packet numbering is used.

DFTWDWSIZE

Specifies the default window size used by the X.25 network.

Element 1: Transmit Window Size

2: The default packet window size is 2; modulus 8 is supported.

transmit-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use modulus 128 packet numbering.

Element 2: Receive Window Size

***TRANSMIT:** The value specified as the default window size for transmission is used as the default for reception.

receive-window-size: Specify the appropriate default window size. Valid values range from 1 through 7 for networks that use modulus 8 packet numbering. Valid values range from 1 through 15 for networks that use modulus 128 packet numbering.

NETCTL

Specifies the name of an existing network controller.

SWTCTLLST

Specifies the names of up to 64 switched asynchronous X.25 controllers that can establish a connection with an X.25 switched virtual circuit (SVC). The controller descriptions must already exist, and have been created using the Create Controller Description (Async) (CRTCTLASC) command. Asynchronous X.25 controllers that are specified as dial-in can be connected to SVCIN or SVCBOTH logical channels. Attaching controllers that specify CNNNBR(*ANY) or ANSNBR(*ANY) may result in a reordering of this list.

***NONE:** No switched controllers are specified.

switched-controller-name: Specify the switched controller name.

NETUSRID

Specifies, for switched lines, whether network user identification (NUI) information is included in the facility field of call request packets sent on this line. The NUI facility identifies calling data terminal equipment (DTE) to the network.

This parameter can be specified as a single value (*NONE) or as a hexadecimal network user identification.

***NONE:** No NUI information is sent over this line.

network-user-identification: Specify the hexadecimal NUI that is sent. The system inserts the NUI facility code (X'C6') and the NUI length in the facility field of a call request packet.

IDLTMR

Specifies the maximum amount of time (in 0.1 second intervals) that the system waits for acknowledgment from the network for each frame sent before re-transmission.

Note: The IDLTMR value should be greater than or equal to the value of the following equation:

```
(2 * P + (MAXPKTSIZE*8)/LINESPEED + D) * 10
```

where "P" is the propagation delay (in seconds) of the medium that connects the user to the network, MAXPKTSIZE is the maximum transmit packet size, and "D" is the DCE (Data Circuit-terminating Equipment) processing overhead (in seconds). Contact the network provider for information about variables P and D.

40: The default value is 40 (4 seconds).

idle-timer: Specify a value ranging from 4 through 600. Each unit represents 0.1 seconds, which provides a timeout value ranging from 0.4 through 60 seconds.

FRAMERTY

Specifies the maximum number of retries attempted for various error conditions on the interface, such as link level time-outs, logical link level re-transmissions and others.

7: Seven retries are attempted.

frame-retry: Specify a value ranging from 0 through 64 retries.

ADRINSERT

Specifies whether the system inserts the local network address in CALL REQUEST or CALL ACCEPTED packets.

***YES:** The local network address is inserted in packets.

***NO:** The local network address is not inserted in packets.

X25DCE

Specifies whether the system communicates via the X.25 data circuit-termination equipment (DCE) support. This allows a system to communicate with another system without going through an X.25 network.

***NO:** The AS/400 system does not communicate via the X.25 DCE support.

***YES:** The AS/400 system communicates via the X.25 DCE support.

THRESHOLD

Specifies the temporary error threshold level being monitored by the system. A permanent error is reported only if the errors occurred consecutively and exceeded the retry limit.

Note: Specifying the THRESHOLD parameter affects all threshold errors. They cannot be specified individually.

***OFF:** No monitoring of errors occurs.

***MIN:** The error threshold is set at a minimum monitoring level.

***MED:** Error thresholding is set to a medium monitoring level.

***MAX:** The error threshold is set at a maximum monitoring level.

CNNNBR

Specifies, for switched lines, the telephone number used to establish a switched connection on this line description. The value specified on this parameter is sent to the autocall unit if automatic calling is used to

establish a switched connection, or it is sent to the V.25 bis modem if V.25 bis automatic dialing is used to establish the connection. When manual dialing is used to establish a switched connection on the line, this number is displayed to the system operator. For X.25, this number is the remote data terminal equipment (DTE) address for the switched virtual circuit (SVC) provided by the X.25 network.

CALLNBR

Specifies the local telephone number of the line used for the V.25 bis call request with identification (CRI) dial command. This parameter is used when the CRI function is needed for V.25 bis. When V.25 bis CRI dialing is used, the system takes the called (connection) number from the CNNNBR parameter of the controller description, adds a separator character (;), and concatenates the calling number at the end. Specify the calling number only if the modem and the network both support the CRI dial command.

***NONE:** The CRN (Call Request Normal) dial command is used by the V.25 bis line.

calling-number: Specify up to 32 characters that represent the local telephone number for V.25 bis CRI (Call Request with Identification) auto-dialing.

MODEM

Specifies the type of modem supported on the communications line. Refer to the modem information to select the appropriate value.

***NORMAL:** No attempt is made to run diagnostic tests on the modem.

***V54:** Certain types of diagnostic tests (as defined by the CCITT recommendations) are run to the modem. The AS/400 system supports CCITT V.54 loop 3 (local loop back) and loop 2 (a remote loop back).

***IBMWRAP:** An IBM modem with wrap test capabilities is used on the communications line.

MODEMRATE

Specifies the speed at which the line operates if the modem has the data rate select feature.

Note: The user is responsible for ensuring that the line speed corresponds to the actual modem rate.

***FULL:** The line operates at the full rate of the modem.

***HALF:** The line operates at one-half the full rate, or at the alternate rate, of the modem.

SWTCNN

Specifies, for switched lines or switched network backup lines, whether the line is used for incoming calls, for outgoing calls, or for both incoming and outgoing calls.

***BOTH:** The line is used for both incoming and outgoing calls.

***ANS:** The line is used for incoming calls only.

***DIAL:** The local system starts the call.

CNNLSTOUT

Specifies, for ISDN switched connections, the name of a connection list object that contains the ISDN assigned numbers for a dial-out operation to the ISDN.

CNNLSTOUTE

Specifies, for ISDN switched connections, the entry name from the connection list used to make a call to the ISDN. The connection list must be specified on the CNNLSTOUT parameter.

CNNLSTIN

Specifies for ISDN switched connections the name of the connection list that is used to retrieve call information (or connection) for identifying authorized incoming calls.

***NETATR:** The connection list used by this line description is taken from the list of system default network attributes that were identified at IPL (Initial Program Load). The Display Network Attributes (DSPNETA) command can be used to see the name of the connection list.

connection-list-name: Specify the name of the connection list used for this line description.

AUTOANS

Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically answers a call from a remote system to establish the connection or whether the user must manually answer the call and place the modem in data mode.

***YES:** The system automatically answers incoming calls.

***NO:** The system operator must manually answer incoming calls.

Note: *YES is valid only if the modem has the automatic answer feature.

AUTODIAL

Specifies, for switched or switched network backup (SNBU) lines, whether the system automatically calls a remote system to establish a connection or whether the system operator must manually place the call.

***NO:** The AS/400 system does not automatically call a remote system.

***YES:** The AS/400 system automatically calls a remote system.

Note: *YES is valid only if the system is using an autocall unit or if the modem used is capable of calling though a command interface.

DIALCMD

Specifies the type of dial command used to establish a switched connection with a remote system.

***NONE:** No dial command type is specified. An automatic call unit establishes the connection.

***V25BIS:** V.25 bis is a recommendation which allows the use of one physical interface for call establishment and data transmission. It is sometimes referred to as a

serial automatic call interface because the digits are presented serially on the link from the system (DTE) to the modem (DCE).

CALLIMMED
Specifies, for switched (CNN(*SWTPP)) lines, whether a call (using the number specified by the CNNNBR parameter) is made immediately after the line description is varied on.

***NO:** The call is not made immediately after the line description is varied on. Instead, it must be initiated by an application program.

***YES:** The call is made immediately after the line description is varied on.

AUTOCALL
Specifies, for switched lines, whether the line has an associated autocall unit which performs automatic calling to the remote system.

***NO:** This switched line does not have an autocall unit.

***YES:** This switched line has an autocall unit.

ACRSRCNAME
Specifies the resource name that describes the automatic call unit port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN02 and the port is 1, the resource name is LIN021.

PREDIALDLY
Specifies how long to wait (in 0.5 second intervals) before dialing.

Note: This parameter is valid only for switched lines (CNN(*SWTPP)).

6: The value of 6 provides a 3-second delay.

predial-delay: Specify a value ranging from 1 through 254 in 0.5-second intervals, or specify 0 to indicate no delay.

REDIALDLY
Specifies the length of time (in 0.5-second intervals) to wait before re-dialing when the call attempt is unsuccessful. This parameter can be specified only if CNN(*SWTPP) is also specified.

120: A 60-second delay occurs before the number to the remote system is redialed.

redial-delay: Specify the length of time before the number to the remote system is redialed, or specify 0 to indicate no delay. Valid values range from 1 through 254 (in 0.5-second intervals).

DIALRTY
Specifies the number of re-dial attempts made by the system before considering the dialing unsuccessful.

Note: Dial retries can only be specified for switched lines (CNN(*SWTPP)).

2: The default number of retries is two.

dial-retry: Specify a value ranging from 0 through 254 for the number of retries.

SWTDSC
Specifies whether the switched connection on this line is disconnected when the last switched virtual circuit (SVC) is cleared.

***YES:** The switched connection is disconnected when the last device is varied off.

***NO:** The switched connection is not disconnected when the last switched virtual circuit is cleared.

SWTDSCTMR
Specifies the settings for the switch disconnect timers that run when switched X.25 lines are disconnected from the network or remote system.

Element 1: Settings for the Minimum Connect Timer

170: The minimum connect timer keeps the connection active for a minimum of 170 seconds. This timer is started when the connection is established.

minimum-connect-timer: Specify the minimum length of time the system keeps the connection active. Valid values for the minimum connect timer range from 0 through 65535 seconds.

Element 2: Settings for the Disconnect Delay Timer

0: The system breaks the switched connection as soon as the following two conditions occur:

1. The line is idle.
2. The minimum connect timer has expired.

disconnect-delay-timer: Specify the number of seconds (ranging from 0 through 65535 seconds) the system waits before breaking the switched line connection. The system breaks the switched connection only when the following conditions occur:

1. The line is idle.
2. The minimum connect timer has expired.
3. The disconnect delay timer has expired.

DSRDRPTMR
Specifies the amount of time the system waits for the modem to exit the Data Set Ready (DSR) state before signaling an error.

6: The system waits 6 seconds in the DSR state before signaling an error.

drop-timer: Specify a value ranging from 3 through 60 seconds.

AUTOANSTYP
Specifies the method the system uses to answer incoming calls.

***DTR:** The system enters the Data Terminal Ready state, signals the modem to answer calls, and waits for the modem to enter the Data Set Ready (DSR) state.

***CDSTL:** The system enters the Connect Data Set to Line (CDSTL) state after monitoring the Ring Indicator to signal the modem to answer the call.

CTSTMR

Specifies the amount of time the system waits for the modem to enter or exit the Clear to Send (CTS) state before signaling an error.

25: The system waits up to 25 seconds for the CTS state to begin or end.

CTS-timer: Specify a value ranging from 10 through 60 seconds.

RMTANSTMR

Specifies the amount of time the system waits for the modem to enter the Data Set Ready (DSR) state after dialing before signaling an error.

60: The system waits 60 seconds before signaling an error.

answer-timer: Specify a value ranging from 30 through 120 seconds.

LINKSPEED

Specifies the link speed in bits per second (bps). This parameter is valid only if APPN is used on the system.

INTERFACE: The following link speeds, based on the physical interface type, are used: 9600 bps for RS-232/V.24 and X.21bis/V.24, 48000 bps for V.35 and X.21bis/V.35, and 64000 bps for X.21 and RS-449/V.36.

***MIN:** A link speed of less than 1200 bps is used.

***MAX:** A link speed greater than 16M bps is used.

link-speed: Specify the link speed. Valid values are: 1200, 2400, 4800, 7200, 9600, 14400, 19200, 48000, 56000, 64000, 112000, 128000, 168000, 192000, 224000, 256000, 280000, 320000, 336000, 384000, 448000, 499000, 576000, 614000, 691000, 768000, 845000, 922000, 998000, 1075000, 1152000, 1229000, 1382000, 1536000, 1690000, 1843000, 1997000, 2048000, 4M, 10M, and 16M.

SWTNWISLCT

Specifies the method used to select network interfaces from the switched network interface list.

***FIRST:** Selection begins with the first network interface specified in the switched network interface list.

***CALC:** The system calculates which network interface is selected.

COSTCNN

Specifies the relative cost of being connected on the line. This parameter is required only if APPN is used on the system.

128: The cost per connect time is 128.

cost-per-connect-time: Specify a value ranging from 0 through 255.

COSTBYTE

Specifies the relative cost per byte for sending and receiving data on the line. This parameter is required only if APPN is used on the system.

128: The cost per byte is 128.

cost-per-byte: Specify a value ranging from 0 through 255.

SECURITY

Specifies the security level of the physical line. This parameter is used only if APPN is used on the system.

***PKTSWTNET:** A packet switched network is used. Data does not always follow the same path through the network.

***NONSECURE:** Normal priority is used.

***UNDGRDCBL:** An underground cable is used.

***SECURECND:** A secure, unguarded conduit (for example, a pressurized pipe) is used.

***GUARDCND:** A guarded conduit, which is protected against physical tapping, is used.

***ENCRYPTED:** Data flowing on the line is encrypted.

***MAX:** A guarded conduit, protected against physical and radiation tapping, is used.

PRPDLY

Specifies the level of propagation delay on the line. This parameter is valid only if APPN is used on the system. The order of the values from shortest to longest delay is *MIN, *LAN, *TELEPHONE, *PKTSWTNET, and *SATELLITE.

***PKTSWTNET:** The packet switched network propagation delay is used.

***LAN:** The local area network propagation delay is used.

***MIN:** The minimum propagation delay is used.

***TELEPHONE:** The telephone propagation delay is used.

***SATELLITE:** The satellite propagation delay is used.

***MAX:** The maximum propagation delay is used.

USRDFN1

Specifies the first of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-1: Specify a value ranging from 0 through 255.

USRDFN2

Specifies the second of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-2: Specify a value ranging from 0 through 255.

USRDFN3

Specifies the third of the three user-defined fields. This field is used to describe unique characteristics of the line that is controlled. This parameter is valid only if APPN is used on the system.

128: The value 128 is used.

user-defined-3: Specify a value ranging from 0 through 255.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the line description, who are not on an authorization list, and whose user group has no specific authority to the line description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the line description.

***USE:** The user can perform basic operations on the line description, such as running a program or reading a file. The user cannot change the line description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

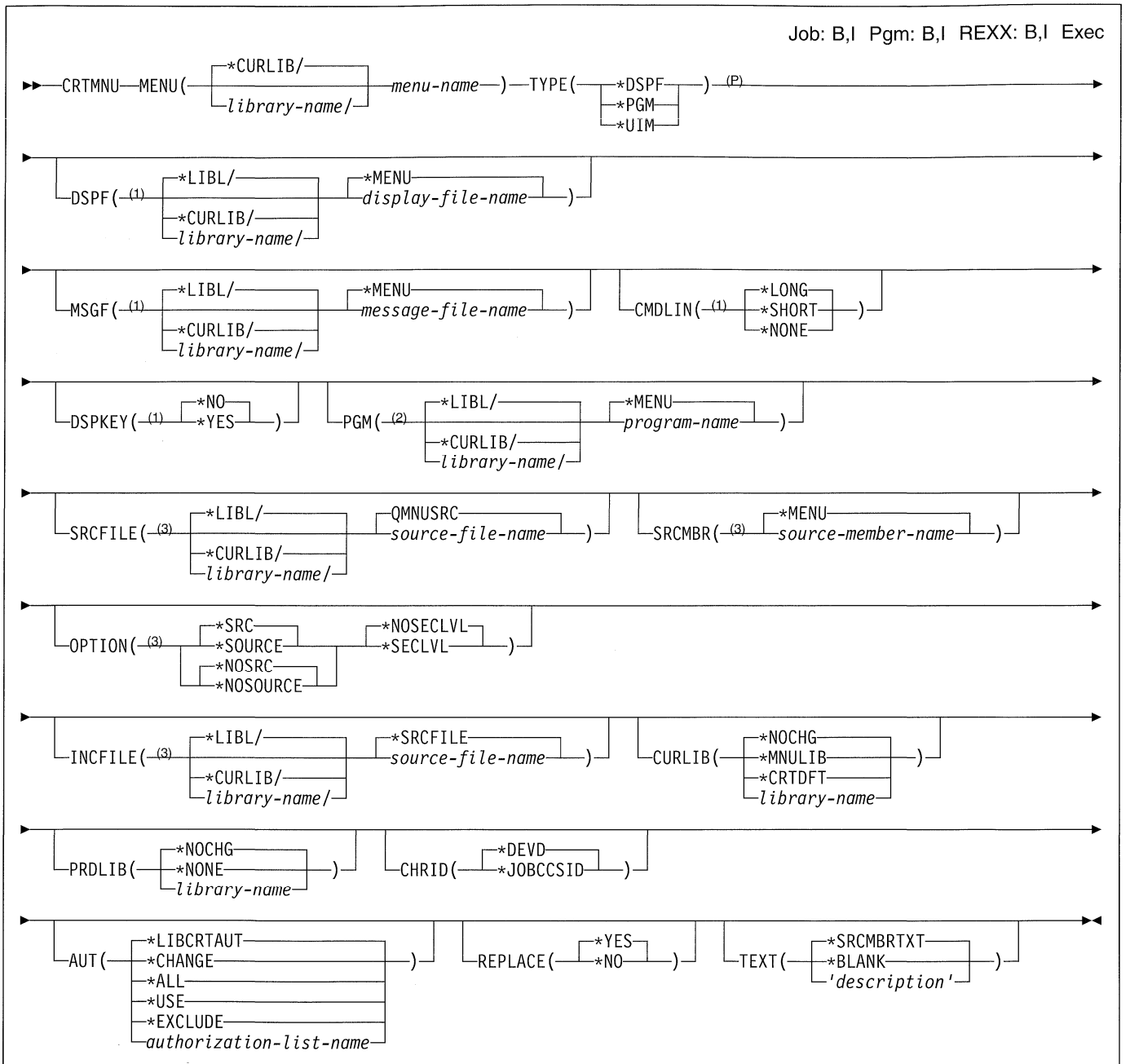
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTLINX25   LIND(X251)  RSRCNAME(LIN011)
  LGLCHLE((111 *PVC CTL1) (222 *SVCIN))
  NETADR(12345)  CNNINIT(*LOCAL)  TEXT('X.25 Line')
```

This command creates an X.25 line (X251) with resource name LIN011, two logical channels (with an attached PVC controller), a network address of 12345, and local connection initiation.

CRTMNU (Create Menu) Command



Notes:

1 The DSPF, MSGF, CMDLIN, and DSPKEY parameters are valid only when TYPE(*DSPF) is specified.

2 The PGM parameter is only valid when TYPE(*PGM) is specified.

3 The SRCFILE, SRCMBR, OPTION, INCFILE, and CHRID parameters are valid only when TYPE(*UIM) is specified.

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Menu (CRTMNU) command creates a menu object. Both Display File (*DSPF) and Program (*PGM) menus can be created by using this command. A menu can be shown by using the Go to Menu (GO) command.

Restriction: The user must have *CHANGE, *READ, and *ADD authority for the library where the menu is being created.

Required Parameters

MENU

Specifies the qualified name of the library where the menu is being created.

The name of the menu can be qualified by one of the following library values:

***CURLIB:** The menu is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the menu is created.

menu-name: Specify the name of the menu.

TYPE

Specifies the type of menu being created.

***DSPF:** An existing display file and message file are used to create a menu like those used in the System/36 environment.

***PGM:** The menu being created calls a program to be run.

***UIM:** The menu is created using the UIM tag language found in the file specified on the SRCFILE and SRCMBR parameters.

Optional Parameters

DSPF

Specifies the qualified name of the display file to use in creating the menu object. The display file must include one record format with the same name as the display file itself; this is called the menu format. Help formats may also be included in the file.

Help formats follow the naming convention #Hxxyy, where xx is the first and yy is the last menu option to which the help format applies. For example, #H0306 would apply to menu options 3 to 6. #H0000 designates the general help for the menu.

The display file must have a separate indicator area (INDARA keyword) and contain no subfile descriptions.

This parameter may be specified only if TYPE(*DSPF) is chosen.

The name of the display file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***MENU:** The display file has the same name as the menu specified on the MENU parameter.

display-file-name: Specify the display file name to use if it is not the same as that specified on the MENU parameter.

MSGF

Specifies the qualified name of the message file that contains the commands to run when a menu option is selected. The MSGIDs of the messages in this file are of the form USRxxxx where xxxx is the menu option number typed on the command line.

Note: If the message file being created is to be used for menus, you must add message file members to the file using the Add Message Description (ADDMSGD) command.

The name of the message file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***MENU:** The message file that contains the commands to run has the same name as the menu specified on the MENU parameter.

message-file-name: Specify the message file name to use if it is not the same as that specified on the MENU parameter.

CMDLIN

Specifies that the user can select a long command line, a short command line, or no command line (an option line).

***LONG:** A 153-byte long command line is used.

***SHORT:** A 73-byte long command line is used.

***NONE:** No command line is used. A 4-byte option line is used.

DSPKEY

Specifies whether the function key legend is shown at the bottom of the menu display.

***NO:** The function key legend is not shown at the bottom of the menu display.

***YES:** The function key legend is shown at the bottom of the menu display.

PGM

Specifies the qualified name of the program to call when the menu is run.

Three parameters are passed to the program:

- The first parameter is the ten-character menu object name that identifies the program to call.
- The second parameter is the ten-character name of the library that contains the menu object.
- The third parameter is a two-character binary return code declared as a variable in the called program. The program must set one of the following return codes:

```
Return
 Code   Hex     Description

   0    0000    Call the program
                (display the menu) again
  -1    FFFF    Exit function requested
  -2    FFFE    Previous function requested
  -4    FFFC    Home function requested
                (display the home menu)
```

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***MENU:** The called program has the same name as the menu specified on the MENU parameter.

program-name: Specify the name of the program to call if it is not the same as that specified on the MENU parameter.

SRCFILE

Specifies the name of the source file containing the menu description source statements.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QMNUSRC: The source file QMNUSRC contains the menu description source statements.

source-file-name: Specify the name of the source file containing the menu description source statements.

SRCMBR

Specifies the member of the source file containing the menu description.

***MENU:** The name specified on the MENU parameter is used for the name of the source member containing the menu description.

source-member-name: Specify the name of the member containing the menu description.

OPTION

Specifies whether a source listing is produced when the menu is created.

Element 1: Source Listing

***SRC:** A source listing is produced.

***SOURCE:** A source listing is produced.

***NOSRC:** No source listing is produced unless errors are detected.

***NOSOURCE:** No source listing is produced unless errors are detected.

Element 2: Second Level Text

***NOSECLVL:** Second level text is not provided with the first level text when the messages are printed at the end of the listing.

***SECLVL:** Second level text is provided with the first level text when the messages are printed at the end of the listing.

INCFILE

Specifies the name of the source file containing the members being included in the menu.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***SRCFILE:** The name specified on the SRCFILE parameter is used.

source-file-name: Specify the name of the source file containing the members to be included on the menu.

Note: If the coded character set identifier (CCSID) of the source file is different than the CCSID of the primary source file specified on the SRCFILE parameter, the CCSID is changed to the CCSID of the primary source file. The CCSID must be the same for all source members of the menu object.

CURLIB

Specifies the name of the library being used as the current library during the processing of this command.

***NOCHG:** The current library for the job does not change for the processing of this menu.

***MNULIB:** The current library for the job is changed to the library containing the menu before it is run. After the program exits the menu, the current library for the job is restored to the value it had before the menu was run.

***CRTDFT:** There is no current library for the job when the menu is run. The current library for the job that is active before the menu is run (if any) is restored after the program exits the menu.

Note: After the program exits the menu, the current library for the job is restored to the value it had before the menu was run.

library-name: Specify the name of the library that replaces the current library entry in the job's library list.

PRDLIB

Specifies the name of the library used as the product library for running the menu.

***NOCHG:** The product library does not change when the menu is run.

***NONE:** The product library entry in the library list is not used while the menu is run.

library-name: Specify the name of the library used as the product library when the menu is run.

Note: After exiting the menu, the product library is restored to the value it had before the menu was run.

CHRID

Specifies whether the character identifier (graphic character set and code page) of the menu object is changed to the CCSID value of the job when the command is run.

***DEVD:** No change occurs. The character identifier of the menu object is the same as the character identifier of the device.

***JOBCCSID:** The character identifier of the menu object is changed from the CCSID of the source file specified on the SRCFILE parameter to the character identifier of the device.

AUT

Specifies the authority given to users who do not have specific authority to the menu, who are not on an authorization list, and whose user group has no specific authority to the menu.

***LIBCRTAUT:** The public authority for the menu is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the menu). The public authority is determined when the menu is created. If the CRTAUT value for the library changes after the menu is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the menu.

***USE:** The user can perform basic operations on the menu, such as running a program or reading a file. The user cannot change the menu. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the menu.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether an existing menu with the same name is replaced.

***YES:** The existing menu is moved to the system library QRPLOBJ and replaced with the new menu.

***NO:** The existing menu is not replaced with the new menu.

TEXT

Specifies text that briefly describes the menu. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text for the menu is obtained from the text associated with the source file member.

***BLANK:** Text is not specified.

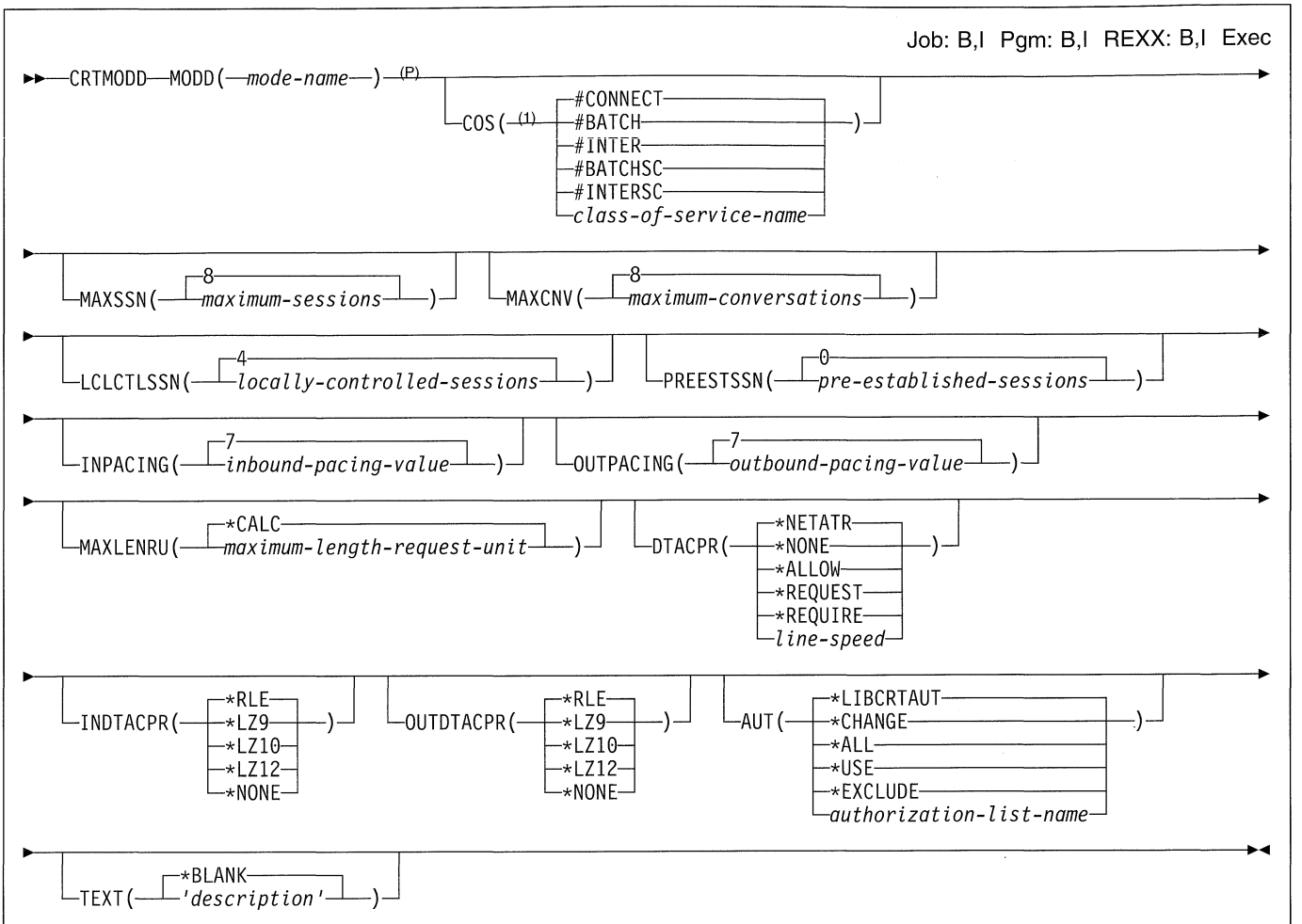
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTMNU   MENU(ARLIB/ARPERS)  TYPE(*PGM)
```

This command creates a menu named ARPERS in library ARLIB. The menu calls a program (also named ARPERS) when the menu is run.

CRTMODD (Create Mode Description) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 This parameter is ignored if the communications line is not using APPN.

Purpose

The Create Mode Description (CRTMODD) command creates a mode description.

Required Parameter

MODD

Specifies the name of the mode description.

Optional Parameters

COS

Specifies the path control network characteristics used by advanced peer-to-peer networking (APPN).

Note: This parameter is ignored if the communications line is not using APPN.

class-of-service-name: Specify the class-of-service name in the mode description. Valid class-of-service names are shown below:

- #CONNECT
- #BATCH
- #INTER
- #BATCHSC
- #INTERSC

MAXSSN

Specifies the maximum number of active sessions that can be established for this mode. This number must be greater than or equal to the number of locally controlled sessions on the local system (as specified in the LCLCTLSSN parameter) plus the number of locally controlled sessions specified at the remote location.

8: The maximum number of active sessions is eight.

maximum-sessions: Specify a value, ranging from 1 through 512, for the maximum number of active sessions.

MAXCNV

Specifies the maximum number of active conversations allowed by the mode. The maximum number of active conversations is the sum of synchronous and asynchronous conversations; this value must be greater than or equal to the value specified by the MAXSSN parameter. A synchronous conversation is a conversation in which both the source and the target programs are communicating. An asynchronous conversation is a conversation in which the source program has detached from the conversation, but there is still data to be read by the target program.

8: Up to eight conversations are allowed.

maximum-conversations: Specify a value, ranging from 1 through 512, for the maximum number of conversations.

LCLCTLSSN

Specifies the minimum number of concurrent locally controlled sessions requested for this mode. This value must be less than or equal to the value specified in the MAXSSN parameter.

4: The minimum number of locally controlled sessions is 4.

locally-controlled-sessions: Specify a value, ranging from 0 through 512, for the number of locally controlled sessions.

PREESTSSN

Specifies the maximum number of locally controlled sessions established when the mode is started. Additional sessions are established when required up to the value specified in the MAXSSN parameter; this value must be less than or equal to the value specified in the LCLCTLSSN parameter.

0: No sessions are established when the mode is started.

pre-established-sessions: Specify a value, ranging from 0 through 512, used to specify the maximum number of concurrent locally controlled sessions established when the mode is started.

INPACING

Specifies the Systems Network Architecture (SNA) pacing value used to schedule the incoming request/response units (RUs).

7: A value of 7 is used as the RU pacing value.

inbound-pacing-value: Specify a value, ranging from 0 through 63, as the limiting value.

OUTPACING

Specifies the SNA pacing value used for outgoing request/response units (RUs).

7: A value of 7 is used as the RU pacing value.

outbound-pacing-value: Specify a value, ranging from 0 through 63, for the limiting value.

MAXLENRU

Specifies the maximum request unit (RU) length (in bytes) allowed.

***CALC:** The system determines the value to use.

maximum-length-request-unit: Specify a value, ranging from 241 through 16384 bytes, for the maximum length of incoming request units. *CALC is the recommended value. Other common values are:

SDLC lines:	256, 512, 1024, 2048
Token-Ring Network lines:	256, 512, 1024, 1985
X.25 (QLLC) lines:	247, 503, 1015
X.25 (ELLC) lines:	241, 497, 1009

More information about this parameter is in the *Communications: Operating System/400* Communications Configuration Reference*.

DTACPR

Specifies whether data compression is used.

***NETATR:** The value from the DTACPR network attributes is used.

***NONE:** Compression is not allowed on the session.

***ALLOW:** Data compression is allowed on the session by the local system if requested by a remote system. The local system does not request compression.

If data compression is requested by the remote system, the data compression levels used by the session are the lower of the requested levels and the levels specified on the INDTACPR and OUTDTACPR parameters.

***REQUEST:** Data compression is requested on the session by the local system. However, the request can be refused or changed to lower compression levels by the remote system. Data compression is allowed on the session if requested by the remote system. The requested compression levels for inbound and outbound data are the levels specified for the INDTACPR and OUTDTACPR parameters.

If data compression is requested by the remote system, the data compression levels used by the session are the lower of the requested levels and the levels specified on the INDTACPR and OUTDTACPR parameters.

***REQUIRE:** Data compression is required on the session. If the remote system does not accept the local system's exact required levels of compression, the session is not established.

The data compression levels that the local system require are the levels specified on the INDTACPR and OUTDTACPR parameters.

line-speed: Specify the maximum line speed at which data is compressed. If the line speed of the link used by the session is less than or equal to this specified line speed, data compression is used for the session as if

*REQUEST is specified. Otherwise, compression is used for the session as if *ALLOW is specified. Valid values range from 1 through 2147483647 in bits per second (bps).

INDTACPR

Specifies the desired level of compression for inbound data. No data compression occurs if DTACPR(*NONE) is specified.

Note: Adaptive dictionary-based compression is a dynamic compression algorithm, similar to Lempel-Ziv, that compresses previously seen strings to 9-, 10-, and 12-bit codes. This algorithm is referred to as LZ in the following parameters.

***RLE:** The Run Length Encoding (RLE) algorithm is used. RLE substitutes a 1- or 2-byte sequence in the data stream for each repeated occurrence of the same character. This algorithm requires no storage and less processing time than the other options.

***LZ9:** The LZ algorithm with the 9-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ9 requires the least storage and processing time of the LZ algorithms; however, it compresses the data stream the least.

***LZ10:** The LZ algorithm with the 10-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ10 table algorithm requires more storage and processing time than the LZ9, but less than the LZ12. The LZ10 compresses the data stream more than the LZ9, but less than the LZ12.

***LZ12:** The LZ algorithm with the 12-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ12 requires the most storage and processing time of the LZ algorithms; however, it compresses the data stream the most.

***NONE:** No compression occurs.

OUTDTACPR

Specifies the desired level of compression for outbound data. No data compression occurs if DTACPR(*NONE) is specified.

***RLE:** The Run Length Encoding (RLE) algorithm is used. RLE substitutes a 1- or 2-byte sequence in the data stream for each repeated occurrence of the same character. This algorithm requires no storage and less processing time than the other options.

***LZ9:** The LZ algorithm with the 9-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ9 requires the least storage and processing time of the LZ algorithms; however, it compresses the data stream the least.

***LZ10:** The LZ algorithm with the 10-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ10 table algorithm requires more storage and processing time than the LZ9, but less than the LZ12. The LZ10 compresses the data stream more than the LZ9, but less than the LZ12.

***LZ12:** The LZ algorithm with the 12-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ12 requires the most storage and processing time of the LZ algorithms; however, it compresses the data stream the most.

***NONE:** No compression occurs.

AUT

Specifies the authority given to users who do not have specific authority to the mode description, who are not on an authorization list, and whose user group has no specific authority to the mode description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the mode description.

***USE:** The user can perform basic operations on the mode description, such as running a program or reading a file. The user cannot change the mode description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the mode description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the mode description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

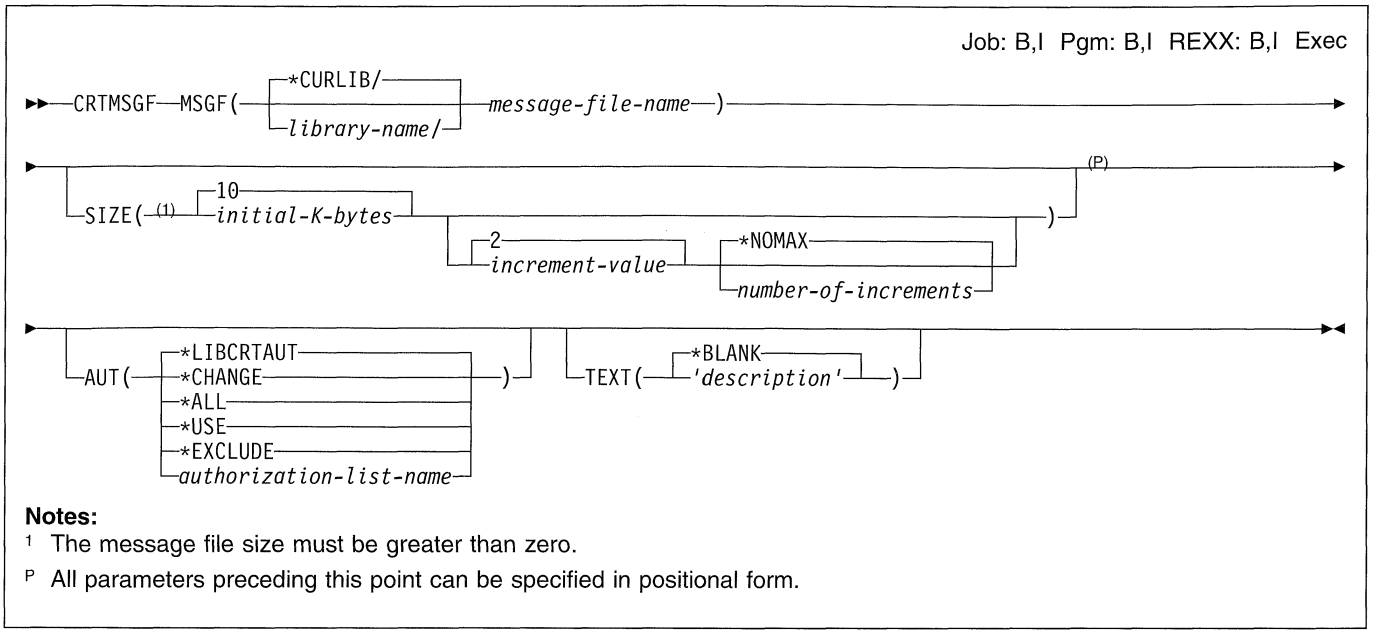
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTMODD   MODD(MODE1)  COS(COSD1)
  TEXT('Mode using COSD1 Class-of-Service')
```

This command creates a mode, MODE1, that specifies class-of-service description COSD1.

CRTMSGF (Create Message File) Command



Notes:
1 The message file size must be greater than zero.
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Message File (CRTMSGF) command creates a user-defined message file for storing message descriptions. The message file should be stored in a library for which all users who use the predefined messages have *USE authority. The system is shipped with the IBM-supplied message files, stored in the system library, QSYS: the CPF message file, QCPFMSG (for the OS/400 system and machine interface messages); and the licensed program message files, such as QRPGMSG (for RPG messages).

Required Parameter

MSGF

Specifies the qualified name of the message file being created.

The name of the message file can be qualified by one of the following library values:

***CURLIB:** The message file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the message file is created.

message-file-name: Specify the name of the message file being created.

Optional Parameters

SIZE

Specifies the initial storage size of the message file, the size of each increment added to increase its storage size, and the number of times the storage size can be incremented. The storage size is expressed in kilobytes (Kbytes). The message file size is increased when source file space is needed for the addition of a message description. The minimum size allowed is 1 Kbyte, the maximum allowed is 16,000 Kbytes (16 Megabytes). If SIZE is not specified, SIZE(10 2 *NOMAX) is assumed.

Element 1: Initial Size

One of the following values is used to specify the initial storage size of the message file.

10: Initially, the message file has 10 Kbytes of storage assigned to it (1 Kbyte equals 1024 bytes of storage).

initial-K-bytes: Specify the initial size of the file, which cannot equal 0.

Element 2: Increment Value

One of the following values is used to specify the amount of storage in kilobytes that is added each time the message file's size is incremented.

2: The message file size is increased by 2 Kbytes of storage for each increment added.

increment-value: Specify the number of kilobytes added for each increment.

Element 3: Maximum Number of Increments

One of the following values is used to specify the maximum number of increments that can be added to increase the message file's size.

***NOMAX:** The system maximum is used.

number-of-increments: Specify the maximum number of increments that can be added to increase the message

file size. Enter a 0 to prevent any increases in the initial size of the file.

AUT

Specifies the authority given to users who do not have specific authority to the message file, who are not on an authorization list, and whose user group has no specific authority to the message file.

***LIBCRTAUT:** The public authority for the message file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the message file). The public authority is determined when the message file is created. If the CRTAUT value for the library changes after the message file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove user ids.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the message file.

***USE:** The user can perform basic operations on the message file. The user is prevented from changing the message file. Use authority provides object operational and read authority.

***EXCLUDE:** The user cannot access the message file.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the message file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

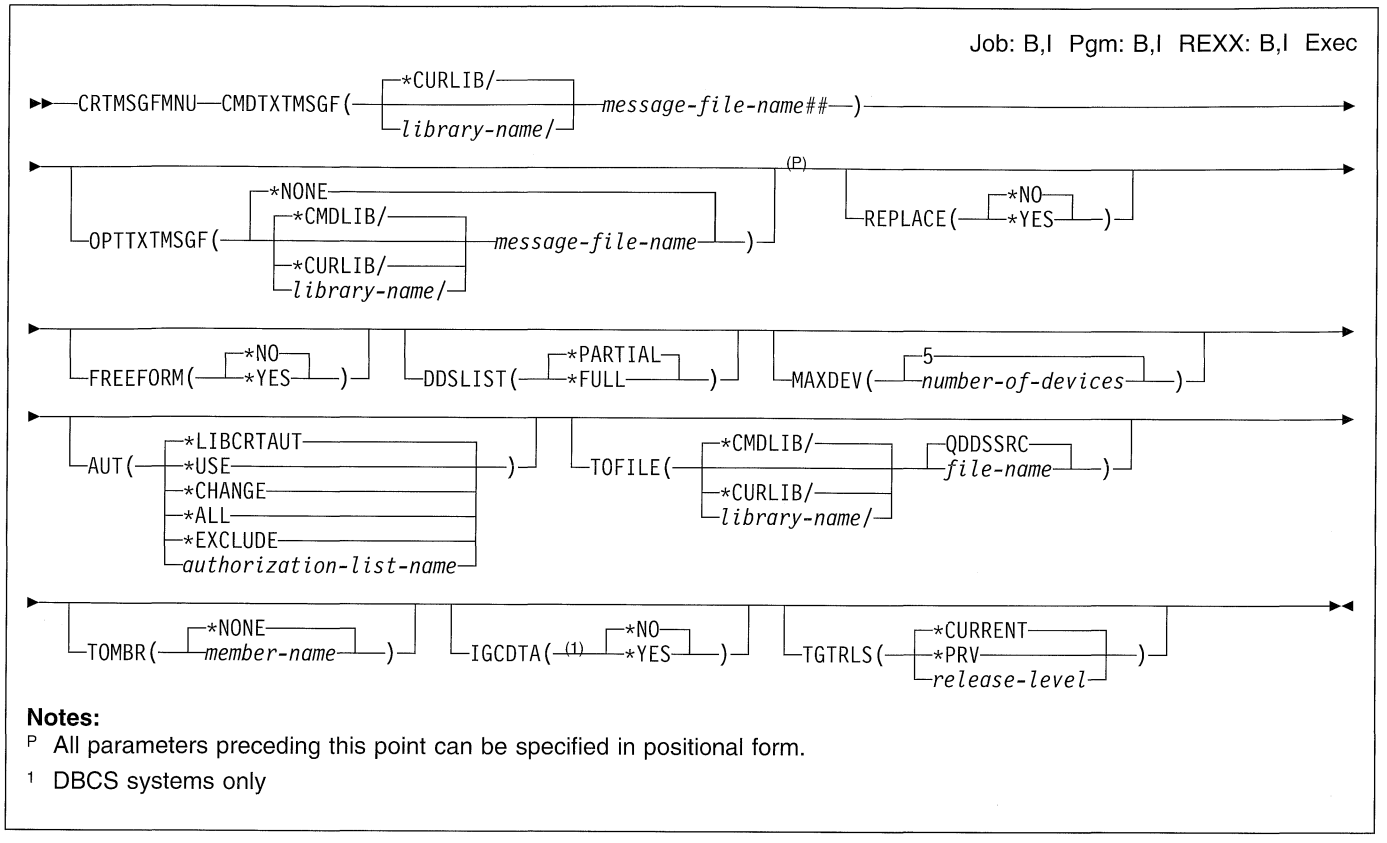
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTMSGF   MSGF(INVLIB/INVMSGS)
  TEXT('Inventory Application Messages')
```

This command creates a message file named INVMSGS in which predefined inventory application messages are stored. The file is stored in the library INVLIB, for which all users of the file must have *USE authority. Because the AUT parameter is defaulted, all users have *CHANGE authority for the file, meaning they can retrieve messages from the file.

CRTMSGFMNU (Create Message File Menu) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 DBCS systems only

Purpose

The Create Message File Menu (CRTMSGFMNU) command creates a menu (display file) from the specified message files. The user can specify that this menu be created either in a fixed-format, with options 1 through 24 arranged in two columns, or in free-format.

To create a menu from the source member, use the Create System/36 Menu (CRTS36MNU) command.

Restriction: The CRTMSGFMNU command can be run either in the System/36 environment or, when library QSSP is installed, on the AS/400 system.

Required Parameter

CMDTXTMSGF

Specifies the qualified name of the command text message file that contains the text for the command that runs when the corresponding option is selected. The CMDTXTMSGF parameter must identify an existing message file and not a Screen File Generator (SFGR), or a System/36 message source member. Trailing ## symbols are required on the CMDTXTMSGF name. The menu (display file) name is the message-file-name without the ## symbols. The run-time menu processor appends the trailing ## symbols to the menu name to determine the message file name. This is the name of the message file that contains messages whose text is the command that is run for any option selected.

The name of the command text message file can be qualified by one of the following library values:

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

Note: The CMDTXTMSGF library is where the menu display file is created. This is a requirement of the run-time menu processor.

message-file-name-##: Specify the name of the message file used to create the menu. Trailing ## symbols are required.

Optional Parameters

OPTTXTMSGF

Specifies the qualified name of the option text message file that contains the text that is displayed on the menu to describe the options that can be selected.

The OPTTXTMSGF parameter must identify an existing message file and not a message source member.

***NONE:** No option text message file is used. The CMDTXTMSGF parameter is used to specify the option text.

The name of the option text message file can be qualified by one of the following library values:

***CMDLIB:** The command library is used to locate the option text message file.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

message-file-name: Specify the name of the option text message file used for the descriptions of the options on the menu being created. The message ID prefix of USR is required unless IGCDTA(*YES) is specified.

REPLACE

Specifies whether the existing display file is replaced by the new file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NO:** No replacement occurs.

***YES:** The existing display file is replaced by the one being created.

Note: A display file is not created if a program, message file, or any other non-display file already exists in the menu library. Only one System/36 "load member" of a given name may exist in any given library.

FREEFORM

Specifies that the menu is created either in free-format or in a fixed-format, using two columns with twelve items in each column.

***NO:** Free format is not used. The menu is created by using a fixed format with two-columns.

***YES:** The menu is created using free-format.

DDSLIST

Specifies whether a partial or full DDS compiler listing is provided.

***PARTIAL:** A partial compiler listing is provided.

***FULL:** A full DDS compiler listing and cross-reference are provided.

MAXDEV

Specifies the maximum number of devices that can use the menu at one time.

5: The maximum number of devices is five.

number-of-devices: Specify the maximum number of devices that can use the menu at one time. The possible values range from 1 through 256.

AUT

Specifies the authority given to users who do not have specific authority to the menu object, who are not on an authorization list, and whose user group has no specific authority to the menu object.

***LIBCRTAUT:** The public authority for the menu object is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the menu object). The public authority is determined when the menu object is created. If the CRTAUT value for the library changes after the menu object is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the menu object, such as running a program or reading a file. The user cannot change the menu object. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the menu object.

***EXCLUDE:** The user cannot access the menu object.

authorization-list-name: Specify the name of the authorization list used.

TOFILE

Specifies the qualified name of the source file (to-file) in which to store the data description specifications (DDS) source that was used to create the object. If the file does not exist, the requester must be authorized to the Create Source Physical File (CRTSRCPF) command so it can be created. This parameter is ignored if TOMBR(*NONE) is specified.

Note: See special case for TOMBR when QS36DDSSRC is specified.

The name of the DDS source file can be qualified by one of the following library values:

***CMDLIB:** The command library is used to locate the source file.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The IBM-supplied source file, QDDSSRC, is used.

file-name: Specify the name of the source file in which to store the DDS source.

TOMBR

Specifies the name of the source file member in which to store the data description specifications (DDS) source. If the member does not exist, it is created. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

***NONE:** The DDS source is not stored in the to-file.

member-name: Specify the name of the source file member in which to store the DDS source.

IGCDTA

Specifies whether the file contains double-byte character set (DBCS) data.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

TGTRLS

Specifies the release level of the operating system on which you intend to use the object being created.

The release level is specified in the format VxRxMx, where Vx is the version, Rx is the release, and Mx is the modification level. The release level V2R3M0 is Version 2, Release 3, Modification 0.

Note: To use the object on the target system, you must save the object to the target release level specified on the create command and then restore it on the target system.

If your system is running V2R3M0 and you want to create a program object for distribution to a V2R2M0 system you must:

1. Create the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
2. Save the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
3. Restore the program on the V2R2M0 system. The program object can also be restored to a V2R3M0 or later system.

***CURRENT:** The object is to be used on the release of the operating system currently running on your system. If V2R3M0 is running on the system, *CURRENT means that you intend to use the object on a system with V2R3M0 installed. You also can use the object on a system with any later release of the operating system installed.

***PRV:** The object is to be used on the previous release with modification level 0 of the operating system. If V2R3M0 is running on the system, *PRV means that you intend to use the object on a system with V2R2M0 installed. You also can use the object on a system with any later release of the operating system installed.

release-level: Specify the release level in the format VxRxMx. The object can be used on a system with the specified release or with any later release of the operating system installed.

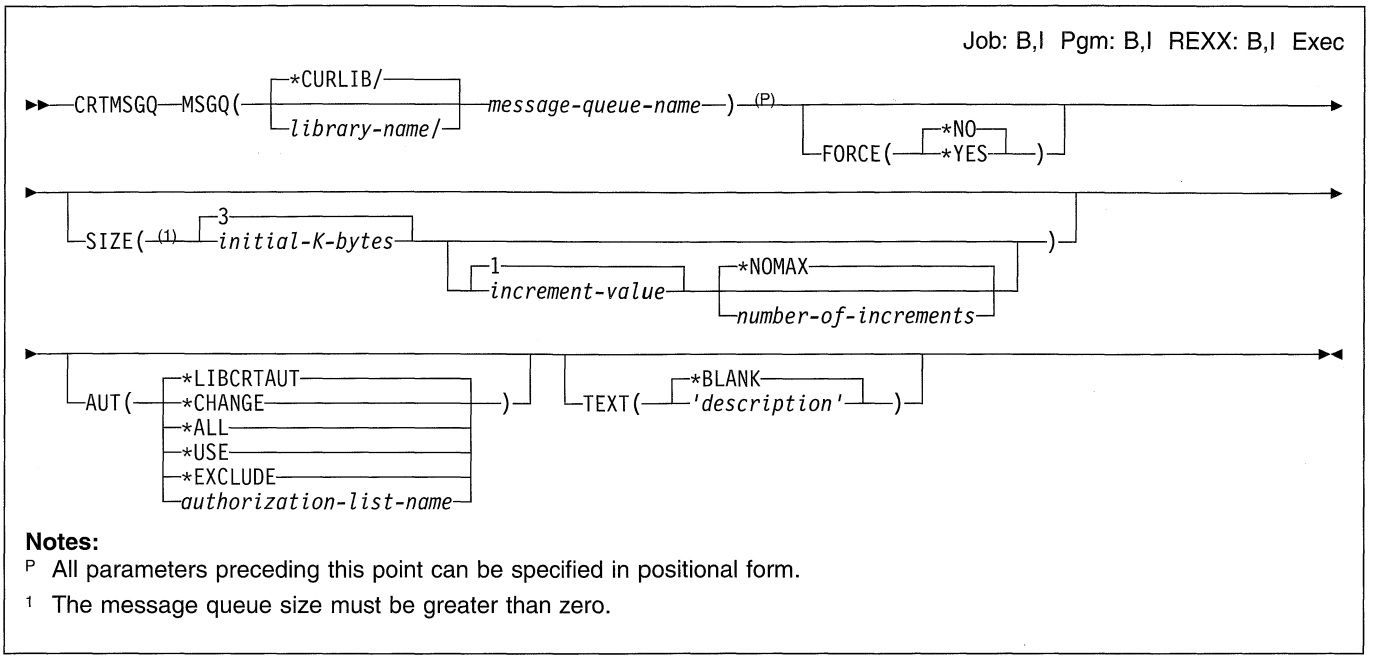
Valid values depend on the current version, release, and modification level, and they change with each new release. Valid values for release V2R3M0 are: V2R3M0, V2R2M0, and V2R1M1.

Example

```
CRTMSGFMNU   CMDTXTMSGF(MYMENU##)
  REPLACE(*YES)  FREEFORM(*NO)
```

This command creates a menu by using a message file named MYMENU. The message file is located in the current library for the job and it contains the commands run for each menu option. Because no OPTTXTMSGF parameter is specified, the text of the command appears on the screen in place of the option text. REPLACE(*YES) specifies that an existing display file is replaced. The created display file is in a fixed-format, with options 1 through 24 arranged in two 12-element columns.

CRTMSGQ (Create Message Queue) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 The message queue size must be greater than zero.

Purpose

The Create Message Queue (CRTMSGQ) command creates a user-defined message queue and stores it in a specified library. The message queue should be put in a library for which all users who are sending messages to and receiving messages from the queue have *USE authority. The messages sent can be either predefined messages or impromptu messages. The message queue has the following attributes initialized when it is created: the DLVRY parameter is set to *HOLD, PGM is set to *DSPMSG, SEV is set to 00, and RESET is set to *NO. These initialized attributes cannot be specified on the CRTMSGQ command and the CHGMSGQ command must be used to change them after the queue is created.

Note: When a message queue fills up with messages, no additional messages can be sent to the queue. A user can either accept the default of *NOMAX for the maximum number of times the message queue can be extended, or create the queue large enough to avoid a space problem. System functions, such as document interchange, make use of this queue for delivering messages.

Required Parameter

MSGQ

Specifies the qualified name of the message queue being created.

The name of the message queue can be qualified by one of the following library values:

***CURLIB:** The message queue is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the message queue is created.

message-queue-name: Specify the name of the message queue being created.

Optional Parameters

FORCE

Specifies whether changes made to the message queue description or messages added to or removed from the queue are immediately forced into auxiliary storage. This ensures that changes to the queue, or messages sent or received, are not lost if a system failure occurs.

***NO:** Changes made to the message queue, including its messages, are not immediately forced to auxiliary storage.

***YES:** Changes to the message queue description and to the messages in the queue are immediately forced to auxiliary storage.

SIZE

Specifies the initial storage size of the message queue, the size of each increment added to its storage, and the number of times an increment of the specified the size can be added. The storage size is expressed in kilobytes (K). The message queue size is increased when a message is sent to the message queue and there is not enough room for it in the queue. If SIZE is not specified, SIZE(3 1 *NOMAX) is assumed.

Element 1: Initial Size

One of the following is used to specify the initial storage size of the message queue.

3: Initially, the message queue has 3K of storage assigned to it (1K equals 1024 bytes of storage).

initial-K-bytes: Specify the value that specifies the initial size of the queue. The specified value cannot equal 0.

Element 2: Increment Value

One of the following is used to specify the amount of storage in kilobytes added to the message queue's size.

1: The message queue size is increased by 1K of storage for each increment added.

increment-value: Specify the number of kilobytes added for each increment.

Element 3: Maximum Number of Increments

One of the following is used to specify the maximum number of storage space increments that can be added to the message queue's size.

***NOMAX:** The system maximum is used.

number-of-increments: Specify the maximum number of increments that can be added to the queue size. A value of 0 prevents any additions to the initial size of the queue.

AUT

Specifies the authority given to users who do not have specific authority to the message queue, who are not on an authorization list, and whose user group has no specific authority to the message queue.

***LIBCRTAUT:** The public authority for the message queue is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the message queue). The public authority is determined when the message queue is created. If the CRTAUT value for the library changes after the message queue is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the message queue.

***USE:** The user can perform basic operations on the message queue. The user is prevented from changing the object. Use authority provides object operational and read authority.

***EXCLUDE:** The user cannot access the message queue.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the message queue. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

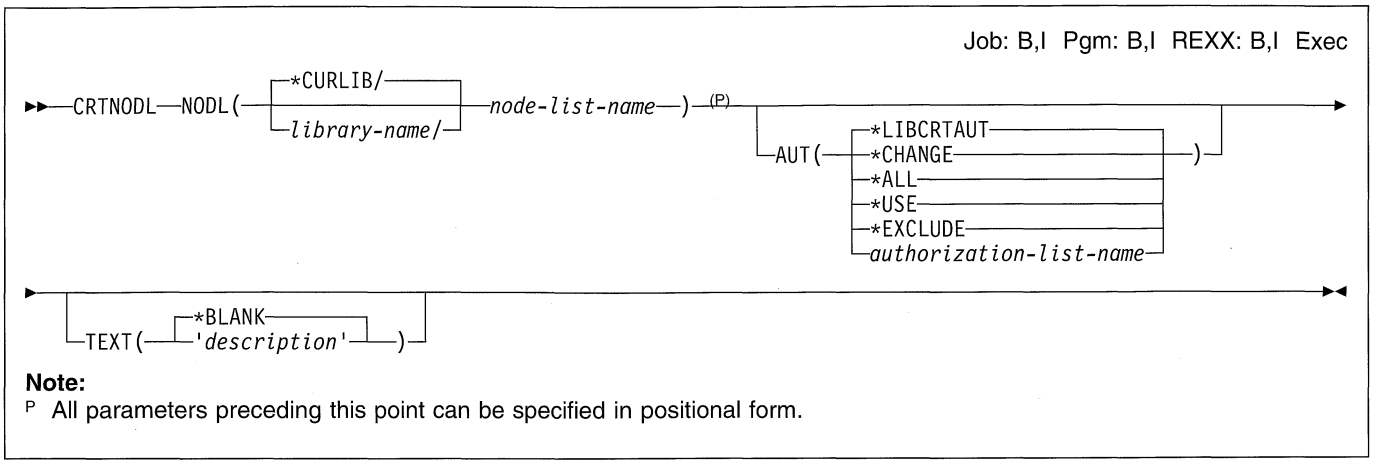
Example

```
CRTMSGQ   MSGQ(MYQ)  SIZE(3 3 *NOMAX)
  TEXT('This message queue is for inventory
  transactions')  AUT(*CHANGE)
```

This command creates the message queue MYQ and stores it in the current library (*CURLIB) by default. All users are authorized to send messages to the queue and to read its messages.

The message queue is created with an initial size of 3 kilobytes (KB) and increased in size in 3 KB increments. The restriction on its maximum size is the system limit for objects, which is about 16,000 KB.

CRTNODL (Create Node List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Node List (CRTNODL) command allows the user to create a node list object. This object is used to store node names that identify a set of systems in a network.

Note: Node lists can be used by system functions to indicate an operation is to be performed on a set of systems.

Required Parameter

NODL

Specifies the qualified name of the node list object that is to be created.

The name of the node list can be qualified by one of the following library values:

***CURLIB:** The node list is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the node list is created.

node-list-object-name: Specify the name of the node list that is to be created.

Optional Parameters

AUT

Specifies the authority given to users who do not have specific authority to the node list, who are not on an authorization list, and whose user group has no specific authority to the node list.

***LIBCRTAUT:** The public authority for the node list is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the node list). The public authority is determined when the node list is created. If the CRTAUT value for the library changes after the node list is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the node list except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the node list, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the node list.

***USE:** The user can perform basic operations on the node list, such as running a program or reading a file. The user cannot change the node list. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the node list.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the node list. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Node List Creation without Text Description

```
CRTNODL   NODL(QGPL/NODL01)
```

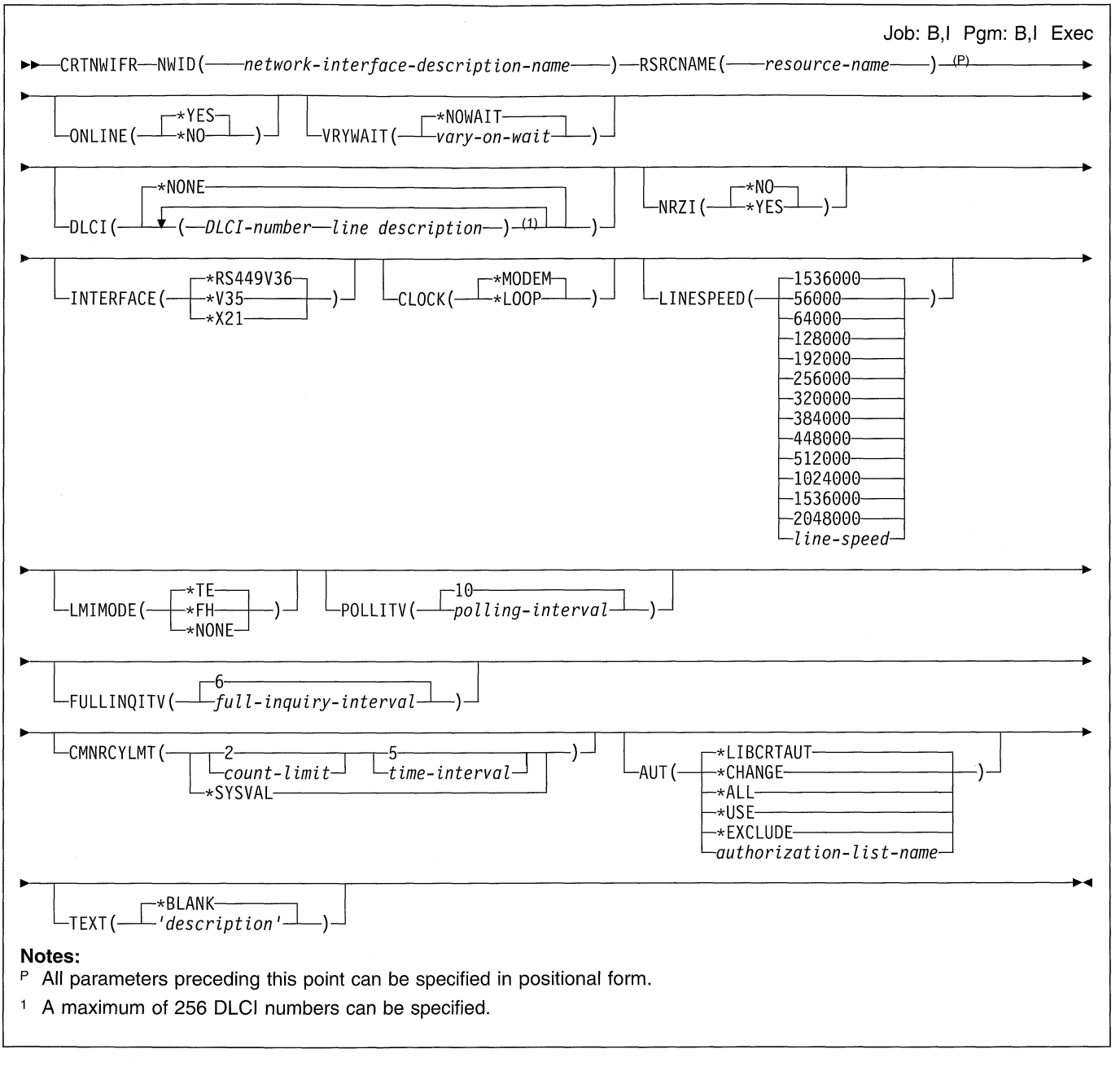
This command creates a node list in library QGPL called NODL01. The node list has the same public authority as that defined for QGPL and it does not have a text description.

Example 2: Node List Creation with Text Description

```
CRTNODL    NODL(MYLIB/NODL02)     AUT(*EXCLUDE)
  TEXT('This is my Node List number 2')
```

This command creates node list NODL02 in library MYLIB with public *EXCLUDE authority. The text description for this node list is 'This is my Node List number 2'.

CRTNWIFR (Create Network Interface (Frame Relay Network)) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 256 DLCI numbers can be specified.

Purpose

The Create Network Interface (Frame-Relay Network) (CRTNWIFR) command creates a network interface for a frame-relay (FR) network. More information about using this command is in the *OS/400* Communications Configuration Reference*.

Required Parameters

NWID

Specifies the name of the network interface description.

network-interface-description-name: Specify the name of a network interface description.

RSRCNAME

Specifies the resource name that identifies the hardware that the description represents.

Note: Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. Specify the resource name of the communications port. The resource name consists of the input/output adapter

(IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 1, then the resource name is LIN011.

resource-name: Specify a resource name.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The network interface is automatically varied on at IPL.

***NO:** This network interface is not automatically varied on at IPL.

VRYWAIT

Specifies whether the network interface is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The network interface is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the network interface is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the network interface description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the network interface or reach the wait-time value.
2. The time required to vary on a network interface is the time it takes to put tasks in place to manage the network interface, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, network interface protocol, and other factors.

DLCI

Specifies the data link connection identifiers of the line descriptions for the frame relay network interface being created.

***NONE:** No data link connection identifier is specified.

Element 1: DLCI Number

DLCI-number: Specify the DLCI number for the line.

Element 2: Line description

line-description: Specify the DLCI line description.

NRZI

Specifies whether non-return-to-zero-inverted (NRZI) data encoding is used for modems that are sensitive to certain bit patterns in the data stream. This ensures that the signal does not stay the same for an extended period of time.

Note: All data communications equipment on the line must use the same transmission method.

***NO:** NRZI data encoding is not used.

***YES:** NRZI data encoding is used.

INTERFACE

Specifies the type of physical interface on the input/output adapter (IOA) port.

***RS449V36:** An RS-499/V.36 physical interface is used. This value is valid only for frame relay and SDLC links.

***V35:** A V.35 physical interface is used. This value is valid only for frame relay, BSC, and SDLC links.

***X21:** An X.21 physical interface is used. This value is valid only for frame relay, X.25, and SDLC links.

CLOCK

Specifies the method in which the clocking function is provided for the network interface.

***MODEM:** The modem provides the clocking.

***LOOP:** The system inverts the clock from the modem and uses it as the transmit clock on the line.

Note: CLOCK(*LOOP) is not valid when INTERFACE(*RS449V36) or INTERFACE(*V35) is specified.

LINESPEED

Specifies the line speed in bits per second (bps)

1536000: The line speed is 1536000 bps.

56000: The line speed is 56000 bps.

64000: The line speed is 64000 bps.

128000: The line speed is 128000 bps.

192000: The line speed is 192000 bps.

256000: The line speed is 256000 bps.

320000: The line speed is 320000 bps.

384000: The line speed is 384000 bps.

448000: The line speed is 448000 bps.

512000: The line speed is 512000 bps.

1024000: The line speed is 1024000 bps.

1536000: The line speed is 1536000 bps.

2048000: The line speed is 2048000 bps.

line-speed: Specify the line speed. Valid values range from 56000 bps through 2048000 bps.

LMIMODE
Specifies whether the local management interface (LMI) for this adapter is configured as terminal equipment or a frame handler.

***TE:** The local system is configured to interface with a frame relay network as terminal equipment.

***FH:** The local system is configured to interface with another system as a frame handler. In this configuration, the local system is performing as the frame relay network.

***NONE:** The local system is configured to interface with the frame relay network or another system without performing any LMI function.

POLLITV
Specifies the rate of the polling cycle. The polling cycle consists of a status enquiry message and a status message exchange. The status message includes the status of the DLCI.

10: A polling interval of 10 seconds is used.

polling-interval: Specify the polling interval to be used within a 5 to 30 second range.

FULLINQITV
Specifies the number of polling cycles that occur before a full status inquiry is requested.

6: A full inquiry interval of 6 polling cycles is used.

full-inquiry-interval: Specify the number of polling cycles for a full status cycle to be requested. Valid values range from 1 through 255.

CMNRCYLMT
Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Recovery Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Value

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT
Specifies the authority given to users who do not have specific authority to the network interface, who are not on an authorization list, and whose user group has no specific authority to the network interface.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the network interface.

***USE:** The user can perform basic operations on the network interface, such as running a program or reading a file. The user cannot change the network interface. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the line description.

authorization-list-name: Specify the name of the authorization list used.

TEXT
Specifies text that briefly describes the line description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

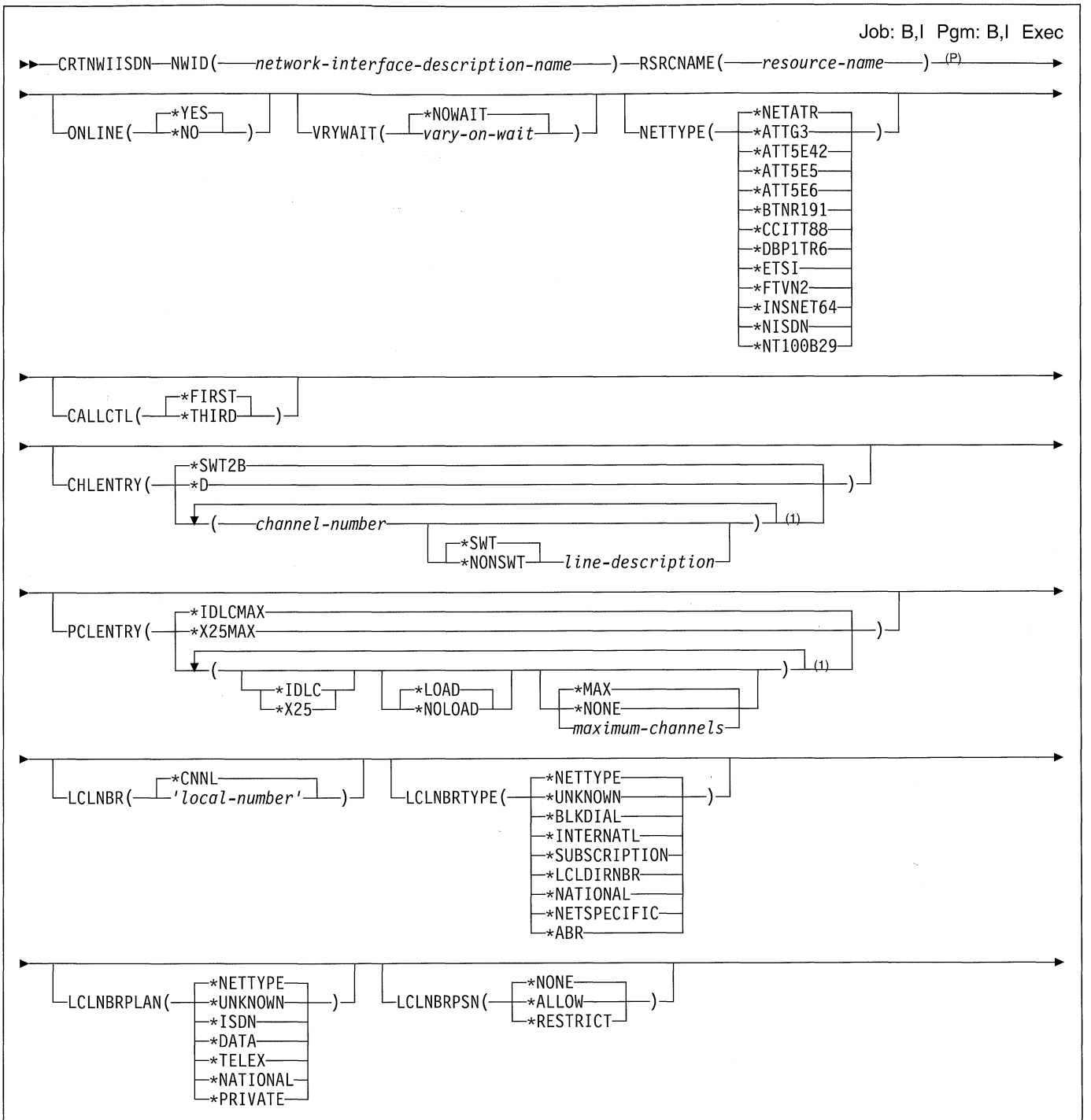
Example

```
CRTNWIFR   NWID(THISONE)  RSRCNAME(LINE031)
  ONLINE(*YES)  VRYWAIT(15)
  DLCI((32 LINEABC) (409 LINEDEF) (94 LINELAST))
  INTERFACE(*V35)  LMIMODE(*TE)
```

This command creates the frame relay network interface THISONE. THISONE represents the resource named LINE031. THISONE is varied on at initial program load (IPL) with a vary on wait time of 15 seconds. It is created with three DLCIs (32, 409, and 94) which refer to line descriptions LINEABC, LINEDEF, and LINELAST respectively. The type of physical interface for the input/output adapter (IOA) port specified by THISONE is *V35. The local management inter-

face mode is configured to interface with a frame relay network as terminal equipment (TE).

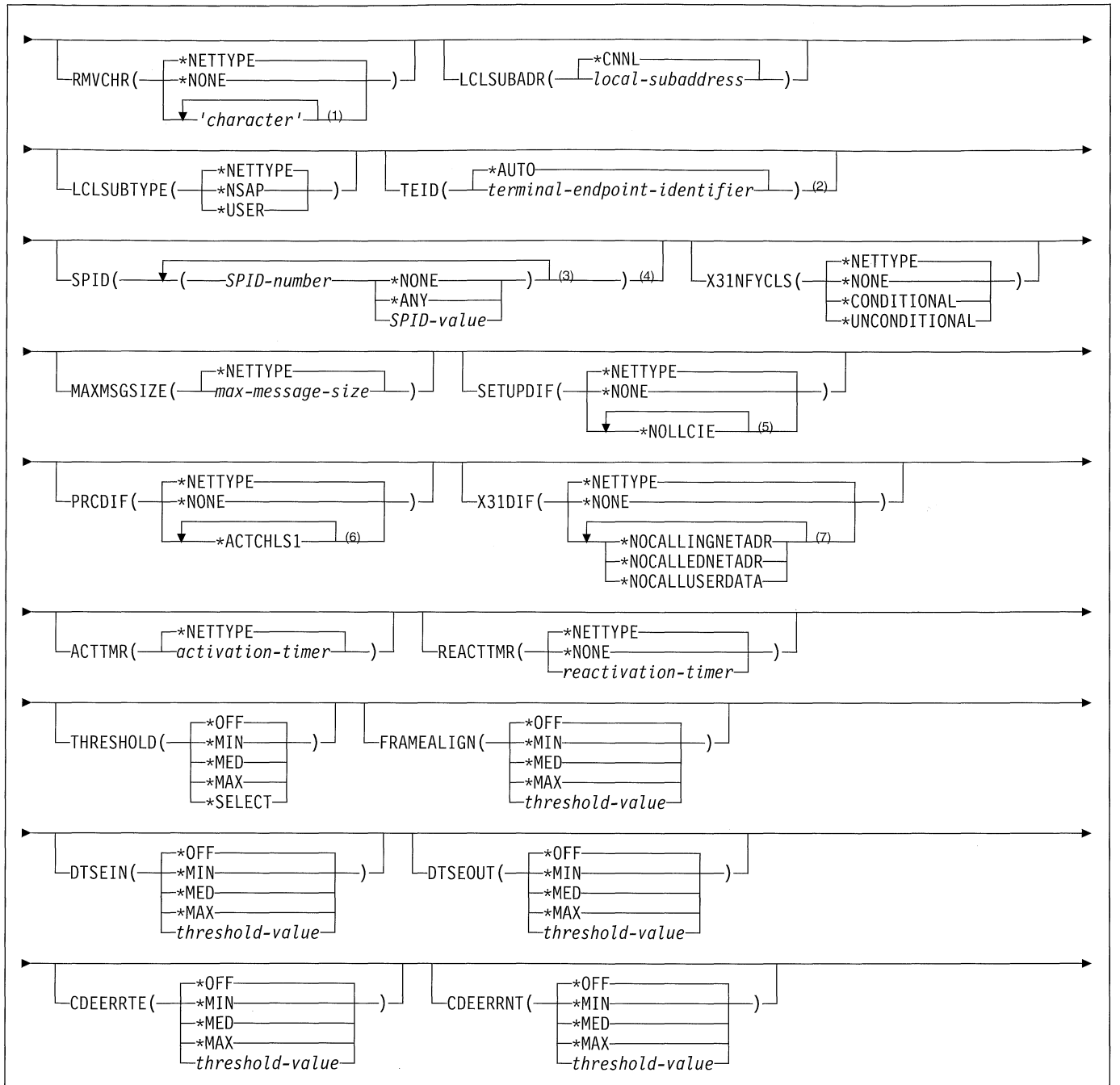
CRTNWIISDN (Create Network Interface Description for ISDN) Command



Notes:

1 A maximum of 2 repetitions.

P All parameters preceding this point can be specified in positional form.



Notes:

1 A maximum of 10 repetitions.

2 This parameter is not valid when NETTYPE(*NISDN) is specified.

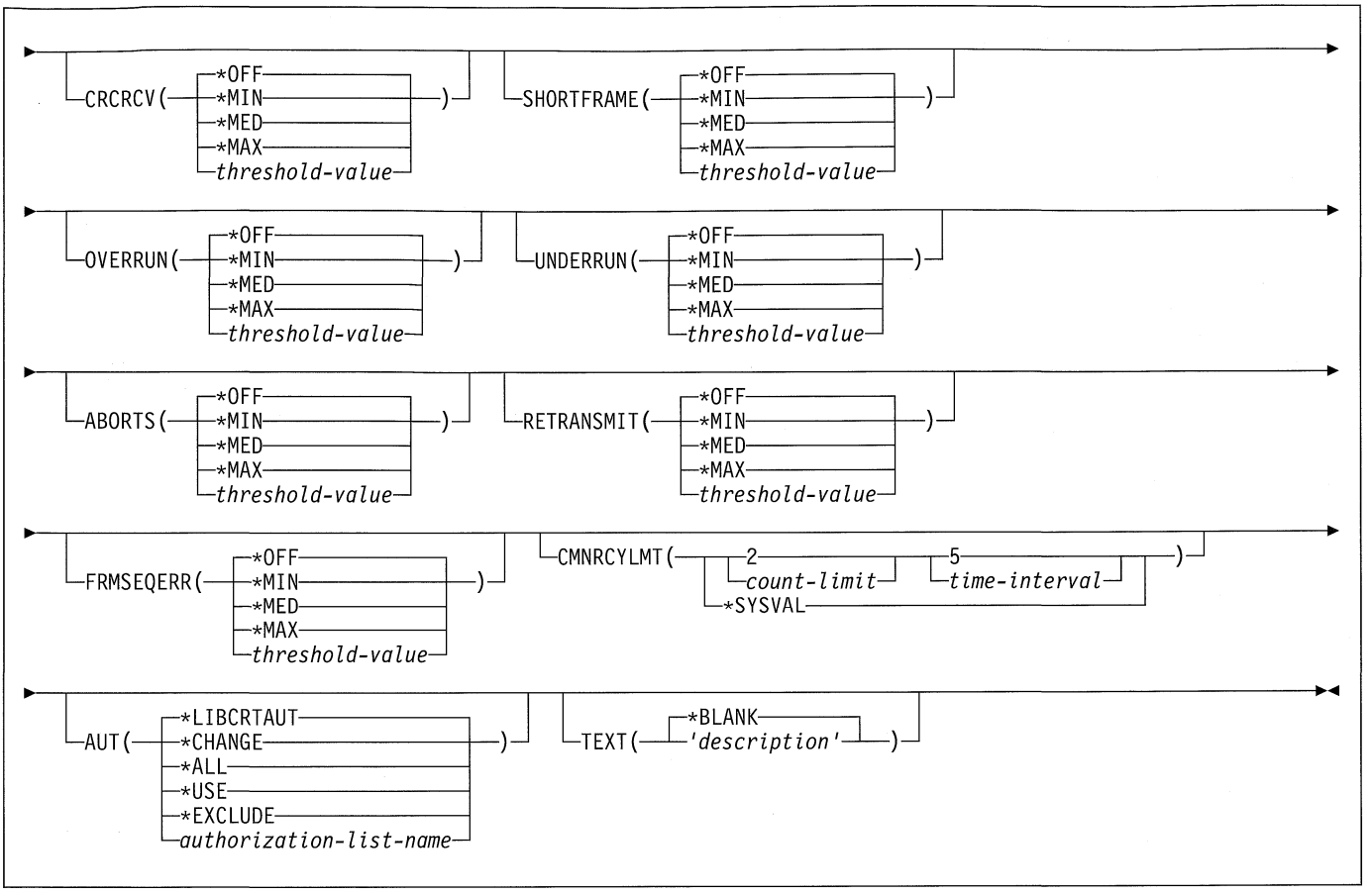
3 A maximum of 2 repetitions.

4 This parameter is valid only when NETTYPE(*NISDN) is specified.

5 A maximum of 4 repetitions.

6 A maximum of 6 repetitions.

7 A maximum of 3 repetitions.



Purpose

The Create Network Interface Description for ISDN (CRTNWIISDN) command creates a network interface description for an integrated services digital network (ISDN) attachment. More information about ISDN is in the *ISDN Guide*.

Required Parameters

NWID

Specifies the name of the network interface description.

RSRCNAME

Specifies the resource name that identifies the hardware that the description represents.

Optional Parameters

ONLINE

Specifies whether this object is automatically varied on at initial program load (IPL).

***YES:** The network interface is automatically varied on at IPL.

***NO:** This network interface is not automatically varied on at IPL.

VRYWAIT

Specifies whether the network interface is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

***NOWAIT:** The system does not wait for the vary on to complete. The network interface is varied on asynchronously.

vary-on-wait: Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the network interface is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

Notes:

1. Specifying a wait time in the network interface description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the network interface or reach the wait-time value.
2. The time required to vary on a network interface is the time it takes to put tasks in place to manage the network interface, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be

longer, depending on the system, network interface protocol, and other factors.

NETTYPE

Specifies the type of integrated services digital network (ISDN) to which the system is attached. The value specified on this parameter is used to determine the defaults for several parameters that are dependent on the type of ISDN to which the system is attached.

***NETATR:** This value indicates that the default network type specified in the network attributes are used. The values of the parameters in the network attributes for the system can be displayed by using the Display Network Attributes (DSPNETA) command. If no value is specified on the DFTNETTYPE parameter, the user should specify one by using the Change Network Attributes (CHGNETA) command.

***ATTG3:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T Definity G3i switching equipment.

***ATT5E42:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T 5ESS version 5E4.2 switching equipment.

***ATT5E5:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T 5ESS version 5E5 switching equipment.

***ATT5E6:** This value is used when attaching to an ISDN in the United States or Canada that uses AT&T 5ESS version 5E6 switching equipment.

***BTNR191:** This value is used when attaching to an ISDN in the United Kingdom controlled by British Telecomm.

***CCITT88:** The ISDN default values recommended by the International Telegraph and Telephone Consultative Committee (CCITT) in 1988 are used.

***DBP1TR6:** This value is used when attaching to an ISDN controlled by Germany's post telephone and telegraph administration (PTT). (Deutsche Bundespost 1TR6).

***ETSI:** This value is used when attaching to an ISDN that uses the European Telecommunications Standards Institute (ETSI, also known as EuroISDN) standard.

***FTVN2:** This value is used when attaching to version 2 of the ISDN controlled by France's PTT. (France Telecomm Numeris VN2).

***INSNET64:** This value is used when attaching to the INSNET64 ISDN controlled by Japan's Nippon Telephone and Telegraph Public Corporation (NTT).

***NISDN:** This value is used when attaching to an ISDN that uses the National ISDN-1 or National ISDN-2 standard for North America.

***NT100B29:** This value is used when attaching to an ISDN in the United States or Canada that uses Northern Telecom DMS100 Version BCS-29 switching equipment.

CALLCTL

Specifies the type of call control performed by the system.

***FIRST:** First-party call control is performed by the system. This allows the AS/400 system to set up and take down calls for which it is a party.

***THIRD:** Third-party call control is performed by the system. This allows the AS/400 system to monitor and control calls of which it is not a party.

CHLENTRY

Specifies a channel entry corresponding to a B channel or D channel associated with the network interface. The user can specify up to two B channel entries or one D channel entry for a basic rate interface. The D channel is always nonswitched, and is used with third-party call control. The B channel is used with first-party call control.

Element 1: Channel Number

***SWT2B:** This value is used for two switched B channels.

***D:** This value is used for one nonswitched D channel.

channel-number: Specify the B-channel number. Valid values are 1 and 2.

Element 2: Channel Connection

***SWT:** This value is used for switched or semi-permanent connections.

***NONSWT:** This value is used for nonswitched (permanent) connections.

Element 3: Line Description

line-description: Specify an existing line description. Names of ISDN data link control (IDLC), network, and X.25 line descriptions can be specified.

Note: A D channel can only be specified with one network line description, while a B channel can be specified with an IDLC or X.25 line description.

PCLENTRY

Specifies a list of protocols used and protocol-specific information.

***IDLCMAX:** Specifies the IDLC protocol. The associated microcode is preloaded, and the maximum number of channels (2) are available for use.

***X25MAX:** Specifies the X.25 protocol. The associated microcode is preloaded, and the maximum number of channels (2) are available for use.

Element 1: Protocol Used

***IDLC:** The IDLC protocol is used.

***X25:** The X.25 protocol is used.

Element 2: Preload Microcode

***LOAD:** Microcode is preloaded. Preloading allows the system to report faster to incoming calls at vary-on time, since the necessary tasks are already in place.

***NOLOAD:** Microcode is not preloaded.

Element 3: Maximum Channels

***MAX:** The maximum number of channels (2) for this network interface description (NWID) are available for use by the specified protocol.

***NONE:** The maximum number of channels is not specified.

maximum-channels: Specify the maximum number of channels to be used. Valid values are 1 and 2.

LCLNBR

Specifies the local number for this system in the ISDN network. The local number can be up to 40 characters long. Special characters are used to delimit the number; see the RMVCHR parameter.

***CNNL:** The system determines the local number by using the connection list object specified for the call.

local-number: Specify up to 40 characters for the local number.

LCLNBRTYPE

Specifies the type of local number specified on the LCLNBR parameter.

***NETTYPE:** The system determines the local number type by using the value specified on the NETTYPE parameter.

***UNKNOWN:** The local number type is not known.

***BLKDIAL:** The local number is a block dial type.

***INTERNATL:** The local number is an international number type.

***SUBSCRIPTION:** The local number is a subscription number type.

***LCLDIRNBR:** The local number is a local directory number type.

***NATIONAL:** The local number is a national address type.

***NETSPECIFIC:** The local number type is specific to the network.

***ABR:** The local number type is abbreviated.

LCLNBRPLAN

Specifies the numbering plan used for the local number.

***NETTYPE:** The numbering plan is determined by the value specified on the NETTYPE parameter.

***UNKNOWN:** The numbering plan is not known.

***ISDN:** The ISDN/telephony numbering plan is used.

***DATA:** The data numbering plan is used.

***TELEX:** The telex numbering plan is used.

***NATIONAL:** The national numbering plan is used.

***PRIVATE:** A private numbering plan is used.

LCLNBRPSN

Specifies the intention of the calling user for the presentation of the local number to the called user. This parameter applies only to outgoing calls.

***NONE:** The local number presentation is not encoded. The network determines whether the local number is presented to the called user.

***ALLOW:** The local number is presented to the called user.

***RESTRICT:** The presentation of the local number to the called user is restricted by the network.

RMVCHR

Specifies the characters to be removed from the local number. Extra characters are removed from the numbers before sending or comparing them. The ability to shorten numbers prior to their use by the system means the user can insert extra characters in numbers to make them more organized and readable.

***NETTYPE:** The system determines the characters to be removed using the value specified on the NETTYPE parameter.

***NONE:** No characters are removed.

'character': Specify up to 10 characters to be removed.

LCLSUBADR

Specifies the local subaddress assigned to a system configured as a terminal endpoint. The subaddress can be up to 40 hexadecimal characters in length.

***CNNL:** The system determines the local number field by using the connection list object specified for the call.

local-subaddress: Specify the local subaddress.

LCLSUBTYPE

Specifies the type of subaddress assigned to a terminal endpoint system.

***NETTYPE:** The subtype is determined by the value specified on the NETTYPE parameter.

***NSAP:** The local subaddress is Network-Layer service access point (NSAP).

***USER:** The local subaddress is user-specified.

TEID

Specifies the terminal endpoint identifier (TEID).

Note: The TEID parameter is not available and the TEID is automatically assigned for network interface descriptions with *NISDN specified on the NETTYPE parameter.

***AUTO:** A terminal endpoint identifier is automatically assigned by the network.

terminal-endpoint-identifier: Specify a number to use as the terminal endpoint identifier. Valid values range from 0 through 63.

SPID

Specifies the service profile identifier (SPID) to be used in an exchange of information between the system and the network when communications are initialized. The SPID value is assigned by and can be obtained from the network provider when the user subscribes to the network.

Element 1: SPID Number

SPID-number: Specify the SPID number. Valid values are 1 and 2.

Element 2: SPID Value

***NONE:** No SPID or TEID is used for the exchange.

***ANY:** An SPID is not specified, but a TEID is used for the exchange.

SPID-value: Specify the service profile identifier. A minimum of 9 characters are required, and no more than 20 characters can be specified.

X31NFYCLS

Specifies the method for the packet handler in the ISDN to notify the system of an incoming packet mode call.

***NETTYPE:** The method of notifying the system is determined by the value specified on the NETTYPE parameter.

***NONE:** The packet handler provides no notification.

***CONDITIONAL:** The packet handler provides notification only if a packet mode call cannot be delivered on an existing connection.

***UNCONDITIONAL:** The packet handler provides notification for every packet mode call.

MAXMSGSIZE

Specifies the maximum size of a Q.931 message to be accepted by the call control layer in the input/output (I/O) processor.

***NETTYPE:** The system determines the maximum size by using the value specified on the NETTYPE parameter.

max-message-size: Specify the maximum message size. Valid values range from 110 through 1024.

Note: Use of the default value is strongly recommended for this parameter.

SETUPDIF

Specifies the defaults used to send **SETUP** messages to the network. Up to four values can be specified on this parameter. The user can follow International Telegraph and Telephone Consultative Committee (CCITT) recommendations by specifying the *NONE value.

***NETTYPE:** The system determines the defaults for sending **SETUP** messages by using the value specified on the NETTYPE parameter.

***NONE:** The CCITT recommended defaults for sending **SETUP** messages are used: High Layer Compatibility IE is not sent; Low Layer Compatibility IE is sent; Bearer Capability IE is sent in the **SETUP** message; and Terminal Capability IE is not sent in the **SETUP** message.

***NOLLCIE:** The Low-Layer Compatibility IE (Information Element) is not sent with the **SETUP** message.

PRCDIF

Specifies the defaults used for the handling of miscellaneous procedures. More than one value can be specified for this parameter. The user can follow International Telegraph and Telephone Consultative Committee (CCITT) recommendations by specifying the *NONE value.

***NETTYPE:** The system determines miscellaneous settings by using the value specified on the NETTYPE parameter.

***NONE:** The CCITT recommended defaults for miscellaneous settings are used: calls are cleared when a **SETUP ACKNOWLEDGE** message is received; a **STATUS** message is sent to the peer when an error is detected; both sides send **RELEASE** messages following a **DISCONNECT** message; **RELEASE COMPLETE** messages are sent to callers.

***ACTCHLS1:** The S1 maintenance channel is activated. The S1 channel allows the system to accept a broader range of RAS messages, resulting in improved network management capabilities.

X31DIF

Specifies network-specific differences in X.31 Case B operations.

***NETTYPE:** The system determines whether X.31 call data is checked depending on the value specified on the NETTYPE parameter.

***NONE:** There are no differences in packet mapping.

***NOCALLINGNETADR:** The call data does not contain a calling network address.

***NOCALLEDNETADR:** The call data does not contain a called network address.

***NOCALLUSERDATA:** The call data does not contain user data.

ACTTMR

Specifies the amount of time in tenths of a second, allowed for the interface activation process to complete.

***NETTYPE:** The system determines the amount of time allowed by using the value specified on the NETTYPE parameter.

activation-timer: Specify a value ranging from 10 through 300 in 0.1-second intervals.

REACTTMR

Specifies the amount of time allowed for interface reactivation to occur following a temporary loss of synchronization.

***NETTYPE:** The system determines the amount of time allowed by using the value specified on the NETTYPE parameter.

***NONE:** The system does not wait for reactivation to occur.

reactivation-timer: Specify a value ranging from 10 through 1800 in 0.1-second intervals.

THRESHOLD

Specifies the threshold for the number of errors beyond which a message is sent informing the user that errors have occurred. All error threshold parameters are set to the value specified, unless *SELECT is specified, in which case each error threshold parameter is set individually. The next eleven parameters are the threshold parameters. They are: Loss of frame alignment, Incoming system access errors, Outgoing system access errors, Code error detected by TE, Code error detected by NT, CRC errors received, Overrun, Underrun, Frame aborts, Frame sequence errors, and Retransmitted frames.

***OFF:** No monitoring of errors occurs.

***MIN:** The error threshold is set at a minimum monitoring level.

***MED:** The system performs a medium amount of error threshold monitoring for all types of errors. The values chosen are different for each threshold error type. See the descriptions on the next eleven THRESHOLD parameters for more specific information.

***MAX:** The error threshold is set at a maximum monitoring level.

***SELECT:** The threshold for the number of errors beyond which a message is sent informing the user that errors have occurred is set individually for each parameter. The next eleven parameters are the threshold parameters. They are: Loss of frame alignment, Incoming system access errors, Outgoing system access errors, Code error detected by TE, Code error detected by NT, CRC errors received, overrun, underrun, Frame aborts, Frame sequence errors, and Retransmitted frames.

FRAMEALIGN

Specifies the threshold for the number of frame alignment errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for frame alignment errors: 9 errors in the first 30 seconds or 270 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for frame alignment errors: 3 errors in the first 30 seconds or 90 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for frame alignment errors.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

DTSEIN

Specifies the threshold for the number of incoming system access errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for incoming system access errors: 9 errors in the first 30 seconds or 270 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for incoming system access errors: 3 errors in the first 30 seconds or 90 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for incoming system access errors.

threshold-value: Specify a number, ranging from 1 through 5000, that corresponds to the number of errors allowed in a 15-minute interval.

DTSEOUT

Specifies the threshold for the number of outgoing system access errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for outgoing system access errors: 9 errors in the first 30 seconds or 270 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for outgoing system access errors: 3 errors in the first 30 seconds or 90 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for outgoing system access errors.

threshold-value: Specify a number, ranging from 1 through 5000, that corresponds to the number of errors allowed in a 15-minute interval.

CDEERRTE

Specifies the threshold for the number of code errors detected by the terminal endpoint (TE) beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring: 9 errors in the first 30

seconds or 270 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for code errors detected by the TE: 3 errors in the first 30 seconds or 90 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for code errors detected by the TE.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

CDEERRNT

Specifies the threshold for the number of code errors detected by the network termination (NT) beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for code errors detected by the NT: 9 errors in the first 30 seconds or 270 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for code errors detected by the NT: 3 errors in the first 30 seconds or 90 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for code errors detected by the NT.

threshold-value: Specify a number, ranging from 1 through 5000, that corresponds to the number of errors allowed in a 15-minute interval.

CRCRCV

Specifies the threshold for the number of Cyclic Redundancy Check (CRC) errors received beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for CRC errors received: 5 errors in the first 30 seconds or 150 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for CRC errors received: 2 errors in the first 60 seconds or 30 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for CRC errors received.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

SHORTFRAME

Specifies the threshold for the level of errors for short frames received beyond which a message is sent informing the user that errors have occurred.

***OFF:** Error threshold monitoring is turned off for short frame errors.

***MIN:** The system performs the minimum amount of error threshold monitoring for receive overrun errors: 6 short X.25 or SDLC frames received in the first 30 seconds or 1 short frame received every second for 14 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for receive overrun errors: 6 X.25 or SDLC frames received in the first 30 seconds or 1 received every 3-4 seconds for 10-14 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for short frame errors.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

OVERRUN

Specifies the threshold for the number of receive overrun errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for receive overrun errors: 2 errors in the first 90 seconds or 20 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for receive overrun errors: 2 errors in the first 300 seconds or 6 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for receive overrun errors.

threshold-value: Specify a number, ranging from 1 through 3000, that corresponds to the number of errors allowed in a 15-minute interval.

UNDERRUN

Specifies the threshold for the number of transmit underrun errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs a minimum amount of error threshold monitoring for buffer overrun errors: 2 errors in the first 90 seconds or 20 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for buffer overrun errors: 2 errors in the first 300 seconds or 6 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for transmit underrun errors.

threshold-value: Specify a number, ranging from 1 through 3000, that corresponds to the number of errors allowed in a 15-minute interval.

ABORTS

Specifies the threshold for the number of frame aborts beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for frame aborts: 5 errors in the first 30 seconds or 150 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for frame aborts: 2 errors in the first 60 seconds or 30 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for frame aborts.

threshold-value: Specify a number, ranging from 1 through 5000, that corresponds to the number of errors allowed in a 15-minute interval.

RETRANSMIT

Specifies the threshold for the number of retransmissions of frames beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for retransmitted frames: 5 errors in the first 30 seconds or 150 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for retransmitted frames: 2 errors in the first 60 seconds or 30 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for retransmitted frames.

threshold-value: Specify a number, ranging from 1 through 10000, that corresponds to the number of errors allowed in a 15-minute interval.

FRMSEQERR

Specifies the threshold for the number of frame sequence errors beyond which a message is sent informing the user that errors have occurred.

***OFF:** No monitoring of errors occurs.

***MIN:** The system performs the minimum amount of error threshold monitoring for frame sequence errors: 2 errors in the first 90 seconds or 20 errors in any 15 minutes. The system is more tolerant of errors than if *MED or *MAX is specified.

***MED:** The system performs a medium amount of error threshold monitoring for frame sequence errors: 2 errors in the first 300 seconds or 6 errors in any 15 minutes.

***MAX:** The system performs the maximum amount of error threshold monitoring for frame sequence errors.

threshold-value: Specify a number, ranging from 1 through 3000, that corresponds to the number of errors allowed in a 15-minute interval.

CMNRCYLMT

Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Count Limit

2: Two recovery attempts are made within the interval specified.

count-limit: Specify the number of recovery attempts to be made. Valid values range from 0 through 99.

Element 2: Time Interval

5: The specified number of recovery attempts is made within a 5-minute interval.

time-interval: Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for *count-limit* is not 0, the value 0 specifies infinite recovery.

Other Single Values

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

AUT

Specifies the authority given to users who do not have specific authority to the network interface description, who are not on an authorization list, and whose user group has no specific authority to the network interface description.

***LIBCRTAUT:** The authority for the object is the same as the create authority for QSYS. The create authority for QSYS can be displayed by using the Display Library (DSPLIB) command. If the create authority is changed with the Change Library (CHGLIB) command, the new authority does not affect existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change

the object, and perform basic functions on the object. The user can change ownership of the network interface description.

***USE:** The user can perform basic operations on the network interface description, such as running a program or reading a file. The user cannot change the network interface description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the network interface description.

authorization-list-name: Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

TEXT

Specifies text that briefly describes the network interface. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

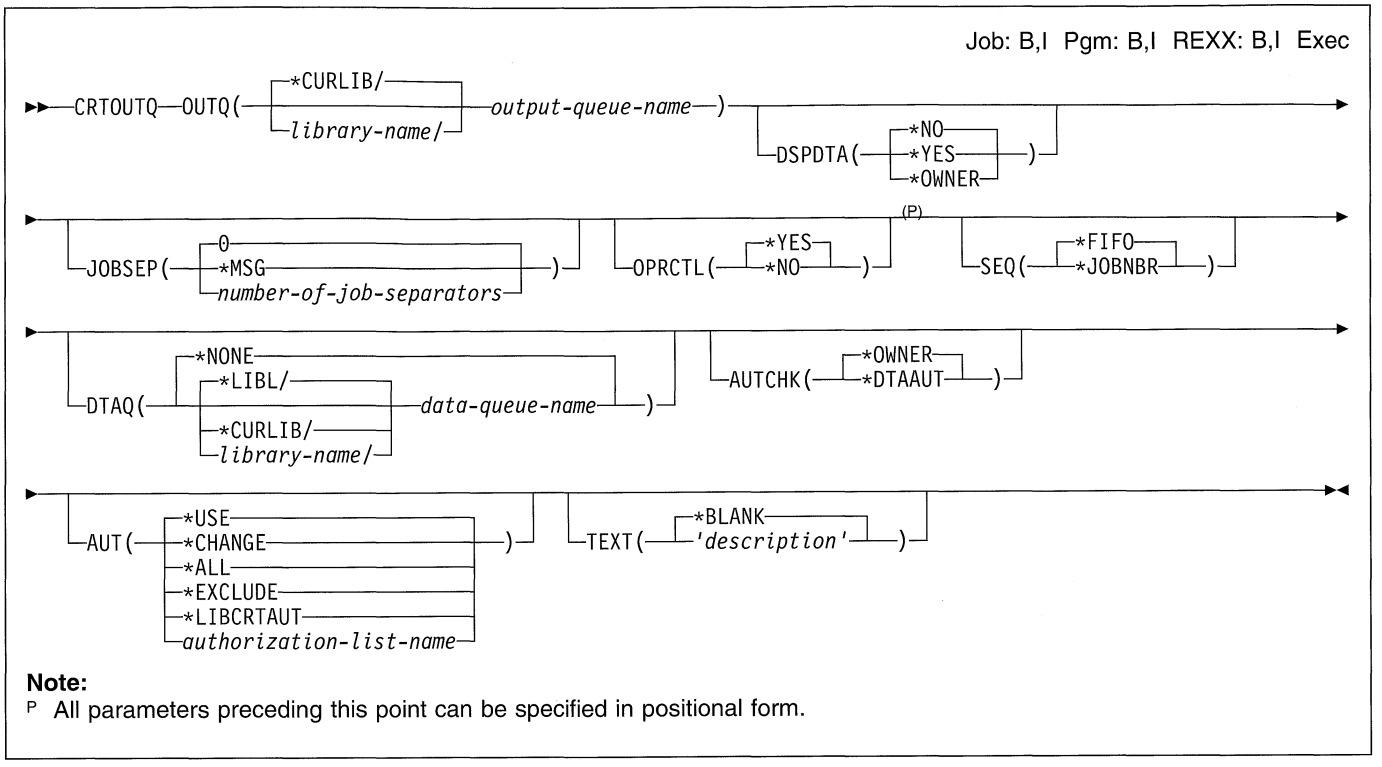
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTNWIISDN   NWID(ISDNNET)  RSRCNAME(LIN011)
  NETTYPE(*CCITT88)  CHLENTRY((1)(2*NONSWT))
```

This command creates a network interface description named ISDNNET which uses the resource associated with resource name LIN011. The network interface description is configured to attach to an ISDN which conforms to the CCITT 1988 recommendations, and uses the first B channel for switched circuit calls and the second B channel for non-switched circuit calls.

CRTOUTQ (Create Output Queue) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Output Queue (CRTOUTQ) command creates a new output queue for spooled files. An entry is placed on the output queue for each spooled file. The order in which the files are written to the output device is determined by the output priority of the file and the value of the SEQ parameter described below.

Required Parameter

OUTQ

Specifies the qualified name of the output queue.

The name of the output queue can be qualified by one of the following library values:

***CURLIB:** The output queue is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the output queue is created.

Note: The temporary library QTEMP is not a valid library name. Output queues must be in permanent libraries.

output-queue-name: Specify the name of the output queue being created.

Optional Parameters

DSPDTA

Specifies whether users who have authority to read the output queue can display the output data of any spooled file on the queue or only the data in their own files.

***NO:** Users authorized to use the queue can display, copy, or send the output data of their own files only unless they have some other special authority.

***YES:** Any user having authority to read the queue can display, copy, or send the data of any file on the queue.

***OWNER:** The owner of the file or a user with *SPLCTL special authority can display, copy, or send the spooled files on the queue.

JOBSEP

Specifies, for each job with files on the output queue, the number of separators placed at the beginning of the output for the job. Each separator contains information that identifies the job, such as the job name, the job user's name, the job number, and the time and date when the job is run.

0: No job separators are printed preceding each job's output.

***MSG:** A message is sent to a message queue notifying the operator of the end of each job. This message queue is specified when a printer writer is started to process the output queue.

number-of-job-separators: Specify a value, ranging from 0 through 9, that specifies the number of separators to be placed preceding each job's output.

OPRCTL

Specifies whether a user (an operator) who has job control authority is allowed to manage or control the files on this output queue. Users have job control authority if SPCAUT(*JOBCTL) is specified in their user profiles.

***YES:** A user with job control authority can control the queue and make changes to the files on the queue.

***NO:** This queue and its files cannot be controlled or changed by users with job control authority unless they also have some other special authority.

SEQ

Specifies the order of the spooled files on the output queue.

***FIFO:** The spooled files are processed first-in first-out within priority groups on the queue. That is, new spooled files are placed after all other entries on the queue of the same priority. The following changes place a queue entry after all others of equal priority on the queue:

- A change of output priority with the Change Job (CHGJOB) or Change Spooled File Attributes (CHGSPLFA) command.
- A change in status from held (HLD), closed (CLO), or open (OPN) to available (RDY).
- A change in status from available (RDY) to not available (HLD, CLO, OPN).
- A spooled file added to the queue when the file is opened.
- Using the Change Spooled File Attributes (CHGSPLFA) command to move a spooled file to an output queue that has SEQ(*FIFO) specified.

***JOBNBR:** Within priority groups, the queue entries for spooled files are sorted by using the job number (actually, the date and time that the job entered the system is used) of the job that created the spooled file.

When using SEQ(*JOBNBR), more spooled files of a job tend to be grouped together even when SCHEDULE(*JOBEND) is not specified.

Also, when using SEQ(*JOBNBR) for jobs of equal priority, the first job always has its spooled files produced first.

DTAQ

Specifies the name of the data queue associated with the output queue. Entries are logged in the data queue when spooled files are in ready (RDY) status on the output queue. A user program can determine when a spooled file is available on an output queue using the Receive Data Queue API (QRCVDTAQ) to receive information from a data queue.

Each time a spooled file on the output queue reaches RDY status, an entry is sent to the data queue. A spooled file can have several changes in status (for example, RDY to held (HLD) to release (RLS) to RDY again) before it is taken off the output queue. These status changes result in entries in the data queue for a spooled file each time the spooled file goes to RDY status.

***NONE:** No data queue is associated with the output queue.

The name of the data queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

data-queue-name: Specify the name of the data queue associated with the output queue.

AUTCHK

Specifies whether the commands that check the requester's authority to the output queue also check for ownership authority or data authority.

***OWNER:** The requester must have ownership authority to the output queue in order to pass the output queue authorization test. The requester can have ownership authority by being the owner of the output queue, sharing a group profile with the queue owner, or running a program that adopts the owner's authority.

***DTAAUT:** The requester must have the appropriate data authority to the output queue (*READ, *ADD, and *DELETE) in order to pass the output queue authorization test.

AUT

Specifies the authority given to users who do not have specific authority to the out queue, who are not on an authorization list, and whose user group has no specific authority to the out queue.

***USE:** The user can perform basic operations on the out queue, such as running a program or reading a file. The user cannot change the out queue. *USE authority provides object operational authority and read authority.

Note: The user can change the output queue description and control files created by other users if AUTCHK(*DTAAUT) is specified.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the out queue.

***EXCLUDE:** The user cannot access the out queue.

***LIBCRTAUT:** The public authority for the out queue is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the out queue). The public authority is determined when the out queue is created. If the CRTAUT value for the library changes after the out queue is created, the new value does not affect any existing objects.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the output queue. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

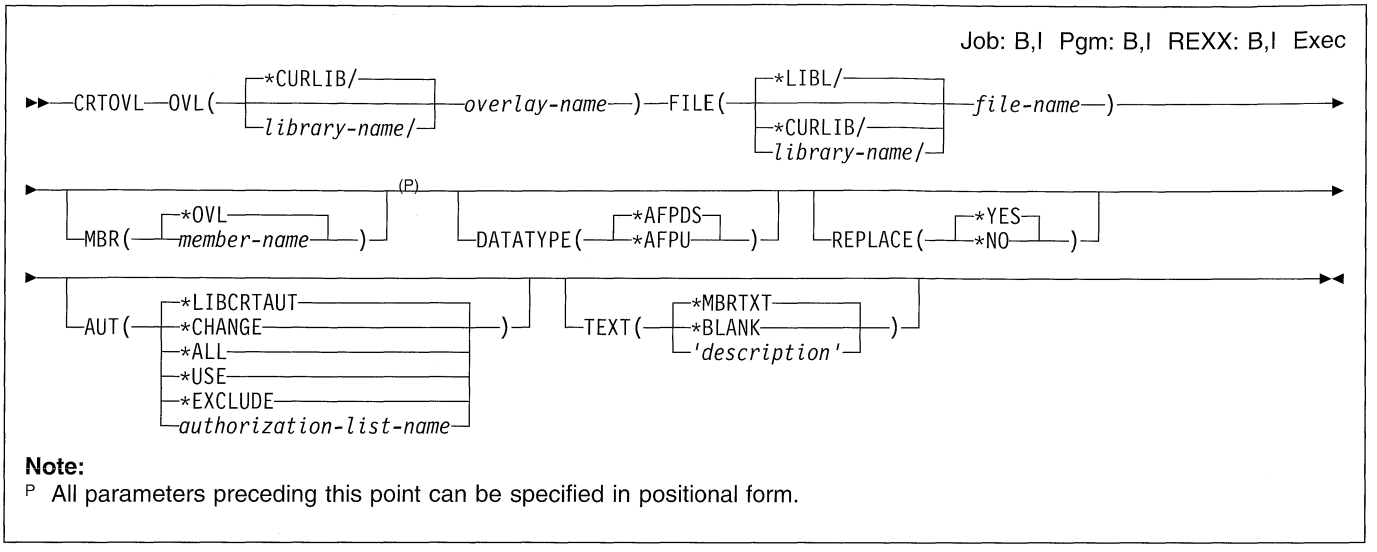
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTOUTQ   OUTQ(DEPTAPRT)  AUT(*EXCLUDE)  SEQ(*FIFO)
  TEXT('SPECIAL PRINTER FILES FOR DEPTA')
```

This command creates an output queue named DEPTAPRT and puts it in the current library. Because AUT(*EXCLUDE) is specified and OPRCTL(*YES) is assumed, the output queue can be used and controlled only by the user who created the queue and users who have job control authority or spool control authority. Because SEQ(*FIFO) is specified, spooled files are placed in first-in first-out order on the queue. If users in Department A are authorized to use this output queue, the Grant Object Authority (GRTOBJAUT) command must be used to grant them the necessary authority. Data contained in files on this queue can be displayed only by users who own the files, by the owner of the queue, by users with job control authority, or by users with spool control authority. By default, no job separator is printed at the beginning of the output for each job.

CRTOVL (Create Overlay) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Overlay (CRTOVL) command creates an overlay resource from a physical file. The physical file contains the overlay resource information. The overlay resource information, can, for example, come from a S/370 host system and be in the Systems Application Architecture (SAA) format.

Required Parameters

OVL

Specifies the qualified name of the overlay being created.

The name of the overlay can be qualified by one of the following library values:

***CURLIB:** The overlay is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the overlay is created.

overlay-name: Specify the name of the overlay being created.

FILE

Specifies the qualified name of the file being used to create the overlay.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file to create the database file.

Optional Parameters

MBR

Specifies the name of the file member used to create the overlay.

***OVL:** The name of the file member that contains the input data is the same as the overlay being created.

member-name: Specify the name of the file member that contains the overlay input data.

DATATYPE

Specifies the source type of the input file.

***AFPDS:** The input is a pre-built *AFPDS file from a System/370.

***AFPU:** The input is an *AFPU source file created with Advanced Function Printing Utilities/400.

REPLACE

Specifies whether the page definition is replaced.

***YES:** The specified page definition is replaced.

***NO:** No replacement occurs.

AUT

Specifies the authority given to users who do not have specific authority to the overlay, who are not on an authorization list, and whose user group has no specific authority to the overlay.

***LIBCRTAUT:** The public authority for the overlay is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the overlay). The public authority is determined when the overlay is

created. If the CRTAUT value for the library changes after the overlay is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the overlay.

***USE:** The user can perform basic operations on the overlay, such as running a program or reading a file. The user cannot change the overlay. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the overlay.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the program and its function. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MBRTXT:** The text is taken from the file member being used to create the overlay.

The user can add or change text for a database source member by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** No text is specified.

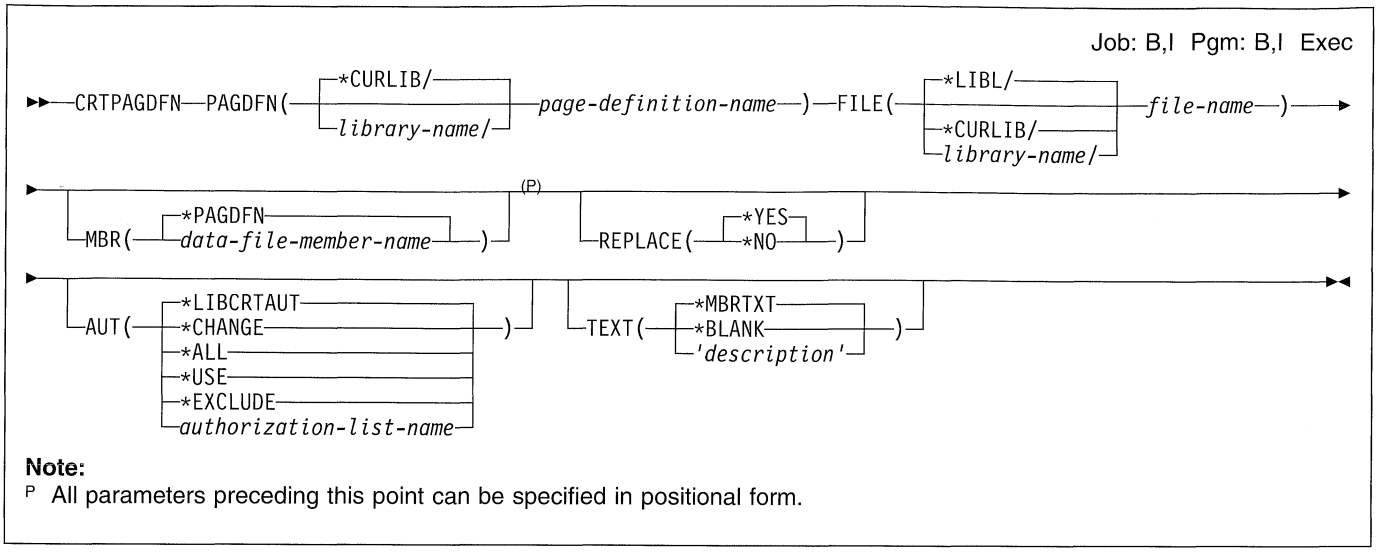
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTOVL  OVL(MYLIB/MYSIGNTR)  FILE(MYLIB/MYSIGNTR)
  MBR(MYSIGNTR)  AUT(*EXCLUDE)
  TEXT('representation of my signature')
```

This command creates the overlay MYSIGNTR into MYLIB. File name MYSIGNTR in library MYLIB with member MYSIGNTR, is used as input. Specifying *EXCLUDE does not allow any other user access to the signature. The text describes the overlay.

CRTPAGDFN (Create Page Definition) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Page Definition (CRTPAGDFN) command creates a page definition by copying a user-supplied AS/400 database file to an internal space object. The user must load the source data into the database from a remote system (such as a System/370) or external medium (usually tape), and put it in the SAA format that can be processed by the OS/400 system.

Restriction: If networking spooled files to a System/370 system, the first two characters of the page definition name must be 'P1'.

Required Parameters

PAGDFN

Specifies the qualified name of the page definition to be created.

The name of the page definition can be qualified by one of the following library values:

***CURLIB:** The page definition is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the page definition is created.

page-definition-name: Specify the up to 8 characters for the name of the page definition.

FILE

Specifies the qualified name of the database file that contains the page definition records sent to this system.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the database file.

Optional Parameters

MBR

Specifies the name of the database file member containing the page definition records.

***PAGDFN:** The name specified on the PAGDFN parameter of this command is used.

data-file-member-name: Specify the name of the database file member.

REPLACE

Specifies whether an existing page definition with the same name as the one being created is replaced.

***YES:** The page definition is replaced.

***NO:** No replacement occurs.

AUT

Specifies the authority given to users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.

***LIBCRTAUT:** The system determines the authority for the object by using the value specified on the CRTAUT parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the

value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the page definition.

***USE:** The user can perform basic operations on the page definition, such as running a program or reading a file. The user cannot change the page definition. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the page definition.

authorization-list-name: Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

TEXT

Specifies text that briefly describes the page description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MBRTXT:** The text is taken from the database file member used to create the page definition.

***BLANK:** No text is specified.

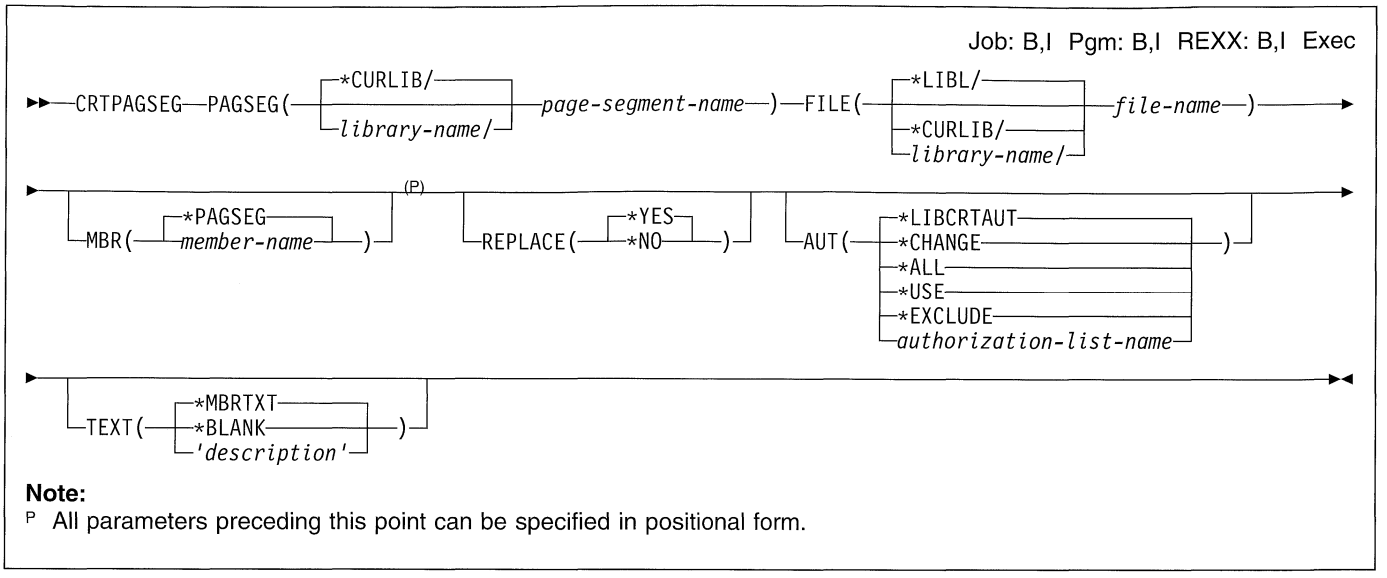
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTPAGDFN   PAGDFN(*CURLIB/P1DFLT)  FILE(*CURLIB/PAGDFNS)
  MBR(*PAGDFN)  AUT(*EXCLUDE)
  TEXT('Default page definition')
```

This command creates page definition P1DFLT in the current library or in library QGPL if there is no current library. Input is taken from source file PAGDFNS with member P1DFLT in the current library. Specifying *EXCLUDE for authority restricts use of the object to the owner. The text describes the page definition.

CRTPAGSEG (Create Page Segment) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Page Segment (CRTPAGSEG) command creates a page segment physical file. The physical file contains the page segment information. The page segment information, can, for example, come from a S/370 host system and be in the Systems Application Architecture (SAA) format.

Required Parameters

PAGSEG

Specifies the qualified page segment name (library-name/page-segment-name) that is being created.

The name of the page segment can be qualified by one of the following library values:

***CURLIB:** The page segment is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the page segment is created.

page-segment-name: Specify the name of the page segment being created.

FILE

Specifies the qualified name of the file being used to create the database file.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file to create the database file.

Optional Parameters

MBR

Specifies the name of the file member being used to create the member.

***PAGSEG:** The name of the file member that contains the input data is the same as the member name.

member-name: Specify the name of the member name that contains the member input data.

REPLACE

Specifies whether the page segment is replaced.

***YES:** The page segment is replaced.

***NO:** No replacement occurs.

AUT

Specifies the authority given to users who do not have specific authority to the page segment, who are not on an authorization list, and whose user group has no specific authority to the page segment.

***LIBCRTAUT:** The public authority for the page segment is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the page segment). The public authority is determined when the page segment is created. If the CRTAUT value for

the library changes after the page segment is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the page segment.

***USE:** The user can perform basic operations on the page segment, such as running a program or reading a file. The user cannot change the page segment. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the page segment.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the page segment. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MBRTXT:** The text is taken from the file member used to create the page segment. Text can be added or changed in a database member by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** No text is specified.

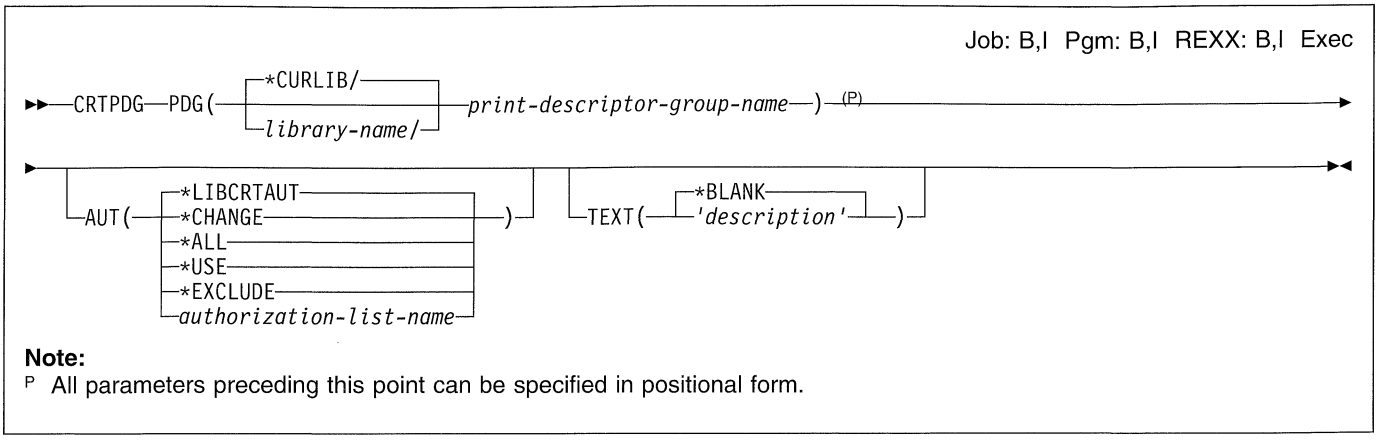
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTPAGSEG   PAGSEG(MYLIB/PAGSEG1)
  FILE(*LIBL/PAGSGMTS)  MBR(*PAGSEG)
  AUT(*ALL)  TEXT('canned paragraph 1')
```

This command creates the page segment PAGSEG1 in MYLIB and uses the PAGSGMNTS member, PAGSEG1 as input to the command. Specifying *ALL for the AUT parameter allows the user to perform most object-oriented commands against it. The text contains the description of the object.

CRTPDG (Create Print Descriptor Group) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Print Descriptor Group (CRTPDG) command creates a space on the AS/400 system into which information about a print descriptor group and its associated print descriptor names can be stored.

Required Parameter

PDG

Specifies the qualified name of the print descriptor group being created.

The name of the descriptor group can be qualified by one of the following library values:

***CURLIB:** The descriptor group is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the descriptor group is created.

print-descriptor-group-name: Specify the name of the print descriptor group being created.

Optional Parameters

AUT

Specifies the authority given to users who do not have specific authority to the print descriptor group, who are not on an authorization list, and whose user group has no specific authority to the print descriptor group. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the print descriptor group is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the print descriptor group). The public authority is determined when the print descriptor group is created. If the CRTAUT value for the library changes after the print descriptor group is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the print descriptor group.

***USE:** The user can perform basic operations on the print descriptor group, such as running a program or reading a file. The user cannot change the print descriptor group. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the print descriptor group.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the print descriptor group. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** No text is specified.

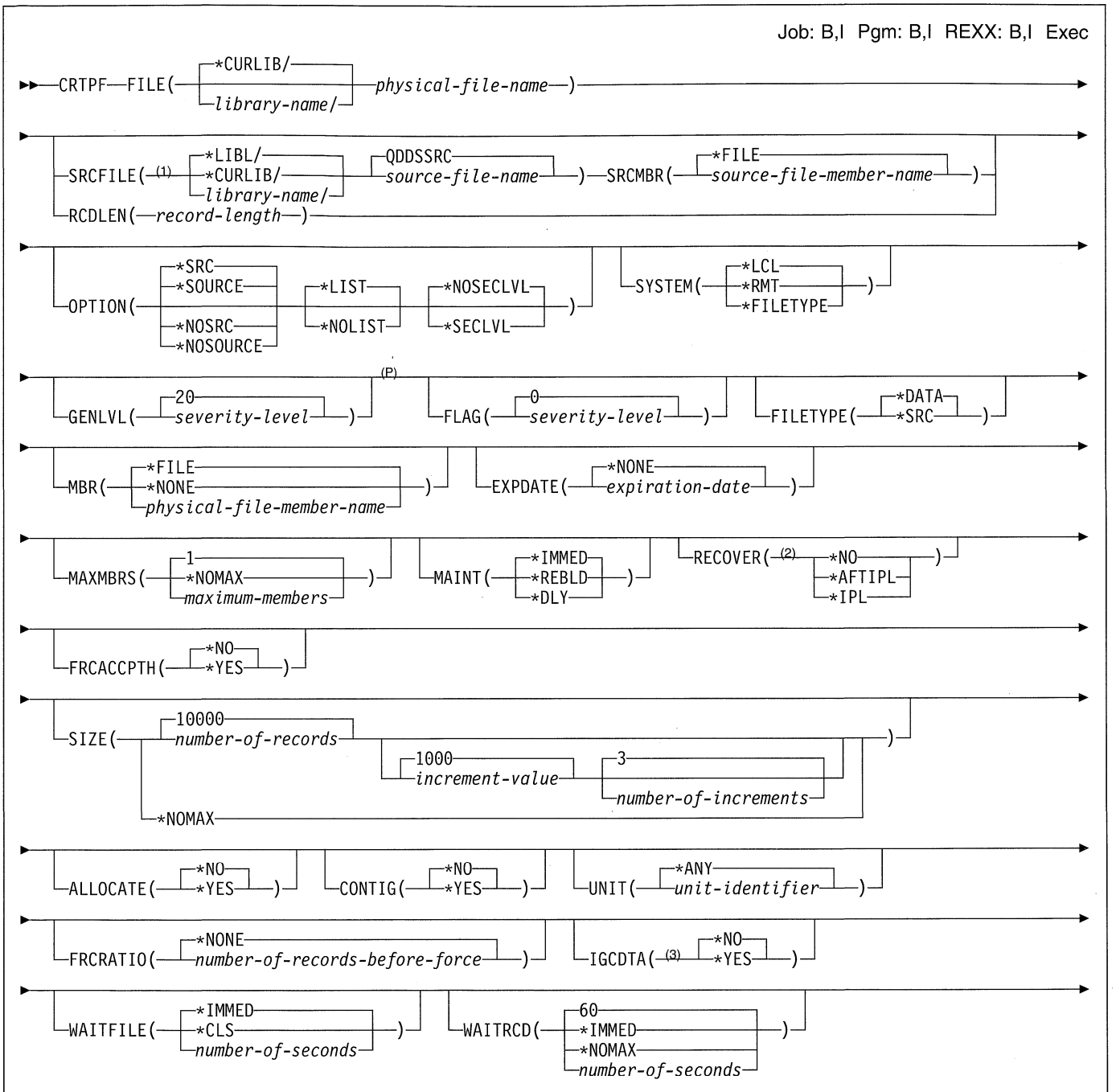
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTPDG    PDG(LETTERS)
```

This command creates print descriptor group LETTERS.

CRTPF (Create Physical File) Command



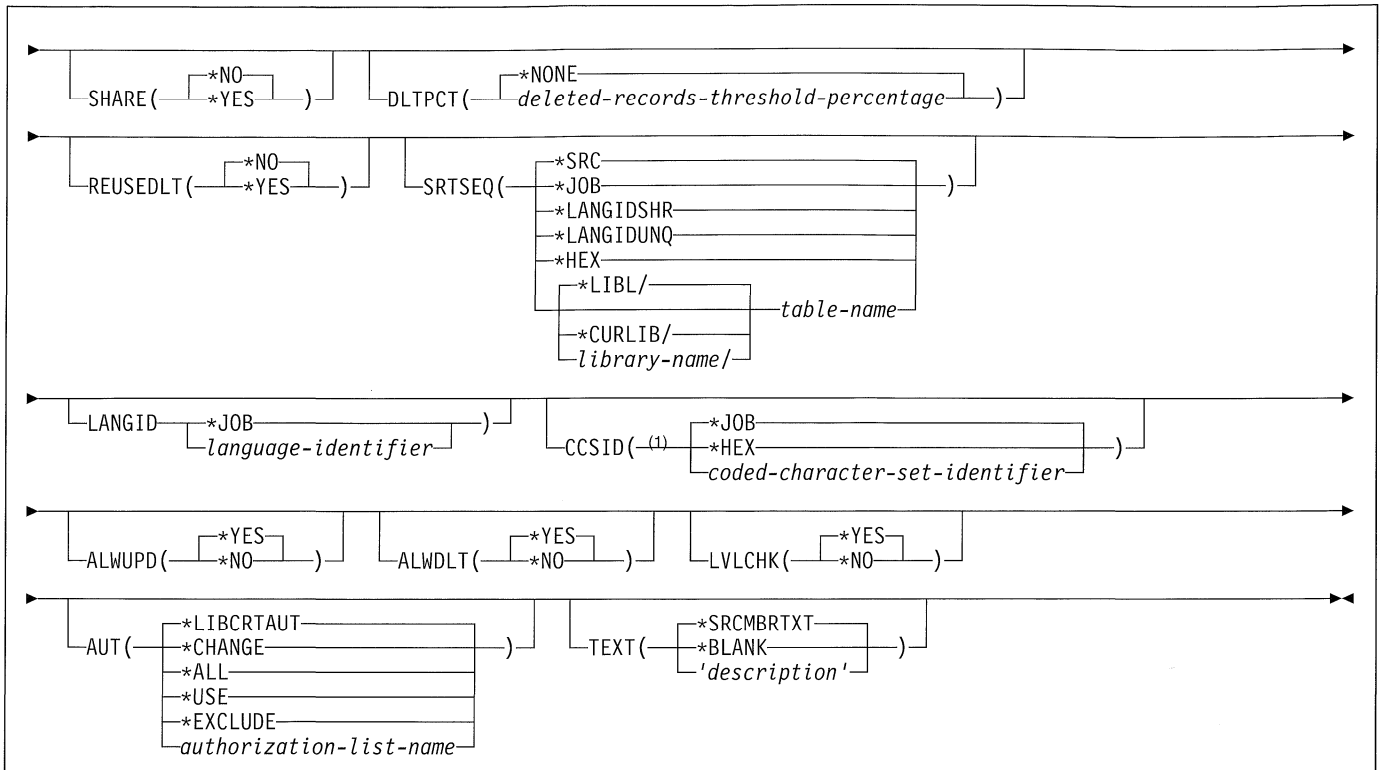
Notes:

1 To code the following parameters positionally, the user must code them in this order, using *N for those not being specified: SRCFILE, SRCMBR, and RCDLEN.

P All parameters preceding this point can be specified in positional form.

2 Refer to the parameter description for the default action for this parameter.

3 DBCS systems only.



Note:
1 If the CCSID parameter is specified, the RCDLEN parameter must also be specified.

Purpose

The Create Physical File (CRTPF) command creates a physical file in the database. A physical file is created from the file description parameters in the CRTPF command and (optionally) from a previously entered data description specification (DDS) source file that contains the source description of the file. If the desired physical file has a record format with only one character field in arrival sequence or if the file is a source file, a DDS source file is not needed. To override attributes of the file after it has been created, use the Override Database File (OVRDBF) command before the file is opened. To change attributes of the file after it has been created, use the Change Physical File (CHGPF) command.

Restriction: A saved keyed physical file can contain up to 3999 members. A saved nonkeyed physical file can contain up to 7999 members. A single save operation can save up to 8000 objects. Information on how many internal objects are saved for each OS/400 system object type is in the *Advanced Backup and Recovery Guide*.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the file can be qualified by one of the following library values:

***CURLIB:** The file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the file is created.

physical-file-name: Specify the name of the physical file.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file used when the physical file is created. The source file contains the specifications that describe the record format and its fields, and the access path for the file and its members. The specifications that are made in DDS are described in the *Database Guide* and *DDS Reference*.

If SRCFILE is specified, the RCDLEN parameter cannot be specified.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The source file, QDDSSRC, contains the DDS used to create the physical file.

source-file-name: Specify the name of the source file that contains the DDS used to create the physical file.

RCDLEN

Specifies the length (in bytes) of the records stored in the physical file. If RCDLEN and FILETYPE(*DATA) are specified, the physical file is created with a record format that has only one field. The file is then restricted to an arrival sequence access path. The record format and the field are both assigned the same name as that of the file, specified in the FILE parameter. A value ranging from 1 through 32766 bytes can be specified for the record length.

If RCDLEN and FILETYPE(*SRC) are specified, the record format has three fields: source sequence number, date, and source statement. The RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and one position for source start, which are required in each record. These fields are defined with fixed attributes and names. If records are copied into the file by the CPYF command and the records are longer than the length specified, the records are truncated on the right.

If RCDLEN is specified, SRCFILE and SRCMBR cannot be specified; RCDLEN is used to specify a fixed record length for the record format when a source file is not needed (when only one field exists in each record or when the file being created is a source file). The high-level language program that processes the file must describe the fields in the record in the program.

Double-Byte Character Set Considerations:

If IGCDTA(*NO) is specified, the field is assigned the data type of character whose length is the same as the record length specified. A value ranging from 1 to 32766 bytes can be specified for the record length. If IGCDTA(*YES) is specified, the field is assigned the data type of DBCS-open and a value ranging from 4 to 32766 can be specified.

The RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and four positions for source start when FILETYPE(*SRC) and IGCDTA(*YES) are specified.

SRCMBR

Specifies the name of the source file member that contains the DDS for the physical file being created; the member is in the source file specified in the SRCFILE parameter (or its default, QDDSSRC). If SRCMBR is specified, RCDLEN cannot be specified.

***FILE:** The source file member name is the same as that of the physical file being created.

source-file-member-name: Specify the name of the member in the source file specified by SRCFILE used to create the physical file.

OPTION

Specifies the type of printout produced when the file is created. Up to three of the following values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value will be used.

Note: The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

Program Creation Options

***SRC** or ***SOURCE:** A printout is created of the source statements used to create the file, and of errors that occur.

***NOSRC** or ***NOSOURCE:** No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the keyword or record format that caused the error.

Source Listing Options

***LIST:** An expanded source printout is created, showing a detailed list of the file specifications that result from the source statements and references to other file descriptions.

***NOLIST:** An expanded source printout is not created.

***NOSECLVL:** The messages section of the DDS printout does not contain the second-level message for the errors found during DDS processing.

***SECLVL:** Second-level message text is included in the source listing.

SYSTEM

Specifies the system on which the physical file is created.

***LCL:** The physical file is created on the local system.

***RMT:** The physical file is created on a remote system using distributed data management (DDM). The physical file name specified on the FILE parameter must be the name of the DDM file (created using the CRTDDMF command). The DDM file contains the name of the physical file being created (RMTFILE parameter on the CRTDDMF command) and the name of the remote system (RMTLOCNAME parameter on the CRTDDMF command) on which the file is created.

***FILETYPE:** If the name specified on the FILE parameter is a DDM file, the physical file is created on the remote system specified by CRTDDMF(RMTLOCNAME) for the DDM file. Otherwise, the name on the FILE parameter cannot be the same as an existing file, since

a physical file of that name is created on the local system.

GENLVL

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

Note: This parameter applies only to messages created while processing the DDS source. Messages created elsewhere in the file creation process are not affected by this parameter.

20: If errors occur in the DDS source file with a severity level greater than or equal to 20, the file is not created.

severity-level: Specify a severity level ranging from 0 through 30. The file is not created if the severity level specified for this parameter equals 0 or is less than the severity level that occurs in the data description specifications (DDS) source. This value must be greater than or equal to value specified on the FLAG parameter.

FLAG

Specifies the minimum severity level of messages to be listed in the DDS source listing.

0: All messages are listed.

severity-level: Specify the minimum severity level of messages to be listed. Valid values range from 0 through 30. The severity level specified must be less than or equal to the severity level specified on the GENLVL parameter.

FILETYPE

Specifies whether each member of the physical file being created contains data records or contains source records (statements) for a program or another file. The file can contain, for example, RPG source statements for an RPG program or DDS source statements for another physical, logical, or device file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: FILETYPE(*SRC) is specified only when including DDS field definitions in the source file. Otherwise, use the Create Source Physical File (CRTSRCPF) command to create a source file.

***DATA:** The physical file contains data records.

***SRC:** The physical file contains source records.

MBR

Specifies the name of the physical file member added when the physical file is created. Other members can be added to the file after it is created by using the Add Physical File Member (ADDPFM) command.

***FILE:** The member being added has the same name (specified in the FILE parameter) as the physical file that contains the member.

***NONE:** No member is added when the file is created.

physical-file-member-name: Specify the name of the member added when the physical file is created.

EXPDATE

Specifies the expiration date. The files cannot be overwritten until the expiration date. The expiration date must be later than or equal to the current date.

Note: An attempt to open a file member that has an expiration date that has been exceeded causes an error message to be sent to the user. The RMVM command is used to remove the member from the file.

***NONE:** No expiration date is specified.

expiration-date: Specify the date after which the physical file member is not used. The date must be in the format specified by the QDATFMT and QDATSEP job attributes. The date must be enclosed in apostrophes if special characters are used in the format.

MAXMBRS

Specifies the maximum number of members that the physical file being created can have at any time.

1: Only one member can be contained in the file.

***NOMAX:** The system maximum is used.

maximum-members: Specify the value for the maximum number of members that the physical file can have. Valid values range from 1 through 32767.

MAINT

Specifies, for files with keyed sequence access paths only, the type of access path maintenance used for members of the physical file.

***IMMED:** The access path is maintained for each physical file member whether the source physical file is opened or closed. The access path is changed whenever a record is updated, added to, or deleted from a member of this file or a logical file member based on a member of this file.

***REBLD:** The access path is completely rebuilt when a file member is opened during the running of the program. The access path is continuously maintained until the member is closed; the access path maintenance is then ended. *REBLD is not valid for access paths that require unique key values.

***DLY:** The maintenance of the access path is delayed until the physical file member is opened for use. Then, the access path is changed only for records that have been added, deleted, or changed since the file was last opened. While the file is *open*, changes made to its members are immediately reflected in the access paths of those members, no matter what is specified for MAINT. To prevent a lengthy rebuild time when the file is opened, *DLY should be specified only when the number of changes to the access path between successive opens are small; that is, when the file is opened frequently or when the key fields in records for this access path change infrequently. *DLY is not valid for access paths that require unique key values.

If the number of changes between a close and the next open reaches approximately 10 percent of the access path size, the system stops saving changes and the access path is completely rebuilt the next time the file is opened.

RECOVER

Specifies, for files having immediate or delayed maintenance on their access paths, when recovery processing of the file is performed after a system failure has occurred while the access path was being changed. This parameter is valid only for a file with a keyed access path.

If *IMMED is specified for the MAINT parameter, the access path can be rebuilt during initial program load (IPL) (before any user can run a job), or after IPL has ended (during jobs running at the same time), or when the file is next opened. While the access path is being rebuilt, the file cannot be used by any job.

During the IPL, an Override Access Path Recovery display lists those access paths that must be recovered and the RECOVER parameter value for each access path. The user can override the RECOVER parameter value on this display. More information on access paths is in the *Advanced Backup and Recovery Guide*.

If *REBLD is specified for the MAINT parameter, the access path is rebuilt the next time its file is opened.

***NO:** The access path of the file is not rebuilt during or after an IPL. The file's access path, if not valid, is rebuilt when the file is next opened.

Note: *NO is the default for all files that do not require unique keys.

***AFTIPL:** The file's access path is rebuilt after the completion of the IPL. This option allows other jobs not using this file to start processing immediately after the completion of the IPL. If a job tries to allocate the file while its access path is being rebuilt, a file open exception occurs.

Note: *AFTIPL is the default for all files that require unique keys.

***IPL:** The file's access path is rebuilt during the IPL. This ensures that the file's access path is rebuilt before the first user program tries to use it; however, no jobs can start running until after all files that specify RECOVER(*IPL) have their access paths rebuilt.

FRCACCPTH

Specifies, for files with keyed access paths only, whether access path changes are forced to auxiliary storage along with the associated records in the file whenever the access path is changed. FRCACCPTH(*YES) minimizes (but does not remove) the possibility that an abnormal job end may cause damage to the access path that requires it to be rebuilt.

***NO:** The access path and changed records are not forced to auxiliary storage whenever the access path is changed.

***YES:** The access path and changed records are forced to auxiliary storage whenever the access path is changed. If FRCACCPTH(*YES) is specified, MAINT(*REBLD) cannot be specified.

FRCACCPTH(*YES) slows the response time of the system if the access path is changed in an interactive job. If the access path is changed frequently, the overall performance of the system is decreased.

SIZE

Specifies the *initial* number of records in each member of the file, the number of records in each increment that can be automatically added to the member size, and the number of times the increment can be automatically applied. The number of records for each file member is specified as the number of records that can be placed in it (this number includes any deleted records).

When the maximum number of records has been reached, a message (stating that the member is full) is sent to the system operator, giving the choice of ending the request or extending the member size. The operator can extend the member by 10% or by the number of records specified as the increment value, whichever is greater, each time the message is received.

A list of three values is specified to indicate the initial size of each member and the automatic extensions that can be added when needed, or *NOMAX can be specified. If SIZE is not specified, SIZE(10000 1000 3) is assumed by the system.

Element 1: Number of Records

Use one of the following to specify the initial number of records in the member before automatic extension of the member occurs. The ALLOCATE parameter determines when the required space for the initial number of records is allocated. If *YES is specified, the space is allocated when a new member is added. If *NO is specified, the initial space is allocated as determined internally by the system.

10000: Initially, up to 10,000 records can be inserted into each member of the file before any extension occurs.

number-of-records: Specify the number of records (ranging from 1 through 16777215) that can be inserted before an automatic extension occurs. If automatic extensions are not wanted, enter zeros for the second and third values in the list.

Element 2: Increment Value

Use one of the following to specify the number of records that can be additionally inserted in the member when the initial member size is exceeded and an automatic extension occurs. The minimum size of an increment is 10% of the size of the member at the time the maximum number of records is reached.

1000: The member size is increased by 10% or 1000 records, whichever is greater.

increment-value: Specify the number of additional records (ranging from 0 through 32767) which, if greater than 10% of the size of the member when the maximum number of records is reached, are to be added to the member during an automatic extension.

If the number specified is not greater than 10% of the member size and not equal to zero, the member size is increased by 10%.

Specify 0 to prevent automatic extensions. This value must be 0 if the value for the number of increments is 0.

Element 3: Maximum Number of Increments

Use one of the following to specify the maximum number of increments that are automatically added to the member. If 0 is specified for the increment amount (element 2), the number of increments are not specified; 0 is the default value instead of 3 (a message is sent to the user issuing the command).

3: Up to three increments is automatically added to the member size.

number-of-increments: Specify the maximum number of increments, ranging from 0 through 32767, that are automatically added to the member. To prevent automatic extensions, specify a value of 0.

Other Single Values

***NOMAX:** The system maximum is used.

ALLOCATE

Specifies whether storage space is allocated for the initial number of records (SIZE parameter) for each physical file member when it is added. The allocation provides enough space to hold the number of records specified by the SIZE parameter. Allocations that occur when a record cannot be added to a member without exceeding its capacity are determined by the system and by the SIZE parameter values.

***NO:** When a new member is added, the system determines whether additional space is needed and allocates that amount.

***YES:** The amount of storage space specified in the first value of the SIZE parameter is allocated each time a new member is added. If that amount of storage space is unavailable, the member is not added, and a message is sent to the user. If this parameter value is used, SIZE(*NOMAX) cannot be specified.

CONTIG

Specifies whether records in the initial allocation in each physical file member are stored contiguously (next to each other) on auxiliary storage. If so, and the necessary contiguous space is unavailable, the system sends a message to the job log and allocates the storage space noncontiguously. The file is still entirely usable. This parameter does not affect additional allocations that might be needed later, which would probably be noncontiguous.

***NO:** The storage space for each member does not have to be contiguous.

***YES:** The system allocates contiguous space for each member of the physical file being added. If it cannot, the user is notified and a message is put in the job log. The affected member is still added, even if the storage space is allocated noncontiguously. The member is just as usable in noncontiguous form. If *YES is specified for CONTIG, then ALLOCATE(*YES) must also be specified.

UNIT

Specifies whether a file is stored on a specific auxiliary storage unit. The system attempts to allocate the storage space for the file and for all its members and their associated access paths on the specified unit. This includes the initial allocation when each member is added and any extensions that occur later for each member in the file. If the system cannot allocate the storage space for each member on the specified unit, it allocates the space on any available unit and sends a message to the job log. The file is entirely usable in all cases.

The unit identifier is a number ranging from 1 through 255, assigned when a new disk device is configured. The configured disk device can be shown and changed by using the Work with Disk Devices display from the Start System Services Tool (STRSST) command. More information on System Services Tool (SST) is in the *Advanced Backup and Recovery Guide*.

***ANY:** The storage space for the file and its members is allocated on any available auxiliary storage unit.

unit-identifier: Specify the storage unit on which the system attempts to allocate the storage space for the file's access path.

If the unit specified is part of any user auxiliary storage pool (ASP) (ASP 2 through 16), the system allocates space from ASP 1, the system auxiliary storage pool.

FRCRATIO

Specifies the number of inserted, updated, or deleted records that are processed before they are forced to auxiliary (permanent) storage.

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions." If this physical file is being journaled, either a large number or *NONE should be used. *NONE may cause long synchronization of the journal and physical files. More information on journal management is in the *Advanced Backup and Recovery Guide*.

***NONE:** There is no force write ratio; the system determines when the records are written to auxiliary storage.

number-of-records-before-force: Specify the number of inserted, updated, or deleted records processed before being explicitly forced to auxiliary storage.

IGCDTA

Specifies whether the program-defined physical files can contain double-byte character set (DBCS) data.

Note: This parameter has no meaning with DDS files, because the use of DBCS data is specified in the DDS.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

Double-Byte Character Set Considerations:

If the user creates a physical file and specifies the RCDLEN parameter, the system creates a default record format.

- If IGCDTA(*YES) is specified, the default record format can contain DBCS data (as if the record were specified with the DBCS-open (O in column 35 of DDS specification) data type).
- If IGCDTA(*NO) is specified, the default record format cannot contain DBCS data (as if the record were specified with the character (A or blank in column 35 of DDS specification) data type).

The system ignores the IGCDTA parameter value if a value for the RCDLEN parameter is not specified.

The user cannot override the IGCDTA value for a physical file.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds a program waits for the file resources to be allocated to the job. Valid values range from 1 through 32767 seconds.

WAITRCD

Specifies the number of seconds that a program waits for a record to be updated or deleted, or for a record read in the commitment control environment with LCKLVL(*ALL) specified. More information on record locking is in the *Database Guide*. If the record is not allocated in the specified wait time, an error message is sent to the program.

60: The program waits for 60 seconds.

***IMMED:** The program does not wait; when a record is locked, an immediate allocation of the record is required.

***NOMAX:** The system maximum is used.

number-of-seconds: Specify the number of seconds a program waits for the file resources to be allocated to the job. Valid values range from 1 through 32767 seconds.

SHARE

Specifies whether the open data path (ODP) for the physical file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

DLTPCT

Specifies the maximum percentage of deleted records allowed for any member in the physical file. The percentage is based on the number of deleted records compared with the total record count in a member. The percentage check is made when a member of the file is closed or a logical file member based on a member of the file is closed. If the number of deleted records exceeds the percentage, a message is sent to the system history log (QHST) to inform the user.

***NONE:** No percentage is specified; the number of deleted records in the file members is not checked when a member is closed.

deleted-records-threshold-percentage: Specify the largest percentage of deleted records in any member in the file. Valid values range from 1 through 100. If this percentage is exceeded, a message is sent to the system history log (QHST) when the file is closed.

REUSEDLT

Specifies whether the space used by deleted data entries is reclaimed by future insert requests.

Note: If a *YES value is specified for this parameter, the key ordering attribute for the physical file must be allowed to default or must be FCFO

instead of FIFO or LIFO. If *YES is specified for this parameter, the key ordering attributes of FIFO and LIFO are not allowed.

***NO:** The file does not reclaim space used by deleted data entries.

***YES:** The file reclaims space used by deleted data entries. More information about the algorithm used to reclaim the deleted data is in the *Database Guide*.

SRTSEQ

Specifies the sort sequence used for this file. The sort sequence is used with the LANGID and CCSID parameters to determine which sort sequence table is used.

***SRC:** The table specified in the data description specification (DDS) on the ALTSEQ keyword is used. If ALTSEQ is not used in the DDS, use the value specified for *JOB on this parameter.

***JOB:** The sort sequence must be resolved when the object is created.

***LANGIDSHR:** The sort sequence table can contain the same weight for multiple characters, and is the shared-weight table associated with the language specified in the LANGID parameter.

***LANGIDUNQ:** The sort sequence table must contain a unique weight for each character in the code page.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.

The name of the table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify a table name.

LANGID

Specifies the language identifier used when *LANGIDSHR or *LANGIDUNQ is specified on the SRTSEQ parameter. The language identifier is used with the SRTSEQ and CCSID parameters to determine which sort sequence table the file will use.

***JOB:** The language identifier specified for the job is used.

language-identifier: Specify a language identifier.

CCSID

Specifies the coded character set identifier (CCSID) used to describe character data in the fields of the file. If this parameter is specified, the RCDLEN parameter and FILETYPE(*SRC) must also be specified.

A CCSID is a 16-bit number identifying a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and additional coding-related information that uniquely identifies the coded graphic representation used.

***JOB:** The current job's CCSID is used.

***HEX:** The CCSID 65535 is used, which indicates that the character data in the fields is treated as bit data and is not converted.

coded-character-set-identifier: Specify the CCSID to be used. More information on valid CCSIDs is in the *National Language Support Planning Guide*.

ALWUPD

Specifies whether records can be updated in the physical file. Records in a logical file can be updated only when the records in each physical file, on which the logical file is based, can be updated.

***YES:** Records can be updated in the physical file.

***NO:** Records cannot be updated in this physical file or in any logical file built over this physical file.

ALWDLT

Specifies whether records can be deleted from the physical file. Records in a logical file can be deleted only when the records in each physical file on which the logical file is based can be deleted.

***YES:** Records can be deleted in this physical file.

***NO:** Records cannot be deleted in this physical file or from any logical file built over this physical file.

LVLCHK

Specifies whether the record format level identifiers in the program are checked against those in the logical file when the file is opened. If so, the record format identifiers in the program must match those in the logical file. This value can be overridden by the Override with Database File (OVRDBF) command at run time.

***YES:** The level identifiers of the record formats are checked when the file is opened. If the level identifiers do not match, an error message is sent to the program requesting the open, and the file is not opened.

***NO:** The level identifiers are not checked when the file is opened.

AUT

Specifies the authority given to users who do not have specific authority to the physical file, who are not on an authorization list, and whose user group has no specific authority to the physical file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the physical file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the physical file). The public authority is determined when the physical file is created. If the CRTAUT value for the library

changes after the physical file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the physical file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the physical file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the physical file.

***USE:** The user can perform basic operations on the physical file, such as running a program or reading a file. The user cannot change the physical file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the physical file.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the physical file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member being used to create the physical file. If the source file is a database file, the text is taken from the source member. Text can be added or changed for a database source member by using the Source Entry Utility or by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the source file is an inline file or a device file, the text is blank.

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a Physical File

```
CRTPF   FILE(PAYLIB/PAYTXS)  SRCFILE(SRCLIB/PAYTXS)
  MBR(*NONE)  MAXMBRS(5)
```

This command creates a physical file named PAYTXS in the PAYLIB library. The source descriptions in the member PAYTXS in source file PAYTXS in the SRCLIB library are used to create the physical file. The file is created without members (*NONE was specified); therefore, no data can be put into the file until a member is added later. As many as five members can be contained in the file.

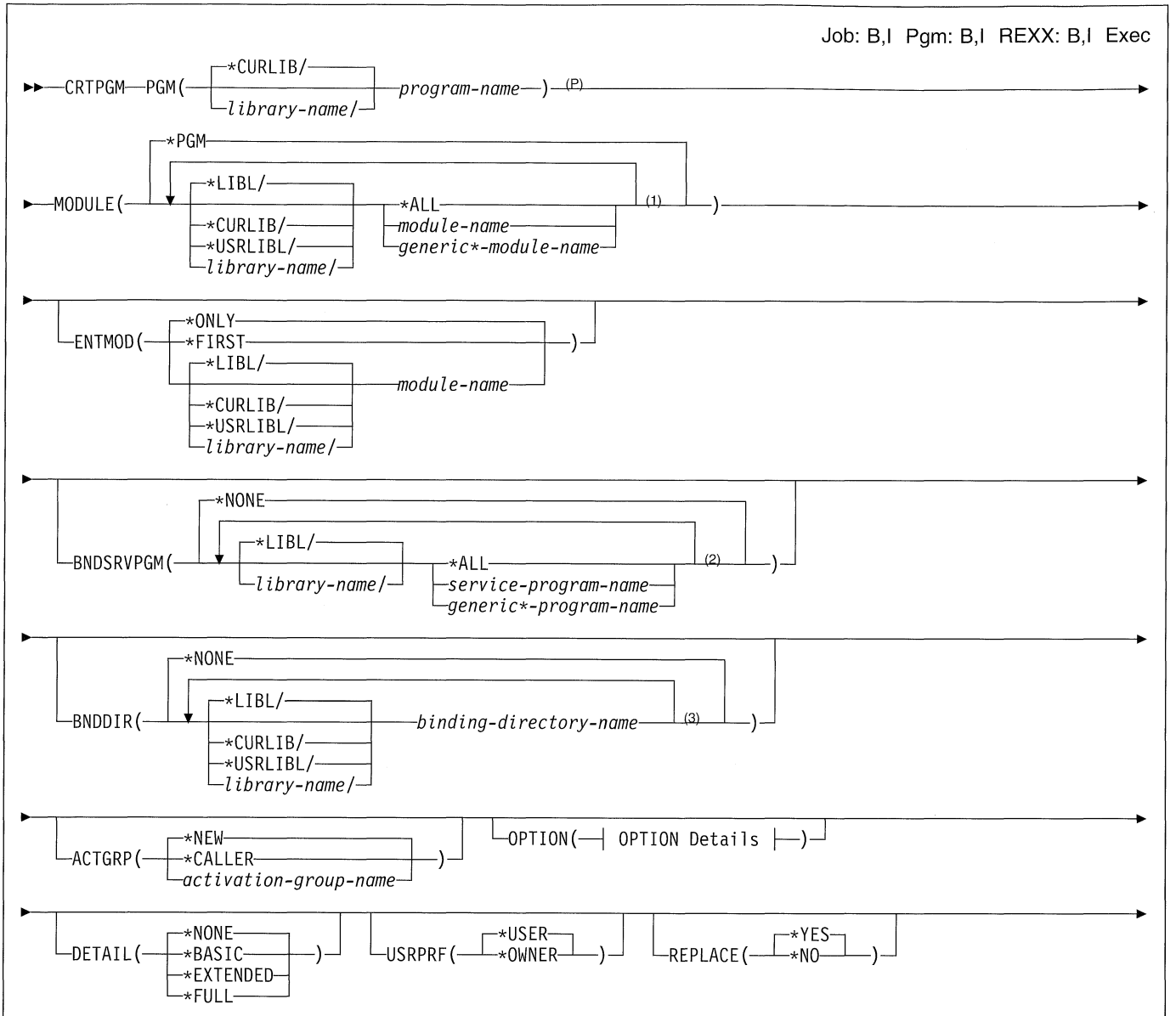
By default, each file member added later will contain data records. The access path of each member is continuously maintained. Each member can have up to 10,000 records before automatic extensions (three increments maximum) occur that add 1000 records to the capacity of the member. Storage space for each member is allocated only as needed, with no restrictions on which unit is used or whether the space is contiguous; there is no initial storage allocation. The public has object operational, read, add, delete, and update authority for the file.

Example 2: Creating a Physical File and Member

```
CRTPF   FILE(ORDERCTL/ORDERS)
  SRCFILE(ORDERCTL/ORDERSRC)
  SRCMBR(MFGORD)  MAXMBRS(50)  SIZE(1000 100 5)
  ALLOCATE(*YES)  UNIT(01)
```

This command creates a physical file and physical file member, both named ORDERS in the ORDERCTL library. The file and its member are created from the MFGORD source member of the ORDERSRC source file in the same library. All records placed in the file are stored on auxiliary storage unit 01, but the space does not have to be contiguous. Up to 50 members can be contained in the file. The initial allocation of storage provides for up to 1000 records, and up to five increments of additional space for 100 records each can be added automatically. These allocation values also apply to each member of this physical file that is added later.

CRTPGM (Create Program) Command



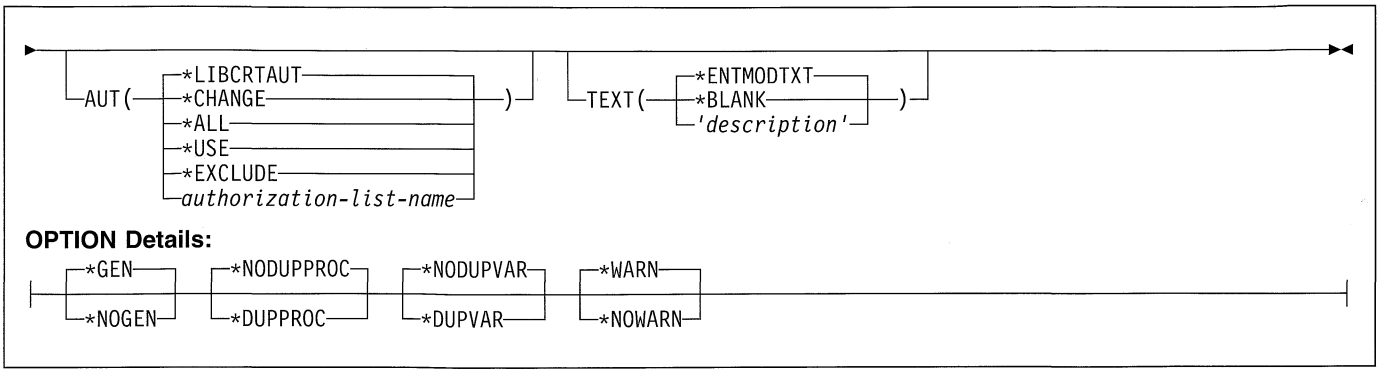
Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 150 module entries.

2 A maximum of 50 service program entries for binding.

3 A maximum of 50 binding directory entries.



Purpose

The Create Program (CRTPGM) command creates a bound program from a set of modules and binding directories.

Restrictions:

1. You must have *CHANGE authority for the library where the program is being created.
2. You must have *USE authority to the specified modules, service programs, and binding directories.

Required Parameter

PGM

Specifies the qualified name of the program object created.

The name of the program object can be qualified by one of the following library values:

***CURLIB:** The program object is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the program object is created.

program-name: Specify the name of the created program.

MODULE

Specifies the list of modules that are copied and bound together to create the program object. If duplicate module and library specifications are found, only the first instance of the duplicate module and library is used. Modules in this list are copied into the final program object.

***PGM:** The name specified on the PGM parameter is used as the module object name. The library specified on the PGM parameter is also used.

The name of the module list can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

***ALL:** Find all module objects in the specified library or libraries.

module-name: Specify the name of the module that is copied to create the program object.

generic-module-name:* Specify all module objects starting with the characters preceding the * in the specified library or libraries.

Optional Parameters

ENTMOD

Specifies the module name that contains the program entry procedure specification to be used for this program.

***ONLY:** Only one module, from the list of modules, can have a specification as the program entry procedure. An error is issued if more than one module is found to have a program entry procedure specification.

***FIRST:** The first module found, from the list of modules, that has a program entry procedure specification is selected as the program entry procedure.

The name of the specified module can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

module-name: Specifies the module containing the program entry procedure specification. If this module is not in the list of modules to be included in this program, it is added to the list of modules. If this module does not have a program entry procedure specification, the program is not created.

BNDSRVPGM

Specifies the list of service program exports to examine to ensure they satisfy any module import request. The service program exports are checked only if there are unresolved module import requests not satisfied by the set of module exports.

***NONE:** No service program is specified.

The name of the service program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

library-name: Specify the name of the library to be searched.

Note: QTEMP is not a valid library name for this parameter.

***ALL:** Find all service program objects in the specified library or libraries.

service-program-name: Specify the name of the service program to be examined during symbol resolution.

generic-program-name:* Specify all service program objects starting with the characters preceding the * in the specified library or libraries.

BNDDIR

Specifies the list of binding directories that are used in symbol resolution. The exports of the modules and service programs in the binding directory are only checked if there are unresolved module import requests that the exports from the modules and service programs (specified in the MODULE or BNDSRVPGM parameters) could not satisfy.

***NONE:** No binding directory is specified.

The name of the binding directory can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

binding-directory-name: Specify the name of the binding directory used in symbol resolution.

ACTGRP

Specifies the activation group this program is associated with when it is called. An activation group provides:

- Run-time data structures to support the running of programs
- Addressing protection
- A logical boundary for message creation
- A logical boundary for application cleanup processing

***NEW:** When this program gets called, a new activation group is created. This called program is then associated with the newly created activation group.

***CALLER:** When this program gets called, the program is activated into the caller's activation group.

activation-group-name: Specify the name of the activation group to be used when this program is called.

OPTION

Specifies options to be used when the program object is created.

Program Objects

***GEN:** A program object is generated.

***NOGEN:** A program object is not generated.

Duplicate Procedure Names

***NODUPPROC:** During the symbol resolution phase of the binding process, each procedure name that is exported from the modules and service programs must be unique.

***DUPPROC:** During the symbol resolution phase of the binding process, the procedure names that are exported from the modules and service programs do not have to be unique. When multiple duplicate procedures are allowed, the first exported procedure in the list of specified modules and service programs that matches the import request is the procedure that is selected.

Duplicate Variable Names

***NODUPVAR:** During the symbol resolution phase of the binding process, each variable name that is exported from the modules and service programs must be unique.

***DUPVAR:** During the symbol resolution phase of the binding process, the variable names that are exported from the modules and service programs do not have to be unique. When multiple duplicate variables are allowed, the first exported variable in the list of specified modules and service programs that matches the import request is the variable that is selected.

Issuing Diagnostic Messages

***WARN:** If duplicate variables or procedures are found, a diagnostic message is issued indicating what duplicates were found.

***NOWARN:** If duplicate variables and procedures are found, diagnostic messages are not issued.

DETAIL
Specifies the level of detail to be printed.

***NONE:** A listing is not generated.

***BASIC:** Contains a listing of the options passed to CRTPGM, and processing statistics. This listing also contains the Brief Summary Table.

***EXTENDED:** In addition to the information provided in the *BASIC listing, this listing contains the Extended Summary Table and the Binding Information Listing.

***FULL:** This listing contains the *EXTENDED listing and the Cross-Reference Listing.

Note: If a printed listing is requested, the printer file *LIBL/QSYSPRT is used to generate the listing.

USRPRF
Specifies whether the authority checking done while this program is running includes only the user who is running the program (*USER) or both the user running the program and the program owner (*OWNER). The profiles of the program user or both the program user and the program owner are used to control which objects can be used by the program, including the authority the program has for each object. Only the program owner or a user with QSECOFR authority can change the user profile attribute.

***USER:** The program runs under the user profile of the program's user.

***OWNER:** The user profiles of both the program owner and the program user are used when the program is run.

REPLACE
Specifies whether the existing program is replaced if a program by the same name already exists in the specified library.

***YES:** Replace the existing program by moving it to the QRPLOBJ library.

Note: Both programs must be owned by the same user for the replace to work.

***NO:** No replacement occurs.

AUT
Specifies the authority given to users who do not have specific authority to the program, who are not on an authorization list, and whose user group has no specific authority to the program.

***LIBCRTAUT:** The public authority for the program is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the program). The public authority is determined when the program is created. If the CRTAUT value for the library changes after the program is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the program.

***USE:** The user can perform basic operations on the program, such as running a program or reading a file. The user cannot change the program. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the program.

authorization-list: The authorization list must exist when the object is created. Users are granted authority to the object as specified by the list.

TEXT
Specifies text that briefly describes the program object. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ENTMODTXT:** The text description of the module specified on the ENTMOD parameter is used.

***BLANK:** Text is not specified.

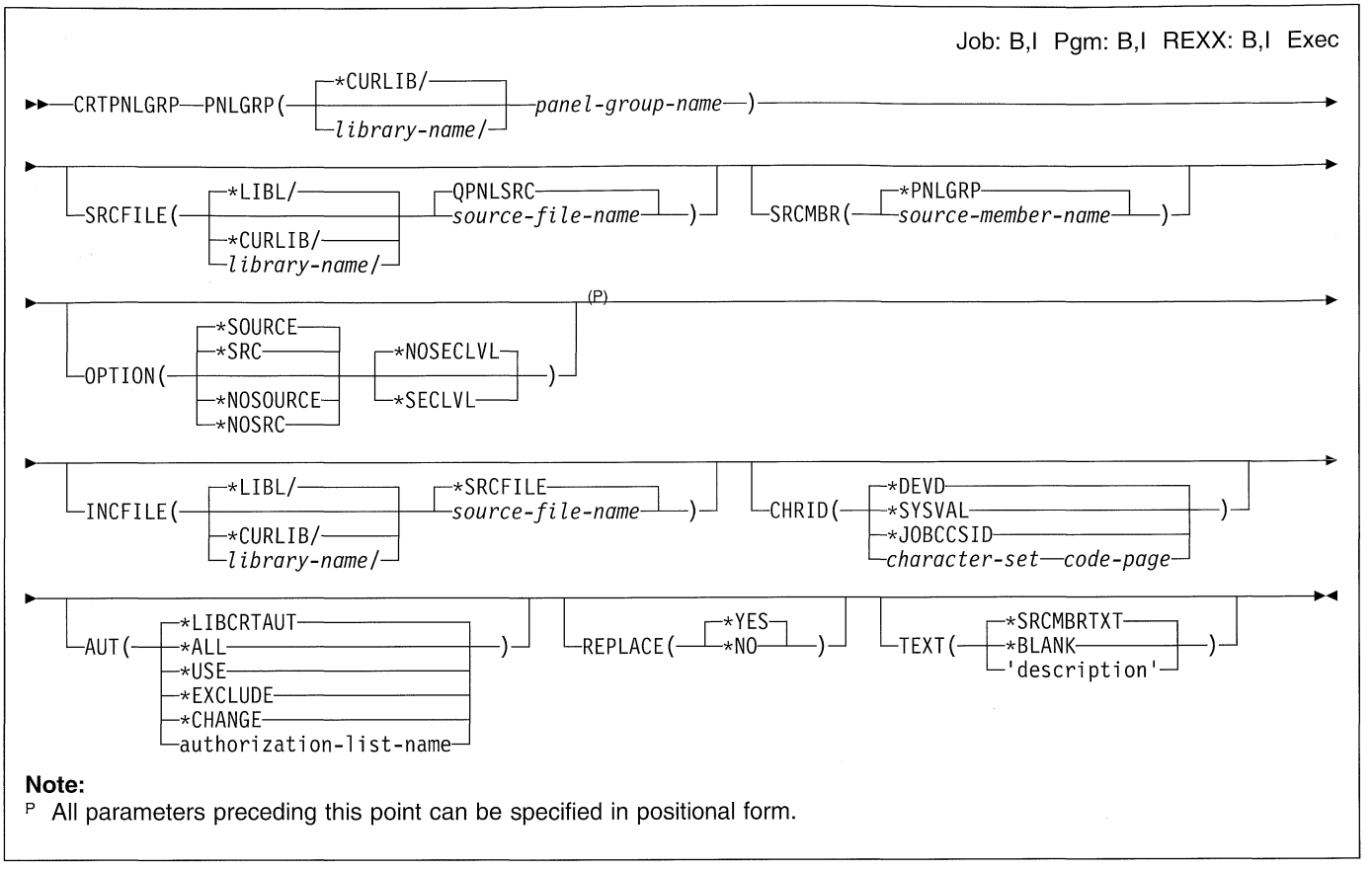
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTPGM   PGM(STAR)
```

This command creates a program object named STAR in the current library or QGPL.

CRTPNLGRP (Create Panel Group) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Panel Group (CRTPNLGRP) command is used to create groups of help panels. Panel groups contain one or more help items used in conjunction with the user's data description specifications (DDS) panels, search index, or CL commands to display help information.

Restrictions: The user must have *READ and *ADD authority for the library where the panel group is being created.

Required Parameter

PNLGRP

Specifies the qualified name of the panel group being created.

The possible library values are:

***CURLIB:** The panel group is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the panel group is created.

panel-group-name: Specify the name of the panel group being created.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file containing the panel group description source statements. The source-file record length ranges from 13 through 92.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QPNLSRC: Source file QPNLSRC contains the panel group description source statements.

source-file-name: Specify the name of the source file that contains the panel group description source statements.

SRCMBR

Specifies the member of the source file containing the panel group description.

***PNLGRP:** The member containing the panel group description has the same name as the panel group specified on the PNLGRP parameter.

source-member-name: Specify the name of the source file member containing the panel group description.

OPTION

Specifies whether a source listing is produced during panel group creation and whether the listing includes message text.

Source Listing Options

***SOURCE or *SRC:** A source listing is produced.

***NOSOURCE or *NOSRC:** No source listing is produced unless errors with a severity level of 30 or greater are detected.

Second-Level Message Text Options

***NOSECLVL:** Second-level message text is not printed with the first-level message text at the end of the listing.

***SECLVL:** Second-level message text is printed with the first-level message text at the end of the listing.

INCFILE

Specifies the name of the source file containing the members being included in the panel group.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***SRCFILE:** The name specified on the SRCFILE parameter is used.

source-file-name: Specify the name of the source file containing the members to be included in the panel group.

Note: If the coded character set identifier (CCSID) of the source file is different than the CCSID of the primary source file specified on the SRCFILE parameter, the CCSID is changed to the CCSID of the primary source file. The CCSID must be the same for all source members of the panel group.

CHRID

Specifies whether the character identifier (graphic character set and code page) of the dialog variables or the panel group is changed to the CCSID value of the job when the command is run.

***DEVD:** No change occurs. The character identifier of the dialog variables is the same as the character identifier of the device.

***SYSVAL:** The character identifier of the dialog variables is changed to the default QCHRID system value for the device.

***JOBCCSID:** The character identifier of the panel group is changed from the CCSID of the source file specified on the SRCFILE parameter to the character identifier of the device.

Element 1: Character Set

character-set: Specify the graphic character set values that match the values of the applications that are to use this panel group, and match the values of the data to be set into dialog variables for the applications.

Element 1: Code Page

code-page: Specify the code page values that match the values of the applications that are to use this panel group, and match the values of the data to be set into dialog variables for the applications.

AUT

Specifies the authority given to users who do not have specific authority to the object, are not on the authorization list, or whose user group has no specific authority for the object.

***LIBCRTAUT:** The authority from the library in which the panel group is being created is used as the authority for the panel group.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the panel group.

***USE:** The user can perform basic operations on the panel group, such as running a program or reading a file. The user cannot change the panel group. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the panel group.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

authorization-list-name: Specify the name of an authorization list.

REPLACE

Specifies whether an existing panel group of the same name in the specified library is replaced.

***YES:** The existing panel group is replaced by moving it to the system library QRPLOBJ.

***NO:** No replacement occurs.

TEXT

Specifies text that briefly describes the panel group. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is obtained from the text associated with the specified source file member.

***BLANK:** Text is not specified.

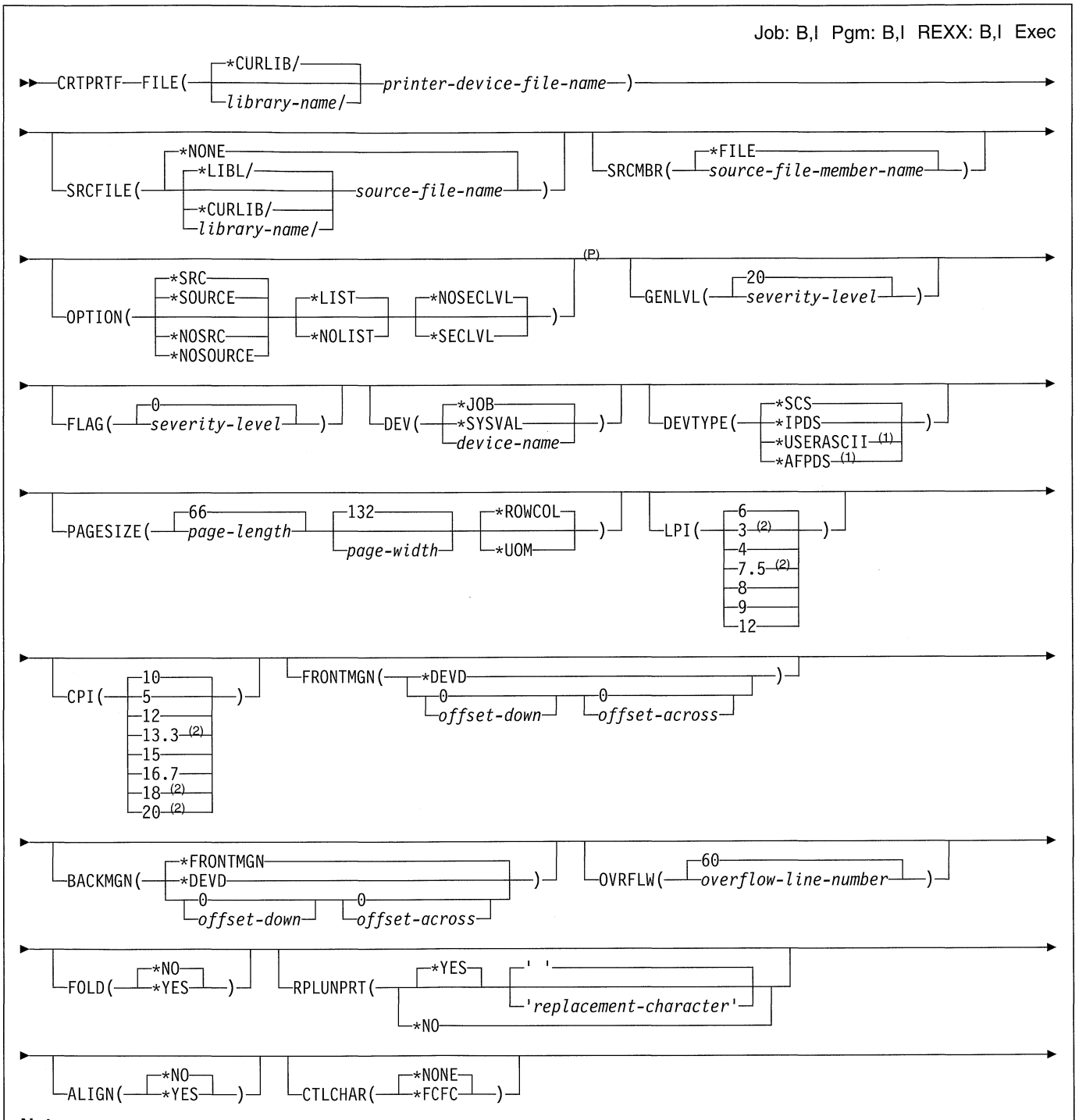
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTPNLGRP   PNLGRP(PAYLIB/PAYROLL)
  SRCFILE(QPNLSRC)  OPTION(*SECLVL)
```

This command creates a panel group named PAYROLL in library PAYLIB, uses source file QPNLSRC in the library list, and prints the second-level message text in the listing.

CRTPRTF (Create Printer File) Command

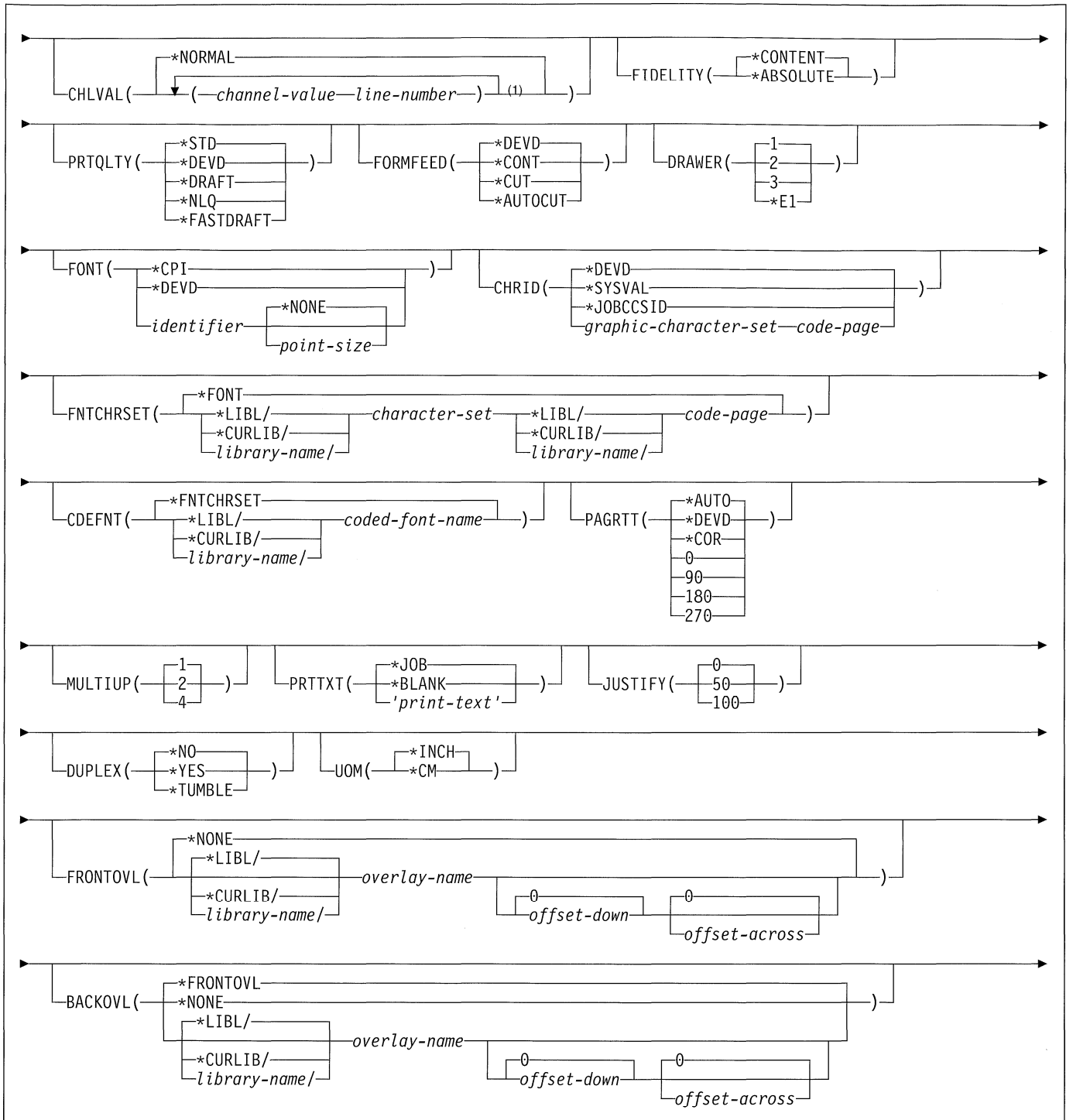


Notes:

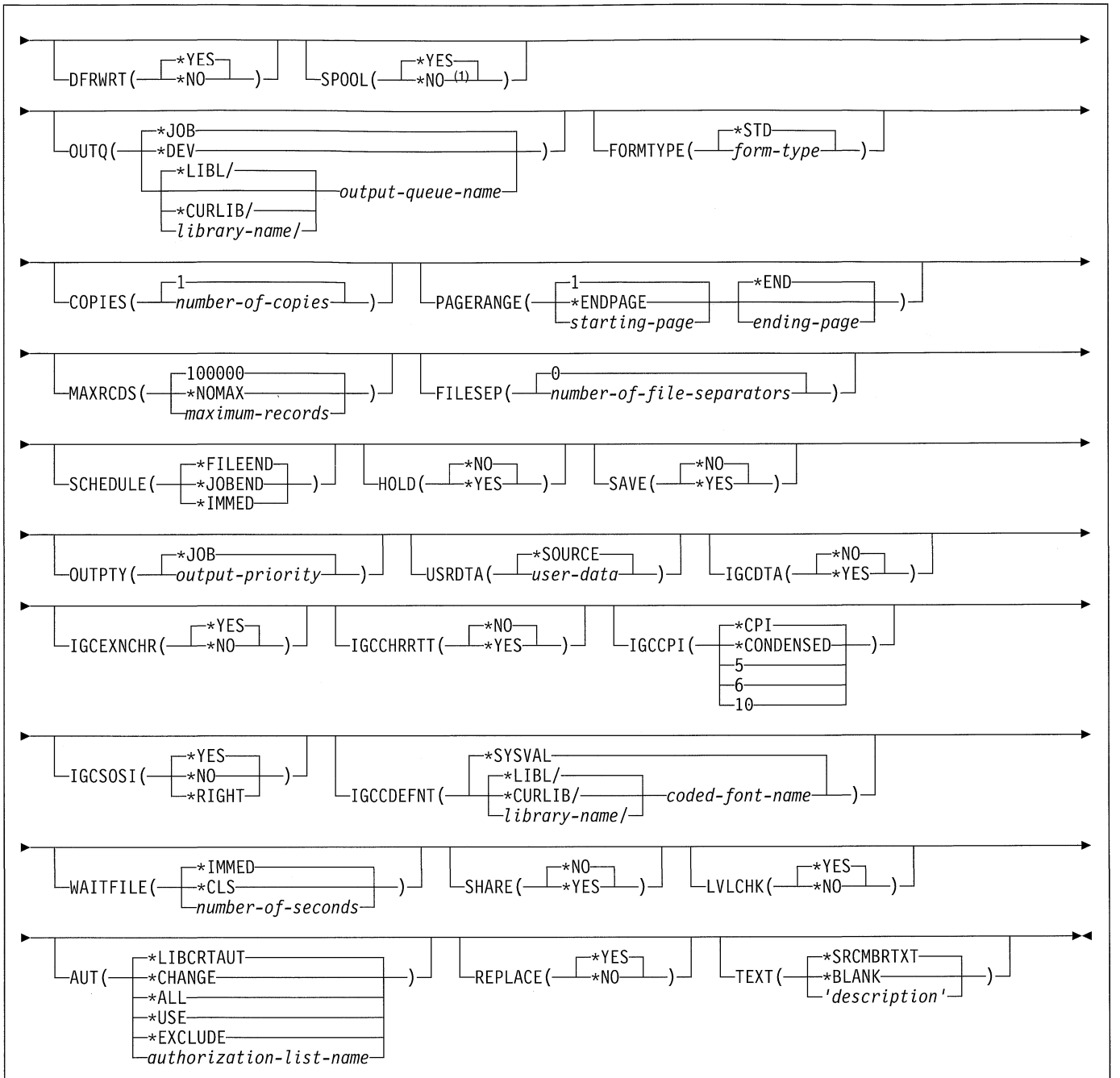
1 This value cannot be specified with SPOOL (*NO).

2 DBCS systems only.

P All parameters preceding this point can be specified in positional form.



Note:
[1] A maximum of 12 repetitions.



Note:
[1] This value cannot be specified with DEVTYPE(*USERASCII) or DEVTYPE(*AFPDS).

Purpose

The Create Printer File (CRTPRTF) command creates a printer device file. The printer file contains the file description, which identifies the device used and specifies the spooling requirements, but does not contain data. The printer file is used to send records to the printer.

The printer file description is made up of information specified in two places: (1) in the source file that contains the data description specifications (DDS), if used; and (2) in the CRTPRTF command. The DDS contains the specifications for each record format in the printer file and for the fields in each record format.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules

of that language; otherwise, the file must be renamed in the program.

The name of the printer file can be qualified by one of the following library values:

***CURLIB:** The printer file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the printer file is created.

printer-device-file-name: Specify the name of the printer file to be created.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file, if one exists, that contains the data description specifications (DDS) for the records in the printer file. More information on the specifications that can be made in DDS is in the *Guide to Programming for Printing* and the *DDS Reference*.

***NONE:** There is no DDS source file for this printer file; the printer file has only one record format with no fields, and the program that uses this printer file must describe the record formats and their fields.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

source-file-name: Specify the name of the source file that contains the DDS for this printer file.

SRCMBR

Specifies the name of the member in the source file that contains the DDS for this printer file.

***FILE:** The source file member name is the same as the device file name specified in the FILE parameter.

source-file-member-name: Specify the name of the member in the source file specified on the SRCFILE parameter that is used to create the printer file.

OPTION

Specifies the type of printout produced when the printer file is created. Up to three of the following values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value is used.

Note: The underlined values for this parameter are *similar* to, but not *actually* default values, and therefore, cannot be changed with the CHGCMDDFT (Change Command Default) command.

Element 1: Source Listing Options:

***SRC** or ***SOURCE:** A printout is created showing the source statements used to create the printer file and any errors that occur.

***NOSRC** or ***NOSOURCE:** No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the parameter or record format that caused the error.

Element 2: Program Listing Options

***LIST:** An expanded source printout is created, showing a detailed list of the file specifications that result from the source statements and references to other file descriptions.

***NOLIST:** An expanded source printout is not created.

Element 3: Second-Level Message Text Options

***NOSECLVL:** The messages section of the printout does not contain the second-level message text for the errors found during processing of DDS source statements.

***SECLVL:** Second-level message text is printed.

GENLVL

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

Note: This parameter applies only to messages issued while processing the DDS source. Messages issued anywhere else in the file creation process are not affected by this parameter.

20: If errors occur in the DDS source with a severity level greater than or equal to 20, the file is not created.

severity-level: Specify a severity level ranging from 0 through 30. The file is not created if the severity level specified for this parameter equals 0 or is less than the severity level that occurs in the data description specifications (DDS) source. This value must be greater than or equal to value specified on the FLAG parameter.

FLAG

Specifies the minimum severity level of messages to be listed in the DDS source listing.

0: All messages are listed.

severity-level: Specify the minimum severity level of messages to be listed. Valid values range from 0 through 30. The severity level specified must be less than or equal to the severity level specified on the GENLVL parameter.

DEV

Specifies the name of a printer device description. For nonspooled output, this identifies the printer device used with the printer file to produce the printed output. For

spooled output, if OUTQ(*DEVD) is also specified, the default output queue for the specified printer is used for the spooled output data. More information is in the *Guide to Programming for Printing*.

***JOB:** The printer device specified in the job description is used.

***SYSVAL:** The value specified in the system value QPRTDEV is used.

device-name: Specify the name of the printer associated with this display station. The printer and the display station must be attached to the same controller. When printing double-byte character set (DBCS) data, specify a DBCS printer (5553 or 5583).

DEVTYPE

Specifies the type of data stream created for a printer file.

***SCS:** An SNA character stream (SCS) is created. This parameter must be specified when using the 3287, 3812 SCS, 3816 SCS, 4214, 4234 SCS, 4245, 5219, 5224, 5225, 5256, 5262, 6252, or 6262 work station printers.

- If *SCS is specified and the spooled printer file is directed to an IPDS printer, the SCS printer file is converted to emulate an IPDS printer file. More information is in the *Guide to Programming for Printing*.

Double-Byte Character Set Consideration:

When using the 5553 and 5583 DBCS-capable printers, DEVTYPE(*SCS) must be specified.

***IPDS:** An intelligent printer data stream (IPDS) is created. This parameter can be specified when using a 3812 IPDS, 3816 IPDS, 3820, 3825, 3827, 3829, 3831, 3835, 3900, 4028, 4224, 4230, or the 4234 IPDS printer.

- If *IPDS is specified and the spooled printer file is directed to a printer other than an IPDS printer, the IPDS printer file is converted to an SCS printer file. More information is in the *Guide to Programming for Printing*.

***USERASCII:** An ASCII data stream is placed on a spooled output queue. The user is responsible for placing the entire hexadecimal data stream in the buffer, since the AS/400 system does not change or validate the values that are passed. This value cannot be specified with SPOOL(*NO).

***AFPDS:** An advanced function print data stream (AFPDS) is created. This parameter can be specified when using the 3812 IPDS, 3816 IPDS, 3820, 3825, 3827, 3829, 3831, 3835, 3900 4028, 4224, 4230, or the 4234 IPDS printer. The printer must be configured with AFP(*YES).

PAGESIZE

Specifies the length and width of the printer forms used by this printer file. The length is specified in lines per page or by the units specified for the UOM parameter. The width is specified in print positions (characters) per line or by the units specified for the UOM parameter.

The page size must be specified with reference to the way the data is printed on the page. For example, if using 8.5 inch wide by 11.0 inch long forms and printing at 6 lines per inch with a 10-pitch font, specify PAGESIZE(66 85) PAGRTT(0). However, to rotate the page, specify the page size for an 11.0 inch wide by 8.5 inch long page and enter PAGESIZE(51 110) PAGRTT(90).

Note: Specify PAGRTT(*AUTO) or PAGRTT(*DEVD) and PRTQLTY(*DRAFT) on this command to enable automatic reduction or rotation if the data does not fit on the paper.

Specify PAGRTT(*COR) on this command to enable automatic reduction whether or not the data fits on the paper.

Element 1: Page Length Value

66: The page length is 66 print lines per page.

page-length: Specify the page length that is used by this printer file. Although a value ranging from 1 through 255 can be specified as the page length, the value specified must not exceed the actual length of the forms used.

More information about the page lengths that are valid for each printer type is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

Element 2: Page Width Value

132: The page width is 132 printed characters per line.

page-width: Specify the page width used by this printer file. The value specified must not exceed the actual width of the forms used.

More information about page width is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

Element 3: Method of Measure

***ROWCOL:** Page length and page width are measured as numbers of rows and columns.

***UOM:** Page length and page width are measured in the units specified on the UOM parameter.

LPI

Specifies the line spacing setting on the printer, in lines per inch, used by this printer file.

The line spacing on the 5256 printer must be set manually. When the lines per inch (LPI) value on this parameter changes (from the value on the previous printer file), an inquiry message is sent to the message queue associated with the printer that requests a change to the LPI value.

The line spacing on the 4214, 4224, 4230, 4234, 4245, and 5262 Printers is set by a print command. These also allow setting the lines per inch spacing on the

control panel of the printer. The lines per inch value must not be set at the printer. If the LPI value is overridden at the control panel, the system overrides the value set with the LPI value of the next printer file received.

More information about the lines per page and lines per inch that are valid for each printer type is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

6: The line spacing on the printer is 6 lines per inch.

3: The line spacing on the printer is 3 lines per inch. This value is valid only for double-byte character set (DBCS) data.

4: The line spacing on the printer is 4 lines per inch.

7.5: The line spacing on the printer is 7.5 lines per inch. This value is valid only for double-byte character set (DBCS) printers.

8: The line spacing on the printer is 8 lines per inch.

Note: When printing double-byte character set (DBCS) data for a file specified with LPI(8), use double spacing. Otherwise, the DBCS data does not print correctly. Alphanumeric data, however, prints correctly in single spacing when LPI(8) is specified.

9: The line spacing on the printer is 9 lines per inch.

12: The line spacing on the printer is 12 lines per inch.

CPI

Specifies the printer character density, in characters per inch (CPI), used by this printer file.

For the printers that support fonts, the value specified in the font special value implies the CPI. If FONT(*CPI) is specified, the font used is based on the CPI value. The following diagram describes the default font ID for each CPI value:

CPI	FONT ID DEFAULT
5	245
10	011
12	087
13.3	204
15	222
16.7	400
18	252
20	281

More information about the characters per page and characters per inch that are valid for each printer type is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

10: Character density is 10 characters per inch.

5: Double-byte character density is 5 characters per inch.

12: Character density is 12 characters per inch.

13.3: Character density is 13.3 characters per inch. This value is valid only for double-byte character set (DBCS) printers.

15: Character density is 15 characters per inch.

16.7: Character density is 16.7 characters per inch.

18: Character density is 18 characters per inch. This value is valid only on double-byte character set (DBCS) printers.

20: Character density is 20 characters per inch. This value is valid only on double-byte character set (DBCS) printers.

FRONTMGN

Specifies the offset, down and across, of the origin from the edge on the front side of the paper. The offsets are in the units of measure specified on the UOM parameter. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.57. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

***DEVD:** The no-print border from the printer is used to place the text on the page when printing to a printer configured as AFP(*YES). A margin of 0 is used for IPDS printers without a no-print border, or which are configured as AFP(*NO).

Element 1: Offset Down

0: No offset of the origin occurs.

offset-down: Specify the offset of the origin from the top of the page.

Element 2: Offset Across

0: No offset of the origin occurs.

offset-across: Specify the offset of the origin from the left side of the page.

BACKMGN

Specifies the offset, down and across, of the origin from the edge on the back side of the paper. The offsets are in the units of measure specified on the UOM parameter. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.57. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

***FRONTMGN:** The offsets specified on the FRONTMGN parameter are used.

***DEVD:** The no-print border from the printer is used to place the text on the page when printing to a printer configured as AFP(*YES). A margin of 0 is used for IPDS printers without a no-print border, or which are configured as AFP(*NO).

Element 1: Offset Down

0: No offset of the origin occurs.

offset-down: Specify the offset of the origin from the top of the page.

Element 2: Offset Across

0: No offset of the origin occurs.

offset-across: Specify the offset of the origin from the left side of the page.

OVRFLW

Specifies the line number on the current page at which overflow to a new page begins. Generally, after the specified line is printed, the printer overflows to the next page before printing continues. Margins specified for the printer file are ignored when determining overflow. More information is in the *Guide to Programming for Printing*.

60: Overflow to a new page begins when the current line is 60, whether or not printing has occurred on that line.

overflow-line-number: Specify the line number on the current page at which overflow to a new page begins, whether or not printing has occurred on that line. The value specified must not be greater than the page length (PAGESIZE). Margins specified for the printer file are ignored when determining overflow.

FOLD

Specifies whether all positions in a record are printed when the record length exceeds the page width (specified by the PAGESIZE parameter). When folding is specified and a record exceeds the page width, any portion of the record that cannot be printed on the first line continues (is folded) on the next line or lines until the entire record has been printed.

The FOLD parameter is ignored under the following conditions:

- When DEVTYPE(*SCS) is not specified.
- When printing through OfficeVision/400.
- When in the S/36 execution environment.

Double-Byte Character Set Considerations:

The system ignores this parameter when printing double-byte character set (DBCS) files. The system assumes that DBCS records fit on a printed line. If the record exceeds the page width, the system continues printing the record on the next line.

***NO:** Records are not folded; if a record is longer than the page width, only the part of the record that fits on one line is printed.

***YES:** Records whose length exceeds the page width are folded on the following lines.

RPLUNPRT

Specifies (1) whether unprintable characters are replaced and (2) which substitution character (if any) is used. An *unprintable* character is a character the printer is unable to print.

Double-Byte Character Set Considerations:

For double-byte character set (DBCS) data, an unprintable character is one that cannot be processed. When using DBCS-capable printers, consider the following:

- If IGCEXNCHR(*YES) is also specified, the system replaces unprintable extension characters with DBCS underline characters. There may be some cases in which the system is unable to replace an unprintable character with a DBCS underline character. In this case, the undefined character is printed.
- If IGCEXNCHR(*NO) is also specified, the device replaces all extension characters with the undefined character. Choosing a blank as the replacement character for alphanumeric characters might improve system performance.

More information is in the *Guide to Programming for Printing*.

Element 1: Replace Character?

***YES:** Unprintable characters are replaced. The program is not notified when unprintable characters are detected. Note the DBCS considerations above.

***NO:** Unprintable characters are not replaced. When an unprintable character is detected, a message is sent to the program.

Element 2: Replacement Character

' ': Text is not specified.

'replacement-character': Specify, if *YES is also specified on this parameter, the replacement character that is used each time an unprintable character is detected. Any printable EBCDIC character can be specified. Valid values range from 40 through 99 and A1 through FE.

ALIGN

Specifies whether the page must be aligned in the printer before printing is started. If ALIGN(*YES) and SPOOL(*NO) are specified, and forms alignment is required, the system sends a message to the message queue specified in the printer device description and waits for a reply to the message. When spool (*YES) is specified on the CRTPRTF command and ALIGN(*FILE) is specified on the STRPRTWTR command, then this parameter is used to determine whether an alignment message is sent by the system.

This parameter is ignored when cut sheets are used (spooled and direct output). Page alignment can be done only for text-only files. Page alignment cannot be done for print jobs containing graphics or bar codes.

***NO:** No page alignment is required.

***YES:** The page is aligned before the output is printed.

CTLCHAR

Specifies whether the printer file supports input with print control characters. Any invalid control characters that are found are ignored, and single spacing is assumed.

***NONE:** No print control characters are passed in the data being printed.

***FCFC:** The first character of every record contains an American National Standards Institute (ANSI) forms control character. If *FCFC is specified, the record length must include one extra position for the first-character forms-control code. This value is not valid for externally described printer files.

CHLVAL

Specifies a list of channel numbers with their assigned line numbers. Use this parameter only if CTLCHAR(*FCFC) has been specified.

Note: If one or more channel-number/line-number combinations are changed, all other combinations must be re-entered.

***NORMAL:** The default values for skipping to channel identifiers are used. The default values are found in Table 16.

Table 37. ANSI First-Character Forms-Control Codes

Code	Action before Printing a Line
' '	Space one line (blank code)
0	Space two lines
-	Space three lines
+	Suppress space
1	Skip to line 1
2-11	Space one line
12	Skip to overflow line (OVRFLW parameter)

Element 1: Channel Number

channel-value: Specify an American National Standard channel number to be associated with a corresponding 'skip to' line number. Valid values for this parameter range from 1 through 12, corresponding to channels 1 through 12. The CHLVAL parameter associates the channel number with a page line number. For example, if you specify CHLVAL(2 20), channel identifier 2 is allocated with line number 20; therefore, if you place the forms-control 2 in the first position of a record, the printer skips to line 20 before printing the line.

Note: If the printer stops and the next record processed has a channel value forms-control number that is the same value as the line number the printer is on, the printer advances to that value (line number) on the next page. However, if the printer is positioned at the top of the page (line number one) and the channel value forms-control value is associated with line number one, the printer does not advance to a new a new page.

If no line number is specified for a channel identifier, and that channel identifier is encountered in the data, a default of 'space one line' before printing is used. Each channel number can be specified only once.

Element 2: Line Number

line-number: Specify the line number assigned for the channel number in the same list. Valid line numbers range from 1 through 255. If no line number is assigned to a channel number, and that channel number is encountered in the data, a default of 'space one line' before printing is used. Each line number can be specified only once.

FIDELITY

Specifies whether printing continues when print errors are found for printers configured with AFP(*YES).

***CONTENT:** Printing continues when errors are found.

***ABSOLUTE:** Printing stops when errors are found.

PRTQLTY

Specifies, for the 3812 SCS, 3816 SCS, 4214, 4224, 4230, 4234, and 5219 printers, the quality of print produced.

For the 5219 Printer, different print qualities are produced by varying the speed at which the print ribbon advances. Quality mode (*STD or *NLQ) results in normal print ribbon advancement. In draft mode (*DRAFT), the ribbon advances at a rate of one-third the distance it advances in quality mode. The 5219 Printer has a conserve ribbon switch that overrides the value of *DRAFT specified by this parameter.

For the 3812 SCS and 3816 SCS Printers, the automatic hardware selection of computer output reduction printing selected through soft switches on the printers occurs only when *DRAFT is specified for PRTQLTY and PAGRTT is *DEVD. If PAGRTT(*COR) is specified, the PRTQLTY parameter does not affect the printed output.

For the 4214, 4224, 4230, and 4234 Printers, standard print quality is produced by varying the density of the dot matrix pattern used to create printable characters. Standard mode (*STD) is the normal mode. Quality mode (*NLQ) requires multiple passes by the printer to produce a line of data. Draft mode (*DRAFT) results in high-speed printing.

More information about the valid values for the 4214, 4224, 4230, 4234, and 5219 Printers is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

Notes:

1. For the 4214 Printer, standard print quality (*STD or *NLQ) is only supported for 10 and 12 characters per inch. If PRTQLTY(*STD or *NLQ) and 5, 15, or 16.7 characters per inch is specified, the data is printed in draft mode.
2. For the 4234 Printer, only a limited character set (62 characters) is supported when PRTQLTY(*DRAFT) is specified. A description of the character set supported with draft print quality is in the *4234 Printer Operator's Guide*.

3. For the 4224 and 4230 printers, the fonts supported are not available for all three print qualities. The OCR-A and OCR-B fonts are supported only with PRTQLTY(*NLQ). The Courier and Essay fonts are available only with PRTQLTY(*NLQ) and PRTQLTY(*STD). The Gothic font is available only with PRTQLTY(*DRAFT) or PRTQLTY(*FASTDRAFT). If there is a mismatch between the print quality and the font selected, the font is changed to match the print quality.
4. Specify PAGRTT(*DEVD) and PRTQLTY(*DRAFT) on this command to enable automatic rotation if the data does not fit on the paper.

***STD:** The output is printed with standard quality.

***DEVD:** The print quality is set on the printer by the user, if it is not set within the data stream.

***DRAFT:** The output is printed with draft quality.

***NLQ:** The output is printed with near letter quality.

***FASTDRAFT:** The output is printed at a higher speed and with lower quality than it would be if you specified *DRAFT. This value is only supported by the 4230 printer.

FORMFEED

Specifies, for the 4214, 5219, and 5553 printers (including ASCII printers that are configured as an SCS 4214 or SCS 5219 printer), and for IPDS printers, the mode in which forms are fed into the device.

***DEVD:** The forms are fed into the printer in the manner specified in the device description.

***CONT:** Continuous forms are used by the printer. The tractor feed attachment must be on the device.

***CUT:** Single-cut sheets are used by the printer. Each sheet must be manually loaded. For cut sheets, the forms alignment message is not sent.

***AUTOCUT:** The sheet-feed attachment must be on the printer. Single-cut sheets are automatically fed into the printer. The forms alignment message is not sent for cut sheets.

DRAWER

Specifies the source drawer used when automatic cut sheet feed mode is used (specified by FORMFEED(*AUTOCUT)).

1: The paper is fed from the first drawer on the sheet-feed paper handler.

2: The paper is fed from the second drawer on the sheet-feed paper handler.

3: The paper is fed from the third drawer on the sheet-feed paper handler.

***E1:** The envelopes are fed from the envelope drawer on the sheet-feed paper handler.

FONT

Specifies the font identifier and point size used with this printer device file. If a font identifier or point size is not specified, the system automatically sets them.

More information about the valid font identifiers, the display value, the characters per inch value implied with each font style, a description of each font style, and whether the font is supported on a particular printer is in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

Note: Some fonts can be substituted by the printer. Consult the various printer reference guides for details.

***CPI:** The identifier of the font with the specified pitch (characters per inch (CPI)) is used.

***DEVD:** The font identifier and point size specified in the device description are used.

Element 1: Font Identifier

identifier: Specify the numeric font identifier associated with this printer. If one is not specified, the system automatically sets one.

Element 2: Point Size

***NONE:** The point size is supplied by the system and is determined by the specified font identifier.

point-size: Specify a point size ranging from 0.1 through 999.9. If one is not specified, the system automatically sets one.

CHRID

Specifies the character identifier (graphic character set and code page) for the file. This parameter allows printing of text that is in different character identifier (graphic character set and code page) coding. The value specified on this parameter is used to instruct the printer device to interpret the hexadecimal byte string to print the same characters that were intended when the text was created. More information about the character identifier is in the *Guide to Programming for Printing*. A list of valid CHRID values and applicable printers is in the "CHRID Values and Applicable Printers (CHRID parameter)" table in Appendix B, "Font, Character Identifier, and Other Values Supported for Different Printers."

***DEVD:** The default CHRID value that the device is designed to handle is used. The *DEVD value means character selection is normal because the file has the same character identifier as the device default.

***SYSVAL:** The recovery limits specified in the QCMNRCYLMT system value are used.

***JOBCCSID:** The character identifier for the printer file is taken from the coded character set identifier (CCSID) of the job.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values that match the attributes of the printer. Valid values range from 1 through 32767.

Element 2: Code Page

code-page: Specify the code page value that matches the attributes of the printer. Valid values range from 1 through 32767.

FNTCHRSET

Specifies a downloaded font consisting of a character set and code page. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

***FONT:** The value specified on the FONT parameter is used. The value specified on the FONT parameter is used.

Element 1: Font Character Set

The name of the font character set can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

character-set: Specify the font character set to use.

Element 2: Code Page Name

The name of the code page can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

code-page: Specify the code page name to use.

CDEFNT

Specifies the coded font that the system uses for single-byte character set (SBCS) printing. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

***FNTCHRSET:** The font specified on the FNTCHRSET parameter is used.

The name of the coded font can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

coded-font-name: Specify the coded font name to use.

PAGRTT

Specifies the degree of text rotation for the 3812, 3816, 4028, 3820, 3825, 3827, 3829, 3831, 3835, and 3900 printers. This parameter allows the user to specify the degree of rotation of the text on the page with respect to the way the form is loaded into the printer. See the note under the PAGESIZE parameter for directions on specifying page size when rotating the page.

Specify *AUTO or *DEVD for this parameter and PRTQLTY(*DRAFT) on this command to enable automatic rotation if the data does not fit on the paper.

***AUTO:** Indicates that automatic rotation of output is done to fit the printed data on the form. If rotation does not accomplish this, computer output reduction is performed automatically (regardless of the print quality being used). This parameter is valid only for printers supporting rotation.

***DEVD:** The operating system sends a device default rotation value to the printer. Page rotation is dependent upon your printer's specifications. See your printer or printer emulation documentation to determine how page rotation is affected.

***COR:** Computer output reduction is done. Computer output reduction allows printed output intended for a 13.2 inch wide by 11.0 inch long form to be printed on an 8.5 inch wide by 11.0 inch long form.

For computer output reduction printing, the following operations are done by the 3812, 3816, 4028, 3820, 3825, 3827, 3829, 3831, 3835, and 3900 printers:

- Automatic rotation to *COR is not done if the file contains graphics, bar codes, variable LPI, variable font, variable page rotations, or variable drawer.
- The text is rotated 90 degrees clockwise from the 0 degree rotation position (lower left corner of the first edge loaded into the printer).

  Note: For landscape paper on a 3835 printer, the rotation is counter-clockwise from the 0 degree rotation position (upper right corner of the first edge loaded into the printer).
- A top and left margin of 0.5 inches is added to the printed output.
- The 12-pitch fonts are changed to a 15-pitch font and 15-pitch fonts are changed to a 20-pitch font. All other font widths are changed to a 13.3-pitch font, except for the 4028 printer where they are changed to a 15-pitch font.

- Vertical spacing (specified by the LPI parameter) is 70 percent of the normal spacing.
- The page size is set to 8.5 inches wide by 11 inches long.

0: No rotation occurs. Printing starts at and is parallel to the edge loaded into the printer.

90: Rotation of the text is done 90 degrees clockwise from the 0 degree writing position.

180: Rotation of the text is done 180 degrees clockwise from the 0 degree writing position.

270: Rotation of the text is done 270 degrees clockwise from the 0 degree writing position.

MULTIUP

Specifies, for spooled output only, the number of pages printed on a single physical page.

Note: Overlays are not reduced when more than one page is printed on a side.

1: One page of output is printed on one physical sheet of paper.

2: Two pages of output are printed on 1 physical sheet of paper.

4: Four pages of output are printed on 1 physical sheet of paper.

PRTTXT

Specifies up to 30 characters of text to be printed at the bottom of each page of output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***JOB:** The value for the current job is used.

***BLANK:** Text is not specified.

'print-text': Specify the character string printed at the bottom of each page. No more than 30 characters of text can be entered, enclosed in apostrophes.

JUSTIFY

Specifies the printing positions of the characters on a page so the right-hand margin of printing is regular. Justification is done to the record length on the printer file opened.

Note: The JUSTIFY parameter is supported only on the 3812 SCS, 3816 SCS, and 5219 Printers.

0: No justification occurs.

50: Spaces are added to the blanks in the text so that the right margin is more closely aligned but not flush.

100: The text is expanded by spaces (added where the blanks already exist) until the right margin is flush.

DUPLEX

Specifies whether output is printed on one side or two sides of the paper.

***NO:** The output is printed on one side of the paper.

***YES:** The output is printed on both sides of the paper with the top of each printed page at the same end of the paper.

***TUMBLE:** The output is printed on both sides of the paper with the top of one printed page at the opposite end of the sheet from the top of the other printed page. This is usually used for output that is bound at the top.

UOM

Specifies the unit of measure that is used.

***INCH:** An inch is used as the unit of measure.

***CM:** A centimeter is used as the unit of measure.

FRONTOVL

Specifies the qualified name of the object that contains both the overlay that is printed on the FRONT side of the page and the offset, down and across, from the point of origin used when the overlay is printed.

***NONE:** No overlay is used.

Element 1: Overlay Name

The name of the overlay can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

overlay-name: Specify the name of the overlay.

Element 2: Offset Down

0: No offset down from the point of origin is used.

offset-down: Specify the offset down from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.57.

Element 3: Offset Across

0: No offset across from the point of origin is used.

offset-across: Specify the offset across from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.57.

BACKOVL

Specifies the object name and library name containing both the overlay that is printed on the BACK side of the page and the offset, down and across, from the point of origin used when the overlay is printed.

***FRONTOVL:** The values that are specified on the FRONTOVL parameter are used.

***NONE:** No overlay is used.

Element 1: Overlay Name

The name of the overlay can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

overlay-name: Specify the name of the overlay.

Element 2: Offset Down

0: No offset down from the point of origin is used.

offset-down: Specify the offset down from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.57.

Element 3: Offset Across

0: No offset across from the point of origin is used.

offset-across: Specify the offset across from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.57.

DFRWRT

Specifies how much output is held in the system buffer before being sent to the printer.

***YES:** The system controls the amount of output that is held in the buffer before being sent to the printer.

If SPOOL(*YES) is specified along with SCHEDULE(*IMMED), output is held in the buffer until a page of output is available or until the system buffer is full.

***NO:** If SPOOL(*NO) is specified, output is not held in the buffer. Output is sent to the printer immediately after the program performs a write operation.

If the spooled output schedule is not immediate, specifying DFRWRT(*NO) has no effect.

SPOOL

Specifies whether the output data for the printer file is spooled. If SPOOL(*NO) is specified, the following parameters in this command are ignored: OUTQ, COPIES, MAXRCDS, FILESEP, SCHEDULE, HOLD, SAVE, OUTPTY, and USRDTA.

***YES:** The data is spooled for processing by a diskette writer or a print writer.

***NO:** The data is not spooled; it is sent directly to the device and printed as the output becomes available.

OUTQ

Specifies, for spooled output only, the qualified name of the output queue.

***JOB:** The output queue associated with the job is used.

***DEV:** The output queue associated with the printer specified on the DEV parameter is used.

The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

output-queue-name: Specify the name of the output queue to which the output data is spooled.

FORMTYPE

Specifies the type of form on which the output is printed. The identifiers used to indicate the type of forms are user-defined and can be a maximum of 10 characters in length.

***STD:** The standard form type is used.

form-type: Specify the identifier of the form type used with this device file for printed output from jobs. Up to 10 alphanumeric characters can be specified. When the device file is opened, the system sends a message identifying the form type to the system operator, and requests that the identified forms be in the printer.

COPIES

Specifies, for spooled files, the number of copies being printed.

1: One copy of the output is printed.

number-of-copies: Specify a value, ranging from 1 through 255, that indicates the number of identical printouts produced when this printer file is used.

PAGERANGE

Specifies the page range to print for each copy of the file to be printed.

Element 1: Starting Page to Print

1: Page 1 is the page on which to start printing.

***ENDPAGE:** Only the ending page is printed.

starting-page: Specify the page on which to start printing.

Element 2: Ending Page to Print

***END:** The last page in the file is printed.

ending-page: Only the ending page is printed.

MAXRCDS

Specifies, for spooled output only, the maximum number of records that can be in the spooled file for jobs using this printer file. If this maximum is reached, an inquiry message is sent to the program message queue.

100000: Up to 100,000 records can be in the spooled file for each job that uses this printer file.

***NOMAX:** The system maximum is used.

maximum-records: Specify a value, ranging from 1 through 999999, that specifies the maximum number of records allowed in the spooled file.

FILESEP

Specifies, for spooled output only, the number of separator pages placed at the start of each printed file, including those between multiple copies of the same output. Each separator page has the following items printed on it: file name, file number, job name, user name, and the job number.

0: No separator pages are used at the beginning of each spooled file produced by this device file.

number-of-file-separators: Specify the number of separator pages used at the start of each printed output file produced by this device file. Valid values range from 0 through 9. If 0 is specified, no separator pages are printed for the file. In this case, the printed output for each file (or copy of a file) starts at the top of a new page.

SCHEDULE

Specifies, for spooled output only, when the spooled file is available to a writer.

***FILEEND:** The spooled file is made available to the writer as soon as the file is closed in the program.

***JOBEND:** The spooled file is made available to the writer only after the entire job is completed.

***IMMED:** The spooled file is made available to the writer as soon as the file is opened in the program.

HOLD

Specifies, for spooled output only, whether the spooled file is held. The spooled file can be released by using the Release Spooled File (RLSSPLF) command.

***NO:** The spooled printer file is not held by the output queue. The spooled output is available to a writer based on the SCHEDULE parameter value.

***YES:** The spooled file is held until released by the Release Spool File (RLSSPLF) command.

SAVE

Specifies, for spooled output only, whether the spooled file is saved (left on the output queue) after the output has been produced.

***NO:** The spooled file data is not saved on the output queue after it has been produced.

***YES:** The spooled file data is saved on the output queue until the file is deleted. After the file is produced, the number of copies (see COPIES parameter) is set to 1, and its status is changed from WTR to SAV. Refer to the Release Spooled File (RLSSPLF) command for information on how to produce the spooled file again.

OUTPTY

Specifies the output priority for spooled output files that are produced by this job. The highest priority is 1 and the lowest priority is 9. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***JOB:** The output priority associated with the job that created the spooled file is used.

output-priority: Specify the output priority. Valid values range from 1 (high priority) through 9 (low priority).

USRDTA

Specifies, for spooled output only, the user-specified data that identifies the file.

***SOURCE:** If the spooled file was created by an application program, the name of the program is used. Otherwise, blanks are used.

user-data: Specify up to 10 characters of text.

IGCDTA

Specifies, for program-described original files, whether the file processes double-byte character set (DBCS) data. For externally described printer files, this parameter specifies DBCS attributes of the file.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

IGCEXNCHR

Specifies whether the system processes double-byte character set (DBCS) extension characters.

***YES:** The system processes DBCS extension characters.

***NO:** The system does not process DBCS extension characters; it prints extension characters as the undefined character.

IGCCHRRTT

Specifies, for the 5553 and 5583 Printers only, whether the printer rotates double-byte characters 90 degrees counterclockwise when printing. The system prints rotated double-byte characters so they appear in a vertical reading sequence. Alphanumeric characters are not rotated.

***NO:** The system does not rotate double-byte characters when printing.

***YES:** The system rotates double-byte characters 90 degrees counterclockwise when printing. The printer rotates each character individually.

IGCCPI

Specifies the printer character density of double-byte character set (DBCS) characters, in characters per inch (CPI).

Note: This parameter does not specify the printer character density of alphanumeric characters. Alphanumeric characters are printed with the value specified on the CPI parameter.

***CPI:** DBCS character density is based on the values specified for the CPI parameter. The system prints one double-byte character for every two alphanumeric characters.

- For CPI(10), DBCS characters print at 5 characters per inch.
- For CPI(12), DBCS characters print at 6 characters per inch.
- For CPI(13.3), DBCS characters print at 6.7 characters per inch (same as IGCCPI(*CONDENSED)).
- For CPI(15), DBCS characters print at 7.5 characters per inch.
- For CPI(18), DBCS characters print at 9 characters per inch.
- For CPI(20), DBCS characters print at 10 characters per inch.

***CONDENSED:** Condensed printing is used in which the system prints 20 DBCS characters every 3 inches. This value is valid only for the 5553 or 5583 Printers.

5: Double-byte character density is 5 characters per inch.

6: DBCS character density is 6 characters per inch. This value is valid for the 5553 and 5583 Printers only.

10: DBCS character density is 10 characters per inch. This value is valid for the 5553 or 5583 Printers only.

IGCSOSI

Specifies, for bracketed DBCS character strings only, how the system prints shift control characters.

***YES:** The system prints shift control characters as blanks.

***NO:** The system does not print shift control characters. These characters do not occupy a position in printed output.

***RIGHT:** The system prints two blanks when printing shift-in characters but does not print shift-out characters.

IGCCDEFNT

Specifies the coded font that the system uses for DBCS printing.

***SYSVAL:** The DBCS-coded font specified in the system value QIGCCDEFNT is used.

The name of the coded font can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

coded-font-name: Specify the DBCS-coded font name to use.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds that the program waits for the file resources to be allocated to the printer file when the file is opened, or the wait time for the device allocated when an acquire operation is performed to the file. Valid values range from 1 through 32767 seconds.

SHARE

Specifies whether the open data path (ODP) for the printer file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

LVLCHK
Specifies whether the record format level identifiers in the program are checked against those in the device file when the file is opened. If so, the record format identifiers in the program must match those in the device file. Because the same record format name can exist in more than one file, each record format is given an internal system identifier when it is created.

***YES:** The level identifiers of the record formats are checked when the file is opened. If the level identifiers do not match, an error message is sent to the program that requested the open, and the file is not opened.

***NO:** The level identifiers are not checked when the file is opened.

AUT
Specifies the authority given to users who do not have specific authority to the printer file, who are not on an authorization list, and whose user group has no specific authority to the printer file.

***LIBCRTAUT:** The public authority for the printer file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the printer file). The public authority is determined when the printer file is created. If the CRTAUT value for the library changes after the printer file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the printer file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the printer file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the printer file.

***USE:** The user can perform basic operations on the printer file, such as running a program or reading a file. The user cannot change the printer file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the printer file.

authorization-list-name: Specify the name of the authorization list used.

REPLACE
Specifies whether an existing file is replaced by the new printer file More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** The existing printer file is replaced by the one being created.

***NO:** The existing file, if any, is not replaced by the printer file.

TEXT
Specifies text that briefly describes the printer file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member used to create the printer file. If the source file is a database file, the text is taken from the source member. If the source file is an inline file or a device file, the text is blank.

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a Printer File

```
CRTPRTF   FILE(DSPHIST)  SRCFILE(PRSNNL/JOBHIST)
  FILESEP(3)
```

This command creates a printer file named DSPHIST using the DDS source file named JOBHIST that is stored in the PRSNNL library. The defaults for the other parameters are assumed, except for FILESEP.

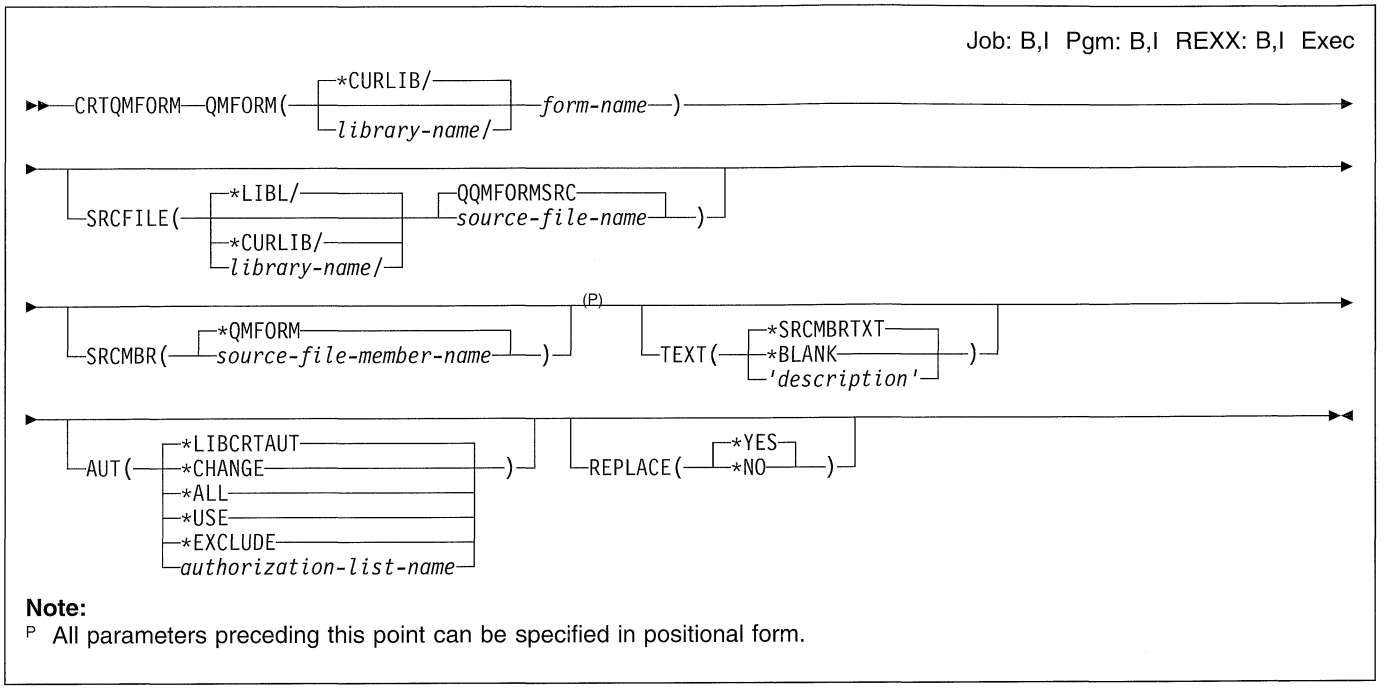
The printer uses standard forms that are 66 lines long and 132 print positions wide. An SCS data stream is used. It prints 6 lines per inch and overflows to a new page after line 60 is printed. The print image specified in the device description is used. Output is spooled to the output queue specified for the job and cannot be printed until the file is closed. The spooled file is not held or saved after printing. One copy of the output is printed, preceded by three separator pages, each containing the file name, the spooled number, and the job name and number. The print text specified in the current job is used.

Example 2: Creating a Printer File Containing DBCS Data

```
CRTPRTF   FILE(IGCLIB/IGCPRT)  FORMFEED(*AUTOCUT)
  IGCDTA(*YES)  IGCCHRRT(*YES)
```

This command creates a printer file, IGCPRT (stored in library IGCLIB) that contains DBCS data. Cut sheets are automatically fed, and double-byte characters are rotated when printing.

CRTQMFORM (Create Query Management Form) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Query Management Form (CRTQMFORM) command allows the user to create a query management form from a specified source file member. The query management form defines how a report is to look when data from running a query is displayed or printed.

Required Parameter

QMFORM

Specifies the name of the query management form to be created.

The possible library values are:

***CURLIB:** The current library for the job is used to locate the form. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the form is located.

form-name: Specify the name of the form to be created.

Optional Parameters

SRCFILE

Specifies the qualified name for the source file containing the source for the form being created. The form is identified by the name specified on the QMFORM parameter.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QQMFORMSRC: The file having the IBM-supplied source file name, QQMFORMSRC, contains the source for the form to be created.

source-file-name: Specify the name of the file containing the source for the form to be created.

SRCMBR

Specifies the name of the source file member containing the form source to be created.

***QMFORM:** The member name is the same as the form name specified on the QMFORM parameter.

source-file-member-name: Specify the name of the member that contains the form source.

TEXT

Specifies text that briefly describes the query management form. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The descriptive text for the object being created is the same as the text description of the source member. If the source member has no text description, or if it is blank, then TEXT(*BLANK) is assumed.

***BLANK:** No text is specified. If this command causes an object to be replaced, then any existing text for the object is replaced with blank text.

'description': Specify no more than 50 characters of text, enclosed in apostrophes. The apostrophes are not part of the 50-character string.

AUT

Specifies the authority given to users who do not have specific authority to the query management form, who are not on an authorization list, and whose user group has no specific authority to the query management form. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the query management form is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the query management form). The public authority is determined when the query management form is created. If the CRTAUT value for the library changes after the query management form is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change the object and perform basic functions on the object. This authority gives object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the query management form. If the object is part of an authorization list, the user cannot add, change, or remove other users.

***USE:** The user can perform basic operations on the object, for example, run a program or read a file. The user is prevented from changing the object in any way. This authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access an object.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether an existing object with the same name and type in the library to receive the output is replaced with the output of this command. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** An existing object is replaced with the output of this command.

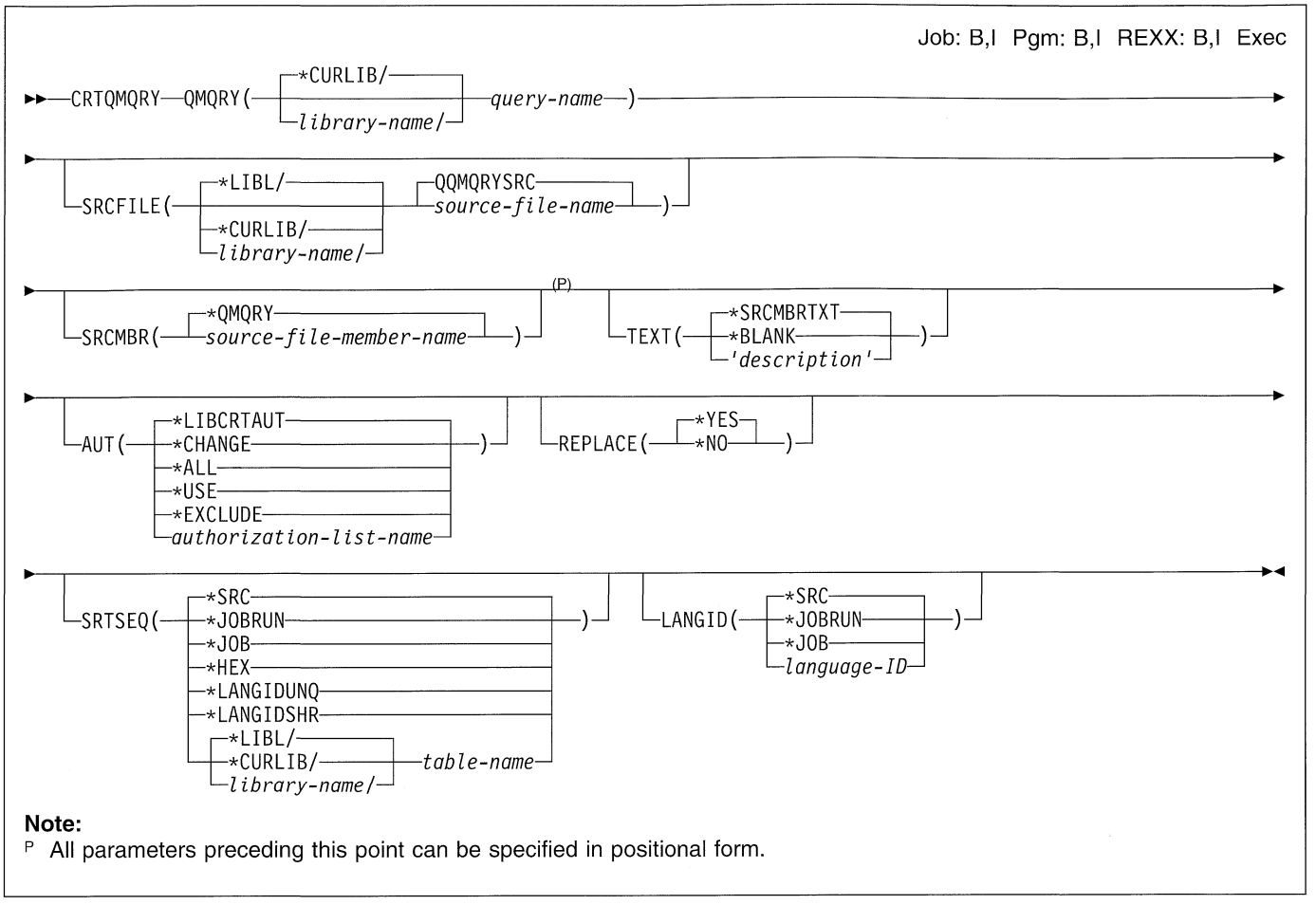
***NO:** An existing object is not replaced with the output of this command.

Example

```
CRTQMFORM   QMFORM(FORMEMP)  SRCFILE(RPTLIB/FORMSRC)
```

This command creates a form named FORMEMP in the current library. The form source is in member FORMEMP, which is located in source file FORMSRC in library RPTLIB.

CRTQMQRY (Create Query Management Query) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Query Management Query (CRTQMQRY) command allows the user to create a query management query from a specified source file member. The source for a query is a single Structured Query Language (SQL) statement that can contain variable substitution values and embedded comments. It can be spread over multiple records in a source file member.

Required Parameter

QMQRY

Specifies the name of the query management query to be created.

The possible library values are:

***CURLIB:** The current library for the job is used to locate the query. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the query is located.

query-name: Specify the name of the query to be created.

Optional Parameters

SRCFILE

Specifies the qualified name for the source file containing the source for the query being created. The query is identified by the name specified on the QMQRY parameter.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QQMQRYSRC: The file having the IBM-supplied source file name, QQMQRYSRC, contains the source for the query to be created.

source-file-name: Specify the name of the file containing the source for the query to be created.

SRCMBR

Specifies the name of the source file member containing the source for the query to be created.

***QMQRY:** The member name is the same as the query name specified on the QMQRY parameter.

source-file-member-name: Specify the name of the member that contains the query source.

TEXT

Specifies text that briefly describes the query management query. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The descriptive text for the object being created is the same as the text description of the source member. If the source member has no text description, or if it is blank, then TEXT(*BLANK) is assumed.

***BLANK:** No text is specified. If this command causes an object to be replaced, then any existing text for the object is replaced with blank text.

'description': Specify no more than 50 characters of text, enclosed in apostrophes. The apostrophes are not part of the 50-character string.

AUT

Specifies the authority given to users who do not have specific authority to the query management query, who are not on an authorization list, and whose user group has no specific authority to the query management query. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the query management query is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the query management query). The public authority is determined when the query management query is created. If the CRTAUT value for the library changes after the query management query is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change the object and perform basic functions on the object. This authority gives object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the query management query. If the object is part of an authorization list, the user cannot add, change, or remove other users.

***USE:** The user can perform basic operations on the object, for example, run a program or read a file. The user is prevented from changing the object in any way. This authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access an object.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether an existing object with the same name and type in the library to receive the output is replaced with the output of this command. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** An existing object is replaced with the output of this command.

***NO:** An existing object is not replaced with the output of this command.

SRTSEQ

Specifies the sort sequence table to be used for string comparisons in this query.

Note: This parameter is valid only when creating a query. This parameter is not valid when creating a FORM.

***SRC:** The source file member contains the sort sequence to be used when creating the query.

***JOBRUN:** The SRTSEQ value for the job at the time the query is run is used.

***JOB:** The SRTSEQ value for the job at the time the query is created is used.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.

***LANGIDUNQ:** The unique-weight sort table for the LANGID specified is used.

***LANGIDSHR:** The shared-weight sort table for the LANGID specified is used.

The name of the sort sequence table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify the name of the sort sequence table to be used when this query is created.

LANGID

Specifies the language identifier to be used when SRTSEQ(*LANGIDUNQ) or SRTSEQ(*LANGIDSHR) is specified.

***SRC:** The source file member contains the language ID to be used when the query is created.

***JOBRUN:** The LANGID value for the job is determined when the query is run.

***JOB:** The LANGID value for the job is determined when the query is created.

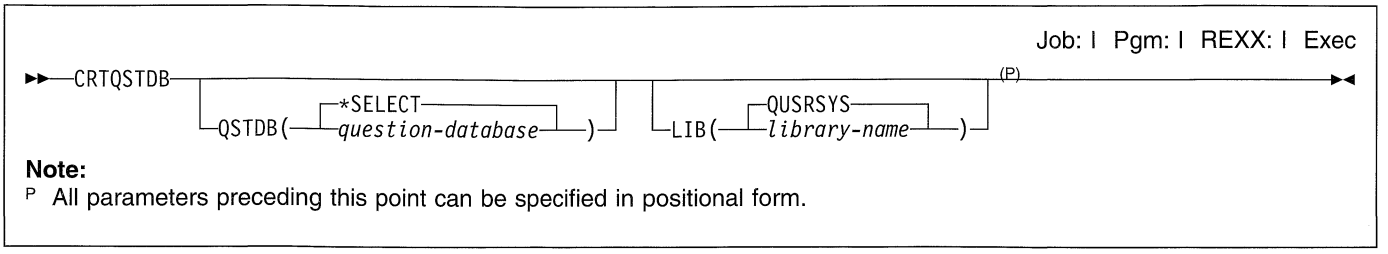
language-ID: Specify the language identifier to be used for the query.

Example

```
CRTQMQRY   QMQRY(NEWQRY)  SRCFILE(RPTLIB/QRYSRC)
```

This command creates a query named NEWQRY in the current library. The query source is in member NEWQRY, which is located in source file QRYSRC in library RPTLIB.

CRTQSTDB (Create Question-and-Answer Database) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Question-and-Answer Database (CRTQSTDB) command allows the user to create a new question-and-answer (Q & A) database. More information is available in the *Q & A Database Coordinator's Guide*.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. The user of this command must have *ADD authority to the library in which the database files are located.

Optional Parameters

QSTDB

Specifies the Q & A database to create. When selecting a Q & A database name, the user must specify 10 or less characters and the name must begin with a letter.

*SELECT: The user is asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database: Specify the name of the Q & A database in which to create.

LIB

Specifies the name of an existing library that will contain the new Q & A database.

QUSRSYS: The Q & A database is created in the QUSRSYS library.

library-name: Specify the name of the library to be searched.

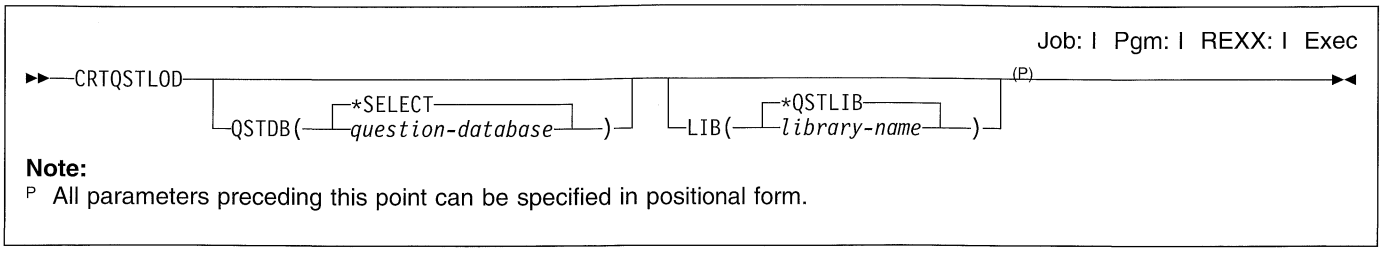
Note: The library must exist on the system.

Example

```
CRTQSTDB
```

This command shows the Create a Q & A Database display.

CRTQSTLOD (Create Question-and-Answer Load) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Create Question-and-Answer Load (CRTQSTLOD) command allows the user to create for distribution a Question-and-Answer (Q & A) database load on an alternative medium, such as tape. More information is in the *Q & A Database Coordinator's Guide*.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. The user must have *READ authority to the Q & A database file referred to by the command.

Optional Parameters

QSTDB

Specifies the Q & A database to distribute.

***SELECT:** The user is asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database: Specify the name of the Q & A database that you want to distribute.

LIB

Specifies the name of the library that contains the Q & A database to be used to create the load.

***QSTLIB:** The library containing the specified Q & A database is searched. If *SELECT is specified on the QSTDB parameter, any Q & A database in any library for which the user is authorized can be selected.

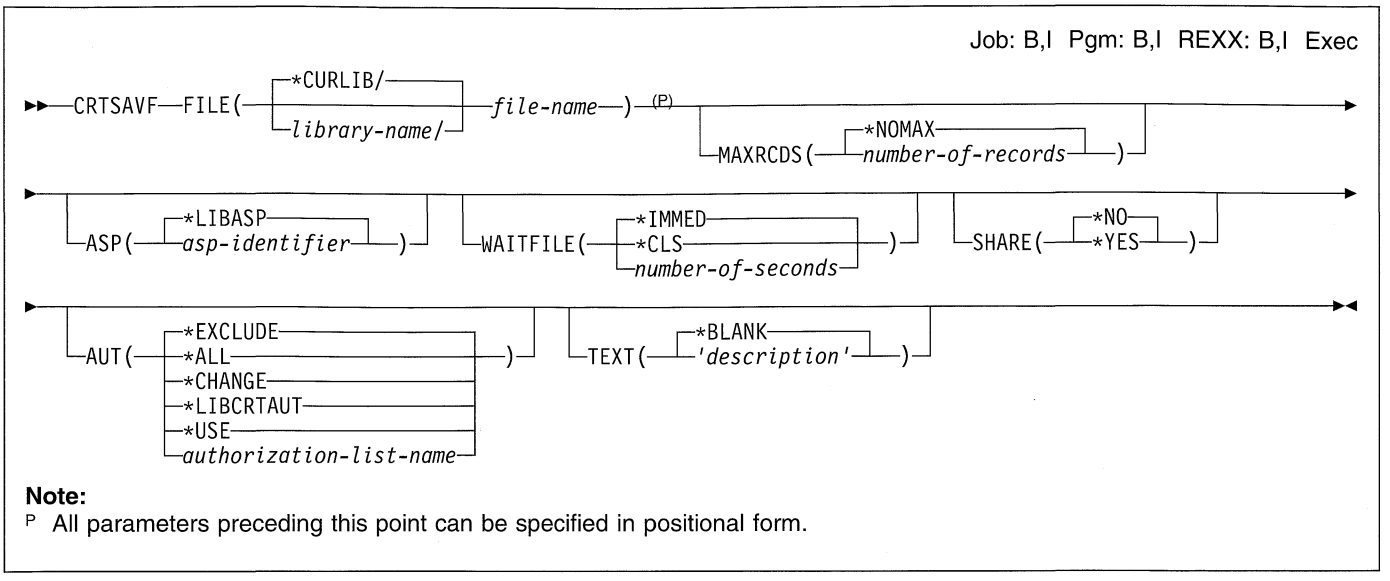
library-name: Specify the name of the library to be searched. If *SELECT is specified on the QSTDB parameter, any database in the library for which the user is authorized can be selected.

Example

```
CRTQSTLOD
```

This command shows the Create a Database Load display.

CRTSAVF (Create Save File) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Create Save File (CRTSAVF) command creates an online save file. An online save file is used with save and restore commands to retain data that would otherwise be written to tape or diskette. The file can also be used like a database file to read or write records that contain save/restore information. An online save file can also be used to send objects to another user on the SNADS network. For more information on sending an online save file, see the Send Network File (SNDNETF) command.

Restrictions:

1. An online save file should only be used for save/restore data. When restoring objects from the save file, the data in the file must have been produced by a save command. When records are written to a save file by using a high-level language program, the records must contain data produced by saving objects into the save file.
2. The records retrieved from a save file contain sequencing checksum information that is validated by the system when records are inserted into a save file. An attempt to insert records that are either out of sequence or have been changed since retrieval from a save file are rejected.
3. A save file can contain the output of only one save operation and one library, and all objects in the save file must have been saved from the same library. If the SNDNETF command is used to send a save file, the maximum size of the save file is approximately 2 billion bytes. The number of objects saved in a save file is identical to the number of objects saved when saving to tape or diskette, as long as the file's maximum size is not exceeded. The exact number of objects saved into a save file depends on the object types and the object contents.
4. The user must have object operational authority to the CRTSAVF command to create a duplicate save file or to restore a save file.
5. The user must have add and read authority to the library in which the save file is located.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

| The name of the save file can be qualified by one of the following library values:

***CURLIB:** The save file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the save file is created.

file-name: Specify the name of the file being created.

Optional Parameters

MAXRCDS

Specifies the maximum number of records the save file can hold. The size of the save file is estimated in bytes about 8192 + (512 × number of records in the save file).

There is room for approximately two thousand 512-byte records in 1 megabyte of space. To ensure that the save file does not exceed approximately 20 megabytes (20 × 2,000), specify MAXRCDS (40000).

***NOMAX:** The system maximum is used.

number-of-records: Specify the maximum number of records that the save file can contain. Valid values range from 1 through 5865293.

ASP

Specifies the auxiliary storage pool (ASP) in which the system creates the save file.

***LIBASP:** The save file is created in the same ASP as the one containing the library holding the file.

ASP-identifier: Specify a value ranging from 1 through 16 for the ASP identifier. These values can only be used if the ASP has already been configured on the system.

Note: The value of 1 is the system ASP and is always configured on the system.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds that the program waits for the file resources to be allocated. Valid values range from 1 through 32767 seconds.

SHARE

Specifies whether the open data path (ODP) for the save file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

This includes several opens in the same program.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

AUT

Specifies the authority given to users who do not have specific authority to the save file, who are not on an authorization list, and whose user group has no specific authority to the save file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***EXCLUDE:** The user cannot access the save file.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the save file.

***CHANGE:** The user can perform all operations on the save file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the save file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***LIBCRTAUT:** The public authority for the save file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the save file). The public authority is determined when the save file is created. If the CRTAUT value for the library changes after the save file is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the save file, such as running a program or reading a file. The user cannot change the save file. *USE authority provides object operational authority and read authority.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the save file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

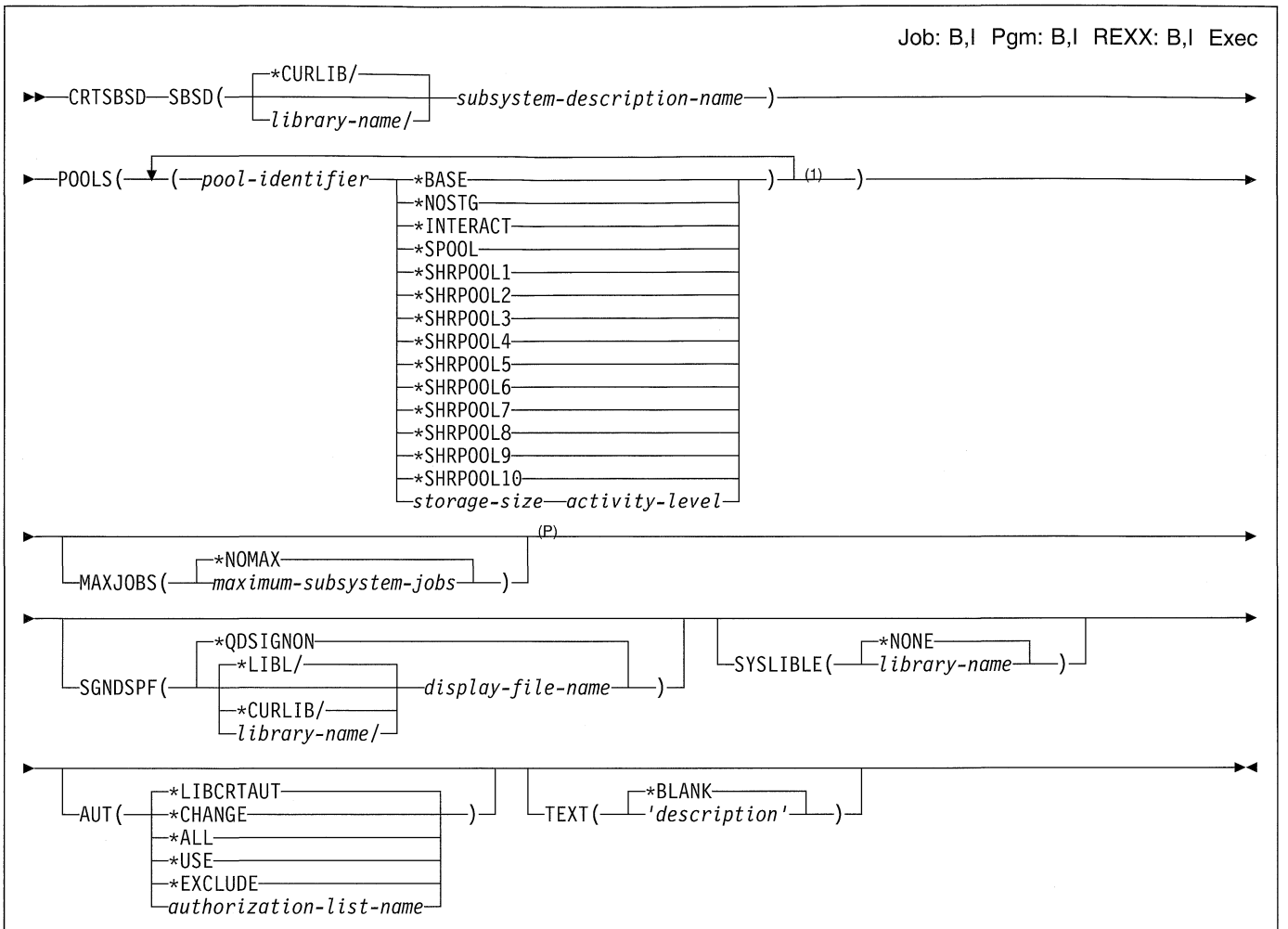
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTSAVF   FILE(ONLINE)  TEXT('Online save file')
```

This command creates an online save file named ONLINE in the current library. The save file is in the system ASP with no maximum number of records. The public has no authority to this file; only the object owner and users that have the object owner user profile as their group profile can use this save file.

CRTSBSD (Create Subsystem Description) Command



Notes:
1 A maximum of 10 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Subsystem Description (CRTSBSD) command creates a subsystem description that defines the operational attributes of a subsystem. After the subsystem description is created, it can be specialized by commands that add, change, and remove work entries and routing entries in the subsystem description.

Restriction: The user of this command must have object operational authority and add authority for the library where the subsystem description is created.

Required Parameters

SBSD

Specifies the qualified name of the subsystem description being created. The subsystem description is stored in the specified library.

More information on the IBM-supplied subsystem descriptions that are shipped with the system is in the *Work Management Guide*.

The possible library values are:

***CURLIB:** The subsystem description is stored in the current library. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the subsystem description is stored.

subsystem-description-name: Specify the name of the subsystem description being created.

POOLS

Specifies one or more storage pool definitions that are in the subsystem description. Each definition designates one storage pool:

- Pool definition identifier: The identifier, inside the subsystem description, of the storage pool definition. The same identifiers can be used for pool definitions in different subsystem descriptions. Valid values range from 1 through 10.
- Size: The size of the storage pool expressed in kilobytes (1K = 1024 bytes). This is the amount of main storage that can be used by the pool.
- Activity level: The maximum number of jobs that can run at the same time in the pool.

Up to 10 storage pool definitions can be specified for the subsystem description being created. Although each subsystem description can have as many as 10 storage pool definitions, there is an operational limitation on how many active storage pools can be in the *system*. In the system, no more than 16 storage pools can be active at one time, including the base storage pool and a machine storage pool. A storage pool for which *NOSTG has been specified is not considered active, and it is not allocated to any subsystem.

If a subsystem is started and all of its storage pools cannot be allocated without exceeding the 16-pool *system* maximum, the pools that can be allocated (up to the limit) are allocated and the remainder are not. Then, for each routing step started by that subsystem that normally is routed into one of the pools that was not allocated, the base pool is used instead. More information about storage pools is in the *Advanced Backup and Recovery Guide* and the *Work Management Guide*.

Element 1: Pool Identifier

pool-identifier: Specify the pool identifier of the storage pool definition to be in this subsystem. Valid values range from 1 through 10. The attributes of the pool must also be specified by one of the following values. As many as 10 sets of values can be specified in the POOLS parameter to define as many as 10 storage pools in the subsystem.

Element 2: Storage Size and Pool Definitions

***BASE:** The specified pool definition is defined to be the base system pool, which can be shared with other subsystems. The minimum size and activity level of the shared system pool are specified in the system values QBASPOOL and QBASACTLVL.

***NOSTG:** No storage and no activity level are assigned to the pool at first (the pool is inactive.)

***INTERACT:** The shared pool for interactive work.

***SPOOL:** The shared pool used for spooled writers.

***SHRPOOL1:** A general purpose shared pool.

***SHRPOOL2:** A general purpose shared pool.

***SHRPOOL3:** A general purpose shared pool.

***SHRPOOL4:** A general purpose shared pool.

***SHRPOOL5:** A general purpose shared pool.

***SHRPOOL6:** A general purpose shared pool.

***SHRPOOL7:** A general purpose shared pool.

***SHRPOOL8:** A general purpose shared pool.

***SHRPOOL9:** A general purpose shared pool.

***SHRPOOL10:** A general purpose shared pool.

storage-size: Specify the storage size (in kilobytes) of the specified storage pool. A value of at least 32 (meaning 32K) must be specified.

Element 3: Activity Level

activity-level: Specify the maximum number of jobs that can run at the same time in the pool.

Optional Parameters

MAXJOBS

Specifies the maximum number of jobs allowed in the subsystem. The maximum applies to all jobs that are started and are waiting or running, except for jobs on the job queue or jobs that have finished running.

***NOMAX:** The system maximum is used.

maximum-subsystem-jobs: Specify the maximum number of jobs allowed in the subsystem.

SGNDSPF

Specifies the qualified name of the sign-on display file that is used when showing sign-on displays at work stations allocated to the subsystem. If the sign-on display file does not exist when the subsystem is created, the user must specify a library qualifier because the sign-on display file name is kept. The sign-on display file must contain a record format by the name of SIGNON. More information is in the *Guide to Programming Displays*.

***QDSIGNON:** The sign-on display file value QDSIGNON in QSYS is used when showing sign-on displays at work stations that are allocated to the subsystem.

The name of the sign-on display file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

display-file-name: Specify the name of the sign-on display file that is used.

SYSLIBLE

Specifies a library that is specified ahead of other libraries on the system library list. This parameter allows the user to create a secondary language library causing messages and displays to appear in the user's spoken language.

Note: The second language library cannot be specified in the QSYSLIBL and QUSRLIBL portions of the library list. To use this parameter, the QSYSLIBL library must contain fewer than 15 libraries. If the prior restrictions are not followed, a message is sent to the job log, but the second language library is not added to the library list.

***NONE:** A secondary language library is not used.

library-name: Specify the name of the library to be used.

AUT

Specifies the authority given to users who do not have specific authority to the subsystem description, who are not on an authorization list, and whose user group has no specific authority to the subsystem description.

***LIBCRTAUT:** The public authority for the subsystem description is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the subsystem description). The public authority is determined when the subsystem description is created. If the CRTAUT value for the library changes after the subsystem description is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove user ids.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the subsystem description.

***USE:** The user can perform basic operations on the subsystem description, such as running a program or reading a file. The user cannot change the subsystem description. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the subsystem description.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the subsystem description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a Description With a Signon Display File

```
CRTSBSD  SBSD(BAKER)  POOLS((1 *BASE)(2 200 4))
  SGNDSPF(*LIBL/NEWSGNON)
  TEXT ('Subsystem for running Baker Department jobs')
```

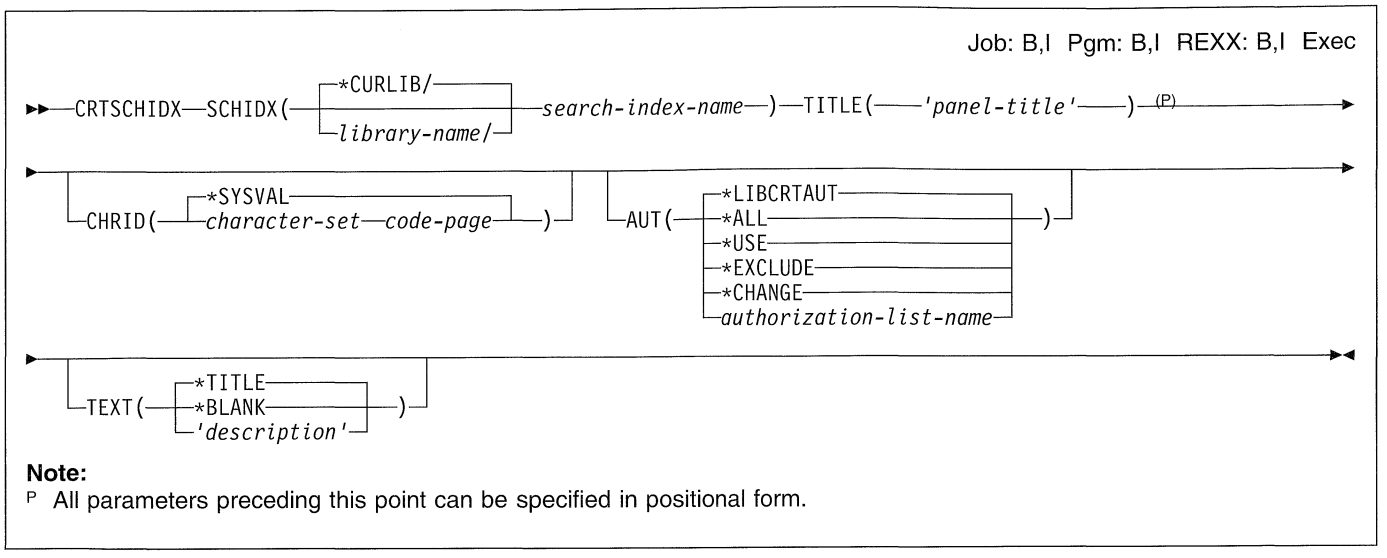
This command creates a subsystem description named BAKER and stores it in the current library. If there is no current library, then it is stored in the general purpose library (QGPL). Storage pool definition 1 specifies that pool 1 is to share the base system pool; the definition of storage pool 2 is to have 200K of storage and an activity level of 4. There is no limit in this subsystem description on the number of jobs that can be active at the same time. The activity levels in the subsystem may, however, be controlled by MAXACT parameters specified in work station entries, job queue entries, and routing entries that are in the subsystem. The sign-on display file is NEWSGNON and is used when showing sign-on displays at work stations allocated to the BAKER subsystem. The user's library list is searched for the NEWSGNON display file.

Example 2: Creating a Description that Contains Three Storage Pool Definitions

```
CRTSBSD  SBSD(MEDLIB/MEDICAL)
  POOLS((1 150 2)(2 *BASE) (3 *NOSTG)
  MAXJOBS(5)  TEXT('Medical files
  Inquiry and update')
```

This command creates a subsystem description named MEDICAL and stores it in the MEDLIB library. The subsystem description contains three storage pool definitions: storage pool 1 is defined to have 150K of storage and an activity level of 2, pool 2 is to share the base system pool, and pool 3 is defined first to be inactive when the other pools are active—it has no storage and no activity level. Up to five jobs can be active at the same time in this subsystem. A text description briefly describes the subsystem.

CRTSCHIDX (Create Search Index) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Search Index (CRTSCHIDX) command is used to create a search index. A search index is a reference to the help information contained in one or more panel groups.

The user can access a search index through data description specifications (DDS), by pressing the Help key, or through the information search function using the Start Search Index (STRSCHIDX) command.

A search index created by the CRTSCHIDX command does not contain any data. The user can add data by using the Add Search Index Entry (ADDSCHIDXE) command.

Restrictions:

1. The user of this command must have *ADD authority for the library where the search index is located.
2. IBM-supplied panel groups cannot be added to a search index created with the CRTSCHIDX command; only panel groups created with the Create Panel Group (CRTPNLGRP) command can be added to a search index created with the CRTSCHIDX command.

Required Parameters

SCHIDX

Specifies the qualified name of the search index being created.

The name of the search index can be qualified by one of the following library values:

***CURLIB:** The search index is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the search index is created.

search-index-name: Specify the name of the search index.

TITLE

Specifies the title to appear at the top of each panel when the search information is presented.

'panel-title': Specify no more than 55 characters of text, enclosed in apostrophes.

Optional Parameters

CHRID

Specifies the graphic character set and code page values used for the search index. The value of TXTCHRID for panel groups added must match the value of this parameter. (TXTCHRID is an attribute on a UIM panel group tag.)

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

Element 1: Character-set

character-set: Specify the graphic character set values that match the code page of the synonyms used in the search index.

Element 2: Code-page

code-page: Specify the code page value that matches the code page of the synonyms used in the search index.

AUT

Specifies the authority given to users who do not have specific authority to the object, who are not on the authorization list, or whose user group has no specific authority to the object.

***LIBCRTAUT:** The authority for the library in which the search index is being created is the authority for the search index.

***ALL:** The user can perform all operations on the search index except those limited to the owner. The user can create, delete, change, and specify security for the search index. The user can transfer ownership of the search index.

***USE:** The user can perform basic operations on the search index, such as running a program or reading a file. The user cannot change the search index. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the search index.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

authorization-list-name: Specify the name of an authorization list from which the authority to the object is taken.

TEXT

Specifies text that briefly describes the search index. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***TITLE:** The first 50 characters of the title are used.

***BLANK:** Text is not specified.

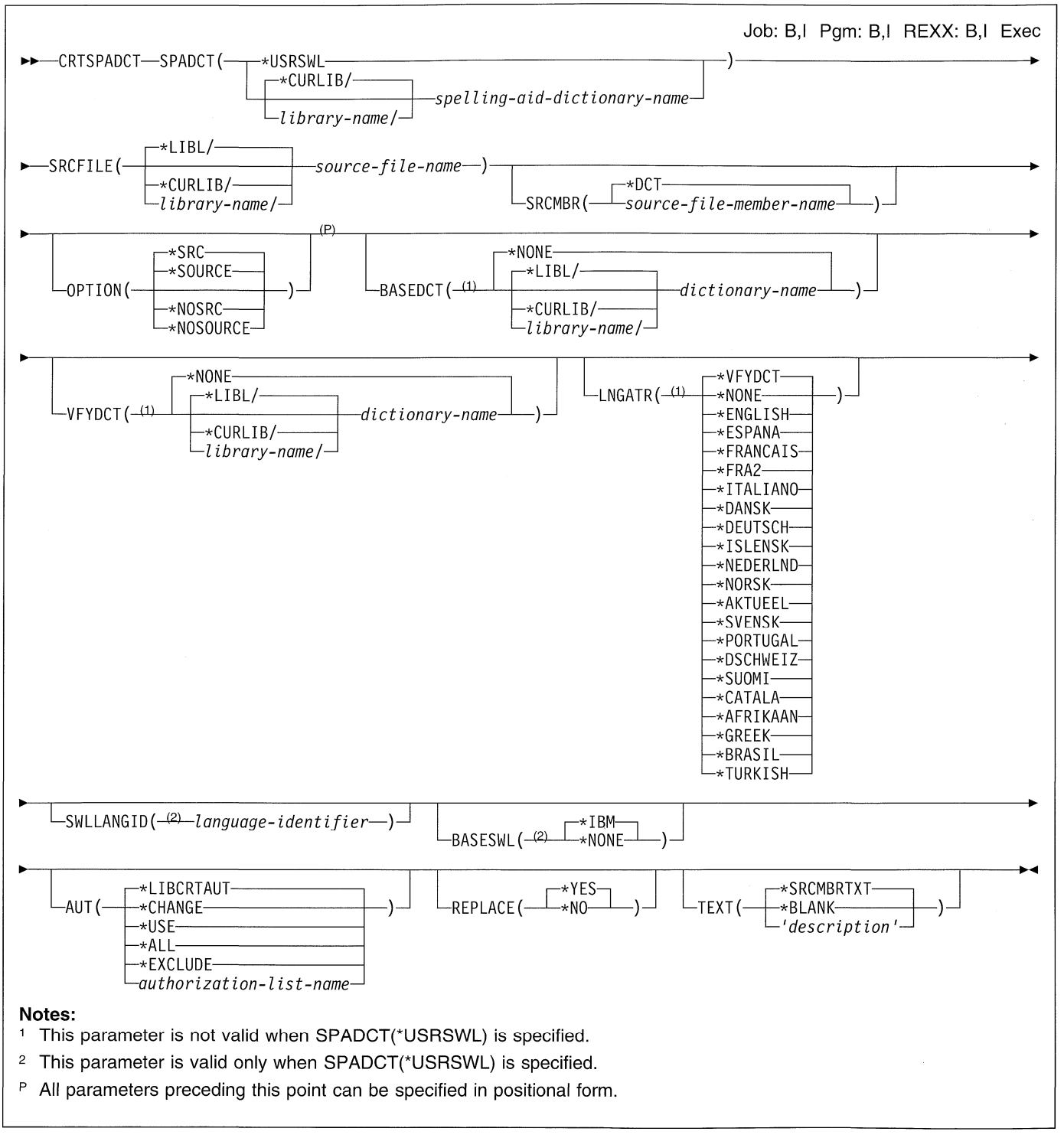
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTSCHIDX   SCHIDX(ACCOUNTING)
  TITLE('Accounting Help Index')
  TEXT('Accounting Help Index')
```

This command creates a search index named ACCOUNTING in the current library.

CRTSPADCT (Create Spelling Aid Dictionary) Command



Notes:

1 This parameter is not valid when SPADCT(*USRSWL) is specified.

2 This parameter is valid only when SPADCT(*USRSWL) is specified.

P All parameters preceding this point can be specified in positional form.

Purpose

The Create Spelling Aid Dictionary (CRTSPADCT) command allows the user to create a user-defined spelling aid dictionary. This dictionary is used with the Create Document (CRTDOC), Check Document (CHKDOC), Edit Document (EDTDOC), or Work with Documents (WRKDOC) commands when doing spell check functions, such as spelling aid and spelling verification.

Spelling aid dictionaries are classified into IBM language dictionaries and user dictionaries. Language dictionaries are IBM-created; user dictionaries are created with this command.

The following names are the IBM-supplied system dictionary names, in alphabetical order, that are in library QDCT. It is recommended that they are not used for user-created dictionaries:

AFRIKAAN
AKTUEEL
BRASIL
CATALA
DANSK
DEUTSCH
DSCHWEIZ
ESPANA
FRANCAIS
FRA2
GREEK
ISLENSK
ITALIANO
LEGAL
MEDICAL
NEDERLND
NORBOK
NORNYN
PORTUGAL
SUOMI
SVENSK
UK
US

This command requires a source file that contains all the words to be put into the spelling aid dictionary. The source member is created by using the Start Source Entry Utility (STRSEU) command. The dictionary source type is SPADCT.

More information on user-defined spelling aid dictionaries is in the *Using OfficeVision/400* Word Processing*.

Required Parameters

SPADCT

Specifies the qualified name of the spelling aid dictionary being created.

***USRSWL:** A user-defined stop word list is created using an IBM-supplied name.

The possible library values are:

***CURLIB:** The current library is used to locate the dictionary. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the dictionary is located.

spelling-aid-dictionary-name: Specify the name of the spelling aid dictionary being created.

SRCFILE

Specifies the qualified name of the source file used when the spelling aid dictionary is created. The source file contains the source member used for creating the dictionary.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

source-file-name: Specify the name of the source file.

Optional Parameters

SRCMBR

Specifies the name of the source file member that contains the words for the dictionary being created; the member is in the source file specified in the SRCFILE parameter. If SRCMBR is not specified, the member name is the same as that of the dictionary being created; the default value *DCT implies that the name of the dictionary being created is used. A member name must be specified when the source file member to be processed does not have the same name as the spelling aid dictionary being created.

***DCT:** The source file member name is the same as that of the dictionary being created.

source-file-member-name: Specify the name of the member in the source file (specified by SRCFILE) used to create the spelling aid dictionary.

OPTION

Specifies the type of output created when the dictionary is created.

Source Listing Options

***SRC or *SOURCE:** A list of the source statements used to create the dictionary, as well as any errors that occur, is created.

***NOSRC or *NOSOURCE:** No list of the source statements is created unless errors occur.

BASEDCT

Specifies the qualified name of the dictionary that contains words added to the dictionary being created. A system dictionary (one created by IBM) cannot be used here.

***NONE:** No words from another dictionary are added into the dictionary being created.

The name of the dictionary can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

dictionary-name: Specify the name of the dictionary that contains words added to the dictionary being created.

VFYDCT

Specifies the qualified name of the dictionary that is searched for each word specified in the source member. Only words not found in the verify dictionary are placed in the new dictionary.

***NONE:** Use every word specified in the source member to create the spelling aid dictionary.

The name of the dictionary can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

dictionary-name: Specify the name of the dictionary that contains words not included in the dictionary being created.

LNGATR

Specifies the language attribute associated with the dictionary being created. Because each language has its own structure, knowing the language for which the new dictionary is intended makes creation and storage of the new dictionary easier and more effective for the system.

***VFYDCT:** The language attribute of the dictionary is the same as the language attribute of the dictionary specified on the VFYDCT parameter.

***NONE:** The dictionary created has no specific language attribute.

***ENGLISH:** The language attribute of the spelling aid dictionary is English.

***ESPANA:** The language attribute of the spelling aid dictionary is Spanish.

***FRANCAIS:** The language attribute of the spelling aid dictionary is French.

***FRA2:** The language attribute of the spelling aid dictionary is French, in which accents are required on uppercase characters.

***ITALIANO:** The language attribute of the spelling aid dictionary is Italian.

***DANSK:** The language attribute of the spelling aid dictionary is Danish.

***DEUTSCH:** The language attribute of the spelling aid dictionary is German.

***ISLENSK:** The language attribute of the spelling aid dictionary is Icelandic.

***NEDERLND:** The language attribute of the spelling aid dictionary is Dutch.

***NORSK:** The language attribute of the spelling aid dictionary is Norwegian.

***SVENSK:** The language attribute of the spelling aid dictionary is Swedish.

***PORTUGAL:** The language attribute of the spelling aid dictionary is Portuguese.

***DSCHWEIZ:** The language attribute of the spelling aid dictionary is Swiss German.

***SUOMI:** The language attribute of the spelling aid dictionary is Finnish.

***CATALA:** The language attribute of the spelling aid dictionary is Catalan.

***AFRIKAAN:** The language attribute of the spelling aid dictionary is Afrikaans.

***GREEK:** The language attribute of the spelling aid dictionary is Greek.

***BRASIL:** The language attribute of the spelling aid dictionary is Brazilian Portuguese.

***TURKISH:** The language attribute of the spelling aid dictionary is Turkish.

SWLLANGID

Specifies the language identifier (ID) for the stop word list.

BASESWL

Specifies whether the IBM-supplied stop word list words are included in the user-created stop word list.

***IBM:** The words from the IBM-supplied stop word list are added to the user-created stop word list.

***NONE:** No words from the IBM-supplied stop word list are added to the user-created stop word list.

Note: Word entries in the IBM-supplied stop word list source file preceded by a dash, (–), are not

added to the user-created stop word list even when BASESWL(*IBM) is specified.

AUT

Specifies the authority given to users who do not have specific authority to the spelling aid dictionary, who are not on an authorization list, and whose user group has no specific authority to the spelling aid dictionary. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the spelling aid dictionary is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the spelling aid dictionary). The public authority is determined when the spelling aid dictionary is created. If the CRTAUT value for the library changes after the spelling aid dictionary is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***USE:** The user can perform basic operations on the spelling aid dictionary, such as running a program or reading a file. The user cannot change the spelling aid dictionary. *USE authority provides object operational authority and read authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the spelling aid dictionary.

***EXCLUDE:** The user cannot access the spelling aid dictionary.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether a dictionary being created has the same name as an existing dictionary. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** The dictionary being created replaces an existing dictionary that has the same name.

***NO:** No replacement occurs.

TEXT

Specifies text that briefly describes the spelling aid dictionary. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member being used to create the spelling aid dictionary. If the source file is a database file, the text is taken from the source member. You can add or change text for a database source member by using the Start SEU (STRSEU), Add Physical File Member (ADDPFM), or Change Physical File Member (CHGPFM) commands. If the source file is an inline or device file, the text is blank.

***BLANK:** Text is not specified.

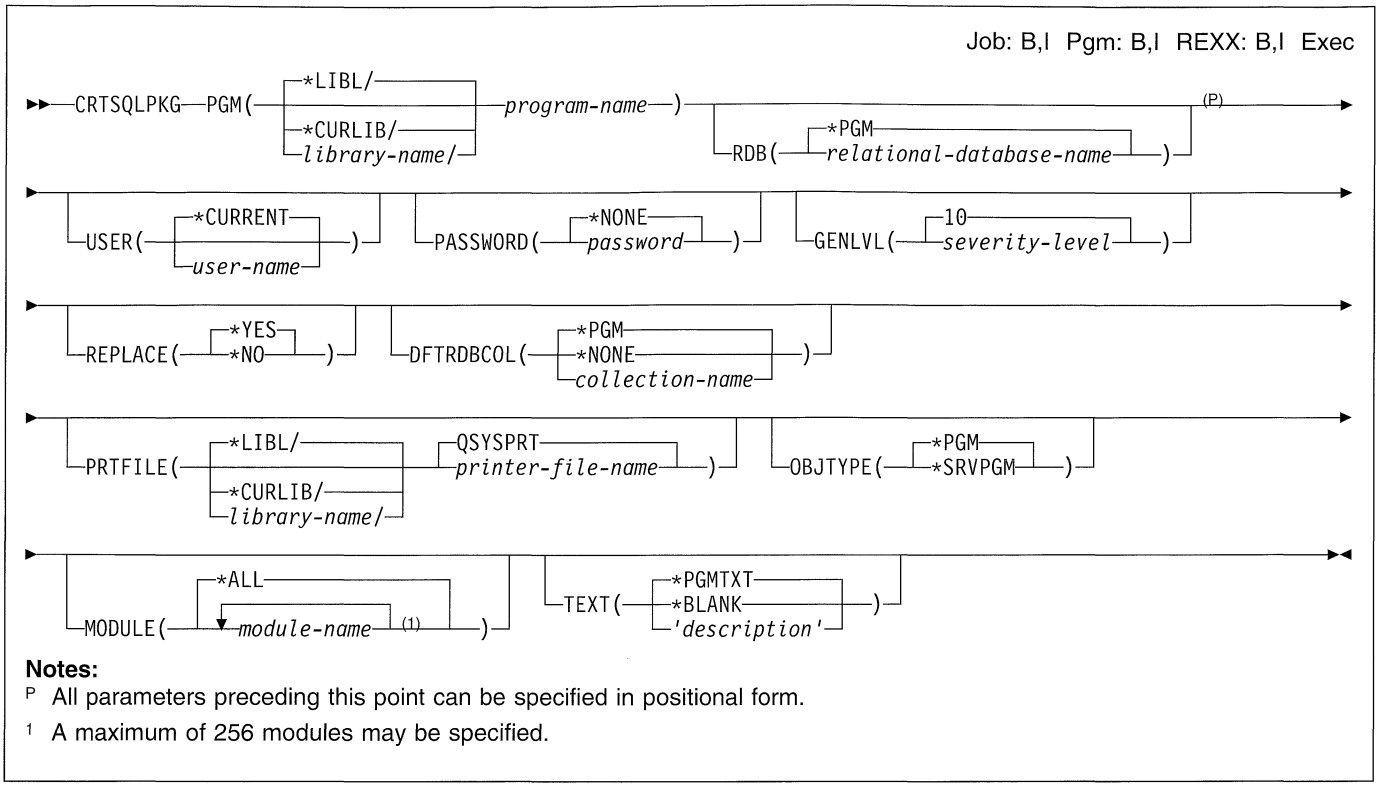
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTSPADCT   SPADCT(MYLIB/MYDCT)  SRCFILE(MYLIB/SRC)
  SRCMBR(WORDS)  OPTION(*NOSRC)  BASEDCT(QGPL/BASDCT)
  VFYDCT(QDCT/US)  LNGATR(*ENGLISH)  REPLACE(*YES)
```

This command creates a spelling aid dictionary named MYDCT in the library MYLIB. The words used in the spelling aid dictionary are from source member WORDS of the SRC source file in MYLIB. The dictionary includes words from a dictionary named BASDCT, but does not contain any words that are found in the dictionary named US. The dictionary being created has the *ENGLISH attribute. If an existing dictionary is named MYDCT, it is replaced.

CRTSQLPKG (Create Structured Query Language Package) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 256 modules may be specified.

Purpose

The Create Structured Query Language Package (CRTSQLPKG) command is used to create (or re-create) an SQL package on a relational database from an existing distributed SQL program. A distributed SQL program is a program created by specifying the RDB parameter on a CRTSQLxxx (where xxx = C, CI, CBL, FTN, PLI, or RPG) command.

Required Parameter

PGM

Specifies the qualified name of the program for which the SQL package is being created. The program must be a distributed SQL program.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program for which the package is being created.

Optional Parameters

RDB

Specifies the name of the relational database where the SQL package is being created.

***PGM:** The relational database name specified for the SQL program is used. The relational database name is specified on the RDB parameter of the distributed SQL program.

relational-database-name: Specify the name of the relational database where the SQL package is to be created. Use the Work with Relational Database Directory Entry (WRKRDBDIRE) command to show the relational database names that are valid on this parameter.

USER

Specifies the user name sent to the remote system when starting the conversation.

***CURRENT:** The user name associated with the current job is used.

user-name: Specify the user name being used for the application server job.

PASSWORD

Specifies the password to be used on the remote system.

***NONE:** No password is sent. If this value is specified, USER(*CURRENT) must also be specified.

password: Specify the password of the user name specified on the USER parameter.

GENLVL

Specifies the maximum severity level allowed for errors detected during SQL package creation. If errors occur at a level that exceeds the specified level, the SQL package is not created.

10: The default severity-level is 10.

severity-level: Specify the maximum severity level. Valid values range from 0 through 40.

REPLACE

Specifies whether an existing package is being replaced with the new package. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** An existing SQL package of the same name is replaced by the new SQL package.

***NO:** An existing SQL package of the same name is not replaced; a new SQL package is not created if the package already exists in the specified library.

DFTRDBCOL

Specifies the collection name to be used for unqualified names of tables, views, indexes, and SQL packages. This parameter applies only to static SQL statements in the package.

***PGM:** The collection name specified for the SQL program is used. The default relational database collection name is specified on the DFTRDBCOL parameter of the distributed SQL program.

***NONE:** Unqualified names for tables, views, indexes, and SQL packages use the search conventions specified on the OPTION parameter of the CRTSQLxxx command used to create the program.

collection-name: Specify the collection name that is used for unqualified tables, views, indexes, and SQL packages.

PRTFILE

Specifies the qualified name of the printer device file to which the create SQL package error listing is directed. If no errors are detected during the creation of the SQL package, no listing is produced.

The name of the printer file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QSYSPRT: If a file name is not specified, the create SQL package error listing is directed to the IBM-supplied printer file QSYSPRT.

printer-file-name: Specify the name of the printer device file to which the create SQL package error listing is directed.

OBJTYPE

Specifies the type of program for which an SQL package is created.

***PGM:** Create an SQL package from the program specified on the PGM parameter.

***SRVPGM:** Create an SQL package from the service program specified on the PGM parameter.

MODULE

Specifies a list of modules in a bound program.

***ALL:** An SQL package is created for each module in the program. An error message is sent if none of the modules in the program contain SQL statements or none of the modules is a distributed module.

Note: CRTSQLPKG can process programs that do not contain more than 1024 modules.

module-name: Specify the names of up to 256 modules in the program for which an SQL package is to be created. If more than 256 modules exist that need to have an SQL package created, multiple CRTSQLPKG commands must be used.

Duplicate module names in the same program are allowed. This command looks at each module in the program and if *ALL or the module name is specified on the MODULE parameter, processing continues to determine whether an SQL package should be created. If the module is created using SQL and the RDB parameter is specified on the precompile command, an SQL package is created for the module. The SQL package is associated with the module of the bound program.

TEXT

Specifies text that briefly describes the SQL package and its function.

***PGMTXT:** The text from the program for which the SQL package is being created is used.

***BLANK:** No text is specified.

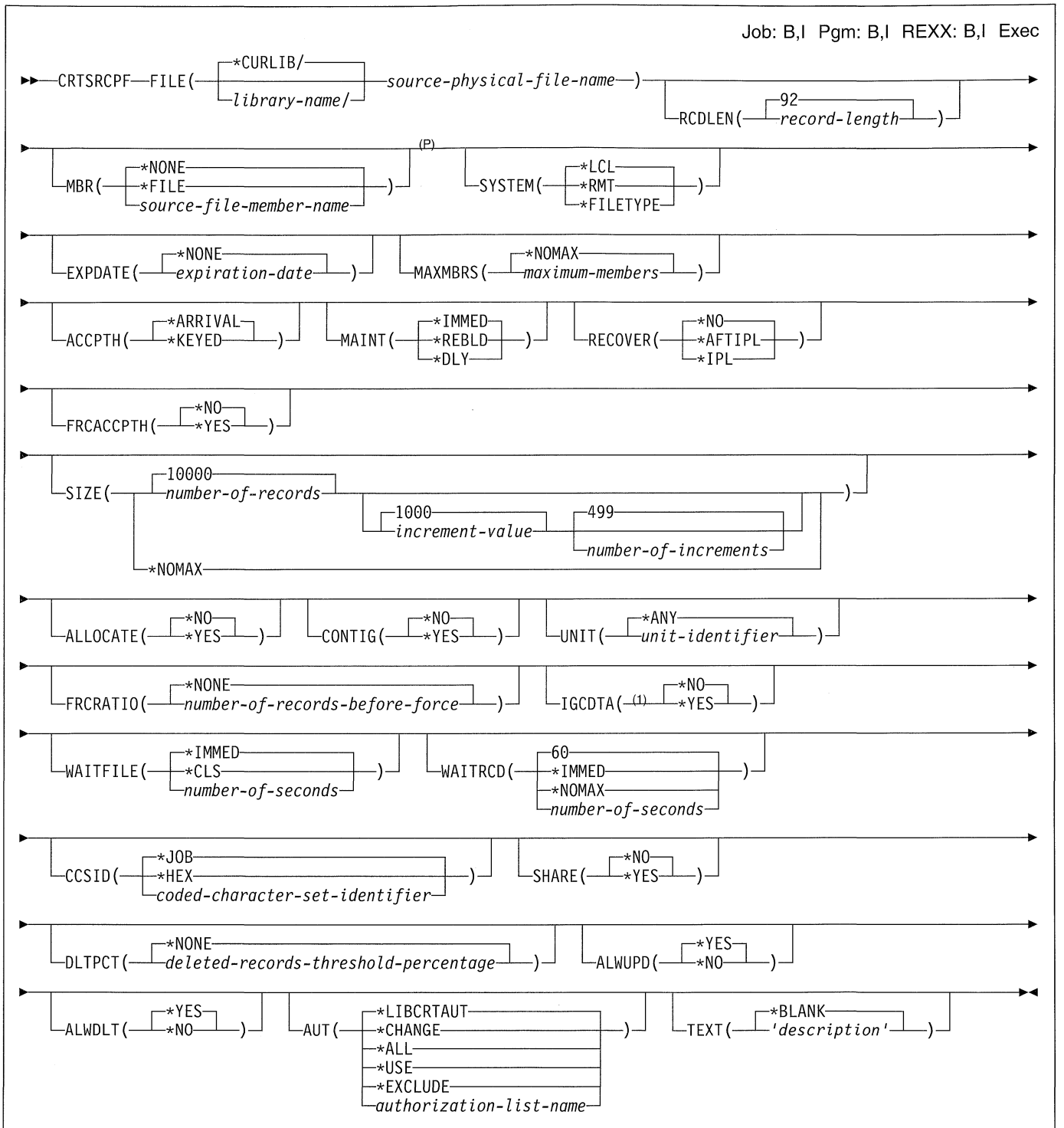
'description': Specify a maximum of 50 characters of text, enclosed in apostrophes.

Example

```
CRTSQLPKG   PAYROLL  RDB(SYSTEMA)
  TEXT('Payroll Program')
```

This command creates an SQL package from the distributed SQL program PAYROLL on relational database SYSTEMA.

CRTSRCPF (Create Source Physical File) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 DBCS systems only

Purpose

The Create Source Physical File (CRTSRCPF) command creates a source physical file in the database. A source physical file is created from the file description parameters in the CRTSRCPF command; it is used to store source records that are used as input to IBM-supplied source processors, such as the data description specifications (DDS) processor, CL compiler, or RPG III compiler. To override attributes of the file after it has been created, use the Override Database File (OVRDBF) command before the file is opened. To change attributes of the file after it has been created, use the Change Source Physical File (CHGSRCPF) command.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the source physical file can be qualified by one of the following library values:

***CURLIB:** The source physical file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the source physical file is created.

source-physical-file-name: Specify the name of the source physical file being created.

Note: If a DDM file is specified, the source physical file (specified in the RMTFILE parameter of the CRTDDMF command) is created on a remote system (specified in the RMTLOCNAME parameter of the CRTDDMF command). See the SYSTEM parameter of this command.

Optional Parameters

RCDLEN

Specifies the length (in bytes) of the records being stored in the source file. The format of each record contains three fields: the sequence number of the record, a date field, and the source statement. The record format name is the same as the file itself, specified in the FILE parameter. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

The RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and at least one position for source start. If the Copy File (CPYF) command is used to copy records into the file, and the records are longer than the length specified, the records are truncated on the right. These fields are defined with fixed attributes and names, and have a keyed access path over the sequence number if ACCPTH(*KEYED) is specified.

92: The default record length is 92 characters. Six characters are for the record sequence number, 6 are for the record date, and the remaining 80 characters are for the source statement.

record-length: Specify the record length of each source record in the file. Valid values range from 13 through 32766.

Double-Byte Character Set Considerations:

If IGCDTA(*YES) is specified, the RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and at least four positions for source start. Valid values range from 16 through 32766.

MBR

Specifies the name of the source file member being added when the source file is created. Other members can be added to the file after it is created by using the ADDPFM command.

***NONE:** No member is added when the file is created.

***FILE:** The member being added has the same name as that of the source file that contains the member (specified in the FILE parameter).

source-file-member-name: Specify the name of the member that is added when the source file is created.

SYSTEM

Specifies the system on which the source physical file is created.

***LCL:** The source physical file is created on the local system.

***RMT:** The source physical file is created on a remote system that is using DDM. The file name specified on the FILE parameter must be the name of a DDM file (created by using the CRTDDMF command). The DDM file contains the name of the source physical file being created (RMTFILE parameter on the CRTDDMF command) on the remote system (RMTLOCNAME parameter on the CRTDDMF command).

***FILETYPE:** If the name specified on the FILE parameter is a DDM file, the physical file is created on the remote system specified by the RMTLOCNAME parameter on the CRTDDMF command for that DDM file. Otherwise, the name on the FILE parameter cannot be the name of an existing file, since a physical file of that name is created on the local system.

EXPDATE

Specifies the expiration date. The files cannot be overwritten until the expiration date. The expiration date must be later than or equal to the current date.

Note: Attempting to open a member for which the expiration date has been exceeded sends an error message to the user.

***NONE:** No expiration date is specified.

expiration-date: Specify the date after which the source file member cannot be used. The date must be in the format specified by the QDATFMT and QDATSEP job attributes. The date must be enclosed in apostrophes if special characters are used in the format.

MAXMBRS

Specifies the maximum number of members that the source file being created can contain at any one time.

***NOMAX:** The system maximum is used.

maximum-members: Specify the maximum number of members that the source file can contain. Valid values range from 1 through 32767 members per file.

ACCPTH

Specifies the type of access path used by all members of the file.

***ARRIVAL:** The access path governs the arrival sequence order. Using this parameter value reduces the size of the file and avoids maintenance of the keyed access path.

***KEYED:** The access path is of keyed sequence order.

More information on keyed access paths and the arrival sequence orders of source file access paths is in the *Database Guide*.

MAINT

Specifies the type of access path maintenance used for all members of the source file that have keyed access paths.

***IMMED:** The access path is maintained for each physical file member whether the source physical file is opened or closed. The access path is changed whenever a record is updated, added to, or deleted from a member of this file or a logical file member based on a member of this file.

***REBLD:** The access path is completely rebuilt when a file member is opened during program running. The access path is continuously maintained until the member is closed, then the access path maintenance ends.

***DLY:** The maintenance of the access path is delayed until the physical file member is opened for use. The access path is changed only for records that have been added, deleted, or changed since the file was last opened. While the file is open, all changes made to its members are immediately reflected in the access path of those members, no matter what is specified for the MAINT parameter. To prevent a lengthy rebuild time when the file is opened, *DLY should be specified only when the number of changes to the access path is small.

If the number of changes between a close and the next open reaches approximately 10 percent of the access path size, the system stops saving changes and the access path is completely rebuilt the next time the file is opened.

RECOVER

Specifies, for files having immediate or delayed maintenance on their access paths, when recovery processing of the file is performed after a system failure has occurred while the access path is being changed. This parameter is valid only for files with a keyed access path.

If *IMMED is specified for the MAINT parameter, the access path can be rebuilt during initial program load (IPL) (before any user can run a job), after IPL has ended (when jobs are running at the same time), or when the file is reopened. While the access path is being rebuilt, the file cannot be used by any job.

During the IPL, an Override Access Path Recovery display lists access paths that must be recovered and what the RECOVER parameter value is for each. The RECOVER parameter values on this display can be overridden. More information on this is in the *Advanced Backup and Recovery Guide*.

If *REBLD is specified for the MAINT parameter, the access path can be rebuilt the next time its file is opened.

***NO:** The access path of the file is not rebuilt during *IPL or *AFTIPL. The file's access path, if not valid, is rebuilt when the file is reopened.

***AFTIPL:** The file has its access path rebuilt after the IPL has been completed. This option allows other jobs not using this file to start processing immediately after the IPL has been completed. If a job tries to allocate the file while its access path is being rebuilt, a file- open exception occurs.

***IPL:** The file has its access path rebuilt during the IPL. This ensures that the file's access path is rebuilt before the first user program tries to use it. However, no jobs can start running until after all files that specify RECOVER(*IPL) have their access paths rebuilt.

FRCACCPTH

Specifies, for files with keyed access paths only, whether access path changes are forced to auxiliary storage along with the associated records in the file whenever the access path is changed. FRCACCPTH(*YES) minimizes (but does not remove) the possibility that an abnormal job end can cause damage to the access path, which requires it to be rebuilt.

***NO:** The access path and updated records are not forced to auxiliary storage whenever the access path is changed.

***YES:** The access path and updated records are forced to auxiliary storage whenever the access path is changed. If FRCACCPTH(*YES) is specified, MAINT(*REBLD) cannot be specified.

FRCACCPTH(*YES) slows the response time of the system if the access path is changed in an interactive job. If the access path is changed frequently, the overall performance of the system is somewhat decreased.

SIZE

Specifies the *initial* number of records in each member of the file, the number of records in each increment that can be automatically added to the member size, and the number of times the increment is automatically applied. The number of records for each file member is specified as the number of records that can be placed in it. This includes deleted records.

When the maximum number of records has been reached, a message, stating that the member is full is sent to the system operator, offering the choice of ending the request or extending the member size. The operator can extend the member by 10% or by the number of records specified as the increment value, whichever is greater, each time the message is received.

A list of 3 values can be specified to indicate the initial size of each member and the automatic extensions that can be added when needed, or *NOMAX can be specified. If SIZE is not specified, SIZE(10000 1000 499) is assumed by the system.

Element 1: Number of Records

One of the following is used to specify the *initial* number of records in the member before an automatic extension of the member occurs. The ALLOCATE parameter determines when the required space for the initial number of records is allocated. If *YES is specified, the space is allocated when a new member is added. If *NO is specified, the initial space is allocated as determined by the system.

10000: Initially, up to 10,000 records can be inserted into each member of the file before an extension occurs.

number-of-records: Specify the number of records (ranging from 1 through 16777215) that can be inserted before an automatic extension occurs. If automatic extensions are not wanted, enter zeros for the second and third values in the list.

Element 2: Increment Value

One of the following is used to specify the number of records that can be additionally inserted in the member when the initial member size is exceeded and an automatic extension occurs. The minimum size of an increment is 10% of the size of the member at the time the maximum number of records is reached.

1000: The member size is increased by 10% or 1,000 records, whichever is greater.

increment-value: Specify the number of additional records (ranging from 0 through 32767) which, if greater than 10% of the size of the member when the maximum number of records is reached, are to be added to the member during an automatic extension.

If the number specified is not greater than 10% of the member size and not equal to zero, the member size is increased by 10%.

Specify 0 to prevent automatic extensions. This value must be 0 if the value for the number of increments is 0.

Element 3: Maximum Number of Increments

One of the following values is used to specify the maximum number of increments that can be automatically added to a file member. If 0 is specified for the increment amount, the number of increments need not be specified; 0 will be the default value instead of 499 and a file-member-full message is sent to the user issuing the command.

499: Up to 499 increments can be automatically added to the member size.

number-of-increments: Specify the maximum number of increments, ranging from 0 through 32767, that can be automatically added to the member. To prevent automatic extensions, specify a value of 0.

Other Single Values

If SIZE(*NOMAX) is specified, ALLOCATE(*NO) must also be specified.

***NOMAX:** The system maximum is used.

ALLOCATE

Specifies whether storage space is allocated for the initial number of records (SIZE parameter) for each source file member as it is added. The allocation provides enough space to hold the initial number of records specified by the SIZE parameter. Later allocations, which occur when a record cannot be added to a member without exceeding its capacity, are determined by the system and by the SIZE parameter values.

***NO:** Minimum storage space is initially allocated when the member is added. The system determines when space allocations are necessary and determines the size of each allocation.

***YES:** Storage space is initially allocated as each member is added. The amount specified in the first value of the SIZE parameter (the number of records) is allocated. If the space cannot be allocated, a message is sent to the user and the affected member is not added. If ALLOCATE(*YES) is specified, SIZE(*NOMAX) cannot be specified.

CONTIG

Specifies whether records in the initial allocation in each source file member are stored contiguously (next to each other) on auxiliary storage. If so, and the necessary contiguous space is not available, the system sends a message to the job log and allocates the storage space separately. The file is still usable. This parameter does not affect additional allocations needed later, which most likely would be noncontiguous.

***NO:** The storage space for each member does not have to be together.

***YES:** The system allocates contiguous space for the added members of the source file. If the members are stored separately, the user is notified and a message is put in the job log. If CONTIG(*YES) is specified, ALLOCATE(*YES) must also be specified.

UNIT

Specifies whether a file is stored on a specific auxiliary storage unit. The system attempts to allocate the storage space for the file and for all of its members and their associated access paths on the specified unit. This includes the initial allocation when each member is added and any extensions that occur later for each member in the file. If the system cannot allocate the storage space for each member on the specified unit, it allocates the space on any available unit and sends a message to the job log. The file is entirely usable in all cases.

The unit identifier is a number ranging from 1 through 255 that is assigned when a new disk device is configured. The user can display and change the configured disk device using the Work with Disk Devices display from the Start System Service Tool (STRSST) command. More information on the System Service Tool (SST) is in the *Advanced Backup and Recovery Guide*.

***ANY:** The storage space for the file and its members is allocated on any available auxiliary storage unit.

unit-identifier: Specify the storage unit where the system attempts to allocate storage space for the file and all of its members.

If the unit specified is part of any user auxiliary storage pool (auxiliary storage pools ranging from 2 through 16), the system allocates space from ASP 1, the system auxiliary storage pool.

FRCRATIO

Specifies the number of inserted, updated, or deleted records processed before being forced to auxiliary (permanent) storage. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

If this physical file is being journaled, a larger force ratio should be specified. More information on journal management is in the *Advanced Backup and Recovery Guide*.

***NONE:** There is no forced write ratio; the system determines when the records are written to auxiliary storage.

number-of-records-before-force: Specify the number of inserted, updated, or deleted records processed before they are explicitly forced to auxiliary storage.

IGCDTA

Specifies whether the file contains double-byte character set (DBCS) data.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds a program waits for the file resources to be allocated to the job. Valid values range from 1 through 32767 seconds.

WAITRCD

Specifies the number of seconds that a program waits for a record to be updated or deleted, or for a record read in the commitment control environment with LCKLVL(*ALL) specified. More information on record locking is in the *Database Guide*. If the record is not allocated in the specified wait time, an error message is sent to the program.

60: A program waits 60 seconds for a requested record.

***IMMED:** The program does not wait; when a record is locked, an immediate allocation of the record is required.

***NOMAX:** The system maximum is used.

number-of-seconds: Specify the number of seconds that a program waits for a record requested for reading, updating, or deleting. Valid values range from 1 through 32767 seconds.

CCSID

Specifies the coded character set identifier (CCSID) being used to describe character data in the fields of the source file.

A CCSID is a 16-bit number identifying a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and additional coding-related information that uniquely identifies the coded graphic representation used.

***JOB:** The current job's CCSID is used.

***HEX:** The CCSID 66535 is used, which indicates that the character data in the fields is treated as bit data and is not converted.

coded-character-set-identifier: Specify the CCSID being used. More information on valid CCSIDs is in the *National Language Support Planning Guide*.

SHARE

Specifies whether the open data path (ODP) for the source physical file is shared with other programs in the routing step. When an ODP is shared, the programs

accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

DLTPCT

Specifies the maximum percentage of deleted records allowed in any member in the source file. The percentage is based on the number of deleted records compared with the total record count in a member. The percentage check is made when any member of the file is closed or any logical file member based on any member of the file is closed. If the number of deleted records exceeds the percentage, a message is sent to the system history log (QHST) to inform the user.

***NONE:** No percentage is specified; the number of deleted records in the file members is not checked when a member is closed.

deleted-records-threshold-percentage: Specify, from 1 through 100, the largest percentage of deleted records allowed in any member in the file. If this percentage is exceeded, a message is sent to the system history log (QHST) when the file is closed.

ALWUPD

Specifies whether records are updated in the physical file. Records in a logical file can be updated only when the records in each physical file on which the logical file is based can be updated.

***YES:** Records can be updated in the physical file.

***NO:** Records cannot be updated in this physical file or in any logical file built over this physical file.

ALWDLT

Specifies whether records can be deleted from the physical file. Records in a logical file can be deleted only when the records in each physical file on which the logical file is based can be deleted.

***YES:** Records can be deleted in this physical file.

***NO:** Records cannot be deleted in this physical file or from any logical file built over this physical file.

AUT

Specifies the authority given to users who do not have specific authority to the source physical file, who are not on an authorization list, and whose user group has no specific authority to the source physical file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the source physical file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the source physical file). The public authority is determined when the source physical file is created. If the CRTAUT value for the library changes after the source physical file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the source physical file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the source physical file, but the user cannot change it. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the source physical file.

***USE:** The user can perform basic operations on the source physical file, such as running a program or reading a file. The user cannot change the source physical file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the source physical file.

authorization-list-name: Specify the name of the authorization list used.

TEXT

Specifies text that briefly describes the source physical file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a File Without Members

```
CRTSRCPF   FILE(SRCLIB/PAYTXS)
```

This command creates a source file named PAYTXS in the SRCLIB library. The file is created without any members; therefore, no data can be put into the file until a member is added later. As many as 32,767 members (*NOMAX) can be added to the file.

Each member can have up to 10000 records before automatic extensions (499 increments maximum) occur that add

1000 records to the capacity of the member. Only minimum initial storage is allocated for each member with no restrictions on which storage unit is used or whether the space is connected. The public has object operational, read, add, delete, and update authority for the file, but no object management or object existence authority.

Example 2: Creating a File With a Member

```
CRTSRCPF   FILE(ORDERCTL/ORDERS)  MBR(*FILE)
  SIZE(100 50 5)  UNIT(01)
```

This command creates a source physical file named ORDERS in the ORDERCTL library. All records placed in the file are stored on auxiliary storage unit 01, but the space does not have to be connected. The initial allocation of storage provides for up to 100 records, and up to five increments of additional space for 50 records each can be added automatically. These allocation values also apply to members of this source file that will be added later.

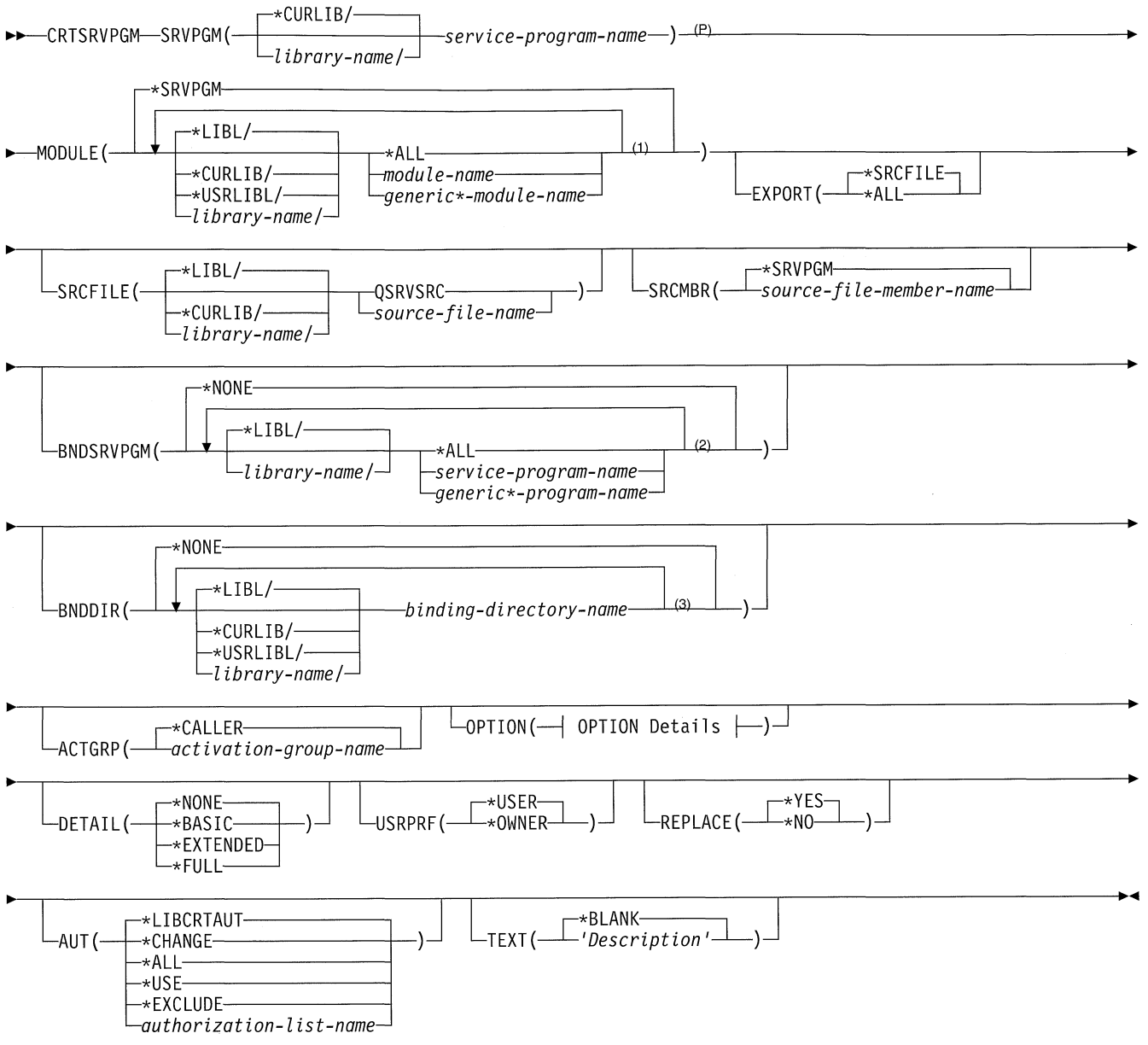
Example 3: Creating a File that Contains DBCS Data

```
CRTSRCPF   FILE(IGCLIB/IGCSRC)  IGCDTA(*YES)
```

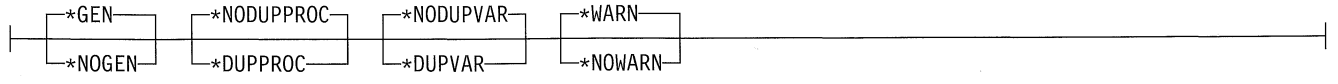
This command creates a source physical file named IGCSRC, which is stored in the library IGCLIB, and can contain DBCS data.

CRTSRVPGM (Create Service Program) Command

Job: B,I Pgm: B,I REXX: B,I Exec



OPTION Details:



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 150 module entries.

2 A maximum of 50 service program entries for binding.

3 A maximum of 50 binding directory entries.

Purpose

The Create Service Program (CRTSRVPGM) command creates a bound service program from a set of modules and binding directories.

Restrictions:

1. You must have *CHANGE authority for the library where the service program is being created.
2. You must have *USE authority to the specified modules, service programs, and binding directories.

Required Parameters

SRVPGM

Specifies the qualified name of the service program object that is created.

The name of the service program object can be qualified by one of the following library values:

<u>*CURLIB:</u> The service program object is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the service program object is created.

service-program-name: Specify the name of the created service program.

MODULE

Specifies the list of modules that are copied and bound together to create the service program object. If duplicate module and library specifications are found, only the first instance of the duplicate module and library is used. Modules in this list are copied into the final service program object.

<u>*SRVPGM:</u> The module and library name specified on the SRVPGM parameter is used. The name specified on the SRVPGM parameter is used as the module name.

The name of the service program object can be qualified by one of the following library values:

<u>*LIBL:</u> All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

***ALL:** Find all module objects in the specified library or libraries.

module-name: Specify the name of the module that is copied to create the service program object.

generic-module-name:* Specify all module objects starting with the characters preceding the * in the specified library or libraries.

Optional Parameters

EXPORT

Specifies the names of the data and procedures this service program exports.

<u>*SRCFILE:</u> The source file and source member specify the data and procedures to export from the service program.

***ALL:** All data and procedures that are exported from the included modules are also exported from the service program.

SRCFILE

Specifies the source file containing the specifications for exporting data and procedures from this service program.

<u>QSRVSRC:</u> The name of the source file containing the specifications for exporting data and procedures.

The name of the source file can be qualified by one of the following library values:

<u>*LIBL:</u> All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

source-file-name: Specify the name of the source file containing the specifications for exporting data and procedures.

SRCMBR

Specifies the name of the source file member containing the specifications for exporting data and procedures from this service program.

<u>*SRVPGM:</u> The source file member name is the same name as the service program name specified for the SRVPGM parameter.

source-file-member-name: Specify the name of the member in the source file containing the specifications for exporting data and procedures.

BNDSRVPGM

Specifies the list of service programs that are examined during symbol resolution.

<u>*NONE:</u> No service programs are provided for symbol resolution.

***LIBL:** Search all of the libraries in the user and system portion of the job's library list for the specified service programs. If one of the service programs has an export that satisfies a module import,

then the library list is searched at run-time to find this service program.

library-name: Specify the name of the library where the service programs can be found. If one of the service programs has an export that satisfies a module import, and a specific library was specified, the specified library is searched at run-time to find this service program.

Note: QTEMP is not a valid library name for this parameter.

***ALL:** Find all service program objects in the specified library or libraries.

service-program-name: Specify the name of the service program to be examined during symbol resolution.

generic-program-name:* Specify all service program objects starting with the characters preceding the * in the specified library or libraries.

BNDDIR

Specifies the list of binding directories that are used in symbol resolution.

***NONE:** No binding directory is specified.

The name of the specified binding directory can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

binding-directory-name: Specify the name of the binding directory used in symbol resolution.

ACTGRP

Specifies the activation group this service program is associated with when it is called. An activation group provides:

- Run-time data structures to support the running of programs
- Addressing protection
- A logical boundary for message propagation
- A logical boundary for application cleanup processing

***CALLER:** When this service program gets called, the program is activated into the caller's activation group.

activation-group-name: Specify the name of the group that is associated with this called service program. If an activation group, by the specified name, currently exists when this service program is called, the service program is associated with the already existing activation group. If an activation group by the specified name does not currently exist when this service program is called, then a new activation group is created and the service program is associated with the newly created activation group.

OPTION

Specifies options to be used when the program object is created.

Program Objects

***GEN:** A program object is generated.

***NOGEN:** A program object is not generated.

Duplicate Procedure Names

***NODUPPROC:** During the symbol resolution phase of the binding process, each procedure name that is exported from the modules and programs must be unique.

***DUPPROC:** During the symbol resolution phase of the binding process, the procedure names that are exported from the modules and programs do not have to be unique. When multiple duplicate procedures are allowed, the first exported procedure in the list of specified modules and programs that matches the import request is the procedure that is selected.

Duplicate Variable Names

***NODUPVAR:** During the symbol resolution phase of the binding process, each variable name that is exported from the modules and programs must be unique.

***DUPVAR:** During the symbol resolution phase of the binding process, the variable names that are exported from the modules and programs do not have to be unique. When multiple duplicate variables are allowed, the first exported variable in the list of specified modules and programs that matches the import request is the variable that is selected.

Issuing Diagnostic Messages

***WARN:** If duplicate variables or procedures are found, a diagnostic message is issued indicating what duplicates were found.

***NOWARN:** If duplicate variables and procedures are found, diagnostic messages are not issued.

DETAIL

Specifies the level of detail to be printed.

***NONE:** A listing is not generated.

***BASIC:** Contains a listing of the options passed to CRTPGM, and processing statistics. This listing also contains the Brief Summary Table.

***EXTENDED:** In addition to the information provided in the **BASIC* listing, this listing contains the Extended Summary Table and the Binding Information Listing.

***FULL:** This listing contains the **EXTENDED* listing and the Cross-Reference Listing.

Note: If a printed listing is requested, the print file *LIBL/QSYSPRT is used to generate the listing.

USRPRF

Specifies whether authority checking is performed only for the user running the program, or for both the user running the program and the program owner.

***USER:** The user profile of the program user is used when the program is run.

***OWNER:** The user profile of both the program owner and the program user is used when the program is run.

REPLACE

Specifies whether the existing program is replaced if a program by the same name already exists in the specified library.

***YES:** Replace the existing program by moving it to the QRPLOBJ library.

Note: Both programs must be owned by the same user for the replace to work.

***NO:** No replacement occurs.

AUT

Specifies the authority given to users who do not have specific authority to the service program, who are not on an authorization list, and whose user group has no specific authority to the service program.

***LIBCRTAUT:** The public authority for the service program is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the service program). The public authority is determined when the service program is created. If the CRTAUT value for the library changes after the service program is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the service program.

***USE:** The user can perform basic operations on the service program, such as running a program or reading a file. The user cannot change the service program. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the service program.

authorization-list: The authorization list must exist when the object is created. Users are granted authority to the object as specified by the list.

TEXT

Specifies text that briefly describes the program object. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

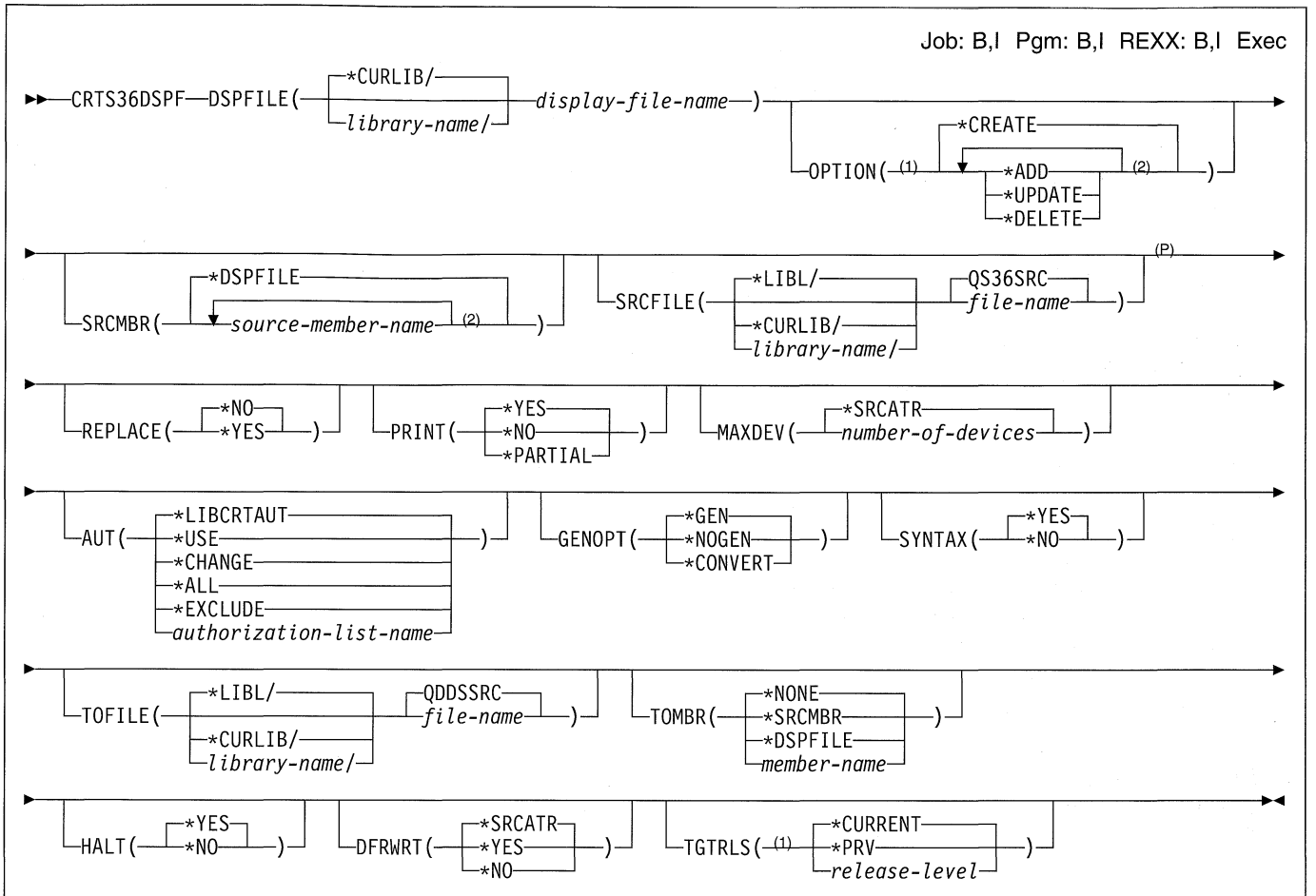
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTSRVPGM   SRVPGM(WORKDOC)
```

This command creates the service program object named WORKDOC in the current library.

CRTS36DSPF (Create System/36 Display File) Command



Notes:

1 TGTRLS(*PRV) is not valid if OPTION(*ADD) or OPTION(*UPDATE) is specified.

2 A maximum of 32 repetitions

P All parameters preceding this point can be specified in positional form.

Purpose

The Create System/36 Display File (CRTS36DSPF) command creates a display file from System/36 Screen File Generator (SFGR) source, and adds, deletes, or updates formats in existing display files.

Note: The data description specifications (DDS) source is saved in the DDS source file QS36DDSSRC, in the same library as the display file. If the QS36DDSSRC source file does not exist, this source file is created with a record length of 92.

Restriction: This command can be run either in the AS/400 system or, when library QSSP is installed, in the System/36 environment. Using the TOFILE and TOMBR parameters, the user can convert the System/36 SFGR source to the AS/400 system data description specifications (DDS) source.

Required Parameter

DSPFILE

Specifies the qualified name for the display file being created.

The name of the display file can be qualified by one of the following library values:

***CURLIB:** The display file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the display file is created.

display-file-name: Specify the name of the display file being created.

Optional Parameters

OPTION

Specifies what to do with the specified display file. A display file can be created, or formats can be added, deleted, or updated in existing display files. Up to 32 options can be specified. A given option applies to the corresponding name in the SRCMBR parameter.

***CREATE:** A display file is created using the specified display file name as defined by all of the specified source members. When *CREATE is specified, no other options may be specified.

***ADD:** One or more display formats are added to the specified display file as defined by the corresponding names specified on the SRCMBR parameter. TGTRLS(*PRV) is invalid if OPTION(*ADD) is specified.

***UPDATE:** One or more display formats are updated in the specified display file as defined by the corresponding names in the SRCMBR parameter. TGTRLS(*PRV) is invalid if OPTION(*UPDATE) is specified.

***DELETE:** A display format is deleted from the specified display file. The format that is deleted is the name specified by the corresponding name specified in the SRCMBR parameter. If the deleted format was the only format in the display file, the entire display file is deleted.

SRCMBR

Specifies the member used in the source file when performing a create, add, or update operation. For the delete option, specify the name of the format being deleted.

***DSPFILE:** A member with the same name as the display file is used for the create, add, delete, or update option.

source-member-name: Specify the source member name that is used. Up to 32 source member names can be specified. The specified name is used for the corresponding option in the OPTION parameter. When the corresponding option is *DELETE, a format name *must* be specified instead of a member name.

SRCFILE

Specifies the qualified name of the source file in which the source members are located.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QS36SRC: The source file, QS36SRC, is used.

file-name: Specify the name of the source file that is used.

REPLACE

Specifies whether an existing display file is replaced. When OPTION(*CREATE) is not specified, this parameter is ignored. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NO:** No replacement occurs.

***YES:** If a display file already exists, it is replaced. Other types of files are not replaced.

Note: A display file is not created if the display has the same name and library as an existing program or message file.

PRINT

Specifies whether the compiler listings are printed.

***YES:** The full compiler listing for both the SFGR source syntax checking and CRTDSPF steps are printed.

***NO:** No compiler listing is printed for the SFGR source syntax checking step. *PARTIAL is assumed for the CRTDSPF step.

***PARTIAL:** If errors are found in the SFGR source syntax checking or CRTDSPF steps, the compiler listing is printed for that step, along with the error and warning messages. If no messages are issued, no compiler listing is printed.

MAXDEV

Specifies the number of devices that can use the display file at one time.

***SRCATR:** The maximum number of devices is taken from the first source member attributes (see the CHGS36SRCA or EDTS36SRCA commands). If the MAXDEV source attribute has not been set or is zero, a value of 5 is used.

number-of-devices: Specify the maximum number of devices that are permitted to use the display file. Valid values range from 1 through 256.

AUT

Specifies the authority given to users who do not have specific authority to the System/36 display file, who are not on an authorization list, and whose user group has no specific authority to the System/36 display file.

***LIBCRTAUT:** The public authority for the System/36 display file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the System/36 display file). The public authority is determined when the System/36 display file is created. If the CRTAUT value for the library changes after the System/36 display file is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the System/36 display file, such as running a program or reading a file. The user cannot change the System/36

display file. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the System/36 display file.

***EXCLUDE:** The user cannot access the System/36 display file.

authorization-list-name: Specify the name of the authorization list used.

GENOPT

Specifies the options used to create the object.

***GEN:** The SFGR source syntax is checked unless SYNTAX(*NO) is specified. The SFGR source is converted to data description specifications (DDS), and an AS/400 system display file is created. When GENOPT(*NOGEN) is specified, SYNTAX(*YES) must be specified.

***NOGEN:** A program object is not created. The compiler checks only the syntax of the statements.

***CONVERT:** The SFGR source syntax is checked unless SYNTAX(*NO) is specified, and the SFGR source is converted to data description specifications (DDS). A value should be specified for the TOMBR parameter to save the DDS.

When the to-member name is the same as that of the display file and the to-file is QS36DDSSRC, in the same library as the display file being created, the DDS is not saved in the specified source file member. Specify some other source file, library, or member name.

SYNTAX

Specifies whether the SFGR source syntax is checked.

***YES:** The source syntax is checked.

***NO:** The source syntax is not checked.

Note: If the SFGR source contains errors, the results are unpredictable.

TOFILE

Specifies the qualified name of the source file in which to store the DDS source that is used to create the display file. The to-file need not already exist. If it does not exist, the requester must be authorized to the Create Source Physical File (CRTSRCPF) command so it can be created. TOFILE is ignored if TOMBR(*NONE) is specified.

The name of the to-file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The source file, QDDSSRC, is used.

file-name: Specify the name of the source file in which to store the DDS source. The file should have a record length of 92 bytes.

TOMBR

Specifies the name of the source file member in which to store the data description specifications (DDS) source. If the member does not exist, it is created. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

***NONE:** The DDS source is not stored in the to-file. The source is saved in the Q36DDSSRC source file.

***SRCMBR:** The first name specified by the SRCMBR parameter is used as the member name.

***DSPFILE:** The display file name is used as the member name.

member-name: Specify the name of the source file member in which to store the DDS source. The member name is added if it does not already exist, and is replaced if it already exists. If any terminating errors are encountered, the member is not added or changed.

HALT

Specifies whether an error message is issued when an error is found.

***YES:** The processing stops, and an error message is issued.

***NO:** The processing stops, and a diagnostic message is issued. The request ends, and a return code of 1008 is set.

DFRWRT

Specifies that the writing of data to the display file is delayed until a read request is made. Control is returned to the requesting program immediately after the data is received for output. This may result in improved performance.

***SRCATR:** The defer write option is taken from the DFRWTR attribute of the first source member (see the CHGS36SRCA or EDTS36SRCA commands). If there is

no source member, or if the DFRWTR source attribute has not been set, *YES is assumed.

***YES:** When a write request is made to the display file, control is returned after the buffer is processed. The data may not be displayed immediately; the display of the data may take place later when a read or combined write/read operation is performed. The buffer is then available for the next read or combined write/read operation.

***NO:** After a write operation to the display file, the entire input/output request is completed before returning control to the requesting program. All input/output feedback information is available.

TGTRLS

Specifies the release level of the operating system on which you intend to use the object being created.

The release level is specified in the format VxRxMx, where Vx is the version, Rx is the release, and Mx is the modification level. The release level V2R3M0 is Version 2, Release 3, Modification 0.

Note: To use the object on the target system, you must save the object to the target release level specified on the create command and then restore it on the target system.

If your system is running V2R3M0 and you want to create a program object for distribution to a V2R2M0 system you must:

1. Create the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
2. Save the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
3. Restore the program on the V2R2M0 system. The program object can also be restored to a V2R3M0 or later system.

***CURRENT:** The object is to be used on the release of the operating system currently running on your system. If V2R3M0 is running on the system, *CURRENT means that you intend to use the object on a system with V2R3M0 installed. You also can use the object on a system with any later release of the operating system installed.

***PRV:** The object is to be used on the previous release with modification level 0 of the operating system. If V2R3M0 is running on the system, *PRV means that you intend to use the object on a system with V2R2M0 installed. You also can use the object on a system with any later release of the operating system installed.

release-level: Specify the release level in the format VxRxMx. The object can be used on a system with the specified release or with any later release of the operating system installed.

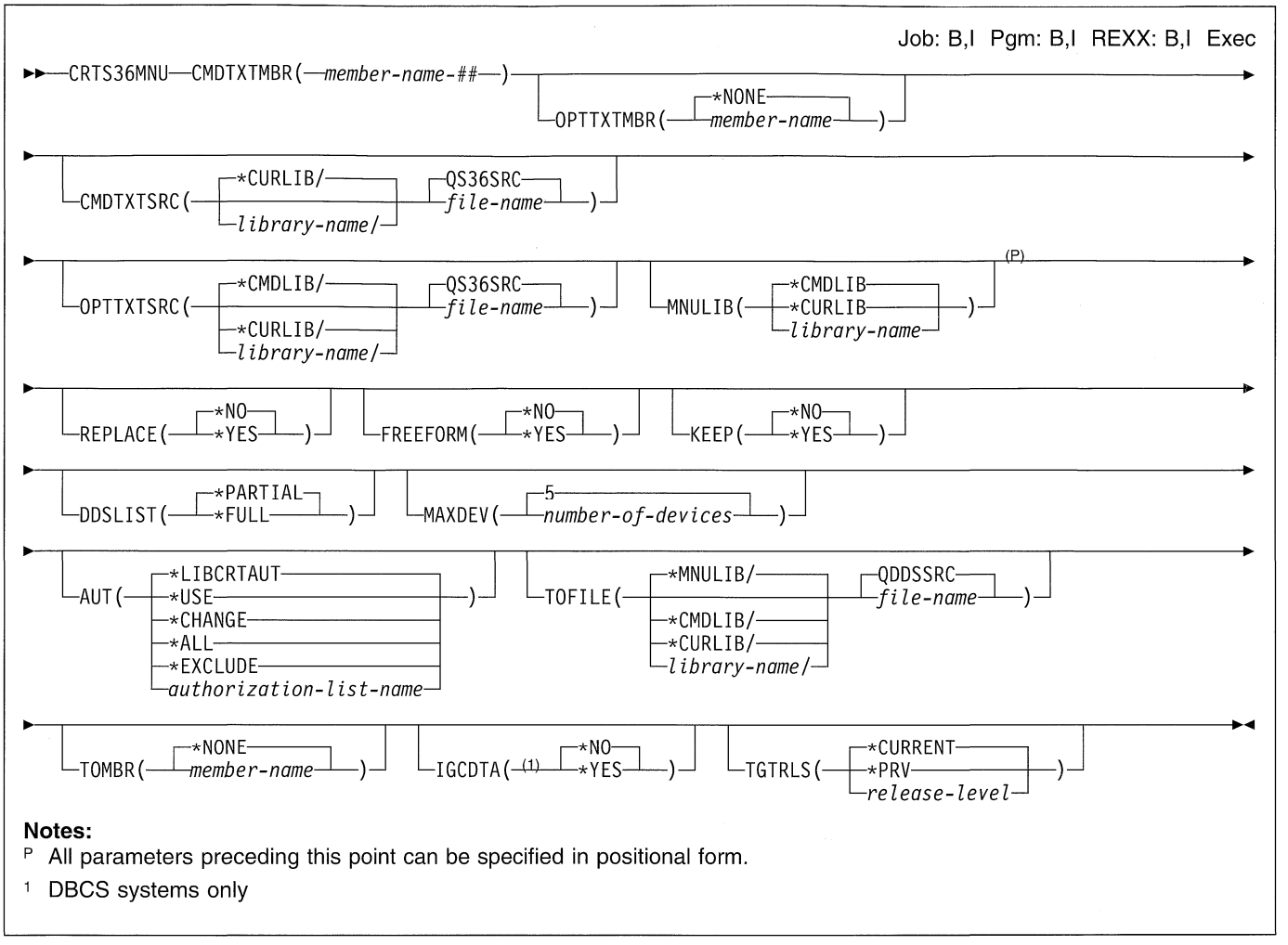
Valid values depend on the current version, release, and modification level, and they change with each new release. Valid values for release V2R3M0 are: V2R3M0, V2R2M0, and V2R1M1.

Example

```
CRTS36DSPF  DSPFILE(MYLIB/MYDISPLAY)  OPTION(*CREATE)
  SRCMBR(SFGRMBR1 SFGRMBR2)  SRCFILE(MYLIB/QS36SRC)
  GENOPT(*NOGEN)
```

This command checks the SFGR source (producing a printer list named QPUTSFGR). No DDS is built, and the display file is not created.

CRTS36MNU (Create System/36 Menu) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 DBCS systems only

Purpose

The Create System/36 Menu (CRTS36MNU) command creates a menu (display file and command message file) from the user's source members. The user can specify that this menu be created either in fixed-format, with options 1 through 24 arranged in two columns, or in free-format.

Restriction: The CRTS36MNU command can be run either in the System/36 environment, or when the QSSP product library is installed, on the AS/400 system.

Required Parameter

CMDTXTMBR

Specifies the source member that contains the System/36 message source used to create the command text message file. The command text message file contains the text for the command that is run when the option is selected for that command. The message IDs must begin with USR. Trailing ## symbols are required by a System/36 on the CMDTXTMBR name. The menu (display file) name is determined by removing the ## symbols.

Note: The message file name specified within the source member must be the same as the source member name.

Optional Parameters

OPTTXTMBR

Specifies the source member used to describe the options on the menu the user is creating.

***NONE:** The option text is taken from the command text message file. If OPTTXTMBR(*NONE) is specified, FREEFORM(*NO) must also be specified.

member-name: Specify the member name of the source to use for the descriptions of the options on the menu the user is creating.

The member may contain either the System/36 SFGR source or the System/36 message source. If the member name is menu-name DT (the name obtained by

removing the ## symbols from the end of the CMDTXTMBR name), the member must contain the System/36 message source. If the member name is the same as the menu name, then the member must contain the System/36 SFGR source. When any other name is specified, the contents of the member is examined to determine the type of source being used. A source member must be specified when FREEFORM(*YES) is specified.

Note: The message file name specified in the source member must be the same as the source member name.

CMDTXTSRC
Specifies the qualified name of the source file in which the CMDTXTMBR is located.

The name of the source file can be qualified by one of the following library values:

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QS36SRC: The source file, QS36SRC, is used.

file-name: Specify the name of the source file that contains the member specified by the CMDTXTMBR parameter.

OPTTXTSRC
Specifies the qualified name of the source file where the OPTTXTMBR is located.

The name of the source file can be qualified by one of the following library values:

***CMDLIB:** The library specified in CMDTXTSRC is used to locate the file.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QS36SRC: The source file, QS36SRC, is used.

file-name: Specify the name of the source file that contains the member specified by the OPTTXTMBR parameter.

MNULIB
Specifies the menu library where both the menu display file and command text message file are created.

The name of the menu can be qualified by one of the following library values:

***CMDLIB:** The menu is created in the command text source (CMDTXTSRC) library.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

REPLACE
Specifies whether an existing display file is replaced. If terminating errors are encountered, the existing display file is not replaced. If the display file already exists, the AUT parameter is ignored, and the authorities for the old display file are copied to the new display file that replaces it. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NO:** No replacement occurs.

***YES:** An existing display file is replaced by the new one being created. Other types of files are not replaced.

Note: A menu is not created if it has the same name and library as an existing program or message file.

FREEFORM
Specifies whether the menu is created in free format or in a fixed format, with two columns each containing twelve items.

***NO:** Free-format is not used. The menu is created using a fixed-format with two-columns. The option text message numbers correspond to the option number. The system supplies the option numbers displayed on the menu.

***YES:** The menu is created using free-format. OPTTXTMBR is required when FREEFORM(*YES) is specified. Message numbers correspond to the row numbers on the screen. FREEFORM(*YES) is required when the OPTTXTMBR contains the screen file generator (SFGR) source. The user's OPTTXTMBR source must supply the option numbers shown on the menu.

KEEP
Specifies whether the option text message file is kept when the compilation is complete.

***NO:** The option text message file is not kept.

***YES:** The option text message file is kept. If the option text source member contains the SFGR source and has the same name as the menu, then KEEP(*YES) is required.

DDSLIST
Specifies whether a full or partial DDS compiler listing is provided.

***PARTIAL:** A partial compiler listing is provided.

***FULL:** A full DDS compiler listing and cross-reference are provided.

MAXDEV
Specifies the maximum number of devices that can use the menu at one time.

5: The maximum number of devices is five.

number-of-devices: Specify the maximum number of devices that can use the menu at one time. Possible values range from 1 through 256.

AUT

Specifies the authority given to users who do not have specific authority to the System/36 menu, who are not on an authorization list, and whose user group has no specific authority to the System/36 menu.

***LIBCRTAUT:** The public authority for the System/36 menu is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the System/36 menu). The public authority is determined when the System/36 menu is created. If the CRTAUT value for the library changes after the System/36 menu is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the System/36 menu, such as running a program or reading a file. The user cannot change the System/36 menu. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the System/36 menu.

***EXCLUDE:** The user cannot access the System/36 menu.

authorization-list-name: Specify the name of the authorization list used.

TOFILE

Specifies the qualified name of the source file (to-file) in which to store the data description specifications (DDS) source that was used to create the object. If the file does not exist, the requester must be authorized to the Create Source Physical File (CRTSRCPF) command so it can be created. This parameter is ignored if TOMBR(*NONE) is specified.

Note: See special case for TOMBR when QS36DDSSRC is specified.

The name of the to-file can be qualified by one of the following library values:

***MNULIB:** The menu library is used to locate the to-file.

***CMDLIB:** The command library is used to locate the to-file.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDDSSRC: The source file, QDDSSRC, is used.

file-name: Specify the name of the source file in which to store the DDS source. The file should have a record length of 92 bytes.

TOMBR

Specifies the name of the source file member in which to store the data description specifications (DDS) source. If the member does not exist, it is created. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

***NONE:** The DDS source is not stored in the to-file.

member-name: Specify the name of the source file member in which to store the DDS source. The member is added if it does not already exist, and replaced if it already exists.

IGCDTA

Specifies whether the file contains double-byte character set (DBCS) data.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

TGTRLS

Specifies the release level of the operating system on which you intend to use the object being created.

The release level is specified in the format VxRxMx, where Vx is the version, Rx is the release, and Mx is the modification level. The release level V2R3M0 is Version 2, Release 3, Modification 0.

Note: To use the object on the target system, you must save the object to the target release level specified on the create command and then restore it on the target system.

If your system is running V2R3M0 and you want to create a program object for distribution to a V2R2M0 system you must:

1. Create the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
2. Save the program with TGTRLS(V2R2M0) or TGTRLS(*PRV).
3. Restore the program on the V2R2M0 system. The program object can also be restored to a V2R3M0 or later system.

***CURRENT:** The object is to be used on the release of the operating system currently running on your system. If V2R3M0 is running on the system, *CURRENT means that you intend to use the object on a system with V2R3M0 installed. You also can use the object on a system with any later release of the operating system installed.

***PRV:** The object is to be used on the previous release with modification level 0 of the operating system. If V2R3M0 is running on the system, *PRV means that you intend to use the object on a system with V2R2M0 installed. You also can use the object on a system with any later release of the operating system installed.

release-level: Specify the release level in the format VxRxMx. The object can be used on a system with the specified release or with any later release of the operating system installed.

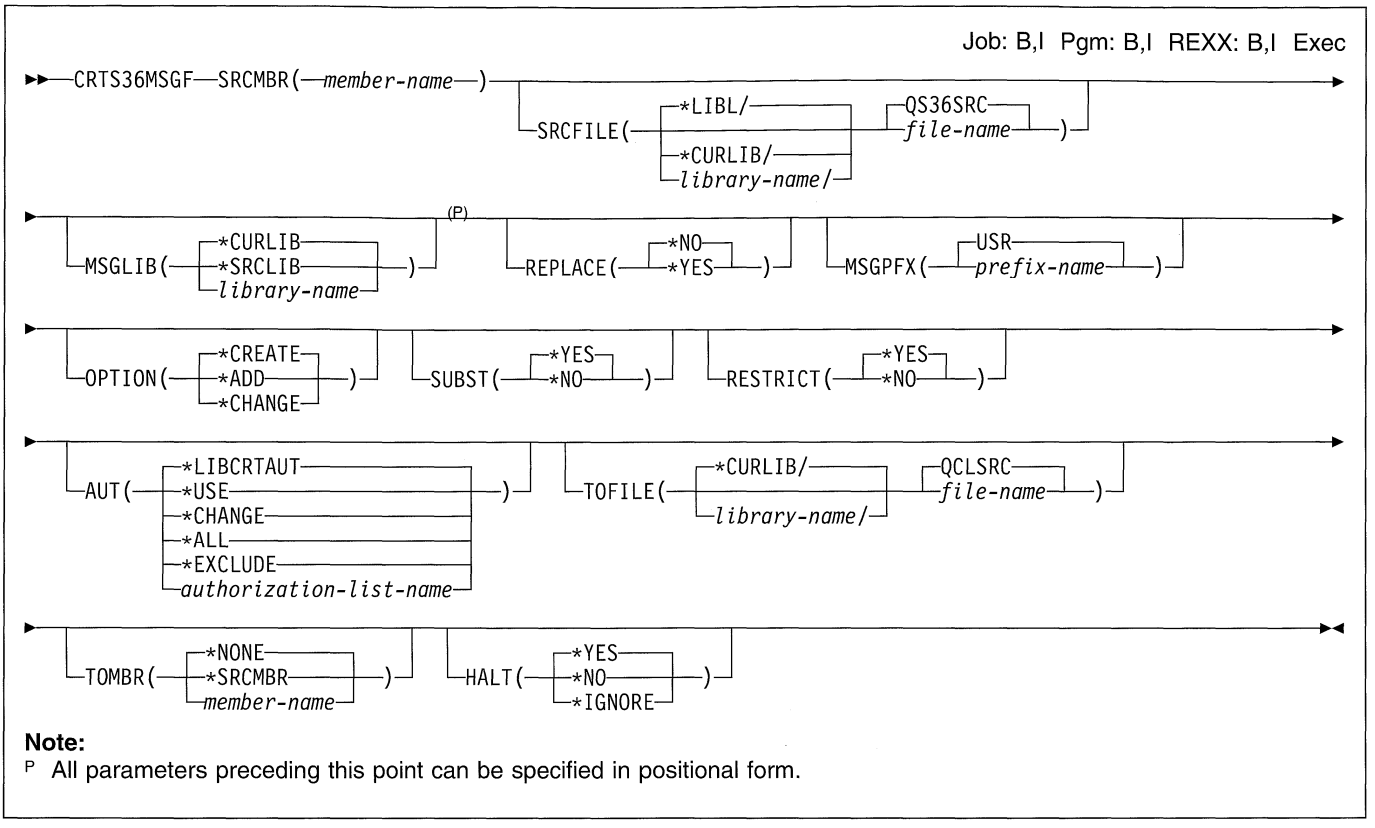
Valid values depend on the current version, release, and modification level, and they change with each new release. Valid values for release V2R3M0 are: V2R3M0, V2R2M0, and V2R1M1.

Example

```
CRTS36MNU  CMDTXTMBR(MENU##)  OPTTXTMBR(SFGRMBR)
  CMDTXTSRC(MYLIB/QS36SRC)  OPTTXTSRC(MYLIB/QS36SRC)
  MNULIB(WORKLIB)  REPLACE(*YES)  FREEFORM(*YES)
  KEEP(*YES)
```

This command creates a free format menu display from the specified SFGR source member, placing the resulting menu into library WORKLIB. Since the SFGR source is being used, FREEFORM(*YES) and KEEP(*YES) are required. The layout of the menu display is controlled by the format definition located in the SFGR source member. No special checking is done to ensure that the display file defined by this source is valid when used by the system as a menu.

CRTS36MSGF (Create System/36 Message File) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Create System/36 Message File (CRTS36MSGF) command creates a message file from the user's System/36 message source member. Using the TOFILE and TOMBR parameters, the user can convert the System/36 message source to an AS/400 system message source.

Restrictions: When the QSSP product library is installed, the CRTS36MSGF command can be run either in the System/36 environment or on the AS/400 system.

System/36 message source consists of 3 types of statements:

1. The message control statement must be the first record in the source. Only one control statement is allowed. The control statement has the following syntax:

   name<,level> <comment>

   where: name is the name of the message file being created or changed. The name must start in column 1 of the source record. Level must be a 1 or 2 or a blank, and is separated from the name by a comma. A level 1 indicates that the source is for message text (first-level messages); a level 2 indicates that the source is for online help information (second-level messages). If level is omitted or blank, a first level message is assumed. Anything after the first blank in the control statement is considered to be a comment and is ignored.

2. Comment records that start with an asterisk (*) in column 1. All comment records are ignored.

3. One or more message text statements may be defined. The message text statement has the following syntax:

   mmmm text

   where: mmmm is the System/36 Message Identification Code (MIC) and must start in column 1 and consist of 4 digits (0-9). All 4 digits are required. The 7-character message identifier is created by adding the message prefix specified by the MSGPFX parameter in front of the MIC. Column 5 is ignored and should be left blank. Text is the text of the message being defined and starts in column 6. The message text statements must be arranged so that the MICs are not in descending order. If there is insufficient room to define the entire message in a single record, the MIC can be repeated in column 1 of the next record and the text can be continued starting in column 6.

   A first level message is restricted to a maximum of 75 characters, and a second level message is restricted to a maximum of 225 characters unless RESTRICT(*NO) is specified. Trailing blanks on the last record for each MIC are not counted. The record length used when processing the SRCMBR is the RCDLEN source attribute for that member. This attribute defaults to the source file record length, but can be set or changed when using the CHGS36SRCA, EDTS36SRCA, or RSTS36LIBM com-

mands. The RCDLEN source attribute is the member's logical record length. If the RCDLEN source attribute is less than the record length of the source file, all characters after the logical record length are ignored. If any record contains any non-blank characters after the logical record length, a diagnostic message is issued as a warning.

When SUBST(*YES) is specified on this command, certain strings of # symbols within the message text are treated as substitution fields. A # symbol string (a series of 1 or more characters defined by code point '7B'X) qualifies as a substitution field only if it is preceded and followed by one of the following delimiter characters:

Char	Hex	Description
	40	Blank
	41	Blank
\	E0*	Backslash
<	4C	Less Than
(	4D	Left Parenthesis
+	4E	Plus
|	4F*	Vertical Bar
!	4F*	Exclamation
&	50	Ampersand
*	5C	Asterisk
)	5D	Right Parenthesis
;	5E	Semicolon
¬	5F*	Not
^	5F*	Caret
-	60	Minus
/	61	Slash
,	6B	Comma
%	6C	Percent
>	6E	Greater Than
?	6F	Question
:	7A	Colon
'	7D	Quote
=	7E	Equals
"	7F	Double Quote

Note: Some hex code points X'4F' and X'5F' are assigned different graphics. The comparison is for the HEX value, not the graphic entered. The backslash ('E0'X) is not a CL delimiter, but is also not allowed in a name. No character below '40'X is allowed in a name.

The delimiter that precedes the # symbol string need not be the same as the delimiter that follows the # symbol string. If a # symbol string occurs at the beginning or end of the message text, it is treated as if it were preceded or followed by a blank, respectively.

Double-Byte Character Set Considerations:

If any double-byte character set (DBCS) text is contained in a message, the first MIC for the message must be preceded by a record containing an MIC of A000. This special MIC resets the current MIC counter so that the non-descending MIC requirement is temporarily suspended, allowing the user to define DBCS versions of previously defined MICs within the same message file. The special A000 MIC also alters the third character of the MSGPFX, changing it to a 'Z'. If DBCS messages will be defined, the user should not specify a MSGPFX that uses 'Z' for the third character or duplicate message ID errors may result.

Required Parameter

SRCMBR

Specifies the source member that is used to create the message file. The first record that is not a comment record in the source member specifies the name of the message file being created or changed. The first record that is not a comment record also specifies whether the text in this source member is for the first-level or second-level message text.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file used to create the message file.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QS36SRC: The source file named QS36SRC is used.

file-name: Specify the name of the source file that contains the member specified by the SRCMBR parameter.

MSGLIB

Specifies the qualified name of the library used to receive the message file that is created.

The name of the message file can be qualified by one of the following library values:

***CURLIB:** The message file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

***SRCLIB:** The source library is used to locate the message file.

library-name: Specify the name of the library where the message file is created.

REPLACE

Specifies whether an existing message file is replaced. This parameter is ignored if OPTION(*CREATE) is not specified. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NO:** No replacement occurs.

***YES:** If a message file already exists, it is replaced by the one being created.

Note: A message file is not created if the message file has the same name and library as an existing program, display file, or other type of file.

MSGPFX

Specifies the message prefix used to create messages. The default prefix of USR must be used if the message is used to build a menu.

USR: The message prefix of USR is used.

prefix-name: Specify the three-character message prefix used in message IDs.

OPTION

Specifies what to do with the specified message file. A message file is created, or messages are added or changed, in existing message files.

***CREATE:** A message file is created by using the message file name specified in the source.

***ADD:** The system adds the new records to the end of the existing records.

***CHANGE:** Messages are updated in the specified message file. If a given message does not exist, it is added.

SUBST

Specifies that System/36 text replacement fields are converted into the notation that represents text replacement fields in AS/400 system message files. Although *YES is the default, not everyone uses text replacement fields and *NO is a way to make sure that a field is not converted by mistake. Replacement fields are represented by a pair of trailing ## symbols in the S/36 source.

***YES:** The text replacement fields are converted.

***NO:** The text replacement fields are not converted.

RESTRICT

Specifies that all System/36 restrictions on message text length are enforced.

***YES:** System/36 restrictions are enforced on the message text length. First-level message text is limited to 75 characters. Second-level message text is limited to 225 characters.

***NO:** System/36 restrictions on the message text length are not enforced.

AUT

Specifies the authority given to users who do not have specific authority to the System/36 message file, who are not on an authorization list, and whose user group has no specific authority to the System/36 message file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the System/36 message file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the System/36 message file). The public authority is determined when the System/36 message file is created. If the CRTAUT value for the library changes after the System/36 message file is created, the new value does not affect any existing objects.

***USE:** The user can perform basic operations on the System/36 message file, such as running a program or reading a file. The user cannot change the System/36 message file. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the System/36 message file.

***EXCLUDE:** The user cannot access the System/36 message file.

authorization-list-name: Specify the name of the authorization list used.

TOFILE

Specifies the qualified name of the source file (to-file) in which to store the data description specifications (DDS) source that was used to create the object. If the file does not exist, the requester must be authorized to the Create Source Physical File (CRTSRCPF) command so it can be created. This parameter is ignored if TOMBR(*NONE) is specified.

The name of the to-file can be qualified by one of the following library values:

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCLSRC: The source file QCLSRC is used.

file-name: Specify the source file name under which to store the source file.

TOMBR

Specifies the name of the source file member in which to store the data description specifications (DDS) source. If the member does not exist, it is created. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

***NONE:** The CL source is not stored in the to-file.

***SRCMBR:** If no member name is specified, the member name specified in the SRCMBR parameter is used.

member-name: Specify the name of the source file member in which to receive the CL source file.

HALT

Specifies whether the processing stops when an error is found in the source.

***YES:** The processing stops and an error message is sent.

***NO:** The request ends and a return code of 2034 is set. A diagnostic message is sent.

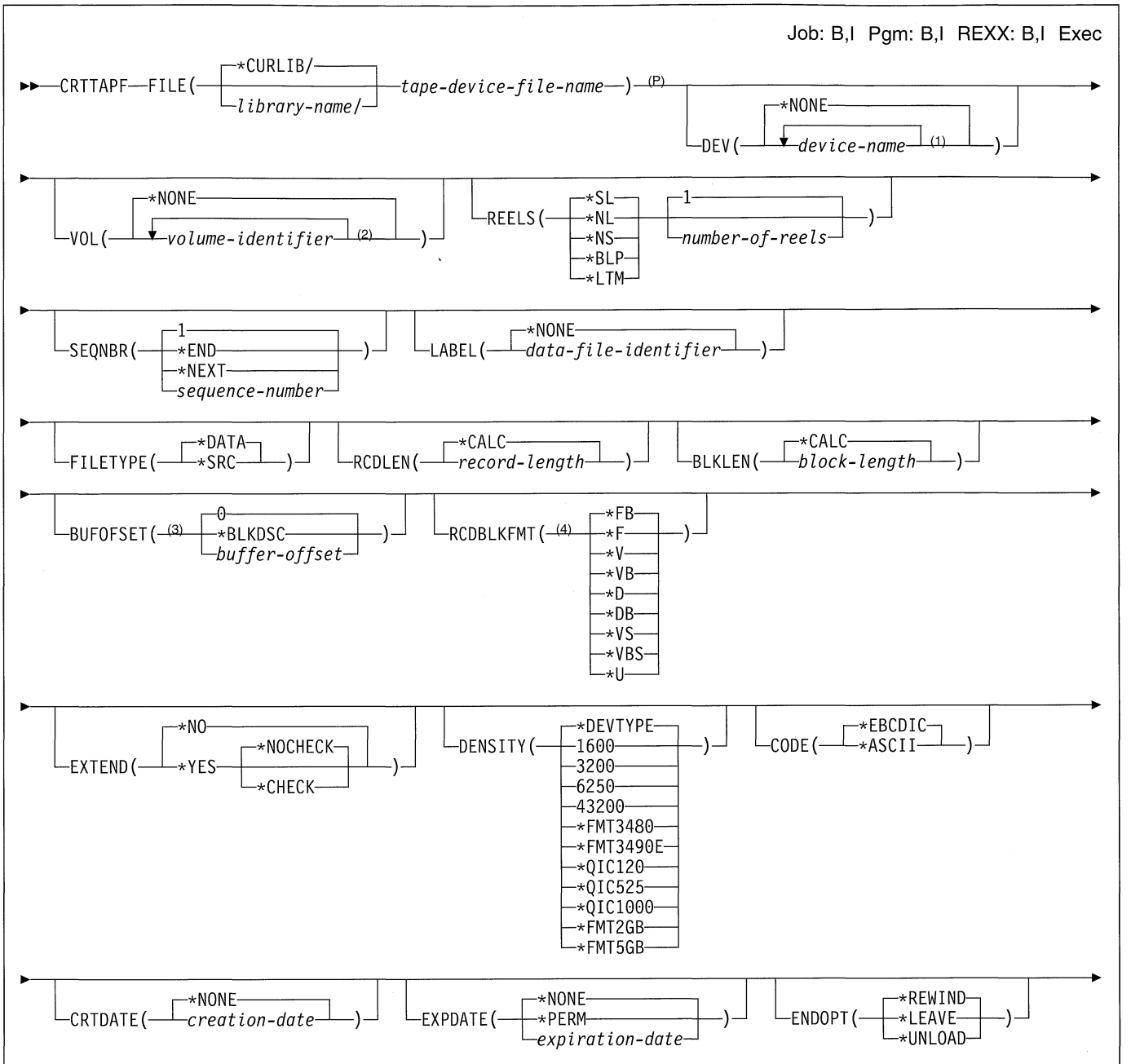
***IGNORE:** A diagnostic message is sent and the current source statement is ignored; processing continues with the next source statement.

Example

```
CRTS36MSGF  SRCMBR(MYMSGSRC)  SRCFILE(MYLIB/QS36SRC)
  MSGLIB(WORKLIB)  REPLACE(*YES)
```

This command creates a message file named MYMSGSRC in library WORKLIB. If a message file with the same name already exists, this new message file replaces the existing one.

CRTTAPF (Create Tape File) Command



Notes:

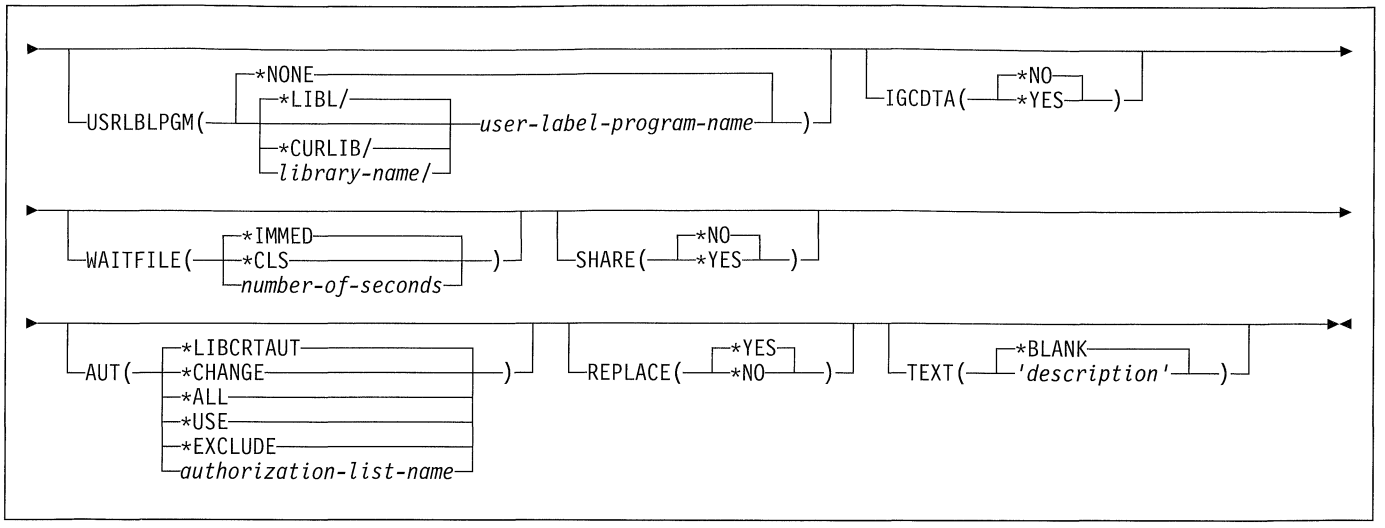
P All parameters preceding this point can be specified in positional form.

1 A maximum of 4 repetitions

2 A maximum of 50 repetitions

3 The value *BLKDSC is valid only for a file with record block format *D or *DB.

4 The values *F, *FB, *BS, *VBS, and *U are valid for both EBCDIC and ASCII codes; *V and *VB are valid only for EBCDIC; *D and *DB are valid only for ASCII.



Purpose

The Create Tape File (CRTTAPF) command creates a tape device file. The device file contains the file description, which identifies the device to be used; it does not contain data. The tape device file is used to read and write records on tape. The same device file can be used for both input and output operations.

Note: This command is not used to create device files for use in save and/or restore operations. User-created device files are not needed for save and/or restore operations.

Tape files have no data description specifications (DDS). The information in the tape file description comes from the command that creates it. The tape file has one record format for input/output operations. The record format consists of one character field containing the input data retrieved from the device or the output data to be written to the device. The program using the device file must describe the fields in the record format so the program can arrange the data received from or sent to the device in the manner specified by the tape file description.

Restriction: Non-labeled tapes cannot be duplicated to 1/4 inch or 8mm cartridge devices.

Required Parameter

FILE

Specifies the qualified name of the file being created. If the file is used by a high-level language (HLL) program, the file name must be consistent with the naming rules of that language; otherwise, the file must be renamed in the program.

The name of the tape device file can be qualified by one of the following library values:

***CURLIB:** The tape device file is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the tape device file is created.

tape-device-file-name: Specify the name of the tape device file being created.

Optional Parameters

DEV

Specifies the names of one or more tape devices used with this tape device file to perform input/output data operations.

***NONE:** No device name is specified. The name of the tape device must be specified later in a CHGDSPF or OVRDSPF command, or in the HLL program that opens the file.

device-name: Specify the names of one or more devices (no more than four) used with this tape device file. The order in which the device names are specified is the order in which tapes on the devices are processed. When more volumes are processed than the number of devices in the DEV list, the devices are used in the same order specified, wrapping around to the first device as needed. Each device name must be known on the system by a device description before this device file is created.

VOL

Specifies one or more volume identifiers used by the file. The volumes must be installed in the same order as the identifiers are specified here (and as they are specified on the DEV parameter). If the file is opened for read backward, then the volume identifiers in the list are processed from last to first (while the devices in the device list are used in first-to-last order). If a list of volume identifiers is provided for the file, operator messages indicate the name of the required volume. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NONE:** No tape volume identifiers are specified for this file. They can be supplied before the device file is opened, either in a CHGTAPF or OVRTAPF command or in the high-level language program. If volume identifiers are not specified before the device file is opened, volume checking is not performed beyond verifying that the correct label type volume is on the device, and volume names are not provided in operator messages. The maximum number of reels processed for an *NL, *NS, *BLP, or *LTM input file when VOL(*NONE) is specified is determined by the REELS(number-of-reels) parameter value.

volume-identifier: Specify the identifiers of one or more volumes in the order in which they are placed on the device. Each volume identifier contains a maximum of 6 alphanumeric characters. Use a blank as a separator character when listing multiple identifiers. Up to 50 volume identifiers can be specified. These identifiers are used in messages sent to the operator during processing. The maximum number of reels processed for an *NL, *NS, *BLP, or *LTM input file is determined by the number of volume identifiers in the list.

Note: If the VOL parameter value used for the file specifies a list of identifiers rather than VOL(*NONE), the number-of-reels part of the REELS parameter is ignored regardless of where it is specified. A description of how the parameter values for the file are determined when overrides are used, the high-level language interface, and the device file when the file is opened is in the *Data Management Guide*. To ensure that the number-of-reels part of the REELS parameter is used (rather than a VOL identifier list) to control the volumes processed by the tape device file, specify VOL(*NONE) in the same command in which the REELS parameter is specified.

REELS

Specifies the type of labeling used on the tape reels and the maximum number of reels processed if both a list of volume identifiers is not specified (VOL parameter) and this device file is used with either *NL, *NS, *LTM, or *BLP input files. When the number of reels is specified as the second element of this parameter, the volume identifiers on the volumes are ignored if labeled tapes are being processed; instead, the order in which the reels are installed on the device must be checked by the operator.

The number-of-reels value is not a limiting value for standard-label or output files. For a standard-label *input* file, the data file labels limit the number of volumes processed by indicating end-of-file. For an *output* file, the number-of-reels value is ignored; the system requests that additional volumes be kept on the device until the file is closed.

The system checks the first record following the load point on the tape to see (1) whether it has exactly 80 bytes for EBCDIC or at least 80 bytes for ASCII and (2) whether the first 4 bytes contain the values VOL and 1. If so, the reel contains a standard-label tape. *SL and *BLP files require standard-label tape volumes. *NL, *NS, and *LTM tape files cannot process standard-label volumes.

Note: The values *SL, *NL, and *LTM can be specified if the device file is used for either reading or writing on tapes. The values *NS and *BLP are valid only if the device file is used to read tapes.

Element 1: Type of Labels

***SL:** The volumes have standard labels. If a list of volume identifiers is specified (with the VOL parameter), the system checks that the correct tape volumes are on the device in the specified sequence.

- If no volume identifier list is given and the file is opened for *output*, any standard-label volumes may be installed on the device.
- If no volume identifier list is given and the file is opened for *input*, the first volume may have any volume identifier, but if the file is continued, the system requires the correct continuation volumes to be processed (verified by checking the data file labels). For an input file, the end-of-file message is sent to the program being used when the labels on the last volume processed indicate that it is the last volume for the data file.

***NL:** The volumes are not labeled. On a nonlabeled volume, tape marks are used to indicate the end of each data file and the end of the volume. For an *input* file, the end-of-file message is sent to the program when the number of volumes specified in the volume list have been processed, or, if no list of volume identifiers is provided, when the number of reels specified in the REELS parameter are processed.

***NS:** The volumes have nonstandard labels. Each volume must start with some kind of label information, optionally preceded by a tape marker and always followed by a tape marker. This nonstandard label information is ignored. The system spaces forward to a point beyond the tape marker that follows the nonstandard labels and positions the tape at the file's data. Each reel must have a tape marker at the end of the file's data. Information beyond this ending tape marker is ignored. Only a single data file can exist on a nonstandard tape. Standard-label volumes *cannot* be processed by using the *NS label processing.

For an *input* file, the end-of-file message is sent to the program using the file when the number of volumes specified in the volume list have been processed, or, if no list of volume identifiers is provided, when the number of reels specified in the REELS parameter are processed.

***BLP:** Standard-label processing is bypassed. Each reel *must* have standard labels. Although each reel is checked for a standard volume label and each file must have at least one standard header label (HDR1) and one

standard trailer label (EOV1 or EOF1), most other label information (such as the data file record length or block length) is ignored. The sequence number of each file on the volume is determined only by the number of tape markers between it and the start of tape (in contrast to *SL processing in which the file sequence number stored in the header and trailer labels of each file are used to locate a data file).

Most of the information in the data file trailer label is ignored, but if an end-of-file (EOF) trailer label is found, the end-of-file message is sent to the program using the tape file. If no end-of-file trailer label is encountered by the time the specified number of volumes or reels have been processed (volume identifier list and REELS parameter), the end-of-file message is immediately sent to the program using the tape file. Bypass label processing can be used when the user does not know the name of the file used or when some file label information is incorrect.

***LTM:** The volumes have no labels but do have a single leading tape marker before the first data file. REELS(*LTM) is processed the same as REELS(*NL) except that when SEQNBR(1) is specified for an output file to create the first data file on the tape, a leading tape marker is written at the start of the tape before the first data block.

Element 2: Number of Reels

1: Only one tape reel is processed for the *NL, *LTM, *NS, or *BLP tape file input operation if no list of volume identifiers is provided (VOL parameter).

number-of-reels: Specify the maximum number of reels to be processed for an *NL, *LTM, *NS, or *BLP input tape operation when a list of volume identifiers is not specified (VOL parameter). If the next reel is not on the device when the end of the currently-processing tape is reached, a message is sent to the operator requesting that the next tape be installed on the next tape device. The number-of-reels value is ignored for a standard-label (*SL) file or for any output file.

SEQNBR

Specifies the sequence number of the data file on the tape being processed.

- When standard-label tapes are used, the four-position file sequence number is read from the first header label of the data file.
- When bypass label processing is used or when standard-label tapes are not used, the system counts the tape markers from the start of the tape to locate the correct sequence number data file to be processed.
- When multiple-file, multiple-volume tapes are processed using REELS(*SL), the file sequence numbers continue consecutively through the volumes; thus, each new data file has a sequence number one greater than the previous file, regardless of its volume location.

1: For standard-label tapes (not using bypass label processing), the data file having the sequence number 1 is processed. For nonlabeled tapes and for bypass label processing of standard-label tapes, the first data file on the tape is processed.

***END:** The file is written on the end of the tape. This value is used only for files that are written to tape.

An error message is shown on the display when a tape device file is used to read from a tape and the *END special value is specified in the tape device file.

***NEXT:** The next file in the sequence is processed. This value is used for files read from tape. If the tape is currently in a position that is prior to the first file, the first file on the tape is processed.

An error message is shown on the display when a tape file is used to write to a tape and the *NEXT special value is specified in the tape file.

sequence-number: Specify the sequence number of the file.

LABEL

Specifies the data file identifier of the data file processed by this tape device file. An identifier is defined only for standard-label tapes and is stored in the header label immediately before the data file.

If a data file identifier is specified for any type of label processing other than *SL, it is ignored.

An identifier is required for a standard label output file, but is optional for an input file because the sequence number uniquely identifies the data file to process.

For an input file or output file with EXTEND(*YES) specified, this parameter specifies the identifier of the data file on the tape. The specified identifier must match the one in the labels of the data file that the SEQNBR parameter specifies; otherwise, an error message is sent to the program using this device file. For output files with EXTEND(*NO) specified, this parameter specifies the identifier of the data file to be created on the tape. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NONE:** The data file identifier is not specified.

data-file-identifier: Specify the identifier (17 alphanumeric characters maximum) of the data file used with this tape device file. If this identifier is for a tape written in the basic exchange format, and is used on a system other than an AS/400 system, up to eight characters or a qualified identifier having no more than eight characters per qualifier must be used.

FILETYPE

Specifies whether the tape device file being created describes data records or describes source records (statements) for a program or another file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***DATA:** The tape file describes data records.

***SRC:** The tape file describes source records.

Note: If *SRC is specified, the system adds 12 bytes to the start of every record (to replace the sequence number and date fields).

RCDLEN

Specifies, in bytes, the length of the records contained in the data file processed with this device file. The system always uses the record length and block length specified in the data file labels for any standard-label input file or output file with EXTEND(*YES) specified (if a second header label (HDR2) is found on the tape and *BLP label processing has not been specified).

***CALC:** No record length is specified for the data file being processed. If *CALC is specified, the system will attempt to calculate an appropriate record length when the file is opened. RCDLEN(*CALC) can be used for nonlabeled tapes or when there is no HDR2 label if a BLKLEN value other than *CALC is specified for the file and RCDBLKFMT does not specify spanned or blocked records. In this case, the system calculates an appropriate record length from the block length, record block format, and buffer offset (for an ASCII file) specified for the file. In any other case, the actual record length must be specified by a CHGTAPF command or OVRTAPF command, or in the high-level language program that opens the device file.

record-length: Specify a value ranging from 1 through 32767 bytes that indicates the length of each record in the data file. The minimum and maximum record length allowed for a file is dependent on the record block format, block length, buffer offset (for an ASCII file), and recording code. The *Record Length Values* table (at the end of this command description) shows the minimum and maximum record length values allowed for each record block format, assuming the block length value is large enough to support the maximum record length.

BLKLEN

Specifies, in bytes, the maximum length of the data blocks transferred to or from the tape for output or input operations. The system uses the block length and record length specified in the data file labels for any standard-label input file or output file with EXTEND(*YES) specified (if a second header label (HDR2) is found on the tape and *BLP label processing has not been specified).

***CALC:** No block length is specified for the data file to be processed. If *CALC is specified, the system attempts to calculate an appropriate block length when the file is opened. BLKLEN(*CALC) can be used for nonlabeled tapes or when there is no HDR2 label if a RCDLEN value other than *CALC is specified for the file and RCDBLKFMT does not specify spanned or blocked records. In this case, the system calculates an appropriate block length from the record length, record block format, and buffer offset (for an ASCII file) specified for the file. In any other case, the actual block length must be specified by a CHGTAPF command or OVRTAPF command, or in the high-level language program that opens the device file.

block-length: Specify a value, not exceeding 32767 bytes, that specifies the maximum length of each block in the data file to be processed. The minimum block length that can be successfully processed is determined by the tape device hardware and AS/400 system machine support functions.

- The maximum block length is always 32767 bytes for an input file, but is limited to 9999 bytes if block descriptors must be created for an ASCII output file.
- The following table shows the minimum and maximum block length values allowed for an output file:

Table 38. Absolute Block Length (BLKLEN) Ranges

CODE	BUFOFSET	Minimum BLKLEN	Maximum BLKLEN
*EBCDIC	Ignored	18	32767
*ASCII	0	18	32767
*ASCII	*BLKDSC	18	9999

BUFOFSET

Specifies the buffer offset value for the start of the first record in each block in the tape data file. A buffer offset value can be used for any record block format ASCII file, and is ignored for an EBCDIC tape file. The system uses the buffer offset specified in the data file labels for any standard-label input file or output file with EXTEND(*YES) specified if a value is contained in the second header label (HDR2) on the tape, and *BLP label processing has not been specified.

The buffer offset parameter specifies the length of any information that precedes the first record in the block. For record block formats *D, *DB, *VS, and *VBS, each record or record segment is preceded by a descriptor that contains the length of the record or segment. A buffer offset value is used to indicate that there is information *ahead* of the descriptor word for the first record in each block, or *ahead* of the data of the first fixed-length record or undefined format record in each block.

This parameter is not needed for a standard-label file processed for input if the tape includes a second file header label (HDR2) that contains the buffer offset value. A buffer offset value must be provided by the Create Tape File (CRTTAPF) command, Change Tape File (CHGTAPF) command, or Override Tape File (OVRTAPF) command, or by the file labels for an input file that contains any information (such as a block descriptor) ahead of the first record in each block. If the user does not specify a buffer offset value when a tape file is created, it is not necessary to specify an offset value when the file is read.

The only buffer offset values allowed for an output file are zero and *BLKDSC. An existing standard-label data file with a buffer offset value in the HDR2 label can be extended only if the buffer offset value is either 0 or 4. A buffer offset value of 0 in the HDR2 label adds data blocks with *no* buffer offset. BUFOFSET(*BLKDSC) must be specified to extend an existing tape data file that contains an offset value of 4 in the HDR2 label.

0: No buffer offset value precedes the first record in each data block.

***BLKDSC:** Creates 4-byte block descriptors in any tape file created by using this device file. Any input file read by using this device file should assume 4 bytes of buffer offset information preceding the first record in each data block. This value is valid only for a record block format *D or *DB file. The contents of the buffer offset information of each output data block when BUFOFSET(*BLKDSC) is specified is the actual length of the data block, expressed in zoned decimal format.

buffer-offset: Specify a value ranging from 0 through 99 bytes that specifies the length of the buffer offset information that precedes the first record in each data block.

RCDBLKFMT

Specifies the type and blocking attribute of records in the tape data file being processed.

Record block format *V and *VB records can be processed only for an EBCDIC file; *D and *DB records can be processed only for an ASCII file. If a standard-label tape (label type *SL or *BLP) is being processed and an inconsistent record block format is specified for the volume code, the correct record type is assumed (V or D) for the volume code and a warning message is sent to the program that opens the file. If the record type and code are inconsistent for a nonlabeled volume (label type *NL, *LTM, or *NS), an error message is sent and the file is *not* opened, because there are no labels to verify the correct volume code.

If a valid record length, block length, and buffer offset value (for an ASCII file) are specified for fixed-length records but the block attribute is incorrect, the correct block attribute is assumed (changing record block format *F to *FB or record block format *FB to *F), and a warning message is sent to the program that opens the file.

If a block length is specified that is longer than required to process a maximum length record, then record block format *V, *D, or *VS is changed to *VB, *DB, or *VBS and a warning message is sent to the program that opens the file.

The *Required RCDLEN/BLKLEN/BUFOFSET Relation* table, at the end of this command description, shows the required relationship between the record length, block length, and buffer offset (for ASCII) file parameters for an output file or an input file where the file parameters are not determined from a second file header label (HDR2).

Note: When BUFOFSET(*BLKDSC) is specified for the file, a value of 4 should be used for the BUFOFSET part of any BLKLEN calculations, unless existing file labels on the tape specify a different value.

***FB:** Fixed-length, blocked, unspanned records in either EBCDIC or ASCII code are processed. The system may change this record block format to *F, based on other file parameters.

***F:** Fixed-length, unblocked, unspanned records in either EBCDIC or ASCII code are processed. The system may change this record block format to *FB, based on other file parameters.

***V:** Variable-length, unblocked, unspanned records in EBCDIC type V format are processed. The system may change this record block format to *VB, *D, or *DB, based on other file parameters.

***VB:** Variable-length, blocked, unspanned records in EBCDIC type V format are processed. The system may change this record block format to *DB, based on the volume code.

***D:** Variable-length, unblocked, unspanned records in ASCII type D format are processed. The system may change this record block format to *DB, *V, or *VB, based on other file parameters.

***DB:** Variable-length, blocked, unspanned records in ASCII type D format are processed. The system may change this record block format to *VB, based on the volume code.

***VS:** Variable-length, unblocked, spanned records in either EBCDIC or ASCII code are processed. The system may change this record block format to *VBS, based on other file parameters. Note that the representation of spanned records on the tape is different for EBCDIC and ASCII files, but the system selects the correct format based on the file code.

***VBS:** Variable-length, blocked, spanned records in either EBCDIC or ASCII code are processed. Note that the representation of spanned records on the tape differs for EBCDIC and ASCII files, but the system selects the correct format based on the file code.

***U:** Undefined format records in either EBCDIC or ASCII code are processed. RCDBLKFMT(*U) records are processed as variable-length records, and each record written or read is in a separate tape block. This format can be useful for processing tape files that do not have the formatting requirements of any other record block format.

EXTEND

Specifies, for output operations to tape, whether new records are added to the end of a data file currently on the tape. The specific data file is identified by the SEQNBR parameter and, for a standard-label file, the LABEL parameter. If the data file is extended, it

becomes the last file on the tape volume; data files that follow it are overwritten as the specified file is extended.

Note: This parameter is not valid for 1/4-inch cartridge tape devices.

Element 1: Adding Records to Data File

***NO:** Records are not added to the end of the specified data file. If there is already a data file with the specified SEQNBR on the tape, a new data file is created by overwriting the existing data file and any files that follow it.

***YES:** New records are added to the end of the specified data file on tape when this device file is used.

Element 2: Checking Active Files

***NOCHECK:** The file is extended without being checked to see whether the file is active.

***CHECK:** Before the file is extended, the file is checked to see whether it is active.

DENSITY

Specifies, in bits per inch, the density of the data that is written on the tape volume when this device file is created. This parameter is used only for tape files being written to tape; it is ignored for tape files being read from the tape (in the case of files being read from tape, the density on the tape is used).

The density of a standard-label volume is specified on the INZTAP command, which initializes tapes as standard-label volumes by writing volume labels on them. If the density specified on this parameter is different than the density of a standard-labeled tape, the density specified on this parameter is used, and a warning message is sent.

***DEVTYPE:** The density is based on the kind of tape device being used.

1600: The data density on this tape volume is 1,600 bits per inch.

3200: The data density on this tape volume is 3,200 bits per inch.

6250: The data density on this tape volume is 6,250 bits per inch.

43200: The data density on this tape volume is 43,200 bits per inch.

***FMT3480:** The data density on this tape volume is formatted to support a 3480 device.

***FMT3490E:** The data density on this tape volume is formatted to support a 3490 device.

***QIC120:** The format of this tape is *QIC120.

***QIC525:** The format of this tape is *QIC525.

***QIC1000:** The format of this tape is *QIC1000.

***FMT2GB:** The format of this tape is *FMT2GB.

***FMT5GB:** The format of this tape is *FMT5GB.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

CRTDATE

Specifies, for tape input data files and for tape output for which EXTEND(*YES) is specified, the date when the data file was created (written on tape).

Note: The data file creation date is stored in file labels on the tape. If a creation date is specified for any type of label processing other than standard-label (*SL), it is ignored. If the creation date written on the tape containing the data file does not match the date specified in this device file description, an inquiry message is sent to the operator.

***NONE:** The creation date is not specified. It is not checked unless it is supplied before the device file is opened, either in a OVRTAPF command or CHGTAPF command, or in the high-level language program.

creation-date: Specify the creation date of the data file used by this device file. The date must be specified in the format defined by the job attributes DATFMT and, if separators are used, DATSEP.

EXPDATE

Specifies, for tape output data files only, and only when standard-labeled tapes are used, the expiration date of the data file used by this device file. If a date is specified, the data file is protected and cannot be overwritten until after the specified expiration date. The files cannot be overwritten until after the expiration date.

Note: The data file expiration date is stored in file labels on the tape. If an expiration date is specified for any type of label processing other than *SL, it is ignored.

***NONE:** No expiration date for the data file is specified; the file is not protected. An expiration date is written in the data file labels so the file can be used as a scratch data file.

***PERM:** The data file is permanently protected. An expiration date of 999999 is assigned.

expiration-date: Specify the date on which the data file expires, after which it can be overwritten with new data. The expiration date must be later than or equal to the current date. The date must be specified in the format defined by the job attributes QDATFMT and, if separators are used, QDATSEP.

ENDOPT

Specifies the operation that is automatically performed on the tape volume after the operation ends. If more

than one volume is included, this parameter applies only to the last tape volume used; all other tape volumes are rewound and unloaded when the end of the tape is reached.

***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE:** The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

This option is used to reduce the time required to position the tape if the next tape file that opens to this device uses a data file that is on this volume.

Note: Even if ENDOPT(*LEAVE) is specified, the next tape file opened to this reel is positioned at the beginning of some data file on the volume (or at the end of a data file, for either read backward or for output that extends an existing data file on the volume). A tape file is always positioned at the start or end of a data file when it is opened.

***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

USRLBLPGM

Specifies the qualified name of the user program that processes user tape labels. On an output file, the user label program will pass the user labels that are written to tape. On an input file, the user labels are passed to the user label program.

***NONE:** There is no user label program for this device file.

The name of the user label program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

user-label-program-name: Specify the name of the user program that processes the user tape labels. If no library qualifier is given, *LIBL is used to find the file.

IGCDTA

Specifies whether the file processes double-byte character set (DBCS) data.

***NO:** The file does not process DBCS data.

***YES:** The file processes DBCS data.

WAITFILE

Specifies the number of seconds that the program waits for the file resources and session resources to be allocated when the file is opened, or for the device or session resources to be allocated when an acquire operation is performed to the file. If those resources are not allocated within the specified wait time, an error message is sent to the program. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: An immediate allocation of the device by the device resource is required when an acquire operation is performed to the file.

***IMMED:** The program does not wait; when the file is opened, an immediate allocation of the file resources is required.

***CLS:** The job default wait time is used as the wait time for the file resources being allocated.

number-of-seconds: Specify the number of seconds that the program waits for the file resources to be allocated to the tape file when the file is opened, or the wait time for the device allocated when an acquire operation is performed to the file. Valid values range from 1 through 32767 seconds.

SHARE

Specifies whether the open data path (ODP) for the tape file is shared with other programs in the routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

More information on shared database files is in the *Database Guide*.

***NO:** The ODP created by the program with this attribute is not shared with other programs in the routing step. Every time a program opens the file with this attribute, a new ODP to the file is created and activated.

***YES:** The ODP created with this attribute is shared with each program in the routing step that also specifies SHARE(*YES) when it opens the file.

Note: When SHARE(*YES) is specified and control is passed to a program, a read operation in that program retrieves the next input record. A write operation produces the next output record.

AUT

Specifies the authority given to users who do not have specific authority to the tape file, who are not on an authorization list, and whose user group has no specific authority to the tape file.

***LIBCRTAUT:** The public authority for the tape file is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the tape file). The public authority is determined when the tape file is created. If the CRTAUT value for the library changes after the tape file is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the tape file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the tape file, but the user cannot change it. Change

authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the tape file.

***USE:** The user can perform basic operations on the tape file, such as running a program or reading a file. The user cannot change the tape file. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the tape file.

authorization-list-name: Specify the name of the authorization list used.

REPLACE

Specifies whether the existing tape file is replaced by the new file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***YES:** The existing tape file is replaced by the one being created.

***NO:** The existing file, if any, is not replaced by the tape file.

TEXT

Specifies text that briefly describes the tape file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Table 39. Record Length Values

Absolute RCDLEN Ranges					
		FILETYPE(*DATA)		FILETYPE(*SRC)	
CODE	RCDFBLKFMT	Minimum RCDLEN	Maximum RCDLEN	Minimum RCDLEN	Maximum RCDLEN
*EBCDIC *ASCII	*F *FB *U *F *FB *U	18 18	32767 32767	30 30	32767 32767
*EBCDIC *ASCII	*V *VB *D *DB	1 1	32759 9995	13 13	32767 10007
*EBCDIC *ASCII	*VS *VBS *VS *VBS	1 1	32759 32759	13 13	32767 32767

Table 40. Required RCDLEN/BLKLEN/BUFOFSET Relation

CODE	RCDBLKFMT	BLKLEN = fcn(RCDLEN,BUFOFSET)
*EBCDIC *ASCII	*F *U *F *U	BLKLEN = RCDLEN BLKLEN=RCDLEN + BUFOFSET
*EBCDIC *ASCII	*FB *FB	BLKLEN = RCDLEN * n BLKLEN = (RCDLEN * n) + BUFOFSET n is the number of records in a maximum-length block
*EBCDIC *ASCII	*V *D	BLKLEN = RCDLEN * 8 BLKLEN = RCDLEN * 4 + BUFOFSET
*EBCDIC *ASCII	*VB *DB	BLKLEN >= RCDLEN + 8 BLKLEN >= RCDLEN + 4 + BUFOFSET
*EBCDIC *ASCII	*VS *VBS *BS *VBS	BLKLEN >= 18 BLKLEN >= 6 + BUFOFSET (18 minimum)

Examples

Example 1: Creating a Description of a Tape Device File

```
CRTTAPF   FILE(BACKHST)  DEV(QTAPE1 QTAPE2 QTAPE3)
  REELS(*BLP 10)  RCDLEN(256)  BLKLEN(1024)
  RCDBLKFMT(*FB)  EXTEND(*YES)  ENDOPT(*UNLOAD)
  WAITFILE(60)
```

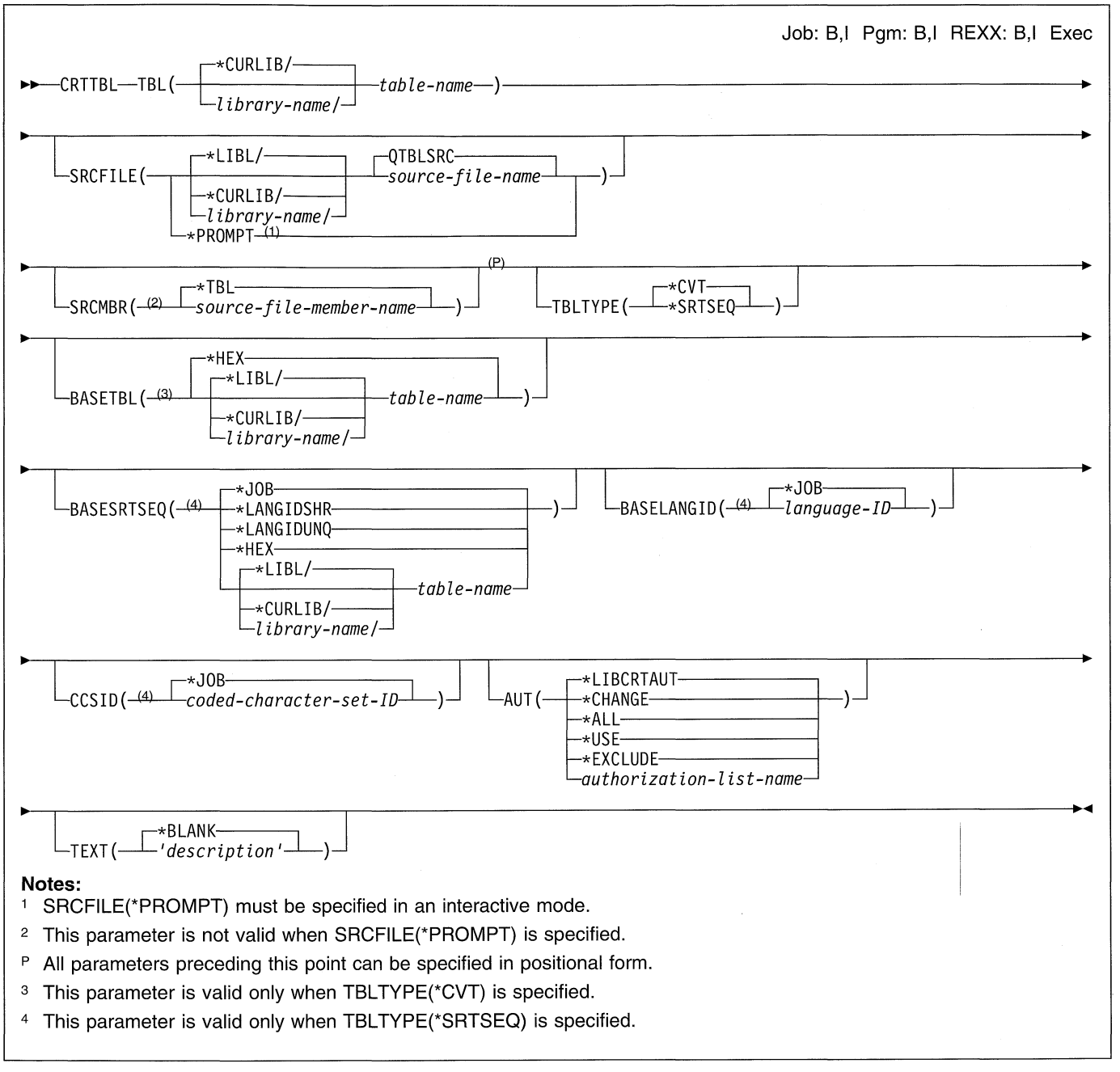
This command creates a description of the tape device file named BACKHST in the current library, to be used with the tape devices QTAPE1, QTAPE2, and QTAPE3. All volumes processed on these devices with this device file must have standard-labels. Each block of data (EBCDIC character code) on the tape volumes contains four records of 256 bytes each. When records are written to the tape, they are added to the end of the data file. No creation or expiration date is specified for this tape, and both unloading and rewinding operations will occur when the device file is closed at the last tape volume processed. The program using this tape device file waits 60 seconds for file resources to be allocated when this file is opened, and this device file is used only when the current program is running.

Example 2: Creating a Tape File Containing DBCS Data

```
CRTTAPF   FILE(IGCLIB/IGCTAP)  LABEL(GENINF)
  IGCDTA(*YES)
```

This command creates the tape file IGCTAP that is stored in library IGCLIB, which is labeled GENINF, and contains DBCS data.

CRTTBL (Create Table) Command



Notes:

1 SRCFILE(*PROMPT) must be specified in an interactive mode.

2 This parameter is not valid when SRCFILE(*PROMPT) is specified.

P All parameters preceding this point can be specified in positional form.

3 This parameter is valid only when TBLTYPE(*CVT) is specified.

4 This parameter is valid only when TBLTYPE(*SRTSEQ) is specified.

Purpose

The Create Table (CRTTBL) command creates a named table. The table can be used for the translation of data being moved between the system and a device (a display, for example). The table can also be used to specify an alternative collating sequence or to specify field translation functions.

You can create a table by prompting for the source information, or by specifying a source member.

To create a table using prompting support, specify *PROMPT on the SRCFILE parameter. A display is shown allowing you to view and change the table values and a new table is created based on the values chosen on the display.

To create a table using a source member, you must specify a source member that contains 8 records. Each record in this member must contain 64 hexadecimal characters (characters after position 64 are not used). This input (512 hexadecimal characters) is converted and stored internally as 256 bytes by the CRTTBL command.

The value you specify for a position within the source member is the same value that is returned by QDCXLATE (or other system program) whenever that position is encountered.

For example, if you specify "C0C1C2C3C4C5C6..." as the first part of record 1 in the source, then a hexadecimal "C0" is returned when given a hexadecimal "00", a hexadecimal "C1" is returned when given a hexadecimal "01" and so on.

Required Parameter

TBL

Specifies the qualified name of the table being created.

The name of the table can be qualified by one of the following library values:

***CURLIB:** The table is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the table is created.

table-name: Specify the name of the table being created.

Optional Parameters

SRCFILE

Specifies the qualified name of the source file that contains the description of the table being created or that prompting support is to be used.

The name of the source file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QTBLSRC: The system source file named QTBLSRC contains the source records that are used with this command to create the table.

source-file-name: Specify the name of the source file that contains the source records that are used with this command to create the table.

Other Single Value

***PROMPT:** Allows the user to access and use source information.

SRCMBR

Specifies the name of the source file member that contains the description of the table being created.

***TBL:** The source file member name is the same as the name of the table.

source-file-member-name: Specify the name of the member in the source file specified on the SRCFILE parameter that is used to create the table.

TBLTYPE

Specifies the type of table object to be created.

***CVT:** A conversion type table is created.

***SRTSEQ:** A sort sequence type table is created.

BASETBL

Specifies the base table to be used for prompting support when creating a conversion table.

***HEX:** A one to one mapping table is used.

The name of the table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify a table object to be used as a base.

BASESRTSEQ

Specifies the base table to be used for prompting support when creating a sort sequence table.

***JOB:** The sort sequence must be resolved when the object is created.

***LANGIDSHR:** A shared weight sort table is used.

***LANGIDUNQ:** A unique weight sort table is used.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence. The CCSID for hexadecimal data is 65535.

The name of the table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify the name of a table object to be used as a base.

BASELANGID

Specifies the base language used for prompting support when creating a sort sequence table.

***JOB:** The sort sequence must be resolved when the object is created.

language-ID: Specify a valid language ID to be used for the sort sequence table being created.

CCSID
Specifies the coded character set ID (CCSID) in which to store the sort sequence table information.

***JOB:** The sort sequence must be resolved when the object is created.

coded-character-set-ID: Specify the CCSID to be used for the sort sequence table information.

AUT
Specifies the authority given to users who do not have specific authority to the table, who are not on an authorization list, and whose user group has no specific authority to the table. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***LIBCRTAUT:** The public authority for the table is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the table). The public authority is determined when the table is created. If the CRTAUT value for the library changes after the table is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the table.

***USE:** The user can perform basic operations on the table, such as running a program or reading a file. The user cannot change the table. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the table.

authorization-list-name: Specify the name of the authorization list used.

TEXT
Specifies text that briefly describes the table. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

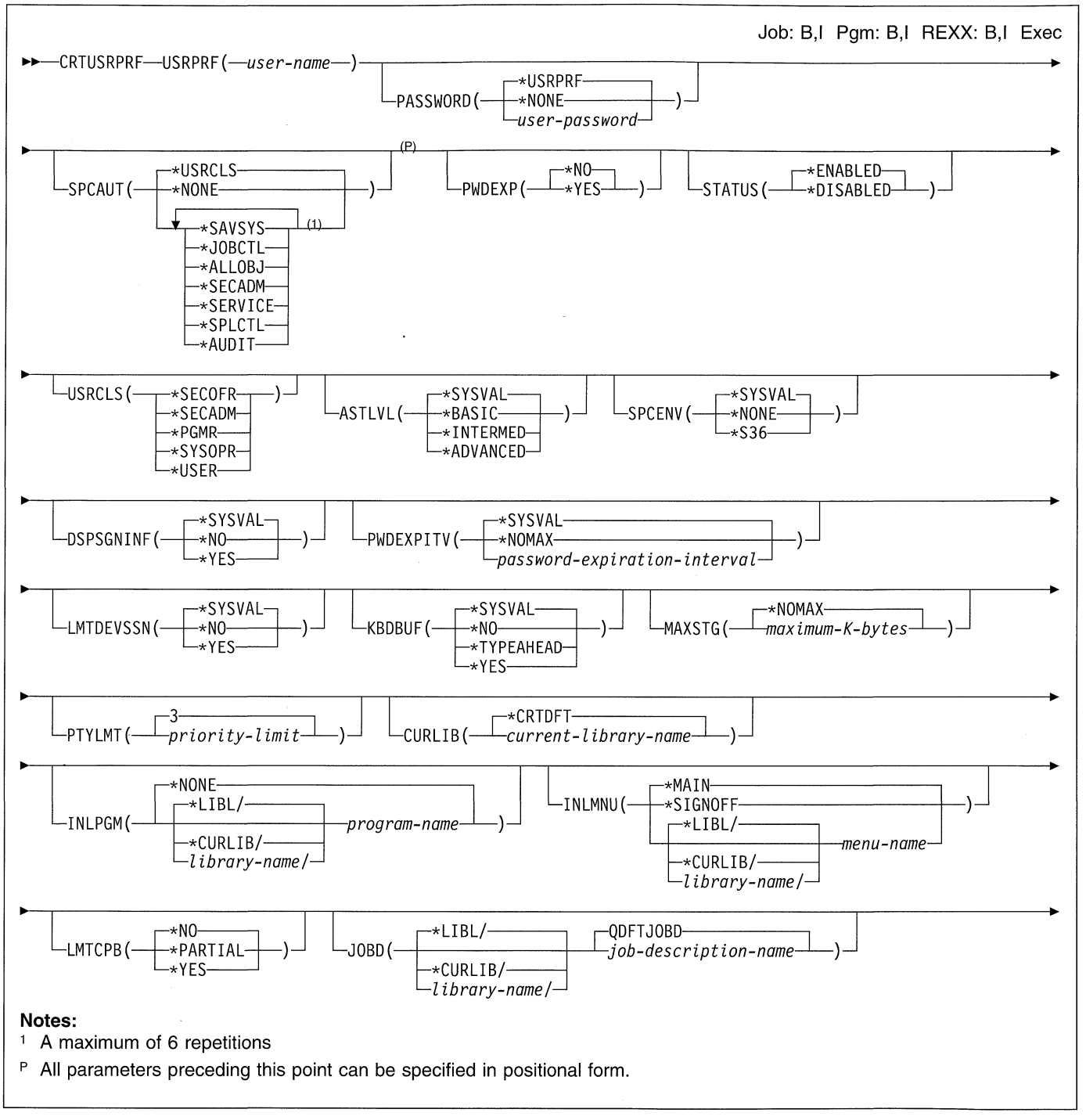
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
CRTTBL   TBL(SCRAMTBL)  SRCFILE(USERTABLES)
  SRCMBR(SCRAMBLE)  TEXT('Translate table for
  scrambling text characters')
```

This command creates a table named SCRAMTBL and (by default) stores it in the current library. The source file named USERTABLES contains the source records used when the table is created; the name of the source file member is SCRAMBLE. The TEXT parameter describes this table as being used as a translate table for scrambling text characters.

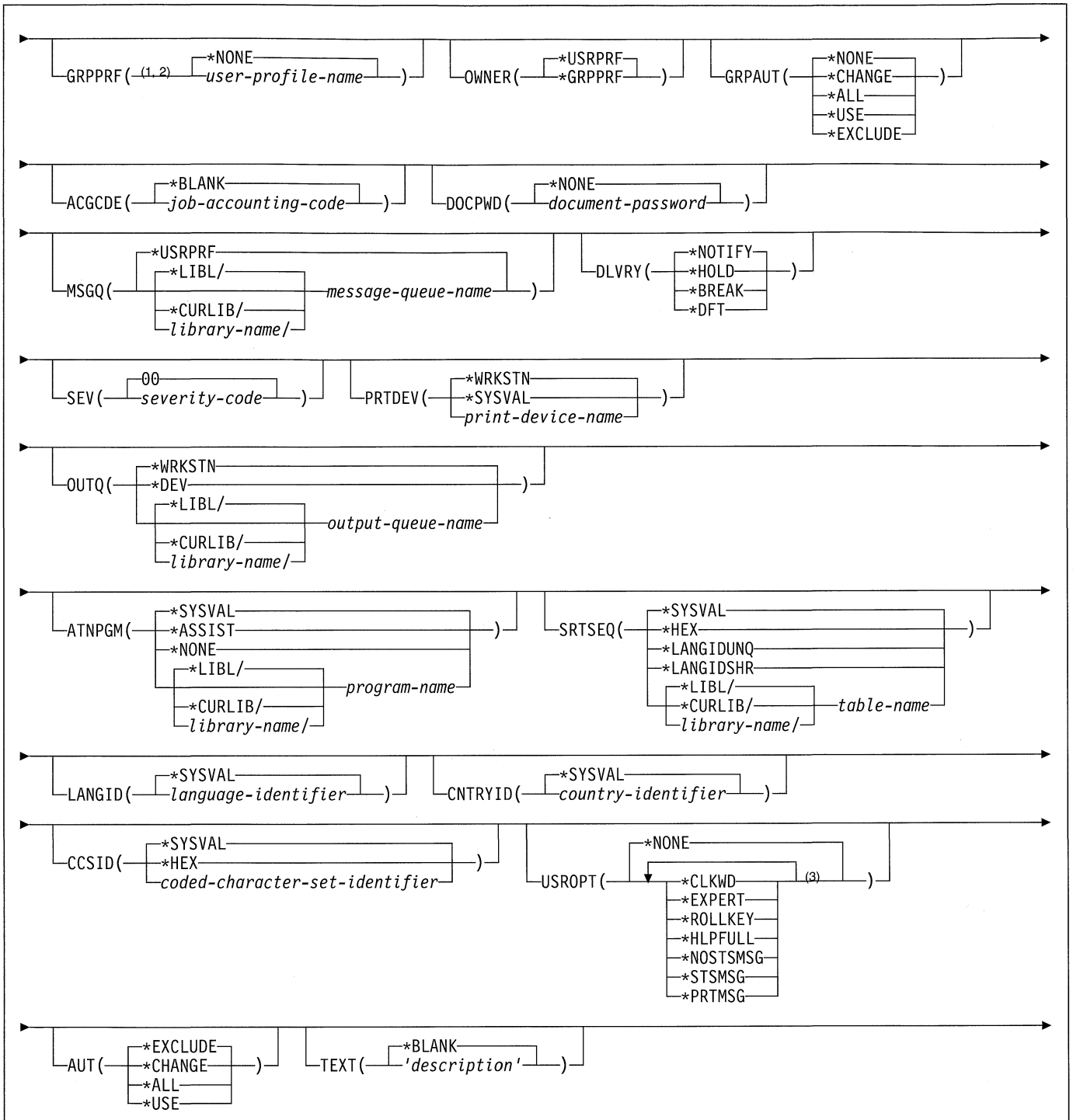
CRTUSRPRF (Create User Profile) Command



Notes:

1 A maximum of 6 repetitions

P All parameters preceding this point can be specified in positional form.



Notes:

1 The following user profiles cannot be specified on the GRPPRF parameter: QRJE, QSRV, QSRVBAS, QSYS, QLPAUTO, QLPINSTALL, QSPL, QDBSHR, QSPLJOB, QDOC, QSNADS, QGATE, QDFTOWN, QDSNX, QTSTRQS, and QFNC.

2 The following user profiles cannot be specified on a group profile: QCSM, QCSMBAS, QSECOFR, and QRJE. Because the IBM-supplied user profiles QSYS, QDBSHR, QSPL, QSPLJOB, QDOC, QSNADS, and QFNC are not allowed on the CHGUSRPRF command, they cannot have a group profile.

3 A maximum of 7 repetitions

Purpose

The Create User Profile (CRTUSRPRF) command identifies a user to the system and creates a user profile containing the values assigned only to that user. The system uses these values to control jobs submitted by the user or jobs that are run under the constraints of this user profile but are submitted by other system users. When the user profile is created, it is stored as part of the internal system, and it appears as it exists in the QSYS (system) library.

When a user identified by this user profile uses the system, the system checks the user profile to determine which objects the user is authorized to use. When the user profile is created, the user is granted *CHANGE and object management authorities for the profile itself, and the user can use only objects, commands and devices that have public authority. Subsequently, the security officer and other object owners can specifically grant to the user authority to use other objects by means of the Grant Object Authority (GRTOBJAUT) command. If the user profile has *ALLOBJ special authority, the user can use all objects, commands, and devices regardless of any private authorities. The profile identifies:

- Objects owned by the user
- Commands and system devices authorized for use by the user
- Objects (including commands and devices) specifically authorized for use by the user; the names of the objects and the authority granted are stored in the profile
- The number of objects owned and/or authorized for use by the user
- Special authorities granted to the user
- The user's class
- The operating environment for the user
- The maximum amount of storage allocated, and the amount currently in use, for the storage of the objects owned by the user
- The current library for the user
- The name of the program called when the user signs on
- The name of the menu displayed when the user signs on
- The maximum scheduling priority for jobs and spooled output, the first program called after user sign-on, and text describing the user's profile
- The group profile associated with the user profile, the owner of objects created by the user's profile (either the job user profile or the group profile), and the authority automatically granted to the group profile when a new object is created
- The accounting code associated with the user's profile
- The job description associated with the user's profile
- The document password that allows a Document Interchange Architecture (DIA) user to prevent personal document distributions from being obtained by other users cooperatively working with authorized programs
- The message queue name associated with the user profile
- The output queue name associated with the user profile
- The ATTN key handling program for this user
- Password expiration information

The password validation rules are not verified by the system when a password is specified on this command. See the *Security Reference* for a description of the password validation rules.

Restrictions:

1. The user of this command must have (1) *SECADM special authority, (2) *USE authority to the initial program, initial menu, job description, message queue, output queue, and attention-key-handling program if specified, and (3) *CHANGE and object management authorities to the group profile if specified.
2. The following user profiles cannot be specified on the GRPPRF parameter: QRJE, QSRV, QSRVBAS, QSYS, QSPL, QDBSHR, QSPLJOB, QDOC, QSNADS, QGATE, QDFTOWN, QDSNX, QTSTRQS, QFNC, QLPINSTALL, and QLPAUTO.

Required Parameter

USRPRF

Specifies the user name for this user profile. The character set for user profile names is the same set used for *SNAME type on the parameter statement of command definitions source statements. A numeric user profile can be specified. If the user profile is numeric, then it must begin with a *Q*.

Optional Parameters

PASSWORD

Specifies the password that lets the user sign on the system. The password is associated with a unique user name used to represent the user in the system. The password must be known only to the user and to the security officer. A numeric password can be specified. If the password is numeric, then it must begin with a *Q*, for example, *Q1234* where *1234* is the password used for signing on the system.

Note: The new password is not checked against the password validation rules. The password validation rules are defined by AS/400 system values. For a description of the password validation rules, see the *Security Reference*.

***USRPRF:** The password for this user is the user ID specified in the USRPRF parameter.

***NONE:** No password is used by this user. Users cannot sign on a system with a profile that has PASSWORD(*NONE) specified.

user-password: Specify the alphanumeric character string (10 characters or less) password that identifies the user and the user's personal profile. The first character must be alphabetic and the other characters must be alphanumeric.

SPCAUT

Specifies the special authorities granted to a user. Special authorities are *required* to perform certain functions on the system. Special authorities cannot be removed from the QSECOFR or the QSYS user profiles.

The following authorities are usually given:

- Save system authority (*SAVSYS) to users who need to operate the system.
- Job control authority (*JOBCTL) to users who need to operate the system.
- Security administrator authority (*SECADM) to users who need to create, change, or delete user profiles.
- All object authority (*ALLOBJ) to users who need to work with system resources.
- Service authority (*SERVICE) to users who need to perform service functions.
- Spool control authority (*SPLCTL) to users who need to perform all spool-related functions.
- Audit authority (*AUDIT) to users who need to perform auditing functions.

Restrictions:

1. The user profile that is creating or changing another user profile must have all the special authorities being granted.
2. A user must have *ALLOBJ and *SECADM special authorities to grant another user *SECADM special authority when using the CRTUSRPRF command.
3. The user must have *ALLOBJ, *SECADM, and *AUDIT special authorities to grant another user *AUDIT special authority when using the CRTUSRPRF command.

***USRCLS:** Special authorities are granted to the user based on the value specified in the USRCLS parameter.

***NONE:** No special authorities are granted to the user.

***SAVSYS:** Save system authority is granted to this user profile. It is normally granted to users who operate the system. The user is given the authority to save, restore, and free storage for all objects on the system, whether or not object management authority has been granted.

***JOBCTL:** Job control authority is granted to the user. It is normally granted to users who operate the system. The user is given the authority to change, display, hold, release, cancel, and clear all jobs running on the system or on a job queue or output queue for which OPRCTL (*YES) is specified. The user also has the authority to load the system, to start writers, and to stop active subsystems.

***ALLOBJ:** All object authority is granted to the user. It is granted to users who work with system resources. The user can access any system resource whether or not the user has private authorizations.

***SECADM:** Security administrator authority is granted to the user. It is granted to users who are the security administrators. The user can create, change, or delete user profiles if authorized to the Create User Profile, Change User Profile, or Delete User Profile commands, and is authorized to the user profile. This authority does not allow the user to grant special authorities that this user profile does not have.

***SERVICE:** Service authority is granted to the user. The user can perform service functions.

***SPLCTL:** Spool control authority is granted to the user. The user can perform all spool functions.

***AUDIT:** Audit authority is granted to the user. This user is given the authority to perform auditing functions. Auditing functions include turning auditing on or off for the system and controlling the level of auditing on an object or user.

PWDEXP

Specifies whether the user's password is set to expired. If the password is set to expired, the user is required to change the password to sign on the system. The Sign-On Information display is shown, and the user can call the Change Password (CHGPWD) command from that display.

***NO:** The password is not set to expired.

***YES:** The password is set to expired.

STATUS

Specifies profile status.

***ENABLED:** The profile created is valid for sign-on.

***DISABLED:** The profile created is not valid for sign-on until an authorized user enables it again.

USRCLS

Specifies the class of user associated with this user profile: security officer, security administrator, programmer, system operator, or user. User class determines which menu options are shown. The special authorities defined by the user class are used only if SPCAUT(*USRCLS) is specified.

***SECOFR:** At all levels of system security, the security officer is granted all object, save system, job control, service, spool control, security administrator, and audit special authorities.

***SECADM:** At system security level 10 or 20, the security administrator is granted all object, save system, security administrator, and job control special authorities.

At system security level 30 or above, the user is granted security administrator, save system, and job control special authorities.

***PGMR:** At system security level 10 or 20, the programmer is granted all object, save system, and job control special authorities.

At system security level 30 or above, the user is granted save system and job control special authorities.

***SYSOPR:** At system security level 10 or 20, the system operator is granted all object, save system, and job control special authorities.

At system security level 30 or above, the user is granted save system and job control special authorities.

***USER:** At system security level 10 or 20, the user is granted all object and save system special authorities. At system security level 30 or above, the user is granted no special authorities.

ASTLVL

Specifies which user interface to use.

***SYSVAL:** The assistance level defined for the system is used.

***BASIC:** The Operational Assistant user interface is used.

***INTERMED:** The system interface is used.

***ADVANCED:** The expert system interface is used. To allow for more list entries, the options keys and the function keys are not displayed. If a command does not have an advanced (*ADVANCED) level, the intermediate (*INTERMED) level is used.

SPCENV

Specifies the special environment in which the user operates after signing on the system.

***SYSVAL:** The system value QSPCENV is used to determine the system environment after the user signs on the system.

***NONE:** The user operates in the AS/400 system environment after signing on the system.

***S36:** The user operates in the System/36 environment after signing on the system.

DSPSGNINF

Specifies whether the sign-on information is displayed when the user signs on. This allows users to see the sign-on information, such as date of last sign-on and sign-on attempts that were not valid. If the password is due to expire in 7 days or less, the number of days until the password expires is shown.

***SYSVAL:** The system value QDSPSGNINF is used to determine whether the sign-on information display is shown when the user signs on the system.

***NO:** The sign-on information display is not shown when the user signs on the system.

***YES:** The sign-on information display is shown when the user signs on the system.

PWDEXPITV

Specifies the interval (number of days from the password changed date) before the password expires.

***SYSVAL:** The system value QPWDEXPITV is used to determine the password expiration interval.

***NOMAX:** The system maximum is used.

password-expiration-interval: Specify the number of days between when the password was last changed and the date when the password expires. Valid values range from 1 through 366.

LMTDEVSSN

Specifies whether the user is limited to one device session. This does not limit use of the System Request menu or prevent a second sign-on.

***SYSVAL:** The system value QLMTDEVSSN determines whether the user is limited to one device session.

***NO:** The user is not limited to one device session.

***YES:** The user is limited to one device session.

KBDBUF

Specifies the keyboard buffering value used when a job is initialized for this user profile. If the type-ahead feature is active, the keystrokes can be buffered. If the attention key buffering option is active, the attention key is buffered like any other key. If the attention key is not active, the attention key is not buffered and is sent to the system even if the display station is input inhibited. The keyboard buffer value can also be set by a user application using the QWSSETWS program. More information is in the *System Programmer's Interface Reference*.

***SYSVAL:** The system value, QKBDBUF, is used to determine the keyboard buffering value for this profile.

***NO:** The type-ahead feature and attention key buffering option are not active for this user profile.

***TYPEAHEAD:** The type-ahead feature is active for this user profile.

***YES:** The type-ahead feature and attention key buffering option are active for this user profile.

MAXSTG

Specifies the maximum auxiliary storage space assigned to store permanent objects owned by this user profile. If the storage maximum is exceeded when an interactive user creates an object, an error message is displayed and the object is not created. If the maximum is exceeded when an object is created in a batch job, an error message is sent to the job log (depending on the logging level of the job) and the object is not created.

When planning maximum storage space for user profiles, consider the following system actions:

- A restore operation first assigns storage space to the user doing the restore operation, and then transfers the object to the owner. For a large restore operation, specify MAXSTG(*NOMAX).
- The user profile that creates a journal receiver is assigned storage space as the receiver size grows. If new receivers are created using JRNRCV(*GEN), the storage continues to be assigned to the user profile that owns the active journal receiver. For a very active journal receiver, specify MAXSTG(*NOMAX).
- User profiles that transfer all created objects to their group profile must have adequate storage in the user profiles to contain any created object before it is transferred to the group profile.

- The owner of the library is assigned storage space for the descriptions of objects stored in a library, even when those objects are owned by another profile. Examples of such objects are text and program references.
- Storage space is assigned to the user profile for temporary objects used when the job is running. Examples of such objects are commit control blocks, file editing space, and documents.

***NOMAX:** As much storage as is required is assigned to this profile.

maximum-K-bytes: Specify the maximum amount of storage in kilobytes (1KB equals 1024 bytes) that can be assigned to this user profile.

PTYLMT

Specifies the highest scheduling priority the user is allowed to have for each job submitted to the system. The value specified for this parameter controls the job processing priority and output priority for any job running under this user profile. This means that values specified in the JOBPTY and OUTPTY parameters of any job command cannot exceed the PTYLMT value of the user profile under which the job is running. The scheduling priority value ranges from 0 through 9, where 0 is the highest priority and 9 is the lowest priority.

3: The user named in this profile can have a priority value no higher than 3 for scheduling jobs on the system. Jobs with this priority value are run before jobs with priority values ranging from 4 through 9, and after jobs with priority values ranging from 0 through 2.

priority-limit: Specify a value, ranging from 0 through 9, for the user's highest job scheduling priority.

CURLIB

Specifies the name of the library being used as the current library during the processing of this command.

Note: If *PARTIAL or *YES is specified for the LMTCPB parameter in the user profile, the user cannot change the current library at sign-on or with the Change Profile (CHGPRF) command.

***CRTDFT:** The user has no current job library. If objects are created in the current library by using *CURLIB on a create command, the QGPL library is used as the default current job library.

current-library-name: Specify the 10-character name of the user's current job library after the user signs on the system.

INLPGM

Specifies, for an interactive job, the name of the program called whenever a new routing step is started that has QCMD as the request processing program. No parameters can be passed to the program. If *PARTIAL or *YES is specified on the LMTCPB parameter in the user profile, the first program value cannot be changed at sign-on or with the Change Profile (CHGPRF) command.

A System/36 environment procedure name can be specified as the initial program if the procedure is a member of the file QS36PRC (in the library list or specified library) and if either of the following conditions are true:

- *S36 is specified for the SPCENV parameter.
- *SYSVAL is specified for the SPCENV parameter and the system value, QSPCENV, is *S36.

***NONE:** No program is called when the user signs on the system. If a menu name is specified in the INLMNU parameter, that menu is displayed.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program called when the user signs on the system.

INLMNU

Specifies the name of the menu displayed when the user signs on the system if the user's routing program is the command processor QCMD. If *YES is specified on the LMTCPB parameter in the user profile, the user cannot change this menu.

A System/36 environment menu can be specified as the initial menu if either of the following conditions are true:

- *S36 is specified for the SPCENV parameter.
- *SYSVAL is specified for the SPCENV parameter and the system value, QSPCENV, is *S36.

MAIN: The main menu is shown when the initial program completes.

***SIGNOFF:** The system signs off the user when the initial program completes. This is intended for users limited to only running the initial program.

The name of the menu can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

menu-name: Specify the name of the menu shown when the user signs on the system.

LMTCPB

Specifies the limits of user control over the initial program, the initial menu, the current library, and the ATTN key handling program values. This parameter is ignored when the security level is 10.

Note: When creating or changing other users' user profiles, users running this command cannot specify values on this parameter that grant greater capabilities to other users than their own user profiles grant to them. For example, if *PARTIAL is specified on the LMTCPB parameter in the user profile of the user running this command, *PARTIAL or *YES can be specified for another user. *NO cannot be specified for another user.

***NO:** The program, menu, and current library values can be changed when the user signs on the system. A user can change the initial program, the initial menu, the current library, or the ATTN key handling program values in the user's user profile by using the Change Profile (CHGPRF) command.

***PARTIAL:** The initial program and the current library cannot be changed on the sign-on display. The menu can be changed and commands can be run from the command line of a menu. A user can change the menu value by using the Change Profile (CHGPRF) command. The initial program, the current library, and the ATTN key handling program cannot be changed by using the Change Profile command.

***YES:** The program, the menu, and the current library values cannot be changed on the sign-on display. Commands cannot be run from the command line of a menu, but can still be run from the command entry display. The user cannot change the first program, the first menu, the current library, or the ATTN key handling program by using the Change Profile (CHGPRF) command.

JOBD

Specifies the name of the job description used. If the job description does not exist when the entry is changed, a library qualifier must be specified because the qualified job description name is retained in the subsystem description. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

The name of the job description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QDFTJOBD: The default system-supplied job description found in the QGPL library is used.

job-description-name: Specify the name of the job description that is used for the work station entries whose job description parameter values indicate the user (JOBD(*USRPRF)).

GRPPRF

Specifies the name of the group profile whose authority is used if the user has no specific authority for the job. The current user of this command must have *OBJMGT and *CHANGE authorities to the profile specified on the GRPPRF parameter. The required *OBJMGT cannot come from the program adopt operation.

Notes:

1. When a group profile is specified, the user is automatically granted *CHANGE and *OBJMGT authority to the group profile.
2. The following IBM-supplied objects are not valid on this parameter:

QDBSHR	QRJE
QDFTOWN	QSNADS
QDOC	QSPL
QDSNX	QSPLJOB
QFNC	QSRV
QGATE	QSRVBAS
QLPAUTO	QSYS
QLPINSTALL	QTSTRQS

***NONE:** This user profile has no group profile.

user-profile-name: Specify the name of a group profile used with this user profile.

OWNER

Specifies whether the user profile or the group profile is the owner of newly created objects.

***USRPRF:** The user profile being used with the job is made the owner of the newly created object.

***GRPPRF:** The group profile is made the owner of newly created objects and is given all authority to those objects. The user profile used with the job does not have specific authority to the object. If *GRPPRF is specified, there must be a user profile name in the GRPPRF parameter, and the GRPAUT parameter cannot be specified.

GRPAUT

Specifies the specific group authority granted to the group profile for newly created objects. If *GRPPRF is specified on the OWNER parameter, specification of this parameter is not allowed.

***NONE:** No group authority is granted.

***CHANGE:** The group profile is granted authority to perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. In addition to the functions allowed with use (*USE) authority, the group profile can change the object. Change authority provides object operational authority and all of the data authorities.

***ALL:** The group profile is granted authority to perform all operations on the object except those limited to the owner or controlled by authorization list management authority. In addition to the functions allowed with use (*USE) and change (*CHANGE) authorities, the group profile can control the object's existence and specify the security for the object. The group profile can transfer ownership of the object. If the object is an authorization list, then the group profile cannot add, change, or remove users.

***USE:** The group profile runs a program or displays the contents of a file. The group profile is prevented from changing the object. Use authority provides object operational authority and read authority.

***EXCLUDE:** The group profile is prevented from accessing the object.

ACGCDE

Specifies the accounting code associated with this user profile. More information on job accounting is in the *Database Guide*.

***BLANK:** A 15-blank job accounting code is assigned to this user profile.

job-accounting-code: Specify the 15-character job accounting code used by jobs that get their accounting code from this user profile. If less than 15 characters are specified, the string is padded on the right with blanks.

DOCPWD

Specifies the document password that allows Document Interchange Architecture (DIA) users to protect personal distributions from being used by others working on their behalf. More information on specifying a document password is in the *Managing OfficeVision/400*.

***NONE:** The user has no document password.

document-password: Specify a document password for use by this user. The password must range from 1 through 8 alphanumeric characters (letters A through Z and numbers 0 through 9). The first character of the document password must be alphabetic; the remaining characters can be alphanumeric. Embedded blanks, leading blanks, and special characters are invalid.

MSGQ

Specifies the name of the message queue used by this user. The message queue is created if it does not already exist. The user profile being created is the owner of the message queue.

***USRPRF:** A message queue with the same name as that specified in the USRPRF parameter is the message queue for this user. This message queue is located in QUSRSYS.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

message-queue-name: Specify the name of the qualified message queue for this user.

DLVRY

Specifies how the messages that are sent to the message queue for the user profile are delivered.

***NOTIFY:** The job to which the message queue is assigned is notified when a message arrives at the message queue. For interactive jobs at a work station, the audible alarm is sounded and the message-waiting light is turned on. The delivery mode cannot be changed to *NOTIFY if the message queue is also being used by another job.

***HOLD:** The messages are held in the message queue until they are requested by the user or program.

***BREAK:** The job to which the message queue is assigned is interrupted when a message arrives at the message queue. If the job is an interactive job, the audible alarm (if present on the display station) is sounded. The delivery mode cannot be changed to *BREAK if the message queue is also being used by another job.

***DFT:** The default reply to the inquiry message is sent. If no default reply is specified in the message description of the inquiry message, the system default reply, *N, is used.

SEV

Specifies the lowest severity code that a message can have and still be delivered to a user in break or notify mode. Messages arriving at the message queue whose severities are lower than the severity code specified on this parameter do not interrupt the job or turn on the audible alarm or the message-waiting light; they are held in the queue until they are requested by using the Display Message (DSPMSG) command. If *BREAK or *NOTIFY is specified on the DLVRY parameter, and is in effect when a message arrives at the queue, the message is delivered if the severity code associated with the message is equal to or greater than the value specified here. Otherwise, the message is held in the queue until it is requested.

00: The default severity level is 00.

severity-code: Specify a value, ranging from 00 through 99, for the lowest severity code that a message can have and still be delivered if the message queue is in break or notify delivery mode.

PRTDEV

Specifies the qualified name of the default printer device for this job. If OUTQ(*DEV) is specified, the file is

placed on an output queue with the same name as the printer.

***WRKSTN:** The output queue assigned to the user's work station is used.

***SYSVAL:** The value specified in the system value QPRTDEV is used.

print-device-name: Specify the name of the printer used to print the output for this user.

OUTQ

Specifies the qualified name of the output queue.

***WRKSTN:** The output queue assigned to the user's work station is used.

***DEV:** The value specified on the DEV parameter of the CRTPRTF, CHGPRTF, or OVRPRTF command is used as the output queue for this user.

The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

output-queue-name: Specify the name of the output queue for this user.

ATNPGM

Specifies the program used for the ATTN key handling program for this user. The ATTN key handling program is called when the ATTN key is pressed in an interactive job. The program is active only when the user routes requests to the QCMD command processor supplied with the system. The ATTN key handling program is set on before a program (if any) is called and is active for both a program and a menu. If a program changes the ATNPGM by using the Set Attention Program (SETATNPGM) command, the new program remains active only for the duration of the program. When control returns and QCMD calls the menu, the original ATTN key handling program again becomes active. If the SETATNPGM command is run from a menu or an application is called from a menu, the new ATTN key handling program specified overrides the original ATTN key handling program. If *YES or *PARTIAL is specified on the LMTCPB parameter in the user profile, the ATTN key handling program cannot be changed with the Change Profile (CHGPRF) command.

***SYSVAL:** The system value QATNPGM is used.

***ASSIST:** QEZMAIN is used.

***NONE:** No ATTN key handling program is used by this user.

The name of the ATTN handling program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the ATTN key handling program used by this user.

SRTSEQ

Specifies the sort sequence table to be used for string comparisons for this user profile.

***SYSVAL:** The system value QSRTSEQ is used.

***HEX:** A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.

***LANGIDUNQ:** A unique-weight sort table is used.

***LANGIDSHR:** A shared-weight sort table is used.

The name of the sort sequence table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

table-name: Specify the name of the sort sequence table to be used with this user profile.

LANGID

Specifies the language identifier used for this user.

***SYSVAL:** The system value QLANGID is used.

language-identifier: Specify the language identifier. More information on valid language identifiers is in the *National Language Support Planning Guide*.

CNTRYID

Specifies the country identifier used for this user.

***SYSVAL:** The system value QCNTRYID is used.

country-identifier: Specify an ISO 3166 Alpha-2 code from the country code table. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

CCSID

Specifies the coded character set identifier (CCSID) used for this user.

A CCSID is a 16-bit number identifying a specific set of encoding scheme identifiers, character set identifiers,

code page identifiers, and additional coding-related information that uniquely identifies the coded graphic representation used.

***SYSVAL:** The system value QCCSID is used.

***HEX:** The CCSID 65535 is used.

coded-character-set-identifier: Specify the CCSID. More information on valid CCSIDs is in the *National Language Support Planning Guide*.

USROPT

Specifies the level of detail the user sees and the default function of the Page Up and Page Down keys. The system shows displays suitable for the inexperienced user. More experienced users must perform an extra action to see more detailed information. When values are specified for this parameter, the system presents detailed information without additional action by the experienced user.

***NONE:** No detailed information is shown.

***CLKWD:** Parameter keywords are shown instead of the possible parameter values when a control language (CL) command is displayed.

***EXPERT:** More detailed information is initially shown when the user is performing display and edit options such as the edit or display object authority.

***ROLLKEY:** The actions of the Page Up and Page Down keys are reversed.

***HLPFULL:** Help text is shown on a full display rather than in a window.

***NOSTSMSG:** Status messages are not shown when they are sent to the user.

***STSMSG:** Status messages are shown when they are sent to the user.

***PRTMSG:** A message is sent to this user's message queue when a spooled file for this user is printed or held by the printer writer.

AUT

Specifies the authority given to users who do not have specific authority to the object, who are not on an authorization list, and whose user group has no specific authority to the object.

***EXCLUDE:** The user cannot access the object.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user performs all operations on the object except those limited to the owner.

***USE:** The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. *USE authority provides object operational authority and read authority.

TEXT

Specifies text that briefly describes the user profile named in the USRPRF parameter. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a User Profile

```
CRTUSRPRF   USRPRF(JJADAMS)  PASSWORD(S1CR2T)
  SPCAUT(*SAVSYS)  INLPGM(ARLIB/DSPMENU)
```

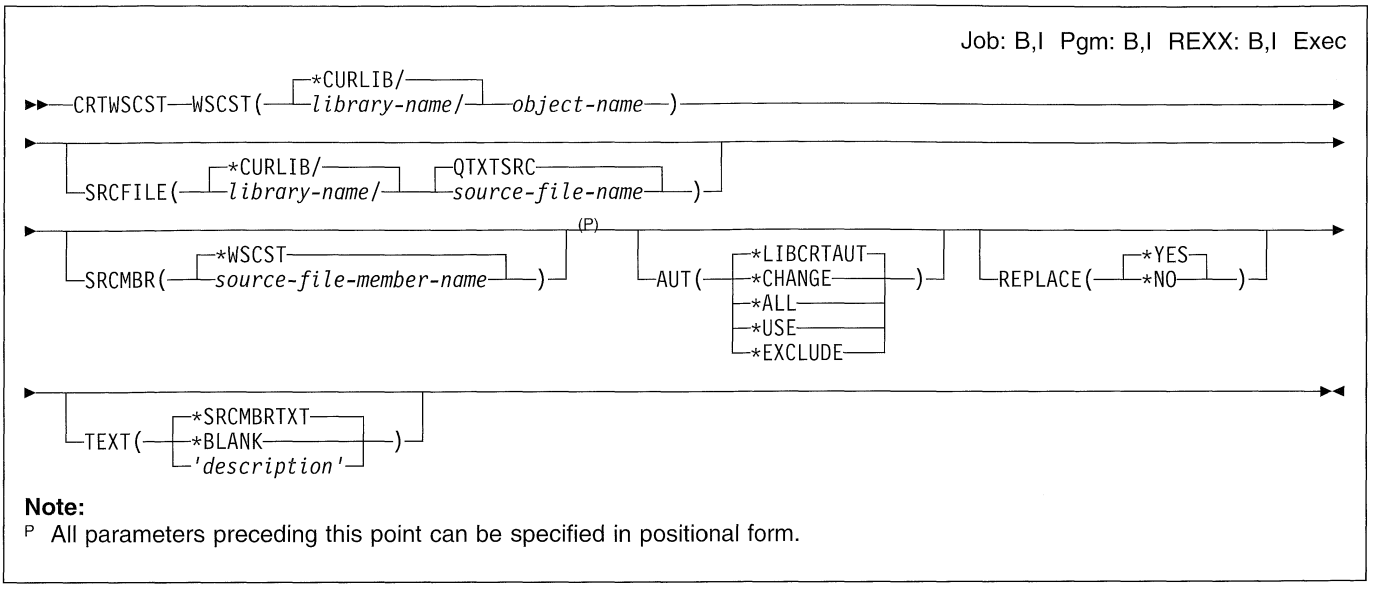
This command creates a user profile with the user name of JJADAMS and a password of S1CR2T. After sign-on, a program called DSPMENU in the ARLIB library is called. The user is granted the save system special authority. Because the other parameters were not specified: (1) The profile has no limit on the amount of storage assigned to it for owned permanent objects; (2) A scheduling priority of 3 is the highest priority that any of the user's jobs can have; (3) The user-defined description text is blank; (4) There is no group profile associated with this user profile; and (5) No authority is granted for the user profile to other users.

Example 2: Creating a User Profile with the Same User Name and Password

```
CRTUSRPRF   USRPRF(TMSMITH)  MAXSTG(10)
  INLPGM(PROGMR/CALC)
  TEXT('Ted M. Smith, Dept 410,
  Application Programs')
```

This command creates a user profile with the user name of TMSMITH; the password is also TMSMITH because the password was not specified. The maximum permanent storage space the user can use for all objects is 10K (or 10,240 bytes). The initial program called following sign-on is CALC, which is located in the library named PROGMR. The text parameter provides the user's name, department, and department name. Default values are assigned to the other parameters.

CRTWSCST (Create Work Station Customizing Object) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Create Work Station Customizing Object (CRTWSCST) command allows the user to create a work station customizing object in a library.

Required Parameter

WSCST
Specifies the name of a work station customizing object to be created.

The possible library values are:

***CURLIB:** The work station customizing object is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the work station customizing object is created.

object-name: Specify the name of the object that is created.

Optional Parameters

SRCFILE
Specifies the name of the source file in which a source file member containing the customizing table attributes resides. If the source file does not exist, an error message is displayed. The coded character set identifier for the source file should be *HEX.

***CURLIB:** The source file is located in the current library for the job.

library-name: Specify the name of the library where the source file is located.

QTXTSRC: The IBM-supplied source file QTXTSRC is used.

source-file-name: Specify the name of the source file.

SRCMBR
Specifies the name of the source file member containing the table attributes.

***WSCST:** Specifies that the work station customizing object name is used as the source member name.

source-file-member-name: Specify the name of the member in the source file specified on the SRCFILE parameter that is used to create the work station customizing object.

AUT
Specifies the authority given to users who do not have specific authority to the work station customizing object, who are not on an authorization list, and whose user group has no specific authority to the work station customizing object.

***LIBCRTAUT:** The public authority for the work station customizing object is taken from the value on the CRTAUT parameter of the target library (the library that is to contain the work station customizing object). The public authority is determined when the work station customizing object is created. If the CRTAUT value for the library changes after the work station customizing object is created, the new value does not affect any existing objects.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the work station customizing object.

***USE:** The user can perform basic operations on the work station customizing object, such as running a program or reading a file. The user cannot change the work station customizing object. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the work station customizing object.

REPLACE

Indicates whether an existing object is replaced.

***YES:** Replace the existing work station customizing object.

***NO:** Do not replace the existing work station customizing object.

TEXT

Specifies text that briefly describes the object. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***SRCMBRTXT:** The text is taken from the source file member used to create the work station customizing object.

***BLANK:** Text is not specified.

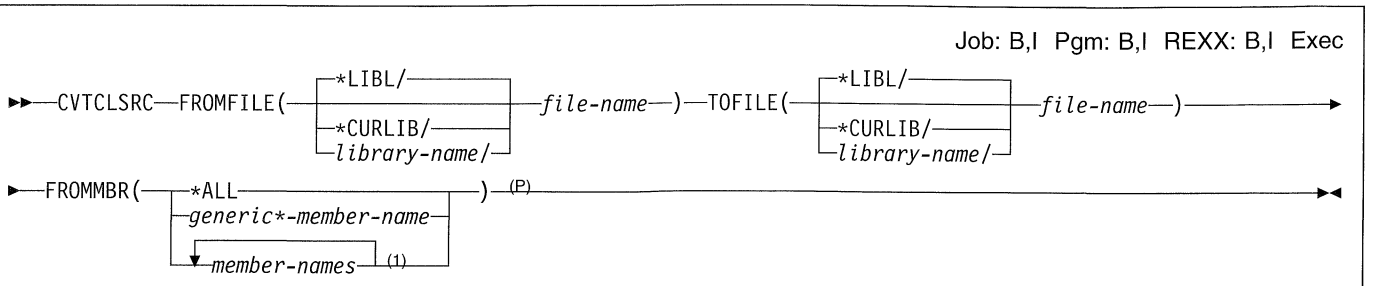
'description': Specify a description of the object.

Example

```
CRTWSCST  WSCST(MYLIB/MYWSCOBJ)
```

This command creates a work station customizing object named MYWSCOBJ.

CVTCLSRC (Convert CL Source) Command



Notes:
1 A maximum of 50 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Convert CL Source (CVTCLSRC) command is used to convert CL source from System/38 syntax to an AS/400 system syntax.

The CVTCLSRC command converts the following to an AS/400 system syntax:

- System/38 *object-name.library name* to AS/400 system *library-name/object-name* qualification
- System/38 *job-name[.user-name(.job-number)]* to AS/400 system *[(job-number/)user-name/]job-name* qualification
- Starting comment delimiters: (/*) to (/*) when necessary
- Command names
- Keyword names and values
- Missing required parameters for AS/400 commands may need to be added.

Conversion of user-defined commands is limited to the reordering of qualified names and adjusting comment syntax.

The CVTCLSRC command creates a report indicating the success or failure of the source file conversion. This report is contained in a printer file with the name 'CVTCLSRC'. Successful conversions of System/38 source are noted in the report with the message:

```
CPF0786   Member has been converted.
```

Error messages are printed for unsuccessful conversions. Some examples of functions which cannot be converted and may be printed as error messages in the report are:

```
stmt#  CPF0785  Command cannot be converted
stmt#  CPF0789  Keyword cannot be converted
```

The user may write a program, perhaps by using the Copy Spooled File (CPYSPLF) command, to process the report based on the success or failure of the conversion.

Restriction: Library QSYS38 must exist on the system for conversion of command names, keywords, keyword values, and detection of unsupported functions by this command. Commands with unsupported command names, keyword names, or keyword values are not converted.

Required Parameters

FROMFILE
Specifies the qualified name of the System/38 source file to convert to AS/400 system syntax.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the System/38 source file to convert to AS/400 system syntax.

TOFILE
Specifies the qualified name of the file in which the converted source is placed. It must be different than the name of the FROMFILE.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name in which the converted source file is placed.

FROMMBR

Specifies the member name of the source file member to convert.

***ALL:** All members of the specified source file are converted to AS/400 system syntax.

generic-member-name:* Specify the generic name of the source file members to convert. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be converted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

member-names: Specify the names of the source file members to convert. Specify no more than 50 names.

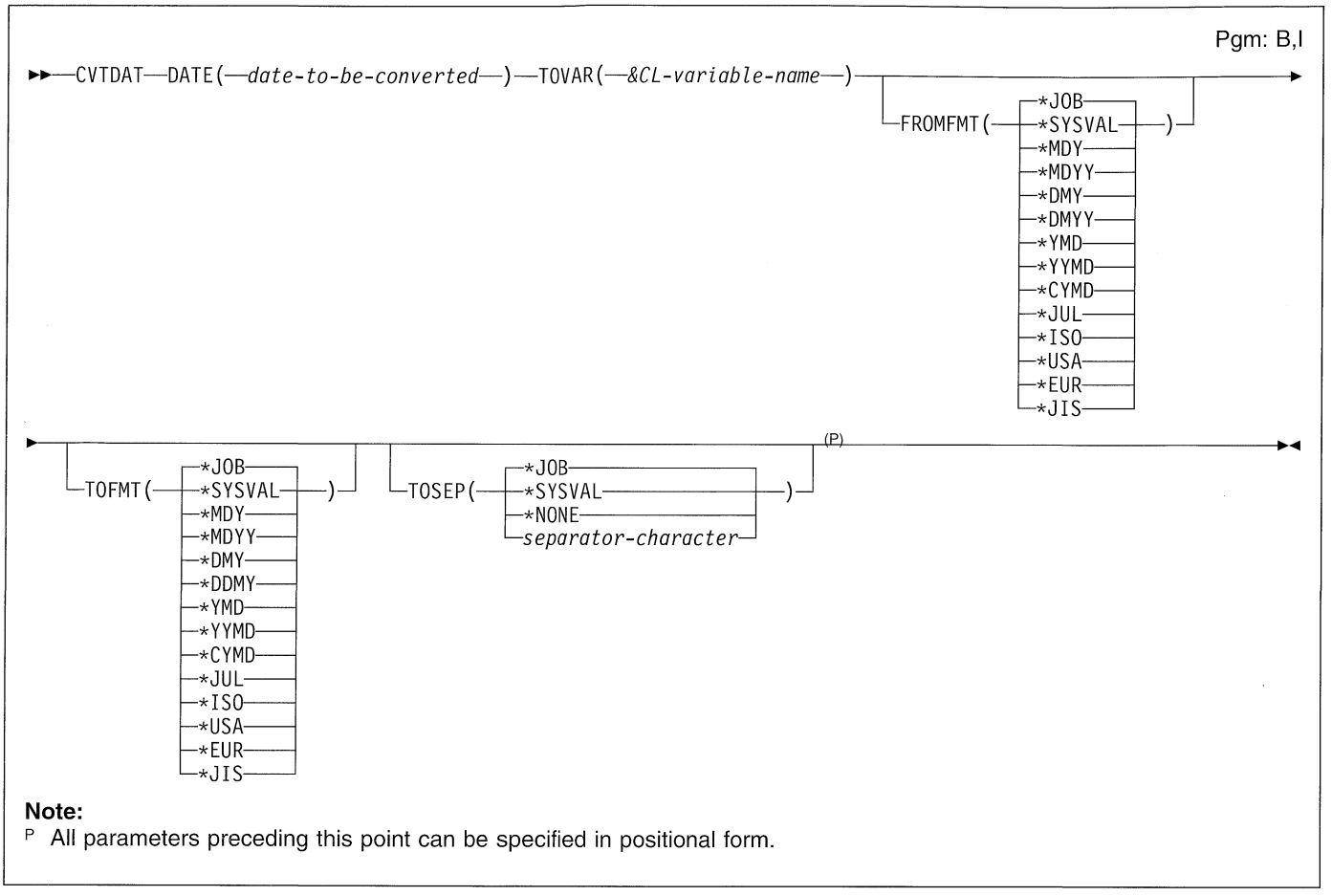
Note that the member name of the *converted* source member is the same as the member name of the unconverted source member in the file specified on the FROMFILE parameter.

Example

```
CVTCLSRC  FROMFILE(OLDLIB/FILEA)  TOFILE(NEWLIB/FILEB)
  FROMMBR(PGM1 PGM2 PGM3)
```

This command converts three members (PGM1, PGM2, PGM3) of a System/38 source file (FILEA) located in library OLDLIB, to an AS/400 system source file. The converted source file members are located in FILEB, in library NEWLIB. The converted members keep their original member names, PGM1, PGM2, and PGM3.

CVTDAT (Convert Date) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Convert Date (CVTDAT) command converts the format of a date value from one format to another, without changing its value. The command ignores any date separators used in the old format, but if separators are included in the converted result, a separator character can be specified on the command. Only valid dates can be converted. The command works in conjunction with the QLEAPADJ system value.

Restriction: This command is valid only within a control language (CL) program.

Required Parameters

DATE

Specifies the constant or CL variable that contains the date to be converted. When a constant is specified that contains separator characters, it must be enclosed in apostrophes (the separator characters are ignored in the conversion). If separators are used in a constant, leading zeros in each part of the date can be omitted (3/3/88 or 03/03/88 are both valid). If a variable is specified, it must be long enough to contain the date type and its date separators, if used. The valid date separators are the slash (/), hyphen (-), period (.), and comma (,).

TOVAR

Specifies the name of the CL variable to receive the converted date value. The variable must be declared with a minimum length of 8 characters if the converted date is to contain date separators (6 are required for the Julian format); the length must be at least 6 characters (5 for Julian) if the converted date does not contain date separators.

For every format except Julian, the month, day, and year subfields in the converted result are each 2 bytes in length, are right-justified, and (if necessary) a leading zero is used as a padding character to fill each 2-byte field. For the Julian format, the day field is 3 bytes long and padded with leading zeros (if necessary), and the year field is 2 bytes long.

Optional Parameters

FROMFMT

Specifies the current format of the date being converted.

***JOB:** The format specified on the job attribute DATFMT is used.

***SYSVAL:** The format specified by the system value QDATFMT is used.

***MDY:** The date has the month, day, year format, **mmddyy**.

***MDYY:** The date has the month, day, year format, **mmddyyyy**.

***DMY:** The date has the day, month, year format, **ddmmyy**.

***DMYY:** The date has the day, month, year format, **ddmmyyyy**.

***YMD:** The date has the year, month, day format, **yymmdd**.

***YYMD:** The date has the year, month, day format, **yyyymmdd**.

***CYMD:** The date has the century, year, month, day format, **cyymmdd**, where c is 0 for years 1940 through 1999 and is 1 for years 2000 through 2039.

***JUL:** The date has the Julian format, **yyddd**.

***ISO:** The date has the International Organization for Standardization (ISO) date format, **yyyy-mm-dd**.

***USA:** The date has the United States date format, **mm/dd/yyyy**.

***EUR:** The date has the European date format, **dd.mm.yyyy**.

***JIS:** The date has the Japanese Industrial Standard date format, **yyyy-mm-dd**.

TOFMT

Specifies the format to which the date is being converted.

***JOB:** The format specified on the job attribute DATFMT is used.

***SYSVAL:** The date is converted to the format specified by the system value QDATFMT.

***MDY:** The date format is converted to the month, day, year format, **mmddyy**.

***MDYY:** The date format is converted to the month, day, year format, **mmddyyyy**.

***DMY:** The date format is converted to the day, month, year format, **ddmmyy**.

***DMYY:** The date format is converted to the day, month, year format, **ddmmyyyy**.

***YMD:** The date format is converted to the year, month, day format, **yymmdd**.

***YYMD:** The date format is converted to the year, month, day format, **yyyymmdd**.

***CYMD:** The date format is converted to the century, year, month, day format, **cyymmdd**, where c is 0 for years 1940 through 1999 and is 1 for years 2000 through 2039.

***JUL:** The date format is converted to the Julian format, **yyddd**.

***ISO:** The date format is converted to the International Organization for Standardization (ISO) date format, **yyyy-mm-dd**.

***USA:** The date format is converted to the United States date format, **mm/dd/yyyy**.

***EUR:** The date format is converted to the European date format, **dd.mm.yyyy**.

***JIS:** The date format is converted to the Japanese Industrial Standard date format, **yyyy-mm-dd**.

TOSEP

Specifies the type of date separators, if any, used in the converted date.

***JOB:** The converted date has the separators specified by the job attribute DATSEP.

***SYSVAL:** The converted date has the separators specified by the system value QDATSEP.

***NONE:** No separator characters are contained in the converted date.

***BLANK:** A blank is used as the separator in the converted date.

separator-character: Specify the character that is used as the date separator in the converted date. The valid separator characters are the slash (/), hyphen (-), period (.), and comma (,).

Examples

Example 1: Converting to DMY Format

```
DCL   VAR(&DATE)  TYPE(*CHAR)  LEN(8)
  •
  •
  •
CVTDAT   DATE('12-24-88')  TOVAR(&DATE)  TOFMT(*DMY)
```

This command converts the date 12-24-88, which is in the MDY format. Because the FROMFMT parameter was not specified, its default, *JOB, indicates that the job attribute DATFMT contains the MDY format. The date is converted to the DMY format, and the separator character specified in the job attribute DATSEP is inserted. If DATSEP contains a slash, the converted result is 24/12/88.

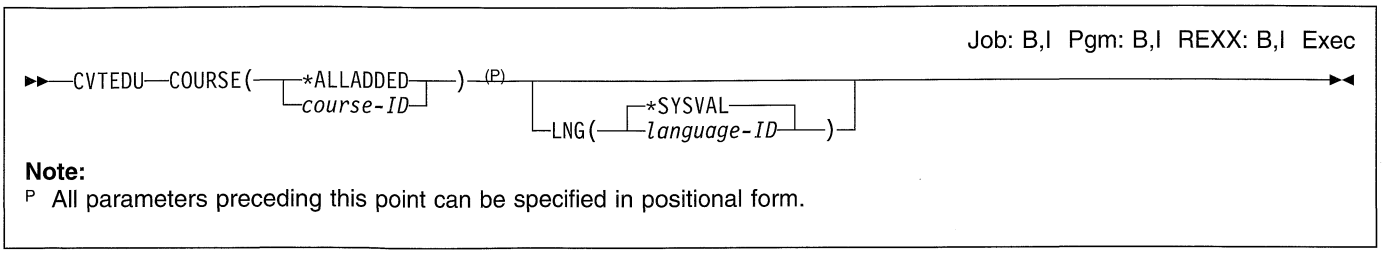
Example 2: Converting to Format Specified by Job Attribute

```
DCL   &PAYDAY  *CHAR  6
DCL   &NEWPDAY  *CHAR  6
  •
  •
  •
CVTDAT   DATE(&PAYDAY)  TOVAR(&NEWPDAY)
  FROMFMT(*YMD)  TOSEP(*NONE)
```

This command converts the format of the date stored in &PAYDAY from year, month, day to the format specified by

the job attribute DATFMT. If, for example, DATFMT contains the MDY format, the format of the converted date is month, day, and year. The converted date is stored in the variable &NEWPDAY. Because &NEWPDAY was declared as a 6-character variable, TOSEP(*NONE) is required; the converted result cannot include separator characters.

CVTEDU (Convert Education) Command



Purpose

The Convert Education (CVTEDU) command converts course modules from ASCII to EBCDIC.

Note: Courses that are valid only for personal computers cannot be converted with this command.

Required Parameters

COURSE
Specifies the name of the course library to be converted.

***ALLADDED:** All of the courses previously added to the system through the education administration system will be converted.

course-ID: Specify the name of the course library to be converted.

Optional Parameters

LNG
Specifies the language (indicated by the language identifier) of the course to be converted.

***SYSVAL:** The identifier of the current primary language for the system will be used.

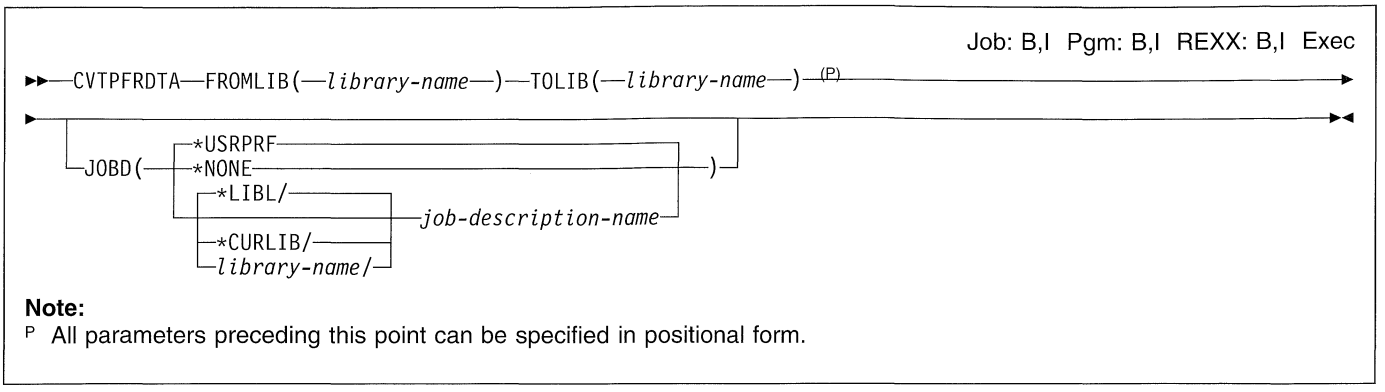
language-ID: Specify the identifier of the language.

Example

```
CVTEDU  COURSE(*ALLADDED)
```

This command converts all of the courses previously added through the education administration system from ASCII to EBCDIC.

CVTPFRDTA (Convert Performance Data) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Convert Performance Data (CVTPFRDTA) command converts performance data from a previous release to the format needed for processing by the current release of Performance Tools/400.

The command first determines the release level at which the data was collected. Then the members of all necessary files are converted. The conversion may be done in the same library where the current data resides. To avoid the risk of destroying the old data if the command ends abnormally, convert the data into a different library (TOLIB), and later, delete the data from the old library (FROMLIB).

If the conversion is done in a different library, the old data remains in the current library (FROMLIB) and the new data resides in the new library (TOLIB). If a new library is specified for the newly converted data, all files are copied to the new library, including those files which do not need to be converted.

Required Parameters

FROMLIB

Specifies the library that contains the files being converted.

TOLIB

Specifies the library where the converted files will be located.

Optional Parameters

JOBD

Specifies the job description used to submit jobs for batch processing. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***USRPRF:** The job description specified in the user's user profile is used.

***NONE:** A batch job is not submitted. Instead, processing continues interactively while the user waits. The user's work station is not available for other use during this time, which could be significant for long jobs.

The name of the job description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

job-description-name: Specify the name of an alternate job description.

Examples

Example 1: Submitting a Batch Job

```
CVTPFRDTA  FROMLIB(MIKE)  TOLIB(TERESA)
```

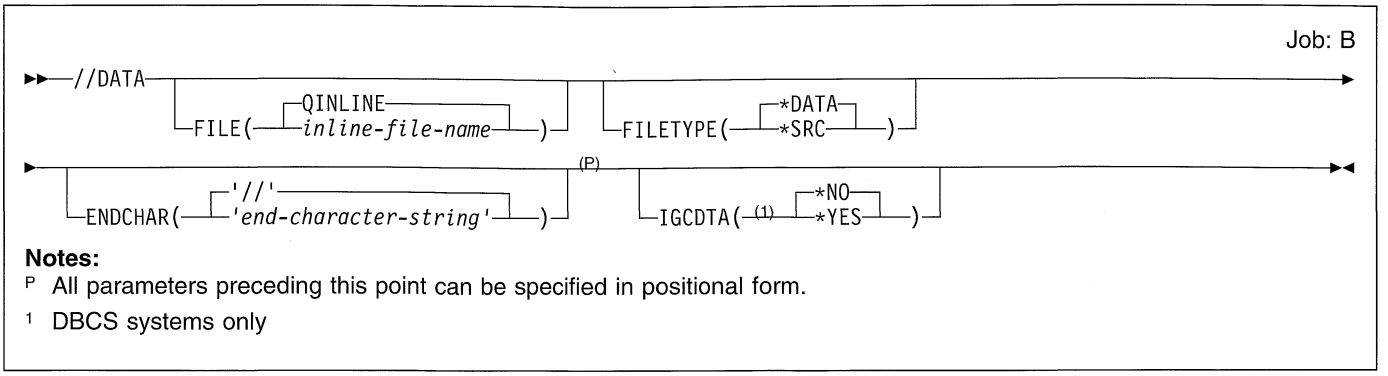
This command converts the performance data in library MIKE and places it in library TERESA.

Example 2: Submitting a Job Interactively

```
CVTPFRDTA  FROMLIB(QPFRDATA)
  TOLIB(QPFRDATA)  JOBD(*NONE)
```

This command converts the performance data in library QPFRDATA and places it in the same library after conversion is complete. This conversion occurs interactively while the user waits.

DATA (Data) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 DBCS systems only

Purpose

The Data (//DATA) command must be used to indicate the beginning of an inline data file in an input stream that is read by a spooling reader. It also specifies the delimiter that must be used to indicate the end of the data file. Inline data files exist only during the current job; they are destroyed after the current job is finished. Unnamed inline files can be used only once in the job.

Restrictions:

1. The DATA command cannot be used from a work station.
2. Two slashes must precede this command name when entering it in the data record, that is, //DATA.
3. The user can separate the slashes from this command name with blank spaces, for example, //bbDATA.

Optional Parameters

FILE

Specifies the name of the inline data file. This name is also specified in the program that processes the file.

QINLINE: The name of the inline data file is QINLINE. The file is processed as an unnamed inline file. An unnamed file can be processed if the program specifies QINLINE as the file name, or if the device file that specifies SPOOL(*YES) is opened for input. Unnamed inline files can be used only once by the job.

inline-file-name: Specify the name of the inline data file used by one or more programs in the job. The file is connected to the program when the program opens the file by specifying its file name. The job can get access to named inline data files more than once.

FILETYPE

Specifies whether the inline data following this command is put in the standard format for source files or in the data file format. The standard source file format is a sequence number (a 6-character source number) followed by the 6-character system date that precedes the data. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***DATA:** The diskette file describes data records.

***SRC:** The inline data is numbered in sequence. It is a source file that can be used to create another file or a program.

ENDCHAR

Specifies a string of characters used to indicate the end of an inline data file. To be recognized, the character string must begin in position 1 of the record. If you specify a character string other than by using // (the default value) as the delimiter, all records up to the end-of-file record (the record containing the specified character string starting in column 1) are treated as data. This allows you to imbed reader commands (//BCHJOB, //DATA, //ENDBCHJOB, or //ENDINP) in the data stream. The end-of-file record for ENDCHAR values (which are not default values) is not put to the data file, and it is not checked to see if it is a valid reader command. It is used only to determine the end of the data stream and then it is discarded.

'//': The default value is two slashes. The command works the same way whether two slashes are coded into the parameter or the parameter itself is defaulted.

Using the default, the slashes in positions 1 and 2 of a record (in either a data file or a source file) identify the first record beyond the file. Thus, the commands //BCHJOB, //DATA, //ENDBCHJOB, or //ENDINP also indicate the end of the inline file.

'end-character-string': A character string (up to 25 characters long and enclosed in apostrophes) can be entered to identify the last record in the file. The character string can contain both alphanumeric and special characters. If a character combination other than '//' is specified on the ENDCHAR parameter, reader commands can be safely imbedded in the data. The reader ignores all other data, including reader commands, while searching for the specified string.

IGCDTA

Specifies whether the file contains double-byte character set (DBCS) data.

***NO:** The file does not process DBCS data.

***YES:** The inline file can contain DBCS data.

Examples

Example 1: Inline Data File in Data File Format

```
//DATA  FILE(FILE1)
```

This command assigns the name FILE1 to the data that follows it, until an end of inline data condition is found (two slashes in positions 1 and 2).

Example 2: Specifying an End Character String

```
//DATA  FILE(FILE2)  ENDCHAR('STOPIT')
```

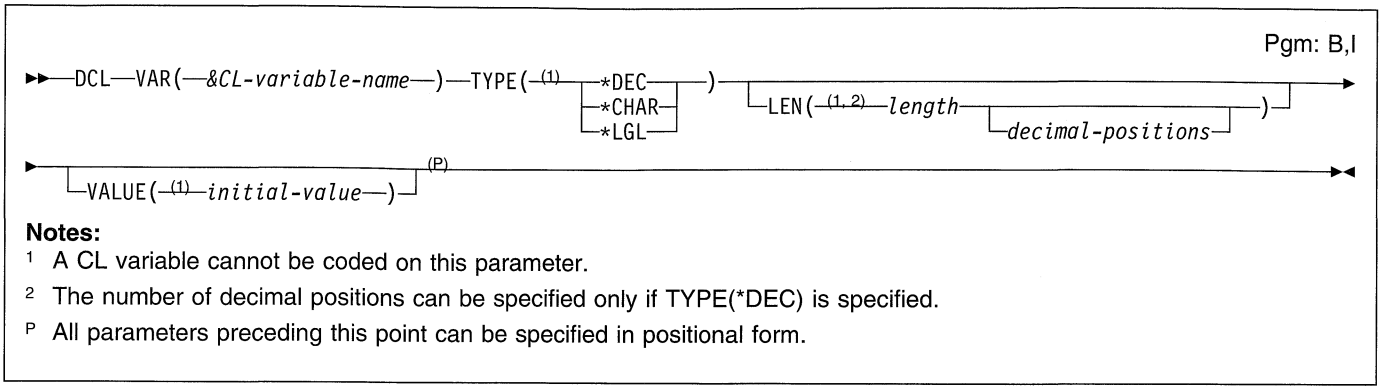
This command assigns the name FILE2 to the data following it; the file continues until a record is found that contains the characters STOPIT in positions 1 through 6. This delimiter allows the //BCHJOB, //ENDBCHJOB, and //DATA commands and records with // in positions 1 and 2 to be embedded in an inline file.

Example 3: Specifying a File Containing DBCS Data

```
//DATA  FILE(FILE3) IGCDTA(*YES)
```

This command assigns the name FILE3 to the data that follows it. This file can contain DBCS data.

DCL (Declare CL Variable) Command



Notes:
1 A CL variable cannot be coded on this parameter.
2 The number of decimal positions can be specified only if TYPE(*DEC) is specified.
P All parameters preceding this point can be specified in positional form.

Purpose

The Declare CL Variable (DCL) command defines control language (CL) program variables used in CL programs. CL variables are used to store and update data and to receive parameters from another program on a call. CL variables are known by name only within the program that declares them. They cannot be used outside a CL program except when they are referred to by some commands (such as the DSPPGMVAR command) used for debugging programs. If a variable is declared, but not referred to by another command in a CL program, the variable is not included in the program when it is compiled. However, the value in the variable can be passed to another program as a parameter. Each DCL command defines the attributes of one CL variable and declares its name in the program in which it is used.

Each CL variable in a program must be identified by one of the two declare commands. The Declare File (DCLF) command declares CL variables for display device files and data areas. The DCL command declares all other CL variables.

Restriction: The DCL command is valid only within a CL program. All declare commands (DCL and DCLF) must follow the PGM (Program) command and must precede all other commands in the program. The two types of declare commands can be intermixed in any order.

Required Parameters

VAR

Specifies the name of the CL variable being declared in the CL program. The variable exists only within the program in which it is defined. It can be passed as a parameter on a call to another program, in which case it can be processed by the called program. The variable name must start with an ampersand (&).

TYPE

Specifies the type of value contained in the CL variable being declared. The value of the variable can be a character constant, a decimal constant, or a logical '1' or '0'. The value for this parameter cannot be specified by a CL variable. Specify one of the following types:

***DEC:** A decimal variable that contains a packed-decimal value.

***CHAR:** A character variable that contains a character string value.

***LGL:** A logical variable that contains a logical value of either '1' or '0'.

Optional Parameters

LEN

Specifies the length of the CL variable being declared. If the variable is a decimal value, the number of decimal digits to the right of the decimal point can be optionally specified. The value for this LEN parameter cannot be specified by a CL variable. The maximum lengths and the defaults for each of the three types are shown in Table 41.

length: Specify the number of characters that the value in this CL variable can have. The length cannot be greater than the maximum for this type of variable.

decimal-positions: This option is valid only for *decimal* variables. The length of the value in the variable includes the number of decimal positions in the value. The maximum length of the decimal value is 15 digits, including the digits to the right of the decimal point. Up to nine decimal positions can be specified. If nine decimal positions are specified, the value to the *left* of the decimal point can never be greater than 999,999 because only 6 of the 15 digits are left for the integer value.

If a length (in digits) is specified for a decimal variable and the number of decimal positions is not specified, 0 decimal positions is assumed.

VALUE

Specifies the initial value assigned to the CL variable when it is declared in the program. The value must be of the type specified by the TYPE parameter. If no value is specified, a character variable is set to blanks, a

decimal variable is set to a value of 0, and a logical variable is set to '0'. The value for the VALUE parameter cannot be specified by a CL variable.

If the name of the declared variable is specified on the PARM parameter of the PGM command in the same program in which the variable is declared, an initial value *cannot* be specified for the variable. In that case, the variable receives its value from the calling program.

Table 41. Variable Lengths

Type	Maximum Length	Default Length[1]
Decimal	15 digits, 9 decimal positions	15 digits, 5 decimal positions
Character	9999 characters[2]	32 characters
Logical	1 character	1 character

1 For decimal and character types, the default length is the same as the length of the initial value, if one is specified in the VALUE parameter.
2 The initial value (specified on the VALUE parameter) of a CL variable can be no greater than 3000 characters.

Examples

Example 1: Specifying the CL Variable Length

```
DCL   &ABLE  *DEC  LEN(5 2)
```

This command declares a CL variable named &ABLE that contains a decimal value. The value can never be greater than 999.99 because LEN specifies up to 5 digits, of which two are to the right of the decimal point. Because the VALUE parameter was not specified, and it is a numeric value, &ABLE is set to a value of zero (000.00).

Example 2: Specifying a Logical Value

```
DCL   &SWITCH   *LGL
```

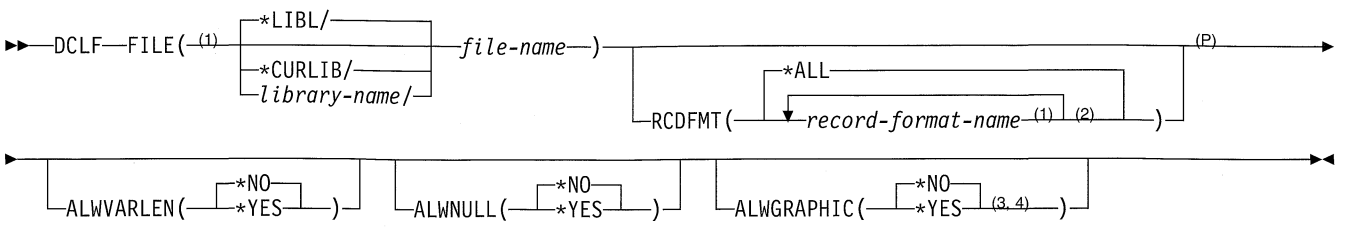
This command declares a CL variable named &SWITCH to contain a logical value. Because the type parameter specifies logical, the variable is one character long and it is set to '0'.

Example 3: Specifying Initial Value of CL Variable

```
DCL   &FILNAM   *CHAR   VALUE(FILEA)
```

This command declares a CL variable named &FILNAM whose value is FILEA. Because the value contains 5 characters and the LEN parameter was not specified, the length of the variable is also 5 characters.

DCLF (Declare File) Command



Notes:

1 CL variables cannot be coded on this parameter.

2 A maximum of 50 repetitions

P All parameters preceding this point can be specified in positional form.

3 The maximum length is 9997 bytes.

4 ALWVARLEN(*YES) must be specified if varying graphic data fields are used.

Purpose

The Declare File (DCLF) command declares one file (by name) to a control language (CL) program. Only one DCLF command is allowed in a CL program. The command specifies the name of the file and the record formats to be used in the program. After the file is declared, the program can contain the data manipulation commands (Send File (SNDF), Receive File (RCVF), Send/Receive File (SNDRCVF), End Receive (ENDRCV), and Wait (WAIT) for display device files, or RCVF for database files) that reference the file. This enables the program to interact with its user by sending data to and receiving data from a work station or database file. Database files can be used for input processing only.

When the CL program is compiled, a CL variable is automatically declared for each field in each record format used in the program. If the file is a nonfield level database file, the record format contains one field with the name of that record format. The field name becomes the variable name prefixed with an ampersand (&). The attributes of each declared field are the same as the attributes of the fields in the file. Fields defined in the record format as numeric are defined as decimal variables. Also, indicators used in the referenced file are declared as logical variables in the form &INnn, in which nn is the indicator number.

Variables automatically declared by the DCLF command can be used in the program the same as the variables declared by a DCL command. For example, indicators can be used in expressions and IF statements because they are declared as logical variables.

The content of the variables, not the variable names, are seen by the user; the display shows one, some, or all of the fields in the record format that can be filled in by the user. DDS determines the display format.

Restrictions:

1. This command is valid only within CL programs. All declare commands (Declare CL Variable (DCL) and Declare File (DCLF)) must follow the Program (PGM) command and precede any other command. The two types of declare commands can be placed in any order.
2. The file must either be a database file with only one record format or be a display file.
3. The file cannot be a mixed file, even if only display devices are defined for that mixed file.
4. The database file can be either physical or logical, and can be either field-level or nonfield level.
5. A file must exist before the program is created.

Because CL variables are automatically declared for each field in a referenced file's record formats, the following restrictions apply:

- If the file is changed (and the file description specifies that level checking is to be performed), the CL program must be recompiled to match the new file description. More information on level checking is in the *Database Guide* and the *Guide to Programming Displays*.
- If any field name is defined in more than one record format of the display file, the attributes in each record format for the commonly named field must match.
- Any CL variable declared in the program by a DCL command with the same name as an automatically declared CL variable (for a referenced field) must also have the same attributes specified in DDS for the referenced field.
- The variables used in the file must have data types supported for CL variables. However, fields defined as packed decimal format, zoned decimal format, or binary format are declared as decimal variables. The variables used in the file must have attributes that are valid for CL variables. For decimal variables, the limits are 15 digits and 9 decimal positions. For character variables, the limit is 9999 characters.

Required Parameters

FILE

Specifies the qualified name of the file used by the CL program. A CL variable name cannot be used to specify the file name.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file used by the CL program.

Optional Parameters

RCDFMT

Specifies the names of one or more file record formats used by the SNDF, RCVF, and SNDRCVF commands in the CL program. Database files can be processed only by RCVF. CL variable names cannot be specified in RCDFMT; only names of record formats can be used. For every field and indicator in each record format specified in RCDFMT, one CL variable is automatically declared in the program.

***ALL:** Every record format in the file, up to 99, has its fields declared in the CL program as variables. If there are more than 99 record formats in the file, only the first 99 are used.

record-format-name: Specify one or more record format names whose fields are being declared as variables in the CL program. No more than 50 record format names can be specified; CL variables cannot be used to specify the names.

ALWVARLEN

Specifies whether variable length fields are allowed in record formats.

***NO:** Record formats cannot contain variable length fields. A message is sent at compile time if variable-length fields are supported in the file.

***YES:** Record formats can contain variable length fields. CL variables declared for variable-length fields are handled as type *CHAR with length equal to 2 bytes plus the maximum field length. Following a RCVF on a variable-length field, the first 2 bytes in the CL variable contain the length of the data. The data received from the field is padded on the right with blanks to the maximum length allowed (9997 bytes).

ALWNULL

Specifies whether a field value of null is allowed.

***NO:** Values of null are not allowed. For each field containing a null value at RCVF time, a diagnostic message is sent with a single escape message for the entire record. Default values are placed in the CL program variables.

***YES:** Values of null are allowed.

ALWGRAPHIC

Specifies whether graphic data fields are allowed in record formats.

***NO:** Record formats cannot contain graphic data fields. A diagnostic message is sent at compile time if graphic data fields are supported in the file.

***YES:** Record formats can contain graphic data fields. CL variables declared for graphic data fields are handled as type *CHAR with length equal (in bytes) to the graphic data field length.

Examples

Example 1: Declaring Fields of All Record Formats as Variables

```
DCLF   FILE(ABLE)
```

This command specifies that the file named ABLE is used by the CL program to pass data between the user and the program. Because no library was specified, the library list is used to locate the file. RCDFMT was not specified; therefore, all the fields and indicators in all the record formats are automatically declared as variables, and data from any field in any record format (up through the first 99) in the file can be passed between the program and the user. No device file routing data is received by the program.

Example 2: Using Multiple Record Formats

```
DCLF   FILE(BAKER)  RCDFMT(REC2 REC6)
```

Display file BAKER is used by the CL program to pass data between the user and the program. Because no library was specified, the library list is used to locate the file. Both the REC2 and REC6 record formats are used. Device file routing data is returned to the program and stored in the CL variable &FB for each RCVF and SNDRCVF command.

Additional Considerations

File processing is handled differently in CL programs, depending on whether the file specified in the DCLF command is a display file or a database file.

The following statements apply if the file is a display device file at compile time:

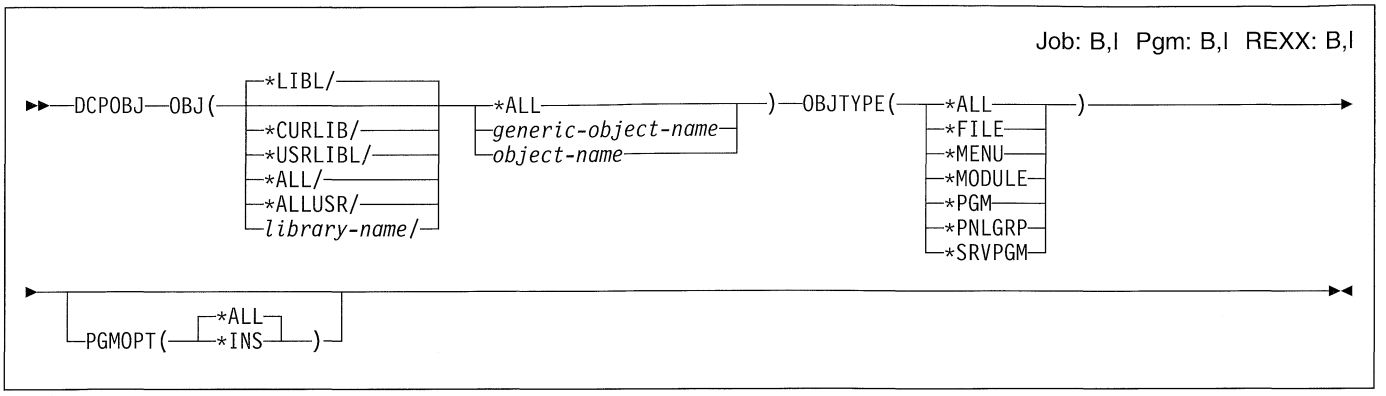
- The file must be a display device file at run time.

- The program may contain any or all of the following commands: SNDF, RCVF, SNDRCVF, ENDRCV, and WAIT.
- The file is opened for input and output.

The following statements apply if the file is a database file at compile time:

- The file must be a database file with a single record format at run time.
- The program may contain only RCVF commands; SNDF, SNDRCVF, ENDRCV, and WAIT commands are not allowed.
- The file is opened for input only.
- The file is opened by the RCVF command, not the OPNDBF command.

DCPOBJ (Decompress Object) Command



Purpose

The Decompress Object (DCPOBJ) command decompresses programs, panel groups, menus, display files, printer files, modules, and service programs.

- *Compressed Objects* are objects that consume less storage space than decompressed objects. When a compressed object is used or a compressed program is called, a decompressed version of the object automatically becomes available to the user.
- *Decompressed Objects* are objects that use the system storage space allocated to them and are in a final, ready-to-use state.
- *Temporarily Decompressed Objects* are temporarily decompressed copies of compressed objects. The system allocates storage space for the temporary copies until the system or the user determines that the temporary storage space needs to be reclaimed.

  Temporary storage is automatically reclaimed when:

  - The RCLTMPSTG command is run
  - The next initial program load (IPL) is run
  - The object is used often enough to cause the system to permanently decompress it

  When an object is permanently decompressed, the compressed version of the object is destroyed as well as any temporary forms of the object; however, compressed versions remain intact as long as the objects are temporarily decompressed.

Restrictions: The user must have object operational and read authorities (*USE) to the objects specified on the command and read authority to the library containing the objects.

Required Parameters

OBJ

Specifies the qualified name of the object to be decompressed.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All compressed objects in the specified library are decompressed.

generic-object-name:* Specify the generic name of the object. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic

(prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be decompressed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

object-name: Specify the name of the object that is decompressed.

OBJTYPE

Specifies the type of the object to be decompressed.

***ALL:** All compressed menus, panel groups, display and printer device files, programs, modules, and service programs with the specified object name are decompressed.

***FILE:** The compressed file with the specified object name is decompressed (display and printer files only).

***MENU:** The compressed menu with the specified object name is decompressed.

***MODULE:** The compressed module with the specified object name is decompressed.

***PGM:** The compressed program with the specified object name is decompressed.

***PNLGRP:** The compressed panel group with the specified object name is decompressed.

***SRVPGM:** The compressed service program with the specified object name is decompressed.

Optional Parameters

PGMOPT

Specifies, for *PGM and *SRVPGM objects, the program option that indicates whether the entire program (instruction stream and observability tables) or only the instruction stream is decompressed.

***ALL:** The entire program or service program is decompressed.

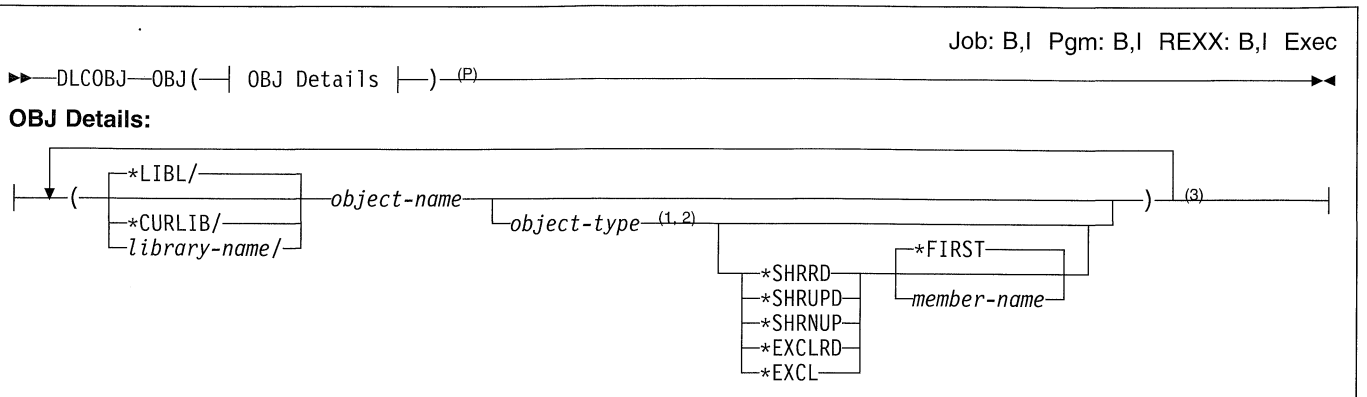
***INS:** Only the instruction stream of the program or service program is decompressed.

Example

```
DCPOBJ    OBJ(QGPL/*ALL)  OBJTYPE(*FILE)
```

This command decompresses all compressed display and printer files in library QGPL.

DLCOBJ (Deallocate Object) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 Refer to the OBJ parameter description for the lock states that are valid for each object type.

2 A list of the valid OS/400 object types for this command is in Appendix A.

3 A maximum of 50 repetitions

Purpose

The Deallocate Object (DLCOBJ) command releases the allocations of the specified objects for the specified lock states. The objects, allocated earlier in the same routing step by one or more Allocate Object (ALCOBJ) commands, are freed for use by other jobs. If the DLCOBJ command is not used, the objects are automatically deallocated at the end of the routing step. If the DLCOBJ command is used when a lock does not exist, no error occurs.

The DLCOBJ command should not be issued for an object that was not explicitly allocated by the ALCOBJ command. If the DLCOBJ command is used this way, internal locks on the object are released, making it possible for the object to be deleted, and for system failures to occur.

To release more than one lock on an object with a single DLCOBJ command, the object name, type, and lock state must be repeated in the list for each lock being released.

This command cannot be used to deallocate peer device descriptions.

When deallocating distributed data management (DDM) files, additional time is required for the command to complete because of the time required for establishing communication and for deallocating files on remote systems.

Required Parameters

OBJ

Specifies the qualified name of one or more operating system objects that are deallocated from the job, the type of each object deallocated, the lock state of each object, and, if the object is a database file or DDM file, an optional member name.

Only some of the operating system object types can be specified on the DLCOBJ command. Of these, some cannot use all of the lock states. Refer to the lock state table under the OBJ parameter of the ALCOBJ command for lock state applicability information.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

Element 1: Name of the Object to be Allocated

object-name: Specify the name of the object that is allocated.

Element 2: Type of Object to be Allocated

object-type: Specify the type of the object to be allocated to the job. Valid object types include the following:

Value	Type of OS/400 Object
*AUTL	Authorization list
*CLD	C locale description
*CLS	Class
*CSI	Communications side information
*CSPMAP	Cross-system product map
*CSPTBL	Cross-system product table
*DEVD	Device description
*DTAARA	Data area
*DTADCT	Data dictionary
*DTAQ	Data queue
*FCT	Forms control table
*FILE	File
*FNTRSC	Font resource
*FORMDF	Form definition
*LIB	Library
*MENU	Menu
*MSGQ	Message queue
*OVL	Overlay
*PAGSEG	Page segment
*PDG	Print descriptor group
*PGM	Program
*PNLGRP	Panel group
*QMFORM	Query management form
*QMQRY	Query management query
*SBSD	Subsystem description
*SCHIDX	Search index
*SPADCT	Spelling aid dictionary
*S36	S/36 machine description
*USRIDX	User index
*USRQ	User queue
*USRSPC	User space

Element 3: Lock States for the Object to be Allocated

lock-state: Specify the lock state for the object. Valid lock states include the following:

Value	Lock State Meaning
*SHRRD	Shared for read
*SHRUPD	Shared for update
*SHRNUP	Shared, no update
*EXCLRD	Exclusive, allow read
*EXCL	Exclusive, no read

Multiple locks can be specified for the same object in the same job with duplicate or different lock states. Each lock is held separately. For example, if an *EXCL lock is already held for an object, and a second *EXCL lock request occurs, the second lock is also acquired. Both locks must be separately released in the job (deallocated with the DLCOBJ command) before another job can get access to the same object. If an object is already allocated with one lock state and the user wants to use a different lock state, first use the ALCOBJ command to request the new lock with the desired lock state and then use the DLCOBJ command to release the old lock (with the old lock state).

Table 42, at the end of this description, shows the system object types that can be specified and the lock states allowed for each object type (A = allowed).

When an exclusive lock is requested on a logical file member, the lock occurs on both the logical file member and the associated physical file members. No other user can use the physical file members (not even through some other logical file member).

The *CL Programmer's Guide* has an explanation of each lock state.

Element 4: Member of the Database File to be Allocated

Note: The following values can only be specified if the object type is a database file.

***FIRST:** The first member of the database file is allocated to the job.

member-name: Specify the name of member to be allocated to the job. If the specified file is a logical file, the physical file members associated with the members of the logical file are also allocated to the job.

Table 42. Lock States

Object Type	*EXCL	*EXCLRD	*SHRUPD	*SHRNUP	*SHRRD
*AUTL	A	A	A	A	A
*DEVD[1]		A			
*DTAARA	A	A	A	A	A
*DTADCT	A	A	A	A	A
*DTAQ	A	A	A	A	A
*FCT	A	A	A	A	A
*FILE[2]	A	A	A	A	A
*LIB		A	A	A	A
*MENU	A	A	A	A	A
*MSGQ	A				A
*PNLGRP	A	A	A	A	A
*PGM	A	A			A
*SBSD	A				
*SCHIDX	A	A	A	A	A
*SSND	A	A	A	A	A
*S36	A	A	A	A	A

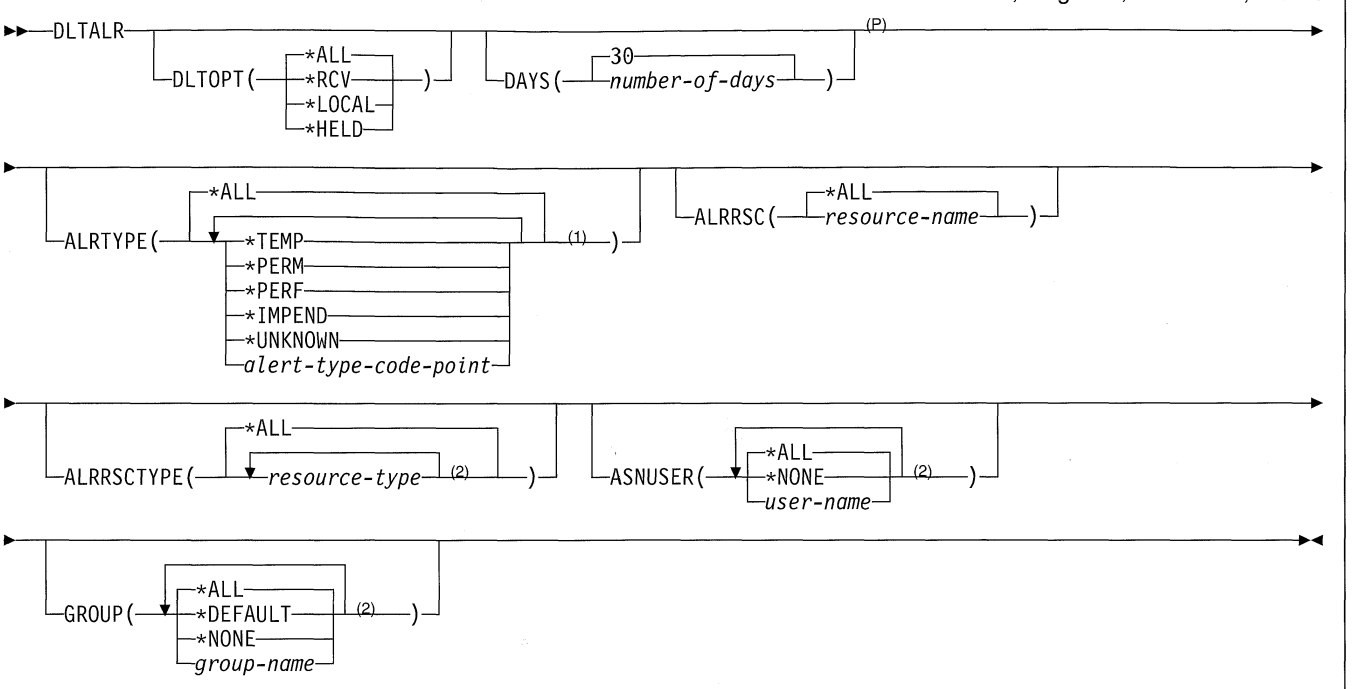
1 APPC and INTRA device descriptions cannot be allocated on the ALCOBJ command.
2 Only database file members can be allocated.

Example

```
DLCOBJ  OBJ((LIBB/FILEA *FILE *SHRRD))
```

This command releases the shared-for-read allocation of the first member of file FILEA in library LIBB.

DLTALR (Delete Alert) Command



Notes:

1 A maximum of 5 repetitions

2 A maximum of 50 repetitions

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Alert (DLTALR) command deletes one or more alerts from the alert data base.

Optional Parameters

DLTOPT

Specifies which alerts are being deleted.

***ALL:** Alerts that meet the selection criteria specified by the remaining parameters are deleted.

***RCV:** Only alerts received from other systems are deleted. Selection criteria for the alerts received can be further specified on the remaining parameters.

***LOCAL:** Only locally created alerts are deleted. Selection criteria for the local alerts can be further specified on the remaining parameters.

***HELD:** Alerts that cannot be sent to the system's focal point and that are marked HELD are deleted. Selection criteria for the held alerts can be further specified on the remaining parameters.

Note: There is a distinction between held alerts that are sent or forwarded by this system, and held alerts that are received by another system. DLTOPT(*HELD) deletes only held alerts that could not be sent or forwarded by this system (alerts that are currently being held based on the ALRHLDCNT network attribute).

DAYS

Specifies that alerts older than this value are deleted. Any alert that is more recent than this value is not deleted.

30: Alerts over 30 days old are deleted.

number-of-days: Specify the number of days after which all alerts are deleted.

ALRTYPE

Specifies which types of alerts are deleted. The alert type indicates the severity of the alert.

***ALL:** All types of alerts are deleted.

***TEMP:** Alerts reporting a temporary problem are deleted.

***PERM:** Alerts reporting a permanent problem are deleted.

***PERF:** Alerts reporting a performance problem are deleted.

***IMPEND:** Alerts reporting an impending problem are deleted.

***UNKNOWN:** Alerts reporting a problem with unknown severity are deleted.

alert-type-code-point: Specify the code point for the alert type. The code point is specified by two (2) hexadecimal digits.

ALRRSC
Specifies the name of resources that are reporting problems.

***ALL:** All resources are reporting problems.

resource-name: Specify the names of the resources reporting problems.

ALRRSCTYPE
Specifies the types of resources that are reporting problems. Each resource name has an associated resource type. For example, diskette (DKT) and tape (TAP) are resource types.

***ALL:** All types of resources are reporting problems.

resource-type: Specify the resource type that is reporting problems.

ASNUSER
Specifies the user to which the alerts being deleted are assigned. This value is taken from the value on the ASNUSER parameter in the Add Alert Action Entry (ADDALRACNE) command.

***ALL:** All alerts are deleted.

***NONE:** The alerts not assigned to a user are deleted.

user-name: Specify the name of the user to which the alerts being deleted are assigned.

GROUP
Specifies the group to which the alerts being deleted are assigned. This value is taken from the value on the GROUP parameter in the Add Alert Selection Entry (ADDALRSLTE) command.

***ALL:** All alerts are deleted.

***DEFAULT:** The alerts assigned to the default group are deleted.

***NONE:** The alerts not assigned to a group are deleted.

group-name: Specify the name of the group to which the alerts being deleted are assigned.

Examples

Example 1: Deleting Temporary and Permanent Alert Types

```
DLTALR  DLTOPT(*LOCAL)  DAYS(10)
        ALRTYPE(*TEMP *PERM)  ALRRSCTYPE(DKT)
```

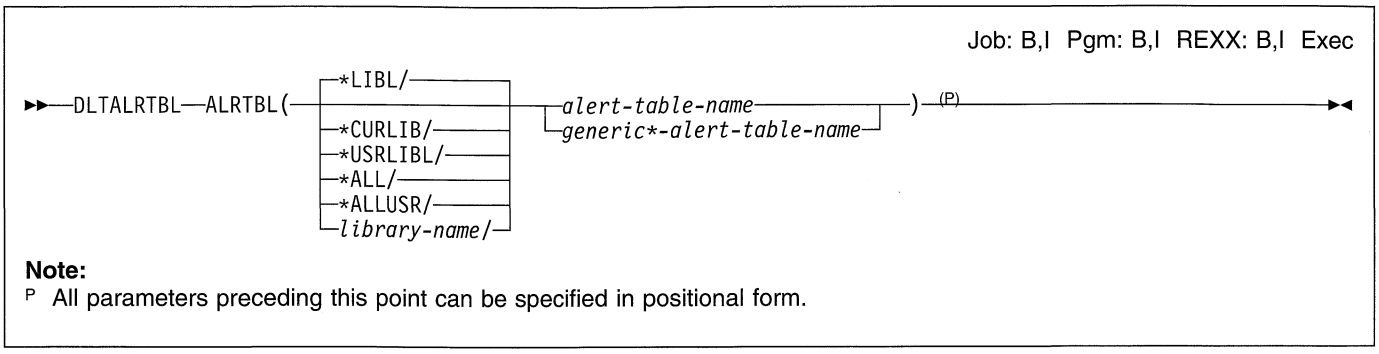
This command deletes temporary and permanent locally created alerts in the alert database. The alerts that are deleted are reporting problems about diskettes. Alerts more than 10 days old that match these criteria are deleted.

Example 2: Deleting Alerts Associated with Diskette Resources

```
DLTALR  DLTOPT(*RCV)  DAYS(0) ALRRSCTYPE(DKT)
```

This command deletes received alerts associated with diskette resources.

DLTALRTBL (Delete Alert Table) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Alert Table (DLTALRTBL) command deletes an alert table from the specified library. More information on the deletion of alert tables is in the *Alerts and DSNX Guide*.

Required Parameters

ALRTBL

Specifies the qualified name of the alert table being deleted.

The name of the alert table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

alert-table-name: Specify the name of the alert table being deleted.

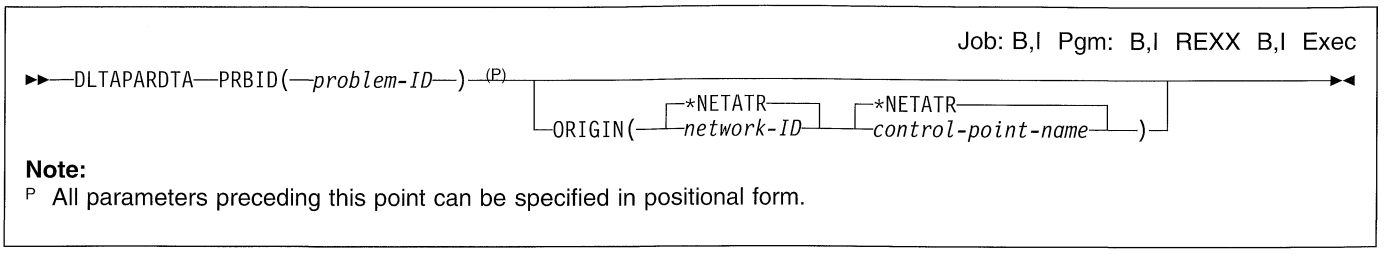
generic-alert-table-name:* Specify the generic name of the alert table being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTALRTBL ALRTBL(ALRTBLLIB/ALRTBLNBR1)
```

This command deletes alert table ALRTBLNBR1 from library ALRTBLLIB.

DLTAPARDTA (Delete APAR Data) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete APAR Data (DLTAPARDTA) command deletes the Authorized Problem Analysis Report (APAR) library and any data created by the Restore APAR Data (RSTAPARDTA) command. This command also updates the problem log entry with new information and removes the reference to the APAR library.

Restrictions: The following user profiles have authority to this command:

- QPGMR
- QSYSOPR
- QSRVBAS
- QSRV

Required Parameters

PRBID
Specifies the identifier (ID) of the problem for which APAR data is deleted.

Optional Parameters

ORIGIN
Specifies the network ID and the control point where the problem occurred.

Element 1: Network ID

***NETATR:** The problem originated on a system with the same local network ID as the one defined on the network attributes of this system.

network-ID: Specify the network ID of the system where the problem originated.

Element 2: Control Point Name

***NETATR:** The problem originated on a system with the same control point name as the one defined on the network attributes of this system.

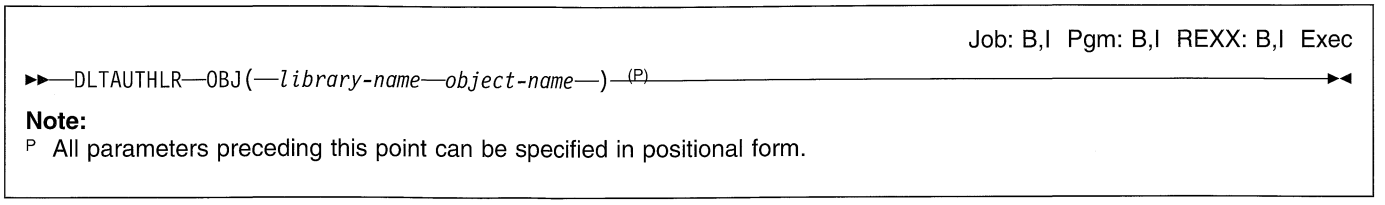
control-point-name: Specify the control point of the system where the problem originated.

Example

```
DLTAPARDTA  PRBID(9202448748)
```

This command deletes an APAR library and the APAR data for the problem ID 9202448748.

DLTAUTHLR (Delete Authority Holder) Command



Purpose

The Delete Authority Holder (DLTAUTHLR) command allows the user to delete an authority holder that secures an object type *FILE. The file must be a program-described database file.

Restriction: You must be the security officer or have all (*ALL) authority to the object to delete the authority holder.

Required Parameter

OBJ

Specifies the qualified name of the authority holder to be deleted.

The name of the object can be qualified by one of the following library values:

library-name: Specify the name of the library to be searched.

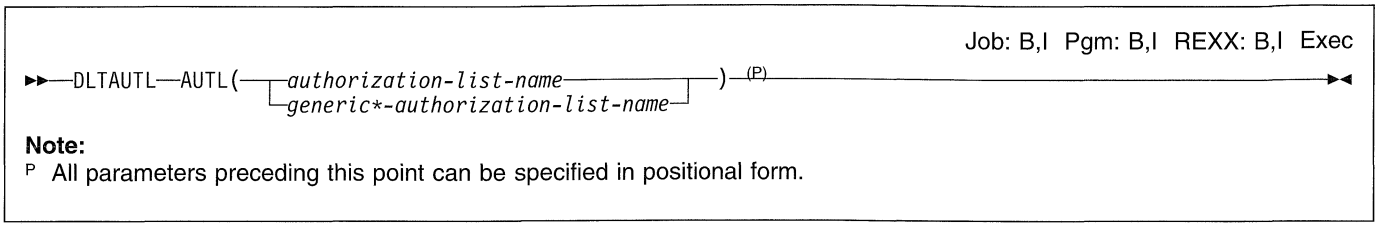
object-name: Specify the name of the authority holder.

Example

```
DLTAUTHLR   OBJ(QGPL/FIL1)
```

This command deletes the authority holder for FIL1 in the QGPL library.

DLTAUTL (Delete Authorization List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Authorization List (DLTAUTL) command allows a user to delete an authorization list. Authorization lists cannot be deleted if they are being used to secure an object. The user deleting an authorization list must have all (*ALLOBJ) authority or be the owner of the authorization list.

Required Parameter

AUTL

Specifies the name or generic name of an authorization list being deleted.

authorization-list-name: Specify the name of the authorization list being deleted.

generic-authorization-list-name:* Specify the generic name of the authorization list being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting an Authorization List

```
DLTAUTL   AUTL(PROGMR)
```

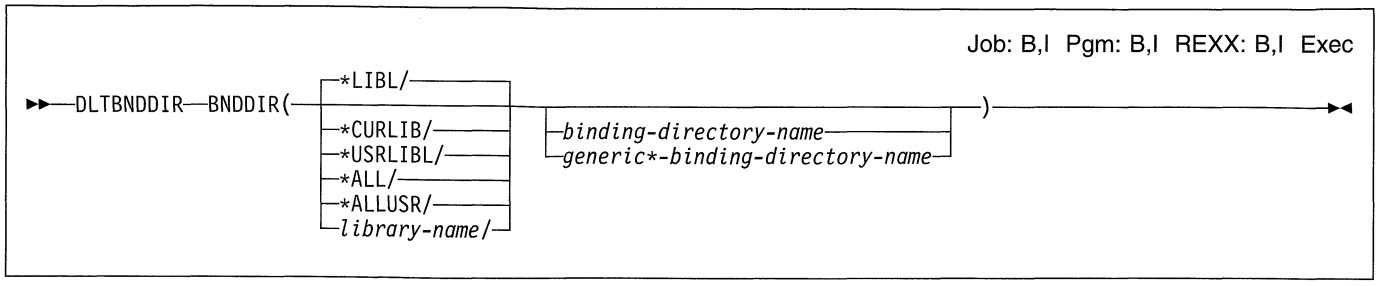
This commands deletes the PROGMR authorization list.

Example 2: Deleting Generic Authorization Lists

```
DLTAUTL   AUTL(FR*)
```

This commands deletes all authorization lists starting with the letters FR.

DLTBNDDIR (Delete Binding Directory)



Purpose

The Delete Binding Directory (DLTBNDDIR) command deletes a binding directory.

Restriction: You must have *USE authority for the library where the binding directory is being deleted and object existence authority to the binding directory.

Required Parameters

BNDDIR

Specifies a binding directory name or a group of binding directories to be deleted.

The name of the binding directory can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

binding-directory-name: Specify the name of the binding directory to be deleted.

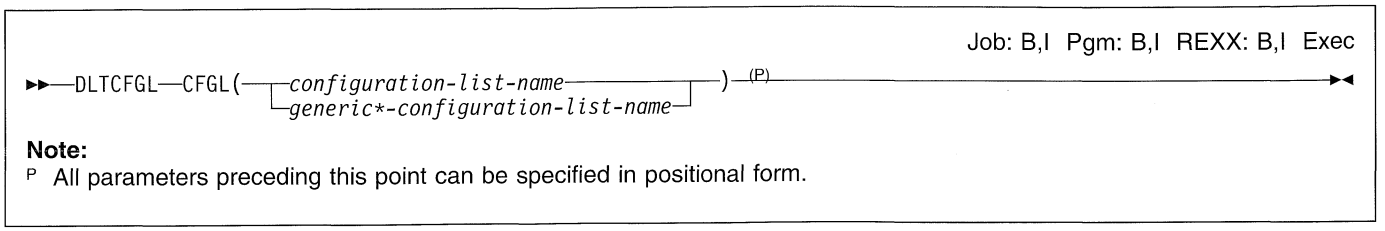
generic-binding-directory-name:* Specify the generic name of the binding directory being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, then all binding directories with names that begin with the generic name, and for which the user has authority, are deleted. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete binding directory name.

Example

```
DLTBNDDIR   BNDDIR(DISPLAYS)
```

This command deletes the binding directory named DISPLAYS from the library list.

DLTCFGL (Delete Configuration List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Configuration List (DLTCFGL) command deletes one or more configuration lists.

Required Parameters

CFGL

Specifies the names of the configuration lists being deleted.

configuration-list-name: Specify the name of the configuration list to be deleted. Use the Work with Configuration List (WRKCFGL) command to get a list of configuration lists on the system.

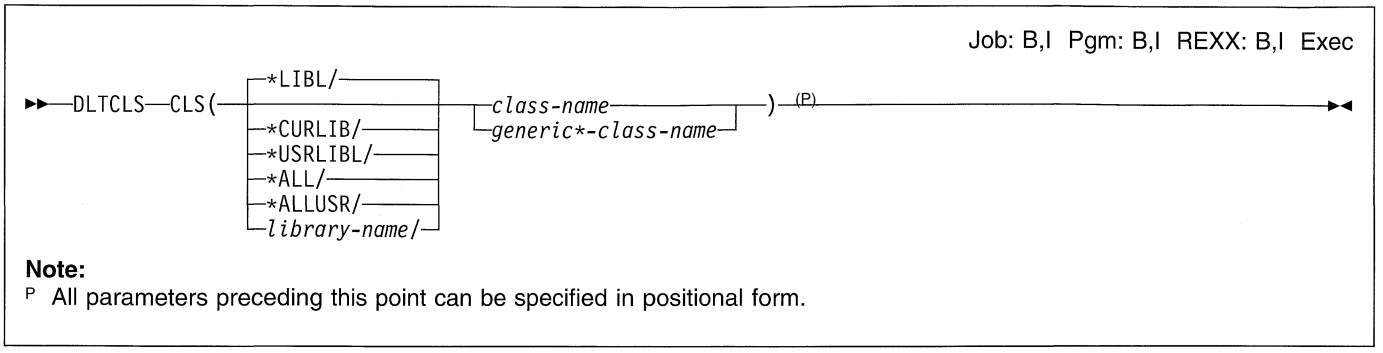
generic-configuration-list-name:* Specify the generic name of the configuration list(s) to be deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCFGL   CFGL(CONFIG01)
```

This command deletes the configuration list CONFIG01 from the system.

DLTCLS (Delete Class) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Class (DLTCLS) command deletes a class object or a group of class objects from the system. Routing steps that are run by using the class are not affected by its deletion. However, additional routing steps using the deleted class cannot be started. If the deleted class is referred to in an existing routing entry, either the routing entry should be changed (to refer to a different class) or another class with the same name should be created. If the subsystem is active when the class is deleted, errors might occur in the subsystem.

Restriction: The user must have object existence authority for the class being deleted.

Required Parameters

CLS

Specifies the name and library of the class descriptions being deleted. A specific class description or a generic class description can be specified; either type can be optionally qualified by a library name.

The name of the class description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

class-name: Specify the name of the class description being deleted.

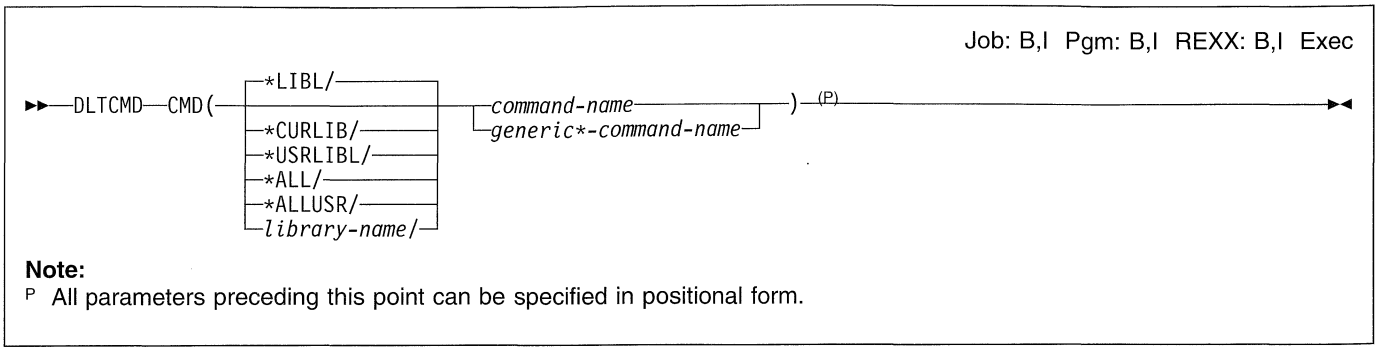
generic-class-name:* Specify the generic name of the class. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCLS  CLS(CLASS1)
```

This command deletes the class named CLASS1 from the system.

DLTCMD (Delete Command) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Command (DLTCMD) command deletes a user-defined command (or group of commands) from the library where it is located. Only the command definition object is removed; the command definition source, the command processing program, and the validity checker are not affected.

Restriction: The user of this command must have object existence authority for the command being deleted.

Required Parameters

CMD

Specifies the qualified name of the commands being deleted. A specific command or a generic command can be specified, and either type can be optionally qualified by a library name.

The name of the command can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

command-name: Specify the name of the command being deleted.

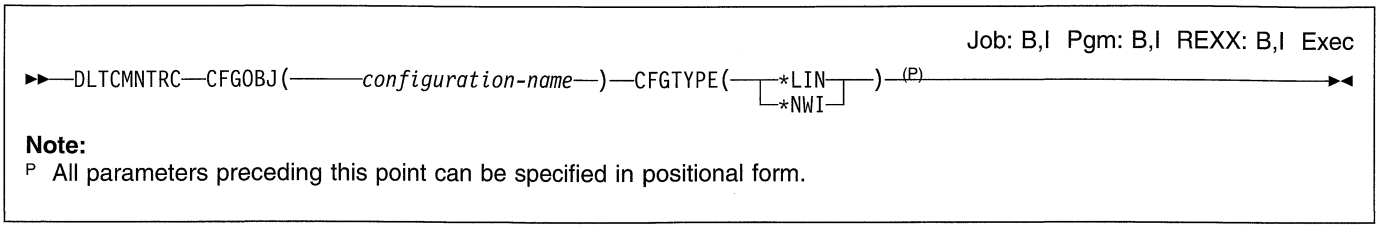
generic-command-name:* Specify the generic name of the commands that are being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCMD  CMD(LIB01/PAYROLL)
```

The command named PAYROLL is deleted from library LIB01. Any authorities of use for the command are removed from the profiles of all authorized users.

DLTCMNTRC (Delete Communications Trace) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Communications Trace (DLTCMNTRC) command deletes the communications trace for the specified line or network interface description. The communications trace can be deleted after the trace has ended.

Restrictions:

1. *SERVICE authority is required to use this command.
2. The following user profiles have authority to this command:
   - QSECOFR
   - QSRV

Required Parameters

CFGOBJ

Specifies the name of the configuration object that was traced. The object is either a line description or a network interface description.

CFGTYPE

Specifies the type of configuration description that was traced.

***LIN:** The type of configuration object is a line description.

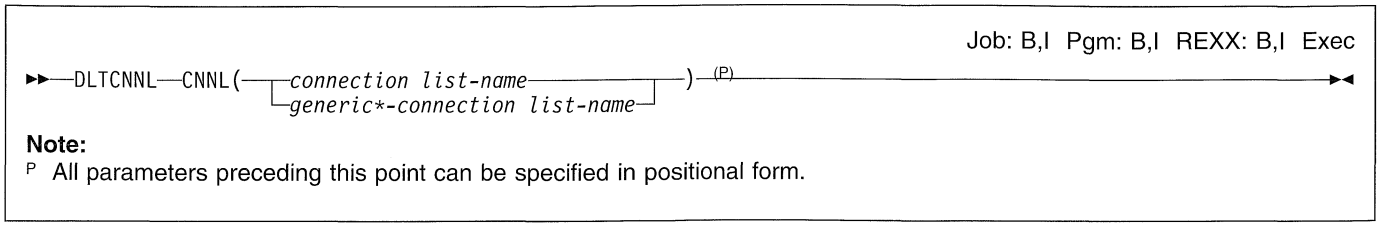
***NWI:** The type of configuration object is a network interface description.

Example

```
DLTCMNTRC  CFGOBJ(*QESLINE)  CFGTYPE(*NWI)
```

This command deletes the communications trace data for line description QESLINE.

DLTCNNL (Delete Connection List) Command



Purpose

The Delete Connection List (DLTCNNL) command is used to delete a connection list.

Required Parameters

CNNL

Specifies the name of the connection list to be deleted.

connection list-name: Specify the name of the connection list to be deleted.

generic-connection list-name:* Specify the generic name of the connection list name. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting a Specified Connection List

```
DLTCNNL   CNNL(CHICAGO)
```

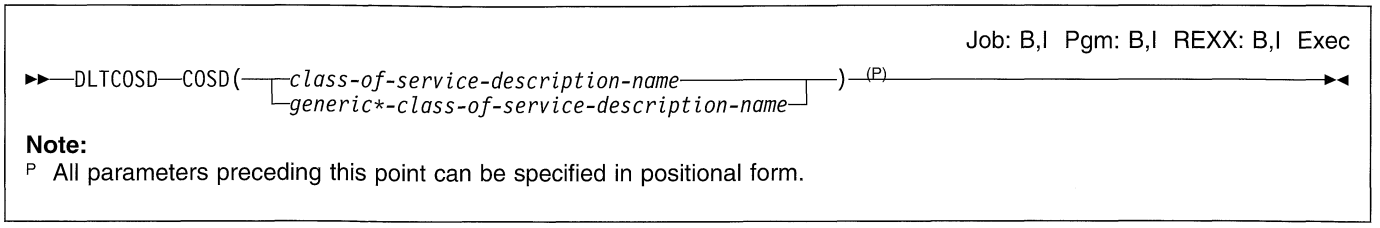
This command deletes the connection list named CHICAGO. Any entries in connection list CHICAGO are also deleted.

Example 2: Deleting Multiple Connection Lists

```
DLTCNNL   CNNL(CHI*)
```

This command deletes all connection lists whose name begins with CHI. All entries that were listed in the connection lists are also deleted.

DLTCOSD (Delete Class-of-Service Description) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Class-of-Service Description (DLTCOSD) command deletes a class-of-service description(s).

Required Parameters

COSD

Specifies the name of the class-of-service description to be deleted.

class-of-service-description-name: Specify the name of the class-of-service description to be deleted.

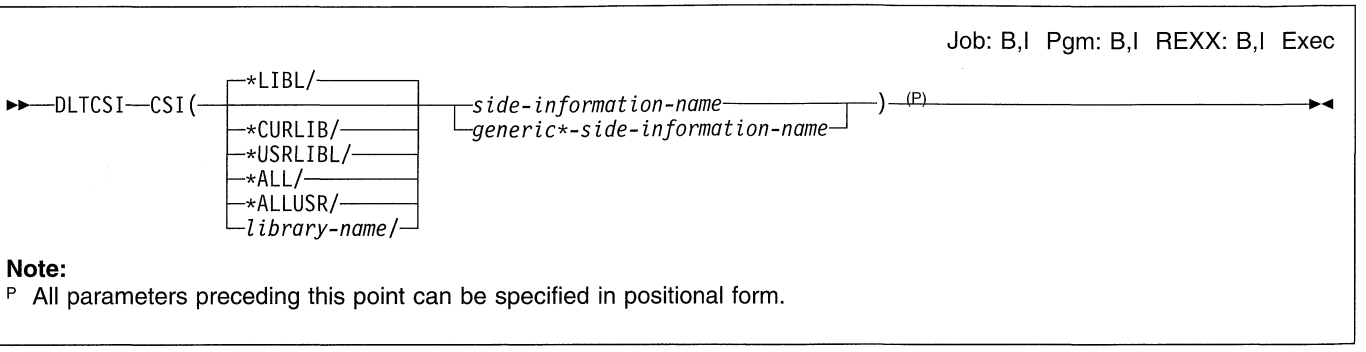
generic-class-of-description-name:* Specify the generic name of the class-of-service description being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCOSD   COSD(COS01)
```

This command deletes the class-of-service description named COS01 from the system.

DLTCSI (Delete Communications Side Information) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Communications Side Information (DLTCSI) command is used to delete side information objects from the specified libraries.

Required Parameters

CSI

Specifies the name of the side information object to be deleted.

The name of the side information object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

side-information-name: Specify the name of the object that contains the desired side information. This is the CPI-Communications symbolic destination name (sym_dest_name).

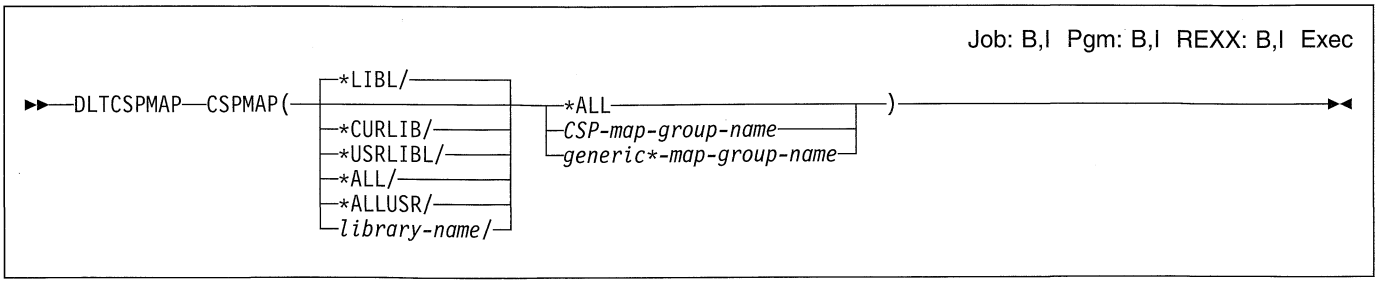
generic-side-information-name:* Specify the generic name of the side information object. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCSI   CSI(QGPL/SIDEOBJ)
```

This command deletes the communications side information object named SIDEOBJ from library QGPL.

DLTCSPMAP (Delete CSP/AE Map Group) Command



Purpose

The Delete CSP/AE Map Group (DLTCSPMAP) command removes the specified Cross-System Product/Application Execution (CSP/AE) map groups from the system. Generic names are allowed for the names of the map groups being deleted.

Restriction: Users deleting a map group must have object existence authority for the map group (*CSPMAP).

Required Parameters

CSPMAP

Specifies the qualified name of a CSP/AE map group or the generic name of several CSP/AE map groups being deleted.

The name of the CSP/AE map group can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All CSP/AE map groups in a library or library list are deleted.

CSP-map-group-name: Specify the name of the map group being deleted.

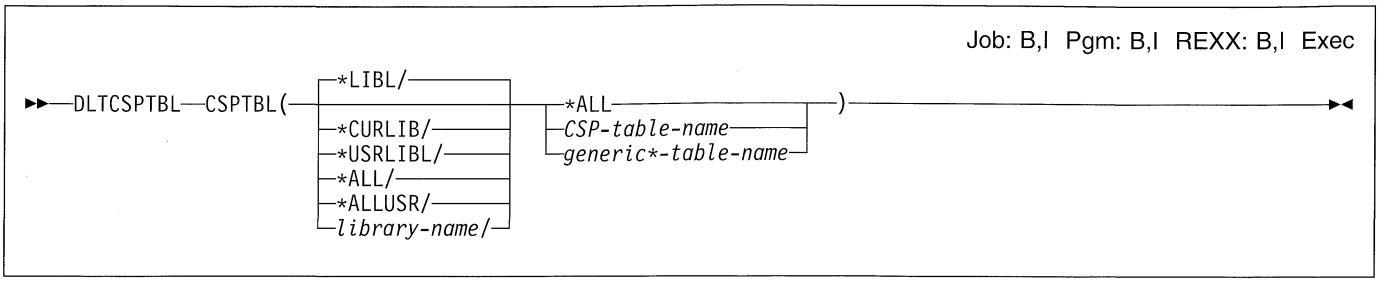
generic-map-group-name:* Specify the generic name of the map group being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCSPMAP  CSPMAP(CSPLIB/MAPD5)
```

This command deletes the map group MAPD5 from library CSPLIB.

DLTCSPTBL (Delete CSP/AE Table) Command



Purpose

The Delete CSP/AE Table (DLTCSPTBL) command deletes the specified Cross System Product/Application Execution (CSP/AE) tables from the system. Generic names can be specified for the names of the tables being deleted.

Restriction: Users deleting a table must have object existence authority for the table (*CSPTBL).

Required Parameters

CSPTBL

Specifies the qualified name of a CSP/AE table or the generic name of several CSP/AE tables to be deleted.

The name of the CSP/AE TABLE can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All CSP/AE tables in a library or library list are deleted.

CSP-table-name: Specify the name of the table being deleted.

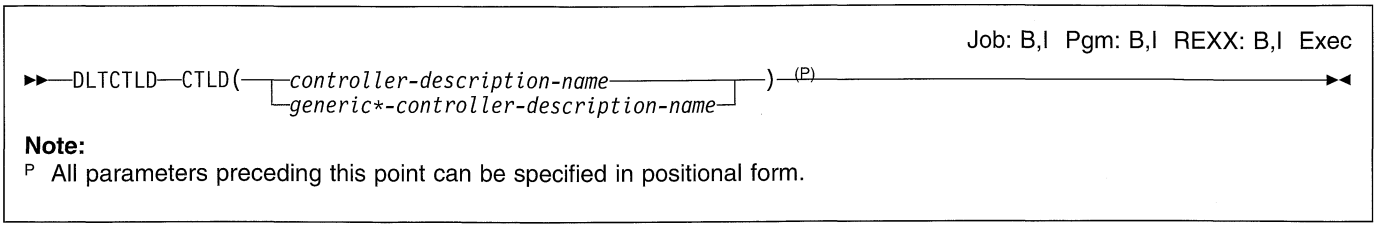
generic-table-name:* Specify the generic name of the table being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTCSPTBL  CSPTBL(CSPLIB/TBL4000T)
```

This command deletes the table TBL4000T from library CSPLIB.

DLTCTLD (Delete Controller Description) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Controller Description (DLTCTLD) command deletes controller descriptions.

Required Parameters

CTLD

Specifies the name of the controller descriptions being deleted. A specific controller description or a generic controller description is specified.

controller-description-name: Specify the name of the controller description to be deleted.

generic-controller-description-name:* Specify the generic name of the controller description to be deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

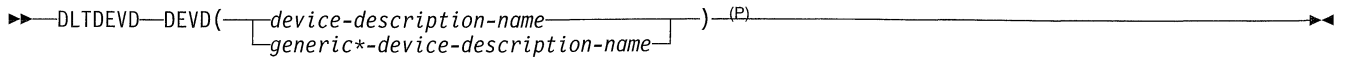
Example

```
DLTCTLD   CTLD(CONTROL01)
```

This command deletes the controller description named CONTROL01 from the system. If the controller description being deleted has any device descriptions associated with it, they are detached and a message containing their names is sent to the system operator.

DLTDEVD (Delete Device Description) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Device Description (DLTDEVD) command deletes the specified device descriptions.

Restriction: The device identified in the device description must be varied off before this command is sent. If the device description is the last device description on a varied on remote controller description, the remote controller description must be varied off.

Required Parameters

DEVD

Specifies the name of the device descriptions being deleted. A specific device description or a generic device description can be specified.

device-description-name: Specify the name of the device description being deleted.

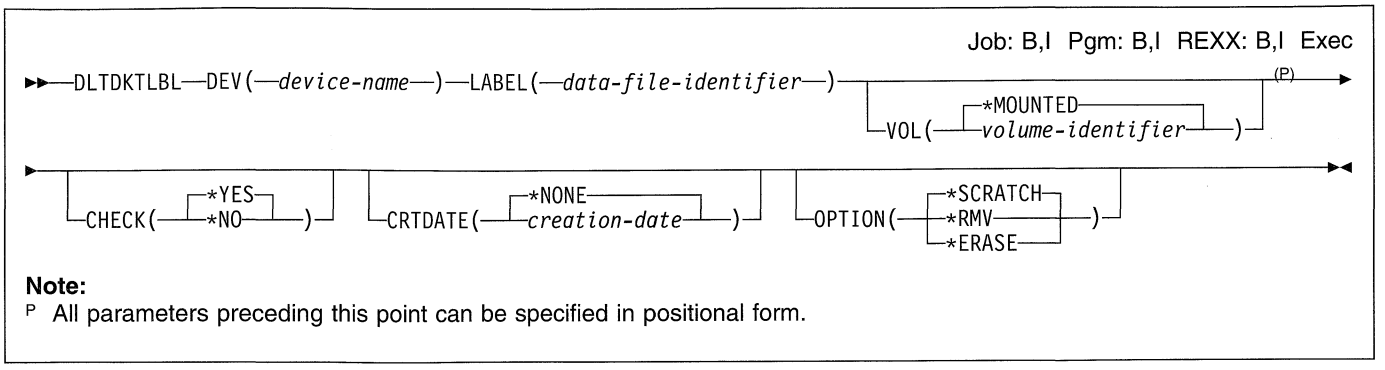
generic-device-description-name:* Specify the generic name of the device description being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTDEVD   DEVD(DSPC01)
```

This command deletes the device description of the device named DSPC01 from the system.

DLTDKTLBL (Delete Diskette Label) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Diskette Label (DLTDKTLBL) command deletes the label (that is, the data file identifier) of a named data file from a diskette. The data in the file can optionally be overwritten with binary zeros. If the file is active (the file end date is later than the system date), a message is sent to the system operator. The operator can then either continue to delete the file or end the command.

Note: Unpredictable results can occur when processing diskettes that do not have IBM standard labels. To initialize the diskette, use the Initialize Diskette (INZDKT) command with CHECK(*NO) specified.

Restriction: If diskettes have an extended label area (not supported by the AS/400 system) the extended label area is not searched for the label of the file being deleted.

Required Parameters

DEV
Specifies the name of the device in which the diskette being checked is located.

LABEL
Specifies the data file identifier of the file being deleted.

Optional Parameters

VOL
Specifies one or more volume identifiers used by the file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: If the CRTDATE parameter is also specified, it must match the date on the file being deleted.

***MOUNTED:** The volume currently placed in the device is used.

volume-identifier: Specify up to 6 characters for the volume identifier of the diskette that contains the file label being deleted; any combination of letters and numbers can be used. If the volume identifiers do not match, a message is sent to the system operator. The operator can then either insert the correct diskette and try again, or stop the command.

CHECK
Specifies whether a check for active files (those with an expiration date later than the system date) is done.

***YES:** If the file is active, an operator message is sent. The operator either can continue or can stop the deletion of the file.

***NO:** The file is deleted without the active file check being done.

CRTDATE
Specifies the creation date of the object.

Note: If CRTDATE is specified and the date on the file being deleted does not match, the file is not deleted and a message is sent to the system operator. The operator can then either try the operation again or stop the command.

***NONE:** No test is made of the date the file was created.

creation-date: Specify the creation date of the file being deleted. The date must be typed in the format specified by the system values QDATFMT or QDATSEP.

OPTION
Specifies how the file is deleted from the diskette.

***SCRATCH:** The expiration date of the file is changed to the current system date. The file can still be referenced for input data. However, when the system writes a new file on the diskette, all files that have expired are deleted to free space for the new files.

***RMV:** The data file identifier is removed from cylinder 0. When the file identifier is gone, the file cannot be referenced for input.

***ERASE:** The data file identifier is deleted from cylinder 0, and data in the file is overwritten with binary zeros.

Examples

Example 1: Deleting File Without an Active File Check

```
DLTDKTLBL  LABEL(FILEA)  DEV(DKT1)  CHECK(*NO)
```

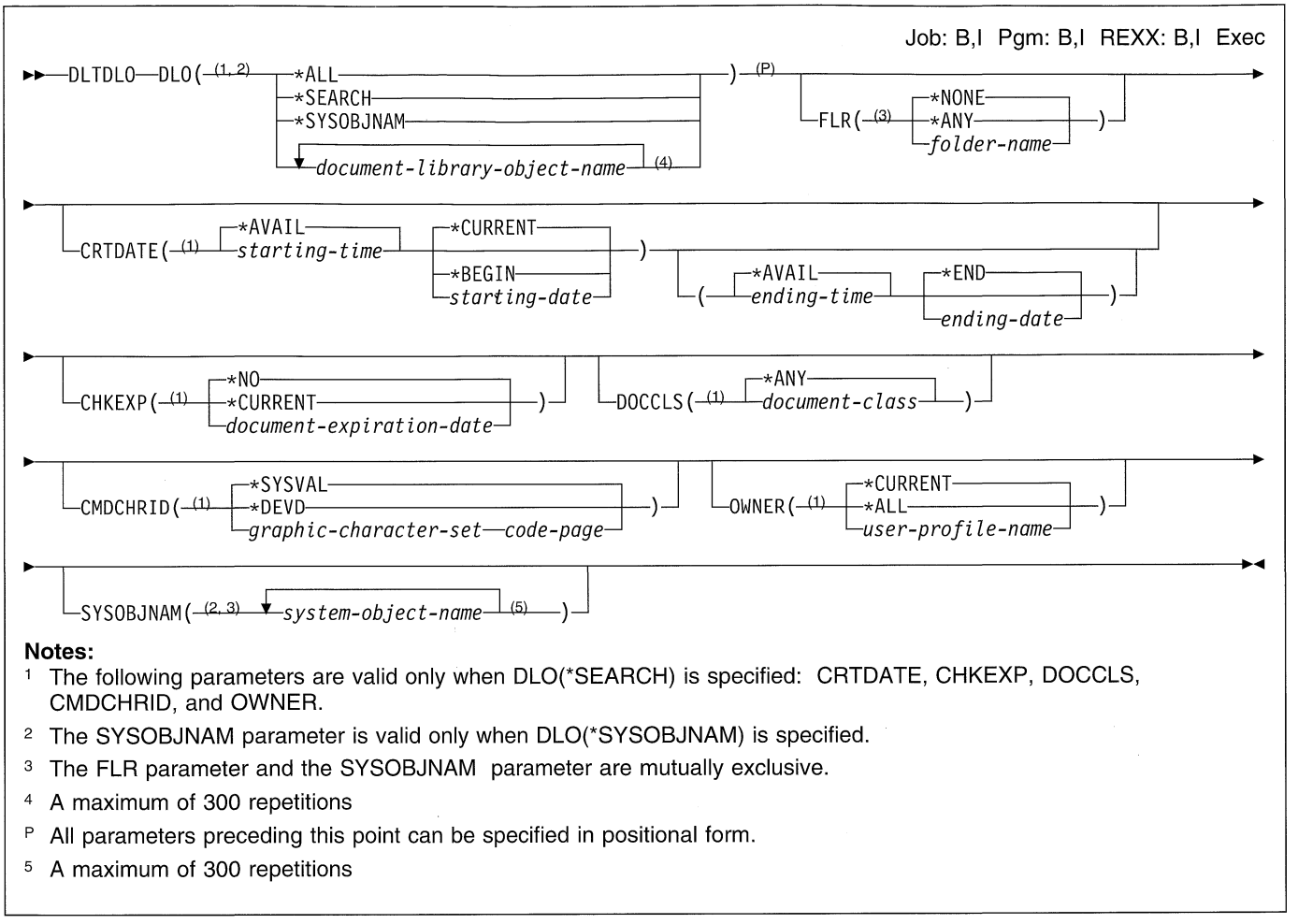
This command scratches (assumed by the system) FILEA on the diskette in device DKT1 without the active file check.

Example 2: Deleting a File Identifier and Overwriting the Data

```
DLTDKTLBL   LABEL(MONDAY)  DEV(DKT2)  OPTION(*ERASE)
```

This command deletes the file identifier MONDAY from the diskette in device DKT2 and overwrites the data with binary zeros if the file MONDAY is not active (CHECK(*YES) is assumed).

DLTDLO (Delete Document Library Object) Command



Notes:

1 The following parameters are valid only when DLO(*SEARCH) is specified: CRTDATE, CHKEXP, DOCCLS, CMDCHRID, and OWNER.

2 The SYSOBJNAM parameter is valid only when DLO(*SYSOBJNAM) is specified.

3 The FLR parameter and the SYSOBJNAM parameter are mutually exclusive.

4 A maximum of 300 repetitions

P All parameters preceding this point can be specified in positional form.

5 A maximum of 300 repetitions

Purpose

The Delete Document Library Object (DLTDLO) command deletes document library objects from the system.

This command is used to recover disk storage space when system storage becomes critical. It is also used to delete documents and folders that are no longer needed, and generates a request to remove the document from the text index if there is a text entry in the index.

Restrictions:

1. The user of this command must have *ALL, *ALLOBJ, or *SAVSYS authority.
2. Pressing the Enter key deletes the objects immediately; there is no prompting before the document or folder is deleted.
3. A document or folder that is in use is not deleted.
4. When more than one document or folder is specified for deletion, but one of the documents or folders is in use or the user is not authorized to it, a message is sent and the other documents specified for deletion are deleted except the one either being used or restricted by lack of authority.
5. While using this command, the user may encounter an error message indicating that internal objects are locked. This means that another user is using a document library function that cannot run at the same time as the DLTDLO command. Therefore, retry this command later. (6) If a folder, and all documents and folders within it, is being deleted, any document or folder that cannot be deleted from the folder remains, causing the original folders to remain as well; all others are deleted. A message is sent indicating the documents and folders that cannot be deleted.
6. Any combination of FLR, CRTDATE, CHKEXP, DOCCLS, CMDCHRID, and OWNER may be used when DLO(*SEARCH) is specified. When more than one of them is specified, only those documents and folders which meet all of the specified criteria are deleted. Though the FLR parameter can be specified under the preceding criteria, it is not limited to this criteria. The FLR parameter and the SYSOBJNAM parameter are mutually exclusive. It can be specified in any combination except when DLO(*SYSOBJNAM) or the SYSOBJNAM parameter is specified.

7. The document descriptors are searched to locate and delete document library objects that match the search values.
8. CRTDATE contains two lists of two elements each. *N must be specified for an element that precedes the values being specified to maintain that element's position within the parameter value sequence when previous elements in the list are deleted.

Required Parameters

DLO

Specifies the document library objects (documents and/or folders) that are deleted. If DLO(*SEARCH) is not specified, CRTDATE, CHKEXP, DOCCLS, CMDCHRID, and OWNER cannot be specified.

***ALL:** All documents or folders currently filed in the folder specified in the FLR parameter are deleted. Only the security officer or a user with *SECADM special authority can specify DLO(*ALL) with FLR(*ANY) or FLR(*NONE).

***SEARCH:** All documents or folders currently filed in the document library and are described by the FLR, CRTDATE, CHKEXP, DOCCLS, and OWNER values are deleted.

***SYSOBJNAM:** The system object name for the documents or folders specified in the SYSOBJNAM parameter is used.

document-library-object-name: Specify one or more AS/400 system document or folder names of specific documents or folders being deleted. Up to 300 documents or folders can be specified. More information on folder names is in "Rules for Specifying Folder and Document Names."

Optional Parameters

FLR

Specifies the name of the folder that contains the document.

Note: More information on folder names is in "Rules for Specifying Folder and Document Names," in Volume 1.

***NONE:** The document or folder is not located in a folder. This implies that DLO(*SEARCH) and DLO(*ALL) apply only to the set of all documents and folders not in folders.

***ANY:** A system-wide search is made for the documents to be deleted. This implies that DLO(*SEARCH) and DLO(*ALL) apply to the set of all documents and folders, including those not found in folders.

folder-name: Specify the name of the folder that contains the documents and/or folders being deleted.

CRTDATE

Specifies that documents or folders filed during the specified time period are deleted. If this parameter is specified, then DLO(*SEARCH) must also be specified. For a document or folder being selected, it must have been created on or after the starting time and date, and before or on the ending time and date. This parameter contains two lists of paired elements, one for the starting time and date and one for the ending time and date as follows:

```
((start-time  start-date) (end-time  end-date))
```

If CRTDATE is not specified, the following values are assumed:

```
((*AVAIL  *CURRENT) (*AVAIL  *END))
```

Element 1: Starting Time

Use one of the following values to specify the starting time. Documents and folders must have been created after this time to be deleted. Documents and folders created before this time are not deleted.

***AVAIL:** Documents and folders created at any time can be deleted.

starting-time: Specify deletion of only those documents and folders that were created at or after the specified time on the starting date. When the start-time is used as a search value, the start-date must not be *BEGIN. The time is specified in 24-hour format with or without a time separator as follows:

- With a time separator, specify a string of 5 or 8 digits where the time separator separates the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command fails.
- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where **hh** = hours, **mm** = minutes, and **ss** = seconds. Valid values for **hh** range from 00 through 23. Valid values for **mm** and **ss** range from 00 through 59.

Element 2: Starting Date

Use one of the following values to specify the starting date. Documents and folders must have been created on or after this date in order to be selected for deletion. Documents and folders created before this date cannot be deleted.

***CURRENT:** The current date is used.

***BEGIN:** Documents and folders created on any date can be deleted. The starting date and time is ignored.

starting-date: Specify deletion of only those documents and folders created on or after the date specified. The date must be entered in the format specified by the system value QDATFMT and, if separators are used, QDATSEP.

Element 3: Ending Time

Use one of the following values to specify the ending time. Documents and folders must have been created at or before this time to be deleted. Documents and folders created after the specified time cannot be deleted.

***AVAIL:** Documents and folders created at any time can be deleted.

ending-time: Specify deletion of documents and folders that were created at or before the specified time on the ending date. When the end-time is used as search values, the end-date must not be *END. See the description of *starting-time* for details about how time can be specified.

Element 4: Ending Date

Use one of the following values to specify the ending date. Documents and folders must have been created on or before this date to be deleted. Documents and folders created after this date cannot be deleted.

***END:** Documents and folders created on any date can be deleted. The ending time is ignored when *END is specified.

ending-date: Specify selection of documents and folders created on or before this date. The date must be specified in the format specified by the system value QDATFMT or, if separators are used, QDATSEP.

CHKEXP

Specifies the expiration date of the documents being deleted. The expiration date is assigned to specify when a document is no longer needed. A document with an expiration date (as specified by the user on the Describe Document display) before the date specified, is deleted. If this parameter is specified, then DLO(*SEARCH) must also be specified.

***NO:** The expiration date for the document is ignored.

***CURRENT:** All documents with an expiration date earlier than today's date are deleted.

document-expiration-date: Specify a date when all documents are to be deleted.

DOCCLS

Specifies the class of documents being deleted. Any character string can be used to specify the document class when creating a document; for example, SPECIFICATION, MEMO, FORM, or PAYROLL. If this parameter is specified, then DLO(*SEARCH) must also be specified.

***ANY:** The DOCCLS parameter is not used to select the documents being deleted. Documents meeting the requirements of the other parameters are deleted.

document-class: Specify a character string ranging from 1 through 16 characters in length to select documents for deletion. The AS/400 system does not define a meaning for document-class; that is, when creating a document, any character string can be entered into the document class. For comparison, the document class specified and the document class of all documents are changed to uppercase where necessary.

CMDCHRID

Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set of the class of documents being deleted. Valid values range from 1 through 999.

Element 2: Code Page

code-page: Specify the code page value used to create the command parameters. Valid values range from 1 through 999.

OWNER

Specifies the owner of the documents and folders being deleted. If this parameter is specified, then DLO(*SEARCH) must also be specified.

***CURRENT:** The user profile under which the current job is running is used.

***ALL:** Documents and folders are selected for deletion without regard to the owners. Only a security officer or a user with *SECADM special authority can specify OWNER(*ALL).

user-profile-name: Specify the name of the user who owns the documents and folders being deleted. Only a security officer or a user with *SECADM special authority is authorized to specify a user profile name other than that belonging to the user who issues this command.

SYSOBJNAM

Specifies the system object name. This parameter is valid only when DLO(*SYSOBJNAM) or DOCL(*SYSOBJNAM) is specified. A full ten characters must be specified.

Examples

Example 1: Deleting a Folder

```
DLTDLO    DLO(ABC)
```

This command deletes folder ABC if it contains no documents or folders.

Example 2: Deleting All Documents and Folders Within a Folder

```
DLTDLO   DLO(*ALL)  FLR(ABC)
```

This command deletes all the documents and folders that the user is authorized to delete from folder ABC. If folder ABC turns out to be empty, then it is also deleted.

Example 3: User is Securing Officer or Has *SECADM Special Authority

```
DLTDLO   DLO(*SEARCH)  CRTDATE((*AVAIL 080187)
  (*AVAIL 083187))  DOCCLS(LETTERS)  OWNER(*ALL)
```

This command deletes all documents in document class LETTERS that were filed in the system during August 1987. Only the security officer or a user with *SECADM special authority is allowed to specify OWNER(*ALL).

Example 4: Deleting Current User's Documents

```
DLTDLO   DLO(*SEARCH)  CRTDATE((*AVAIL *BEGIN)
  (*AVAIL *END))  DOCCLS(*ANY)  OWNER(*CURRENT)
```

This command deletes all documents belonging to the current user.

Example 5: Deleting User's Documents Filed Today

```
DLTDLO   DLO(*SEARCH)
```

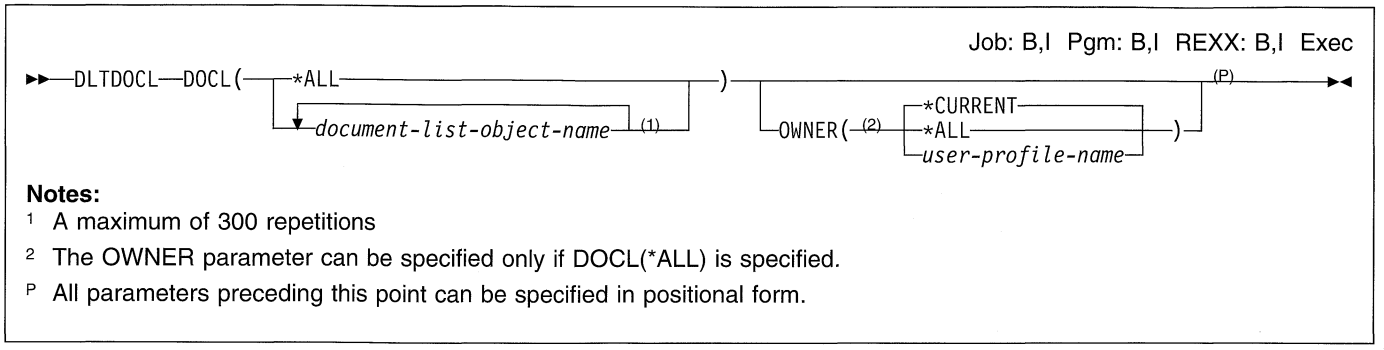
This command deletes all documents belonging to the specified user that were filed on that day.

Example 6: Deleting Documents and Folders With Document Descriptors

```
DLTDLO   DLO(*SEARCH)  CRTDATE((*AVAIL *BEGIN)
  (*AVAIL *END))  DOCCLS(*ANY)  OWNER(*ALL)
```

This command deletes all documents and folders with document descriptors from the system. If the document descriptors are not synchronized with the document and folder objects, then they are the same as DLO(*ALL). The *Advanced Backup and Recovery Guide* has more information about synchronizing document descriptors.

DLTDOCL (Delete Document List) Command



Purpose

The Delete Document List (DLTDOCL) command deletes a document list from the system. All document lists are stored in library QUSRSYS.

Restrictions:

1. To delete a document list, the user must have object existence authority for the document list, *ALLOBJ special authority, or *SECADM special authority.
2. To delete a document list for another user, the user must have *ALLOBJ or *SECADM special authority.

More information on deleting document list names is in the *Communications: Distribution Services Network Guide*.

Required Parameters

DOCL

Specifies the document lists that are being deleted from the system.

***ALL:** All document lists for the specified owner are deleted.

document-list-object-name: Specify the system object name of the AS/400 system document list that is being deleted. Up to 300 document lists can be specified.

Optional Parameters

OWNER

Specifies the owner of the document lists that are being deleted. This parameter is ignored if a list of document list names is specified on the DOCL parameter.

***CURRENT:** The user profile under which the current job is running is used.

***ALL:** All document lists for all owners are deleted.

user-profile-name: Specify the name of the user who owns the document lists that are being deleted. All document lists owned by this user are deleted.

Examples

Example 1: Deleting Current User's Document Lists

```
DLTDOCL   DOCL(*ALL)  OWNER(*CURRENT)
```

This command deletes all document lists owned by the current user.

Example 2: Deleting All Users' Document Lists

```
DLTDOCL   DOCL(*ALL)  OWNER(*ALL)
```

This command deletes all document lists owned by all users.

Example 3: Deleting Document Lists Owned By a Specified User

```
DLTDOCL   DOCL(*ALL)  OWNER(ANN)
```

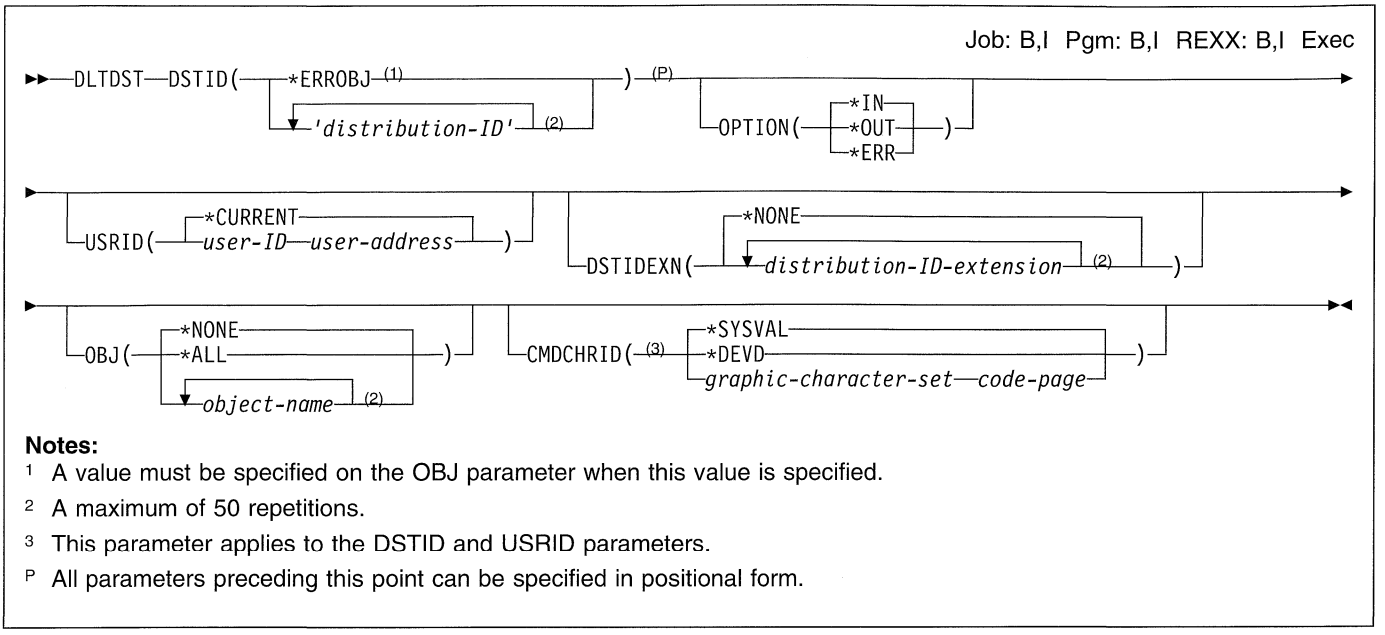
This command deletes all document lists owned by ANN.

Example 4: Deleting a Specific Document List Owned By User

```
DLTDOCL   DOCL(AN8T475237)
```

This command deletes document list AN8T475237, owned by the user running this command. Displaying the user's owned objects will show the system object name for a specific document list.

DLTDST (Delete Distribution) Command



Notes:

1 A value must be specified on the OBJ parameter when this value is specified.

2 A maximum of 50 repetitions.

3 This parameter applies to the DSTID and USRID parameters.

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Distribution (DLTDST) command allows users to delete their own distributions or distributions for another user if the authority to work on behalf of the other user has been given. This command also allows system operators to delete damaged distribution objects and dangling distribution objects (distribution objects that are not controlled by a distribution tracking object).

Outgoing distributions represent the outgoing confirmation request. This deletes the confirmation entries associated for the particular mail from the confirmation log. The deletion affects only the confirmation request. The request to deliver the mail remains in effect. Also, the request can delete error status information from the undelivered status.

Restrictions:

1. To work on behalf of other users, a user must have special permission given with the Grant User Permission (GRTUSRPMN) command.
2. The requester of the command (the user who is signed on) must be enrolled in the system distribution directory.
3. Personal distribution cannot be deleted if the requester is working on behalf of another user.
4. Damaged and dangling distribution objects can be deleted only by the user who signs on and has *ALLOBJ authority.

Required Parameter

DSTID

Specifies the distributions to be deleted.

***ERROBJ:** The distributions to be deleted are damaged or dangling distribution objects.

Note: To delete a damaged or dangling distribution, a value must be specified also on the OBJ parameter.

distribution-identifier: Specify the unique identifier of the distribution to be deleted. The distribution identifier is assigned to the distribution by the system that created it.

Up to 50 distribution IDs can be specified. The distribution identifier consists of the sender's address (padded on the right with blanks up to 8 characters), the sender's user ID (padded on the right with blanks up to 8 characters), and a 4-digit zoned sequence number with leading zeros. For example,

```
'NEWYORK SMITH   0204'   or
UADDRESSUSERIDID0099
```

The apostrophes are needed if there are blanks or special characters in the distribution identifier. The distribution identifier is specified this way because blank characters are valid in a user ID or address.

The distribution identifier can be determined with the Query Distribution (QRYDST) command.

Note: The distribution specified cannot be damaged or dangling.

Optional Parameters

OPTION

Specifies whether incoming, outgoing, or error distribution information is deleted.

***IN:** Information about incoming distribution is deleted. An incoming distribution is a distribution being sent to a user. If confirmation of delivery is requested by the

sender, notice of the deleted status is returned to the sender.

***OUT:** Information about an outgoing distribution is deleted. An outgoing distribution is the status saved by the system for a distribution that was sent to one or more users with a request for delivery confirmation.

***ERR:** Information about an error or undelivered distribution is deleted. A distribution is sent that was undelivered because an error occurred while routing the distribution.

USRID

Specifies the user ID and address of the user making this request.

***CURRENT:** The user profile under which the current job is running is used.

Element 1: User ID

user-ID: Specify the user ID of the user for whom the distribution information is deleted.

Element 2: User Address

user-address: Specify the user address of the user for whom the distribution information is deleted.

DSTIDEXN

Specifies the extension of the distribution identifier specified by the DSTID parameter. This extension identifies duplicate distributions. This extension is a 2-digit extension that ranges from 00 through 99. For incoming distributions, this extension ranges from 01 through 99. For outgoing and undelivered status distributions, this extension is always 00. For example, if the distribution ID is *'NEWYORK SMITH 0204'* and two copies of this distribution were sent to a user, the user has two distributions with the same distribution ID. To distinguish the two distributions, an extension is added to each distribution ID and one extension is *NEWYORK SMITH 020401* and the other one is *NEWYORK SMITH 020402*. If there are no duplicates, the extension defaults to 01. These extensions map one-to-one with the distribution ID specified on the DSTID parameter.

***NONE:** There is no duplicate distribution. This is equivalent to an extension of 00 for outgoing distributions and 01 for incoming distributions.

distribution-ID-extension: Specify the extension associated with the distribution. This is used to identify duplicate distribution IDs. Up to 50 distribution IDs can be specified.

OBJ

Specifies the damaged or dangling distribution objects to be deleted.

***NONE:** No damaged or dangling distribution objects are deleted.

***ALL:** All damaged and dangling distribution objects are deleted.

object-name: Specify the name of the damaged or dangling distribution object to be deleted.

The object named can be a local or remote object.

- Local object names consist of the sender's address (padded on the right with blanks to a maximum of 8 characters), the sender's user ID (padded on the right with blanks to a maximum of 8 characters), and a 4-digit zoned sequence number with leading zeros. For example:

  ```
  'NEWYORK SMITH   0204'   or
  UADDRESSUSERIDID0099
  ```

  The apostrophes are needed if there are blanks or special characters in the distribution identifier. The distribution identifier is specified this way because blank characters are valid in a user ID or address.
- Remote object names consist of the remote system name and the system date and time. For example:

  ```
  'QOSRDIST091112509152355'
  ```

  where QOSRDIST is the remote district name and 091112509152355 is the system date and time.

The distribution identifier for damaged or dangling distributions can be determined when the Save Document Library Object (SAVDLO) command is run and damaged or dangling distribution objects are identified.

A maximum of 50 objects can be named to be deleted.

CMDCHRID

Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

Note: This parameter translates the USRID and DSTID parameters to character set and code page of '930 500'. The *Communications: Distribution Services Network Guide* contains the character set and code page table for '930 500'.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the command parameter.

Element 2: Code Page

code-page: Specify the code page. Valid values range from 1 through 9999.

Examples

Example 1: Deleting Incoming Distribution Sent to User

```
DLTDST  DSTID(('RCH38P  BAKER  0019'))  USRID(*CURRENT)
  DSTIDEXN(02)
```

This command deletes an incoming distribution sent to the user.

Example 2: Deleting Status of Outgoing Distribution

```
DLTDST  DSTID(('SYSTEM20  BRUCE  1361'))  USRID(*CURRENT)
  DSTIDEXN(00)  OPTION(*OUT)  CMDCHRID(101 37)
```

This command deletes the status associated with an outgoing distribution. The DSTID parameter value is encoded with a character identifier where the code page is 37 and the character set is 101. The system performs translation on the DSTID parameter value before trying to locate the distribution on the distribution recipient index.

Example 3: Deleting Incoming Distribution to Signed-On User

```
DLTDST  DSTID(('CHICAGO JONES   0013'))
```

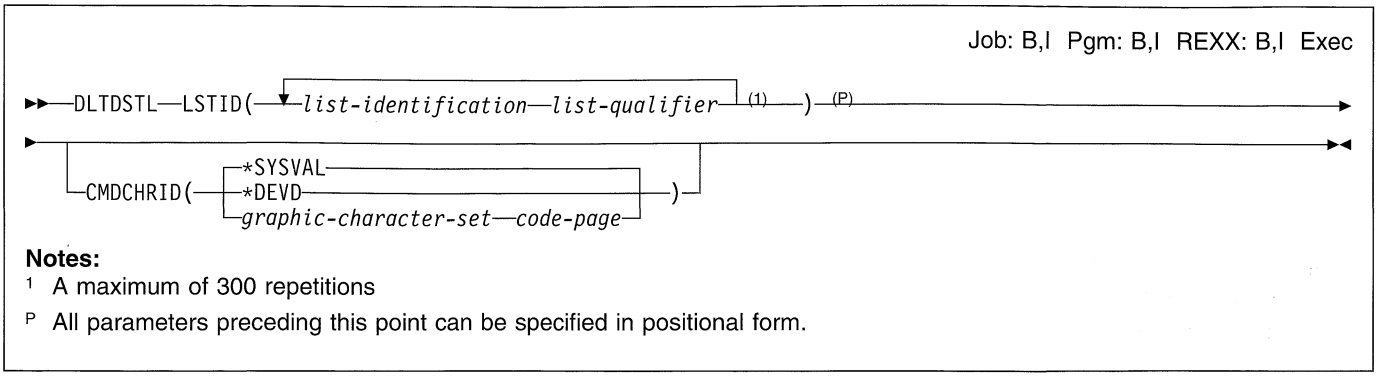
This command deletes an incoming distribution sent to a user who is signed on. The distribution extension defaults to '01' for incoming distributions. If duplicate distributions were sent, this command would only delete the first one.

Example 4: Deleting All Damaged Distributions

```
DLTDST  DSTID(*ERROBJ) OBJ(*ALL)
```

This command deletes all damaged and dangling distributions on the system.

DLTDSTL (Delete Distribution List) Command



Purpose

The Delete Distribution List (DLTDSTL) command deletes an existing distribution list from the system distribution directory. Up to 300 lists can be deleted at a time.

Restriction: The user of this command must have *SECADM special authority to delete distribution lists owned by another user. Users can delete distribution lists that they create.

Required Parameters

LSTID

Specifies the two-part list identifier of the distribution list being deleted. Up to 300 list IDs (both parts) can be specified.

Element 1: List Identifier

list-identifier: Specify the list identifier (ID) of a distribution list.

Element 2: List Qualifier

list-qualifier: Specify the list qualifier of the distribution list.

Note: The distribution list identifier/list qualifier is in two parts, separated by at least one space. If any lowercase characters are specified, the system changes them to uppercase.

The naming rules for the two-part list ID are identical to the rules for the user ID and address. A complete description of these rules is in the *Distribution Services Network Guide*.

Optional Parameters

CMDCHRID

Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the command parameter. Valid values range from 1 through 9999.

Element 2: Code Page

code-page: Specify the code page. Valid values range from 1 through 9999.

Example

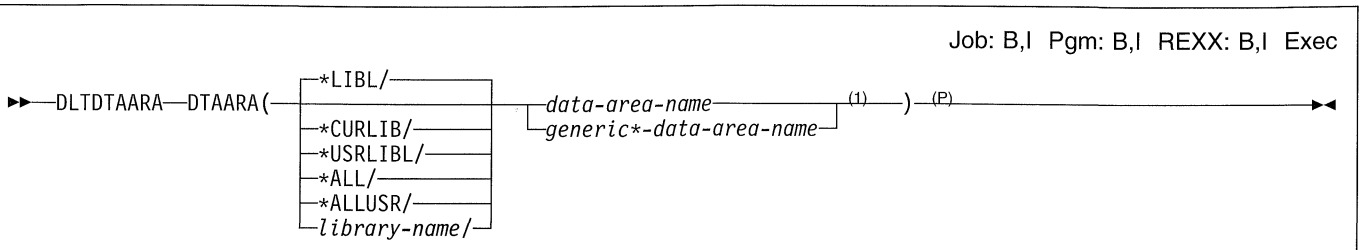
```
DLTDSTL  LSTID((DEPT48K DLIST) (ISMGRS DSTLIST))
```

This command deletes the two distribution lists, DEPT48K DLIST and ISMGRS DSTLIST, if they exist.

Additional Considerations

If a specified distribution list identifier does not exist in the directory, an error message is returned. If several list identifiers are given, each is processed individually. An error message is returned each time a list identifier is found not to exist or if the current user of this command does not have the authority to delete the list.

DLTDTAARA (Delete Data Area) Command



Notes:
1 A variable cannot be coded on this parameter.
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Data Area (DLTDTAARA) command deletes the specified data areas from a library.

Restriction: The user of this command must have object existence authority for the data area. Local data areas cannot be deleted.

Required Parameters

DTAARA

Specifies the qualified name of the data areas being deleted. A specific data area or a generic data area can be specified; either type can be optionally qualified by a library name.

The name of the data area can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

data-area-name: Specify the qualified name of the data area being deleted.

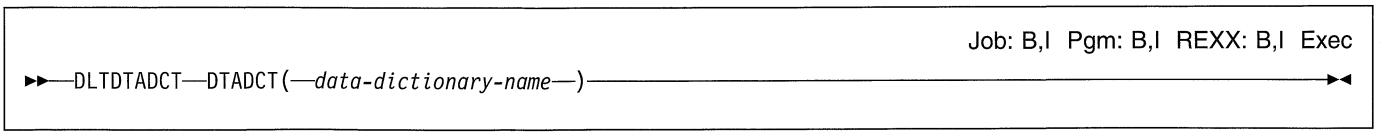
generic-data-area-name:* Specify the generic name of the data area being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTDTAARA   DTAARA(MYLIB/MYDATA)
```

This command deletes the data area named MYDATA from the library MYLIB if the user has the proper authority for the data area and the library.

DLTDTADCT (Delete Data Dictionary) Command



Purpose

The Delete Data Dictionary (DLTDTADCT) command allows a user to delete a data dictionary.

All program-described files linked to definitions in the dictionary must be unlinked before the dictionary can be deleted.

Required Parameters

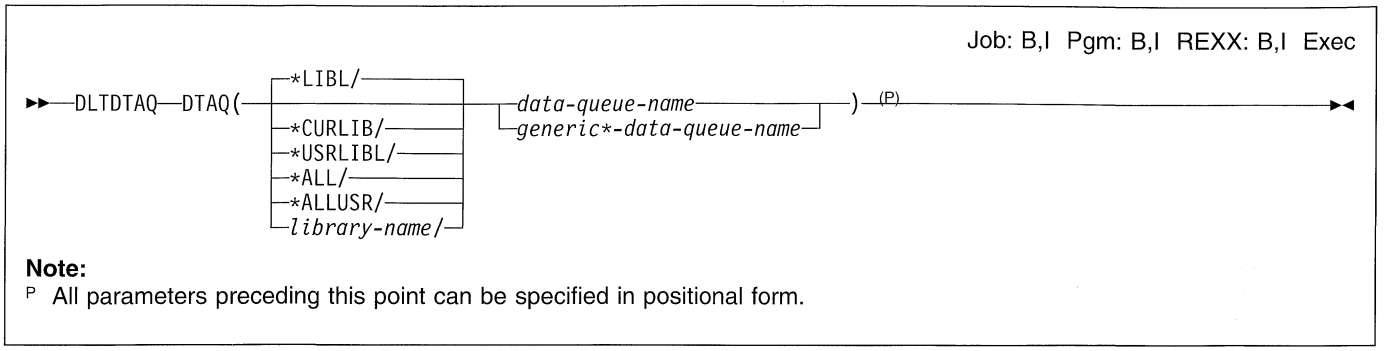
DTADCT
Specifies the name of the data dictionary being deleted.

Example

```
DLTDTADCT   DTADCT(DEPT547)
```

This command deletes the DEPT547 data dictionary in library DEPT547.

DLTDTAQ (Delete Data Queue) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Data Queue (DLTDTAQ) command deletes the specified data queues from the system.

Restriction: The user must have object existence authority for the queue.

Required Parameters

DTAQ

Specifies the qualified name of the data queues being deleted. A specific data queue or a generic data queue can be specified, and either type can be optionally qualified by a library name.

The name of the data queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

data-queue-name: Specify the name of the data queue being deleted.

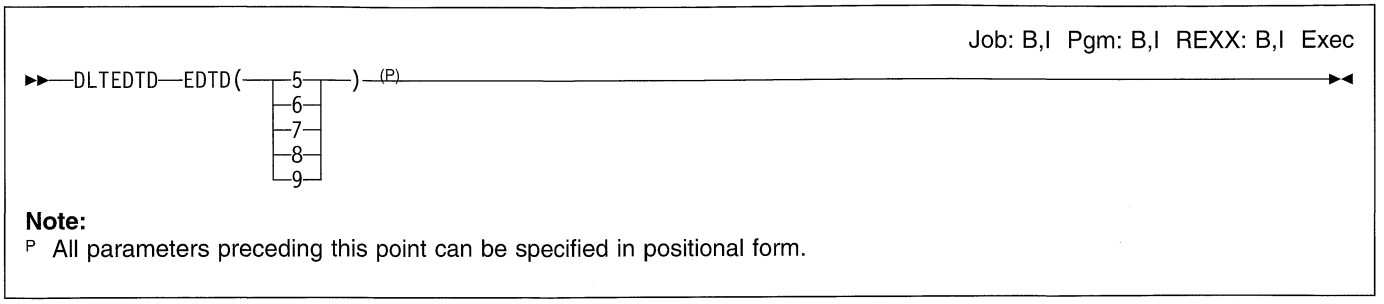
generic-data-queue-name:* Specify the generic name of the data queue being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTDTAQ   DTAQ(DEPTADTAQ)
```

This command deletes the data queue named DEPTADTAQ from the system.

DLTEDTD (Delete Edit Description) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Edit Description (DLTEDTD) command deletes a specified user-defined edit description.

Note: Any data description specifications (DDS), or high-level language programs that have already been created, are not affected.

Required Parameters

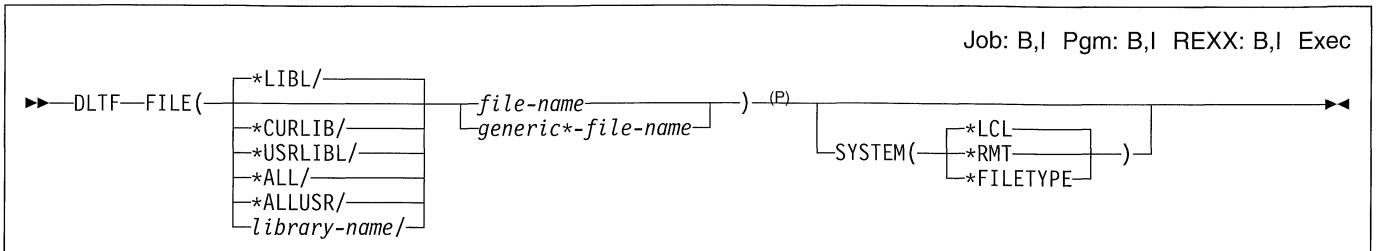
EDTD
Specifies a digit code (5, 6, 7, 8, or 9) that identifies the user-defined edit description being deleted.

Example

```
DLTEDTD    EDTD(5)
```

This command deletes the user-defined edit description 5 from the system.

DLTF (Delete File) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete File (DLTF) command deletes the specified database files or device files from the system. Deleting the files also frees the storage space allocated to the file. If a database file (physical or logical) is deleted, all members contained in the file are also deleted. If the file is in use, the file is not deleted.

A delete operation by generic name deletes every file with that generic name that is eligible to be deleted. For each file that cannot be deleted, a diagnostic message is sent. Files can be ineligible to be deleted for the following reason:

- Logical files must be deleted before the physical files on which they are based can be deleted.
- SQL views must be deleted before the physical files, SQL tables, or SQL views on which they are based can be deleted.

Restrictions:

1. To delete a file, the user must have object existence authority and object operational for the file and read authority for the library that contains the file.
2. If a physical file is being deleted and a logical file is using the data in the physical file, the logical file must be deleted first.
3. If the DLTF command is entered when debugging and UPDPROD(*NO) was specified on the Start Debug (STRDBG) or Change Debug (CHGDBG) command, a physical file that contains data and is in a production library cannot be deleted.

Required Parameters

FILE

Specifies the name of the files being deleted. A specific file or a generic file name is specified; either type optionally can be qualified by a library name.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

file-name: Specify the full name of the file being deleted.

generic-file-name:* Specify the generic name of the file being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Optional Parameters

SYSTEM

Specifies whether the file is deleted from the local or a remote system.

***LCL:** The file is deleted from the local system.

***RMT:** The file is deleted from a remote system. A DDM file must be specified on the FILE parameter; the DDM file identifies the file being deleted and the remote system.

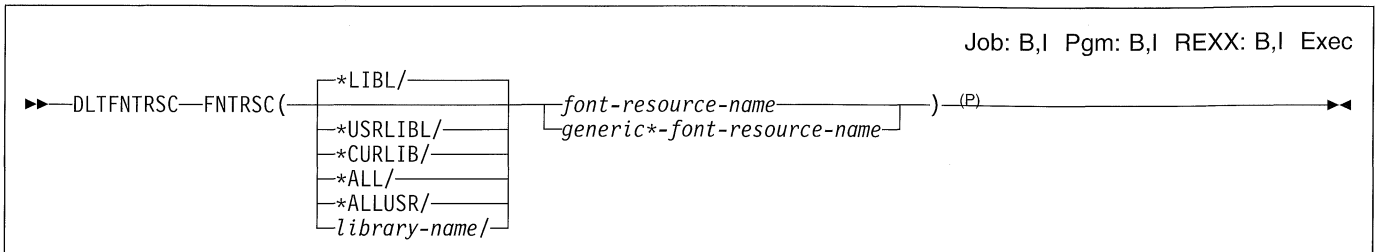
***FILETYPE:** The file is deleted from the local system if a DDM file is not specified on the FILE parameter. If a DDM file is specified on the FILE parameter, the remote file identifier in the DDM file is deleted from the remote system.

Example

```
DLTF   FILE(BILLING/ORDERS)
```

This command deletes the file named ORDERS in library BILLING. Only the BILLING library is searched for the file.

DLTFNTRSC (Delete Font Resources) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Font Resources (DLTFNTRSC) command deletes a font resource from the specified libraries. If the font resource is not found, a message is sent to the user stating that the font resource could not be found.

Required Parameters

FNTRSC

Specifies the qualified name of the font resource.

The name of the font resource can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

font-resource-name: Specify the name of the font resource being deleted.

generic-font-resource-name:* Specify the generic name of the font resource being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting a Font Resource in the Current Library

```
DLTFNTRSC  FNTRSC(*CURLIB/FNTRSC1)
```

This command deletes the font resource FNTRSC1 if it is in the current library.

Example 2: Deleting Font Resources in All Libraries

```
DLTFNTRSC  FNTRSC(*ALLUSR/FNTRSC1)
```

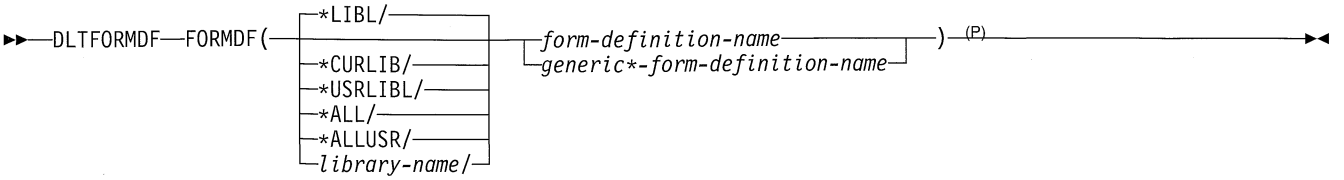
This command deletes all the font resources named FNTRSC1 in all the user's libraries. Libraries starting with a Q, except for QPGL, are not searched.

Example 3: Deleting Font Resources that Begin with FD

```
DLTFNTRSC  FNTRSC(*LIBL/FD*)
```

This command deletes all the font resources that begin with FD in the job's library list.

DLTFORMDF (Delete Form Definition) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Form Definition (DLTFORMDF) command deletes a form definition from the specified library. If the form definition is found, it is deleted. If the form definition is not found, a message is sent to the user stating that the form definition could not be found.

Required Parameters

FORMDF

Specifies the qualified name of the form definition being deleted.

The name of the form definition can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

form-definition-name: Specify the name of the form definition being deleted.

generic-form-definition-name:* Specify the generic name of the form definition. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting the First Occurrence of a Specified Form Definition

```
DLTFORMDF  FORMDF(*LIBL/FORMDF1)
```

This command deletes the first occurrence of FORMDF1 if it was found in the library list.

Example 2: Deleting All Occurrences of a Specified Form Definition

```
DLTFORMDF  FORMDF(*ALLUSR/FORMDF1)
```

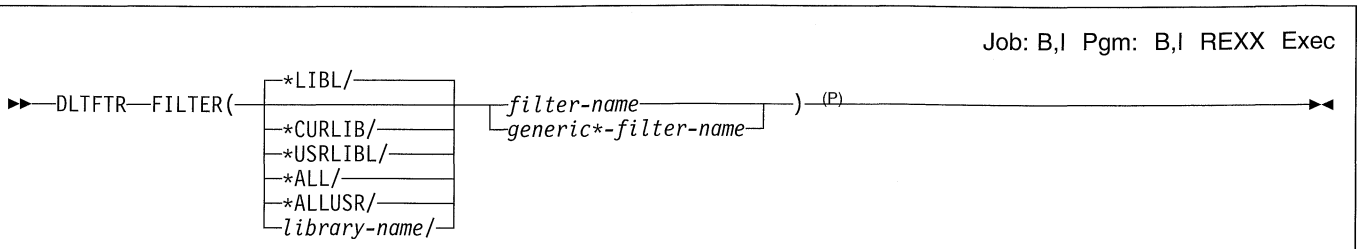
This command deletes all the form definitions named FORMDF1 in all user libraries. Libraries beginning with a Q, except for the QGPL library, are not searched.

Example 3: Deleting Form Definitions That Begin With FD

```
DLTFORMDF  FORMDF(*USERLIBL/FD*)
```

This command deletes all the form definitions that begin with FD in all the user libraries.

DLTFTR (Delete Filter) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Filter (DLTFTR) command deletes a filter object from the specified library.

Required Parameters

FILTER
Specifies the qualified name of the filter being deleted.

The name of the filter can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

filter-name: Specify the name of the filter being deleted.

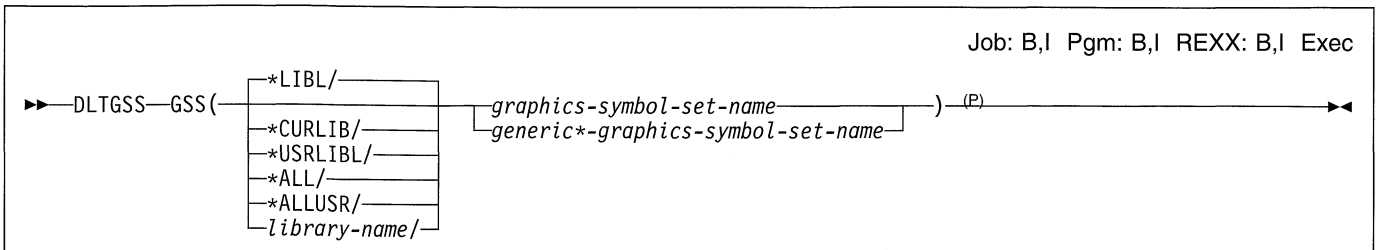
generic-filter-name:* Specify the generic name of the filter being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTFTR   FILTER(MYLIB/MYFILTER)
```

This command deletes filter MYFILTER from library MYLIB.

DLTGSS (Delete Graphics Symbol Set) Command



Note:

| (P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Graphics Symbol Set (DLTGSS) command deletes a graphics symbol set or group of graphic symbol sets from one or more libraries.

Restriction: The user must have object existence authority for the graphics symbol set being deleted.

Required Parameters

GSS

Specifies the qualified name of the graphics symbol set or sets being deleted. A specific graphics symbol set or a generic graphics symbol set can be specified; either type can be optionally qualified by a library name. If no library type is given, *LIBL/ is used to find the graphics symbol set.

| The name of the graphics symbol set can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

graphics-symbol-set-name: Specify the name of the graphics symbol set being deleted.

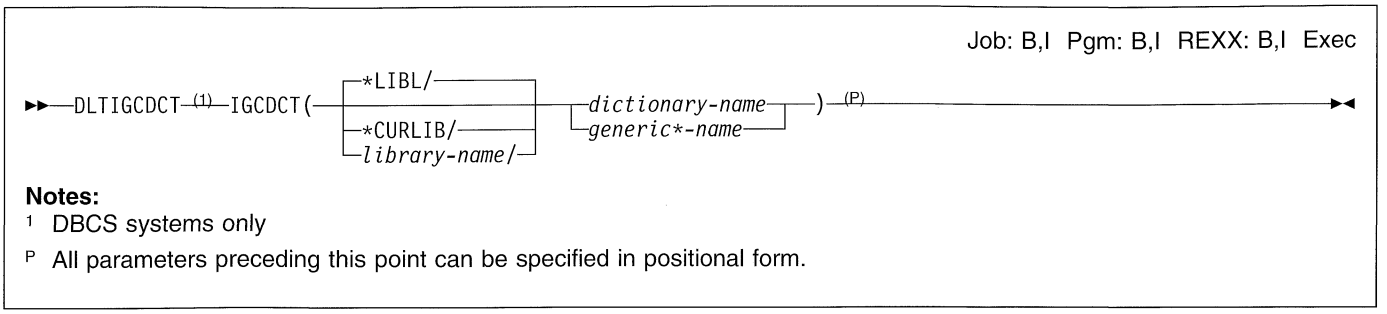
generic-graphics-symbol-set-name:* Specify the generic name of the graphics symbol set that is being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTGSS   GSS(ADMUVTIP)
```

This command deletes graphics symbol set ADMUVTIP from the system.

DLTIGCDCT (Delete DBCS Conversion Dictionary) Command



Notes:
1 DBCS systems only
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete DBCS Conversion Dictionary (DLTIGCDCT) command erases the specified double-byte character set (DBCS) conversion dictionary from the system. The dictionary contains alphanumeric entries and their related words. The system refers to DBCS conversion dictionaries when doing DBCS conversion.

Restrictions: You must have the following authority to use this command:

- Object operational authorities for this command
- Object operational and object existence authorities for the dictionary
- Object operational authorities to the library in which the dictionary is stored

Required Parameters

IGCDCT

Specifies the qualified name of the DBCS conversion dictionary being deleted. If you do not specify a library name, the system deletes the dictionary found when searching the library list.

The name of the dictionary can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

dictionary-name: Specify the DBCS conversion dictionary being deleted.

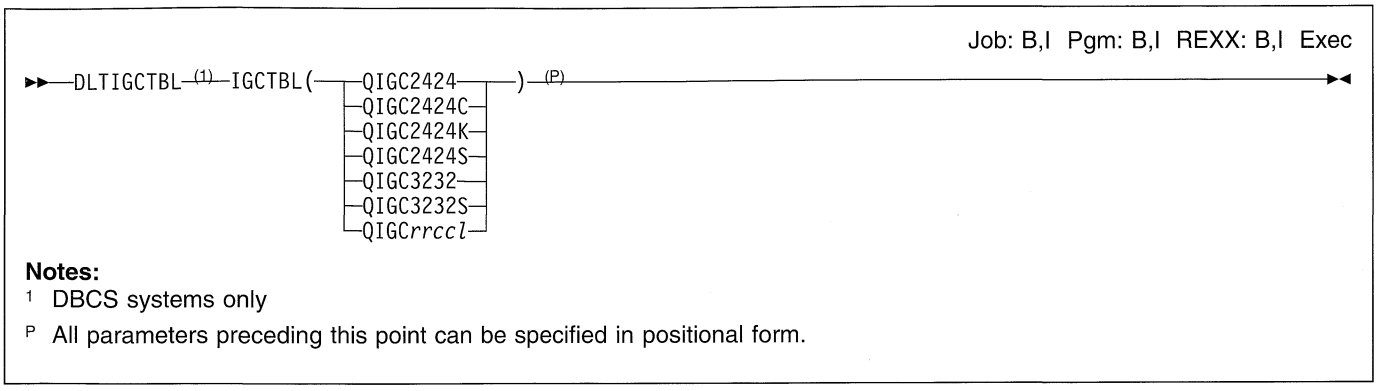
generic-name:* Specify the generic name of the DBCS conversion dictionary being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTIGCDCT   IGCDCT(DBCSLIB/IGCDCT*)
```

This command causes the system to delete each DBCS conversion dictionary whose name starts with the characters IGCDCT in library DBCSLIB.

DLTIGCTBL (Delete DBCS Font Table) Command



Notes:
1 DBCS systems only
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete DBCS Font Table (DLTIGCTBL) command deletes the specified double-byte character set (DBCS) font table from the system. DBCS font tables contain the images of extension characters used on the system and are used for displaying and printing these characters in various matrix sizes, depending on the matrix used by the device.

Consider deleting a table that displays and prints characters in a dot matrix pattern not used by your system. Deleting the table saves system storage. Before deleting a table, it can be copied to tape or diskette for future use. To copy a table to tape or diskette, see the CPYIGCTBL (Copy DBCS Font Table) command.

Required Parameters

IGCTBL

Specifies the name of the DBCS font table being deleted. Choose one of the following table names:

QIGC2424: The Japanese DBCS font table used for displaying and printing extension characters in a 24-by-24 dot matrix image.

QIGC2424C: The Traditional Chinese DBCS font table used for printing extension characters in a 24-by-24 dot matrix image.

QIGC2424K: The Korean DBCS font table used for printing extension characters in a 24-by-24 dot matrix image.

QIGC2424S: The Simplified Chinese DBCS font table used for printing extension characters in a 24-by-24 dot matrix image.

QIGC3232: The Japanese DBCS font table used for displaying and printing extension characters in a 32-by-32 dot matrix image.

QIGC3232S: The Simplified Chinese DBCS font table used for printing extension characters in a 32-by-32 dot matrix image.

QIGCrrccl: The name of the DBCS font table to be deleted must always be in the format QIGCrrccl, where *rr* is the table row matrix size, *cc* is the table column matrix size, and *l* is an optional language identifier.

Example

```
DLTIGCTBL   IGCTBL(QIGC2424)
```

This command deletes the Japanese DBCS font table QIGC2424.

Additional Considerations

Do not delete a DBCS font table if any device attached to the system and currently varied on uses that table. If you delete the table, the system sends you a message identifying the following:

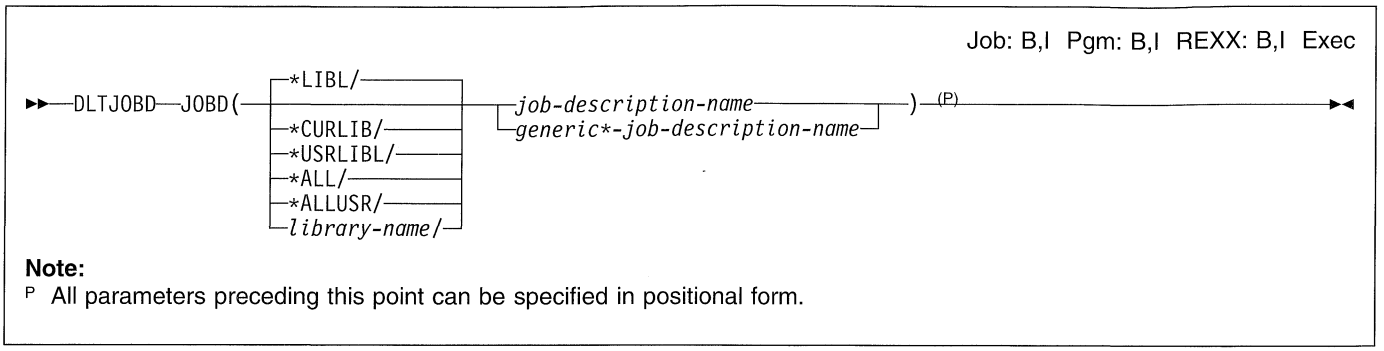
- Devices using the deleted tables
- Devices attached to the same controllers such as devices using the table being deleted
- Controllers that will be damaged the next time you try to print or display extension characters on those devices.

If such a problem occurs, do the following:

1. Vary off the affected devices (VRYCFG command).
2. Vary off the affected control unit.
3. Vary on the affected control unit.
4. Vary on the affected devices.
5. Continue normal system work.

For example, do not delete QIGC2424 when a Japanese 5555 display is varied on.

DLTJOBD (Delete Job Description) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Job Description (DLTJOBD) command deletes the specified job descriptions from the system. Jobs that are running using the specified job description are not affected by its deletion.

Restriction: The user must have object existence authority for the job descriptions being deleted and read authority for the libraries that contain the job descriptions.

Required Parameters

JOBD

Specifies the qualified name of the job descriptions being deleted. A specific job description or a generic job description can be specified. Either type can be optionally qualified by a library name.

The name of the job description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

job-description-name: Specify the name of the job description being deleted.

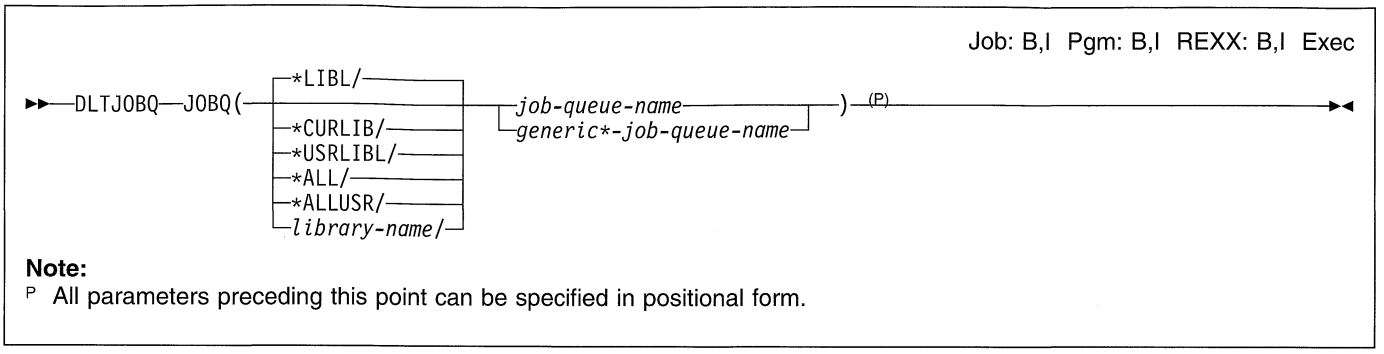
generic-job-description-name:* Specify the generic name of the job description being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTJOBD  JOBD(MYLIB/MYJOBD)
```

This command deletes the job description named MYJOBD from library MYLIB.

DLTJOBQ (Delete Job Queue) Command



Note:

| (P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Job Queue (DLTJOBQ) command deletes the specified job queue or queues from the system.

Restrictions:

1. The job queue being deleted cannot contain any entries; all jobs on the queue must be completed, deleted, or moved to a different job queue.
2. A subsystem cannot be active to the job queue.

Required Parameters

JOBQ

Specifies the (optionally) qualified name of the job queue or queues being deleted. A specific job queue name or a generic job queue name can be specified and either type can be optionally qualified by a library name.

| The name of the job queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

job-queue-name: Specify the name of the job queue being deleted

generic-job-queue-name:* Specify the generic name of the job queue being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

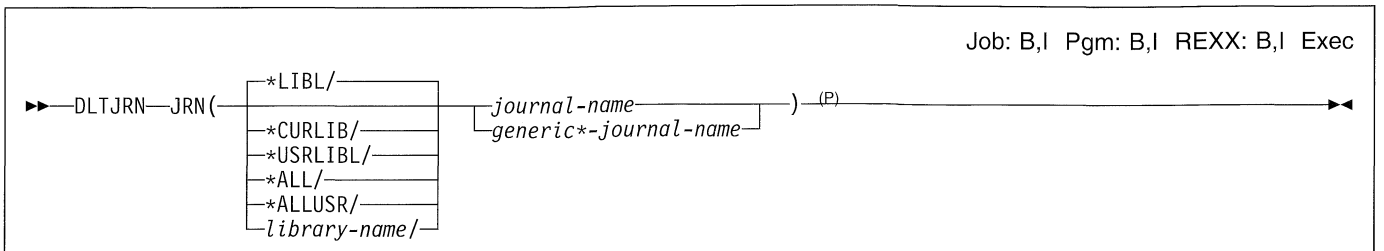
The libraries searched for the job queues being deleted depend on the library qualifier that is specified or assumed.

Example

```
DLTJOBQ  JOBQ(SPECIALJQ)
```

This command deletes the job queue SPECIALJQ from the system.

DLTJRN (Delete Journal) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Journal (DLTJRN) command deletes the specified journal or journals from the system.

Restrictions:

1. Journal entries must not be in the process of being journaled when the command is issued. To determine whether the journal is being used, issue the Work with Journal Attributes (WRKJRNA) command. If any objects are being journaled, issue the End Journaling Access Path (ENDJRNAP) and End Journaling Physical File Changes (ENDJRNPF) commands to end journaling.
2. To delete a journal, the user must have object existence and object operational authority to the journal.

Required Parameter

JRN

Specifies the qualified name of the journal or journals to be deleted. A specific journal name or a generic journal name can be specified; either type of journal name can be optionally qualified by a library name.

Depending on the library name that is specified or assumed, the following libraries are searched for the journals to be deleted:

The name of the journal can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

journal-name: Specify the qualified name of the journal being deleted.

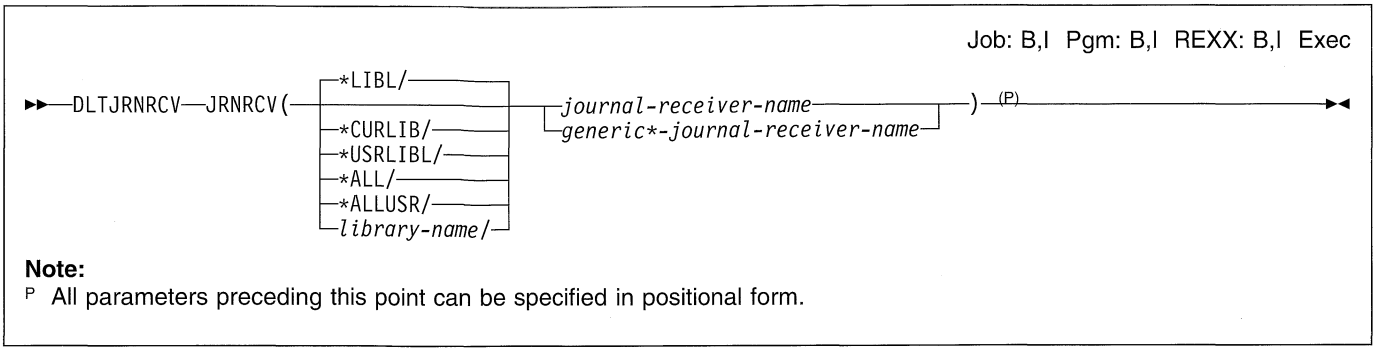
generic-journal-name:* Specify the generic name of the journal being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTJRN    JRN(MYLIB/JRNLA)
```

This command deletes the journal named JRNLA in library MYLIB from the system.

DLTJRNRCV (Delete Journal Receiver) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Journal Receiver (DLTJRNRCV) command deletes the specified journal receivers from the system, which frees the storage space allocated to the journal receivers.

The user can also delete receivers by selecting the delete option on the Work with Receiver Directory display of the Work with Journal Attributes (WRKJRNA) command.

Restrictions:

1. The journal receiver must not be attached to a journal at the time the command is issued.
2. The journal receiver must not be in the middle of a chain of online receivers unless it is damaged. (The receivers must be deleted in the same order in which they were detached, to prevent gaps from occurring in the range of receivers. This restriction does not apply if the journal receiver is damaged or if the second receiver of the pair is not damaged.)
3. If an attempt is made to delete a journal receiver that has not been saved, an inquiry message (CPA7025) is sent to the requesting work station. If the request is made from a batch job, the message is sent to the system operator. This inquiry message can be automatically replied to. For more information, refer to the INQMSGRPY parameter of the Create Job Description (CRTJOBD) command.
4. Users deleting a journal receiver must have object existence, object operational authority, and some data authority to the journal receiver, and object operational authority to the journal attached to the journal receiver.

Required Parameter

JRNRCV

Specifies the qualified name of the journal receivers being deleted. A specific journal receiver or a generic journal receiver name can be specified; either type can be optionally qualified by a library name. If no library name is given, *LIBL is used to find the journal receiver. The system puts receiver names in sequential order so that receivers are not deleted from the middle of a chain.

The name of the journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

journal-receiver-name: Specify the name of the journal receiver being deleted.

generic-journal-receiver-name:* Specify the generic name of the journal receiver being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name.

If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTJRNRCV    JRNRCV(MYLIB/JRNRCLA)
```

This command deletes the journal receiver JRNRCLA in library MYLIB from the system.

DLTLIB (Delete Library) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Library (DLTLIB) command deletes a specified library from the system after all objects in the library have been deleted. If a library being deleted contains objects, this command first deletes all of the objects and then deletes the library. If the user submitting this command does not have the authority to delete every object in the library, only those for which the user has the authority are deleted. In this case, the library and all the other objects in the library do not change. If any object in the library is in use (locked to a program) when this command is entered, the object cannot be deleted.

If a library has been damaged, the user should not delete it without first trying to resolve the damage. In most cases, the damage can be resolved by starting the initial program load (IPL) sequence to rebuild a user library (including the QGPL library), or by reinstalling Operating System/400 to rebuild the QSYS library. Then, if the library is still damaged, it should be deleted. Either a saved version of the library can be restored in its place or the library can be recreated.

Restrictions:

1. To delete a library, the user must have use and object existence authority for the specified library and object existence authority for every object in it. If the user does not have object existence authority for the library, nothing is deleted. If the user does not have object existence authority for one or more objects in the library, those objects and the library are not deleted.
2. A library cannot be deleted if it is in the library list of any job that is active on the system.
3. This command cannot be used to delete the QSYS, QTEMP, QSPL, and QRECOVERY libraries.
4. If the library QDOC is deleted, it is created again during the next IPL.

Required Parameters

LIB

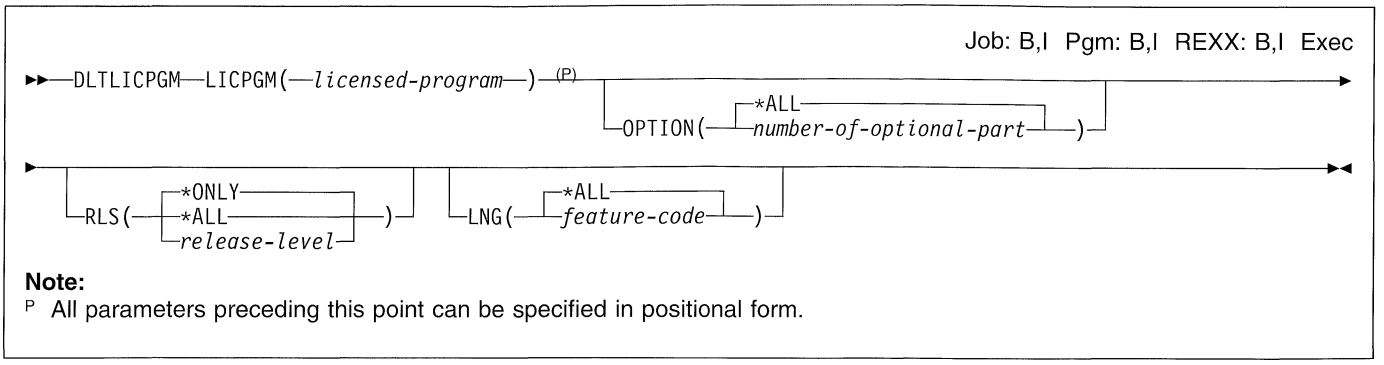
Specifies the name of the library being deleted.

Example

```
DLTLIB   LIB(W)
```

This command deletes library W after all of its objects have been deleted. If library W contains objects and the user has the authority to delete all of those objects, library W and all of the objects are deleted. If the user does not have authority to delete all of the objects, only those for which he has authority are deleted and the library is not deleted.

DLTLICPGM (Delete Licensed Program) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Licensed Program (DLTLICPGM) command is used to delete the objects contained in the licensed program.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. To use this command, you must have *ALLOBJ authority.
3. Some licensed programs can be deleted only if the user is enrolled in the system distribution directory.
4. 5738SS1 *BASE (operating system) and 5738SS1 option 1 (extended base support) cannot be deleted.
5. You must use the LICPGM menu to delete the secondary language library for the operating system and option 1 of the operating system.

Required Parameters

LICPGM

Specifies the 7-character identifier of the licensed program to be deleted. The user can select only one licensed program to delete.

A list of the IBM-supplied licensed programs is the *Licensed Programs and New Release Installation Guide*.

Optional Parameters

OPTION

Specifies which of the parts of the licensed program specified on the LICPGM parameter are deleted.

***ALL:** All of the optional parts and base parts associated with the licensed program listed in the LICPGM parameter are deleted.

Note: If licensed program 5738SS1 is specified, only the optional parts are deleted except Option 1 (Option 10 on a 9406 system). If any other licensed program is specified, all parts are deleted.

number-of-optional-part: Specify the number of the optional part of the listed licensed program being deleted.

RLS

Specifies which version, release, and modification level of the product is deleted.

***ONLY:** Delete the version, release, and modification level installed for the product option.

***ALL:** All installed versions of the product are deleted.

release-level: Specify the release level in the format V*x*R*x*M*y*, where V*x* is the version number, R*x* is the release number, and M*y* is the modification number. Valid values for *x.* are the numbers 0 through 9. Valid values for *y.* are the numbers 0 through 9 and the letters A through Z.

LNG

Specifies which national language version (NLV) objects are deleted for the licensed program specified on the LICPGM parameter.

***ALL:** All NLV objects and program objects for the licensed program specified on the LICPGM, OPTION, and RLS parameters are deleted.

feature-code: Specify the NLV identifier of the language that is deleted. Only the specified NLV is deleted. The program objects for the product are not deleted. The IBM-supplied language feature codes are listed in the *Licensed Programs and New Release Installation Guide* or can be displayed using GO LICPGM, option 20.

Examples

Example 1: Deleting all Licensed Program Objects:

```
DLTLICPGM  LICPGM(5738AP1)
```

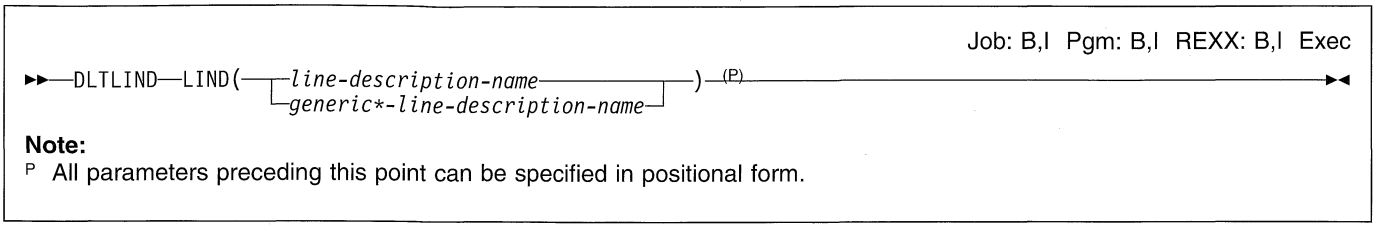
This command deletes all of the objects associated with the 5738AP1 licensed program.

Example 2: Deleting Specific Licensed Program Objects:

```
DLTLICPGM  LICPGM(5738CBI) LNG(2929)
```

This command deletes only the German (NLV 2929) objects for all options of the licensed program 5738CBI.

DLTLIND (Delete Line Description) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Line Description (DLTLIND) command deletes one or more specified line description. The lines identified in the line description must be varied off before this command is issued.

If any controllers are attached to the lines, they (as well as their associated devices) must also be varied off before the DLTLIND command is issued. The controllers are detached when the command is started; a controller cannot be used until it is associated with another line. If a new line description is created, the controllers are either specified when the line is created, or the controllers are attached to other lines that already have line descriptions. In this case, the controller descriptions are deleted and created again, and the new line names are given in the LINE (if just one) or SWTLINLST (if several, for switched network) parameter of each controller command.

Note: If you are deleting a line attached to a frame relay network interface, the network interface (NWI) must also be varied offline.

If a switched line or a nonswitched line with the switched network backup feature is deleted, the name of the deleted line description is automatically removed from every controller description that contains that name in its SWTLINLST parameter.

Required Parameters

LIND

Specifies the name of the line descriptions being deleted. A specific line description or a generic line description is specified.

line-description-name: Specify the name of the line description being deleted.

generic-line-description-name:* Specify the generic name of the line description being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

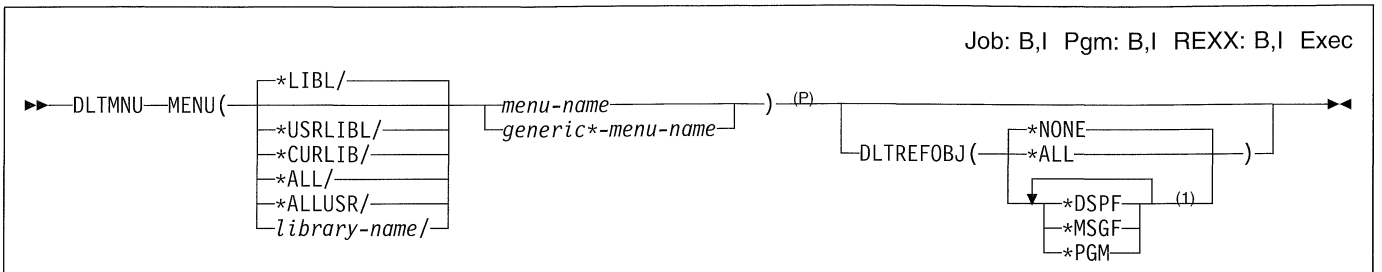
Example

```
DLTLIND   LIND(LINE01)
```

This command deletes the line description of the line named LINE01 from the system.

If the line description being deleted has any controller descriptions associated with it, they are detached and a message containing those controller names is sent to the system operator. The detached controller descriptions are associated with a new line description if their names are specified in the command that creates the line description.

DLTMNU (Delete Menu) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 3 repetitions

Purpose

The Delete Menu (DLTMNU) command deletes the specified menu from the system. Note that this command does not delete the display and message files associated with a Display File (*DSPF) menu, nor does it delete the programs associated with a Program (*PGM) menu.

Restriction: The user must have *OBJEXIST authority for the menu and *USE authority for the library that contains the menu to use this command.

Required Parameters

MENU

Specifies the qualified name of the menu being deleted. A particular menu or a generic menu can be specified, and either type can be optionally qualified by a library name.

The name of the menu can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

menu-name: Specify the name of the menu being deleted.

generic-menu-name:* Specify the generic name of the menu being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Optional Parameters

DLTREFOBJ

Specifies that the display file, message file, or program objects referenced by a menu are deleted. Only the *DSPF menu type specified on the CRTMNU command can reference a display file or a message file, and only the *PGM program type specified on the CRTMNU command can reference a program.

***NONE:** The objects referenced by the menu are not deleted.

***ALL:** All display file, message file, and program objects that are referenced by the menu are to be deleted.

A maximum of three of the following values can be specified:

***DSPF:** The display file that is referenced by the menu is to be deleted.

***MSGF:** The message file that is referenced by the menu is to be deleted.

***PGM:** The program that is referenced by the menu is to be deleted.

Examples

Example 1: Deleting the Menu Only

```
DLTMNU  MENU(FINLIB/ARMENU)
```

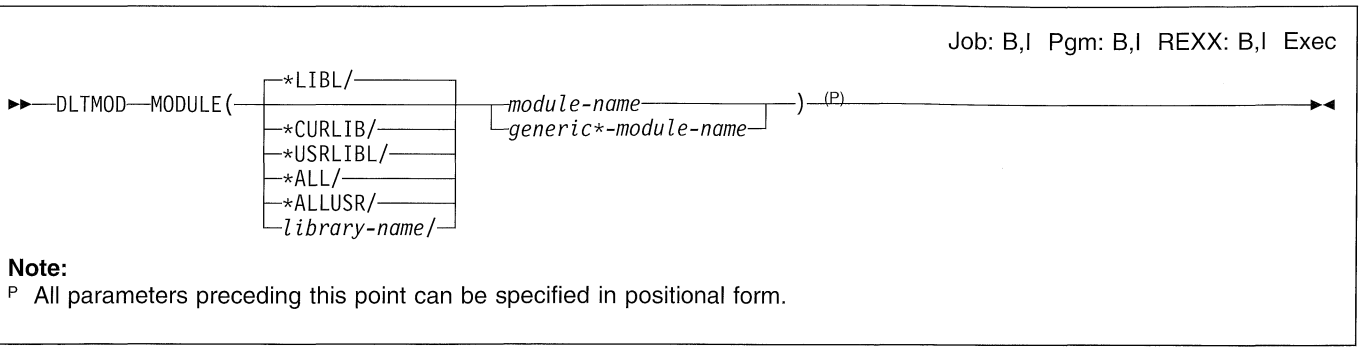
This command deletes the menu named ARMENU from the library named FINLIB.

Example 2: Deleting the Display and Message File

```
DLTMNU  MENU(EXLIB/TEST)  DLTREFOBJ(*DSPF *MSGF)
```

This command deletes the menu named TEST from the library named EXLIB. It also deletes the display file and message file associated with the menu named TEST.

DLTMOD (Delete Module) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Module (DLTMOD) command deletes a compiled module or group of modules. If the module is associated with a binding directory, the entry in the binding directory for this module is not valid after deletion.

Restriction: To use this command, you must have *ALL authority to the module and *READ authority to the library in which the module is stored.

Required Parameters

MODULE

Specifies the qualified name of the modules being deleted. A specific module or a generic module can be specified; either type can be optionally qualified by specifying a library name.

The name of the module can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

module-name: Specify the name of the module being deleted.

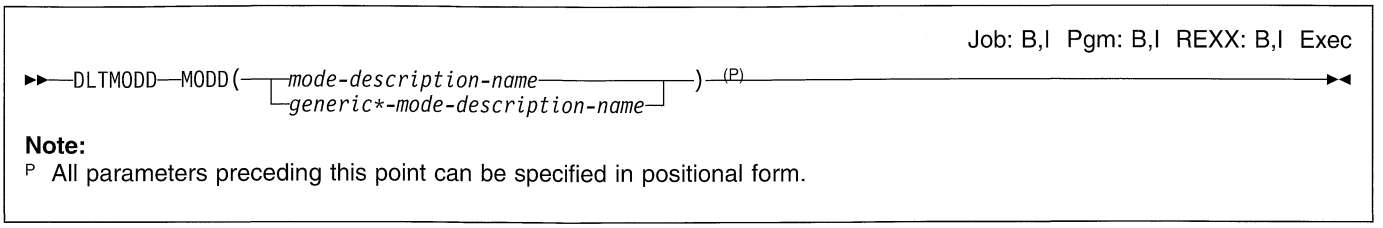
generic-module-name:* Specify the generic name of the module being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, then all modules with names that begin with the generic name, and for which the user has authority, are deleted. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete module name.

Example

```
DLTMOD   MODULE(XYZ/M*)
```

This command deletes all modules in library XYZ that begin with the letter M.

DLTMODD (Delete Mode Description) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Mode Description (DLTMODD) command deletes a mode description.

Required Parameters

MODD

Specifies the name of the mode descriptions being deleted.

mode-description-name: Specify the name of the mode description being deleted.

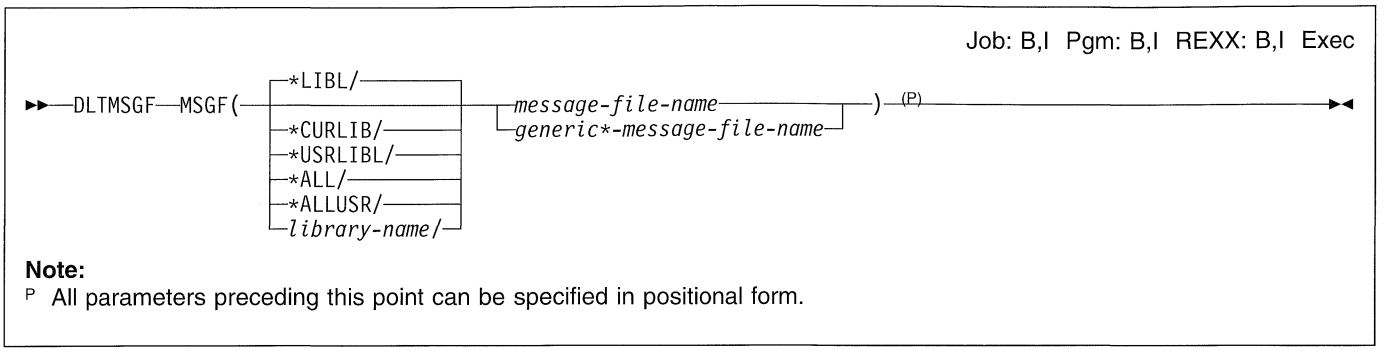
generic-mode-description-name:* Specify the generic name of the mode description being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTMODD   MODD(MODE01)
```

This command deletes the mode description MODE01 from the system.

DLTMSGF (Delete Message File) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Message File (DLTMSGF) command deletes the specified message files from the system, including all the message descriptions stored in the file. If any messages that use this file exist on queues, the file is deleted and no message text is available for those messages.

Restrictions: To delete the specified message file, you must have object existence authority for the file. The IBM-supplied message files, QCPFMSG (for CPF and machine interface messages) and the licensed program message files (such as QRPGMSG), cannot be deleted (unless authorized by the security officer).

Required Parameters

MSGF

Specifies the qualified name of the message files being deleted. A specific message file or a generic message file can be specified; either type can be optionally qualified by a library name.

The name of the message file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

message-file-name: Specify the name of the message file being deleted.

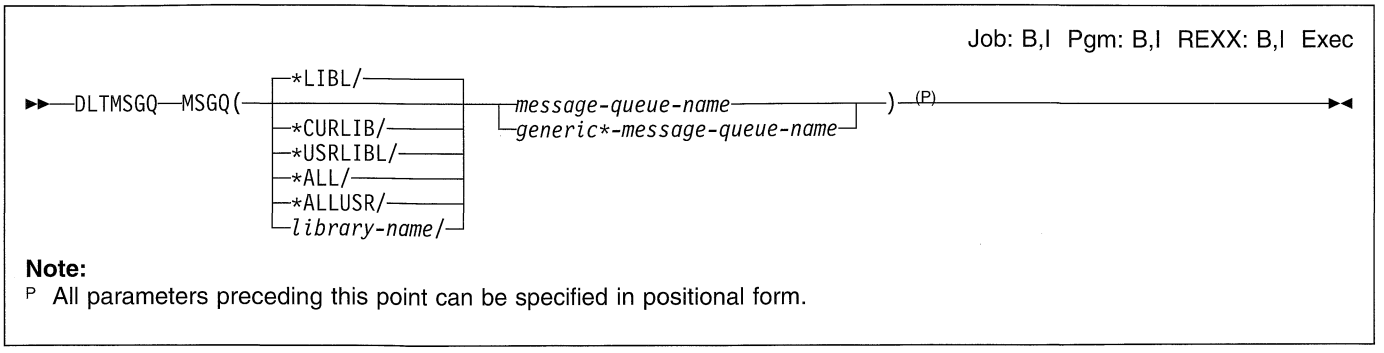
generic-message-file-name:* Specify the generic name of the message file being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTMSGF   MSGF(INV)
```

This command deletes the message file named INV. All message descriptions stored in INV are also removed.

DLTMSGQ (Delete Message Queue) Command



Note:

| (P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Message Queue (DLTMSGQ) command deletes the specified message queues and the messages they contain. Messages in the queue that require a reply are answered with the default reply supplied by that message. Message queues being used by another job cannot be deleted.

| **Restrictions:**

1. The user must have use, existence, and delete authority for the message queue.
2. The user must have read authority to the library in which the message queue is located.
3. The system operator message queue (QSYSOPR) and work station message queues cannot be deleted.

Required Parameters

MSGQ

Specifies the qualified name of the message queues being deleted. A specific message queue or a generic message queue can be specified; either type can be qualified by a library name.

| The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

message-queue-name: Specify the name of the message queue being deleted.

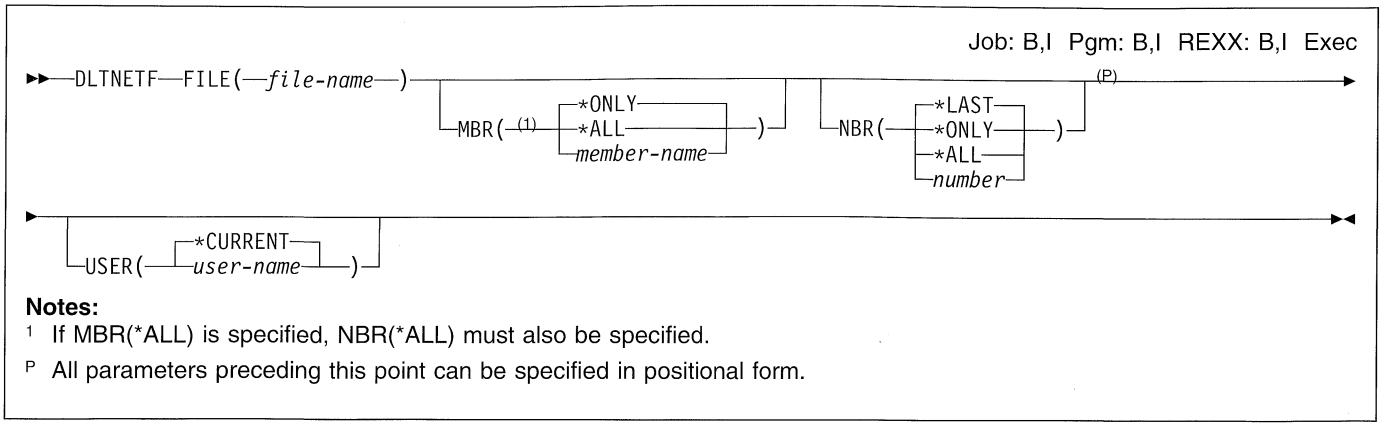
generic-message-queue-name:* Specify the generic | name of the message queue being deleted. A generic | name is a character string of one or more characters fol- | lowed by an asterisk (*); for example, ABC*. The | asterisk substitutes for any valid characters. A generic | name specifies all objects with names that begin with the | generic prefix for which the user has authority. If an | asterisk is not included with the generic (prefix) name, | the system assumes it to be the complete object name. | If the complete object name is specified, and multiple | libraries are searched, multiple objects can be deleted | only if *ALL or *ALLUSR library values can be specified | for the name. For more information on the use of | generic functions, refer to "Rules for Specifying Names."

Example

```
DLTMSGQ   MSGQ(JONES)
```

This command deletes the message queue named JONES. Messages stored in the JONES queue are also removed. The library list is used to find the message queue.

DLTNETF (Delete Network File) Command



Purpose

The Delete Network File (DLTNETF) command deletes a file from a user's queue of network files.

Restriction: A user with security officer authority can delete files sent to any user. Users other than the security officer can delete only those files that are sent to them or to their group profile.

Required Parameter

FILE

Specifies the name of the network file from which one or more members are being deleted. Specify the name of the network file.

Optional Parameters

MBR

Specifies the name of the file member being deleted.

***ONLY:** Only one file member is deleted. This value is valid if there is only one member for the specified network file, or if the NBR parameter is used to uniquely identify a single member.

***ALL:** All file members available for this file are deleted. If MBR(*ALL) is specified, NBR(*ALL) must also be specified.

member-name: Specify the name of the file member being deleted. A file member name cannot be specified if the file is a save file.

NBR

Specifies the number of the network file member that is deleted. This number is used to identify the member deleted when there is more than one member of the same name in the file.

***LAST:** The last copy of the network file member identified by the FILE and MBR parameters is deleted. The last file member is determined as the last member to arrive at the user's system.

Note: The file member that arrived last at the user's system may not have been the last one sent by the sending user. The network does not guarantee the arrival sequence of separately sent files.

***ONLY:** Only one file member of the specified file name is deleted. If there is more than one member of the specified name available, an escape message is sent, and the command is not run.

***ALL:** All members that are identified by the FILE and MBR parameters are deleted. If a member name is specified, all members of that name are deleted. If *ALL is specified for the MBR parameter, all members of the file are deleted.

number: Specify the spool number of the file member being deleted. This value is valid only if a single name is specified on the MBR parameter.

USER

Specifies the user to whom the deleted files were sent.

***CURRENT:** The user profile under which the current job is running is used.

user-name: Specify the name of the user to whom the files were sent. Only users with security officer authority can specify a name other than their own files or group profiles.

Examples

Example 1: Deleting a Single Member

```
DLTNETF  FILE(APPOINTMNT) MBR(FRIDAY)
```

This command deletes member FRIDAY of file APPOINTMNT.

Example 2: Deleting a File

```
DLTNETF  FILE(SCHEDULE) USER(NETUSER1) NBR(708926)
```

This command deletes file SCHEDULE, number 708926, from the network files for user NETUSER1. This command

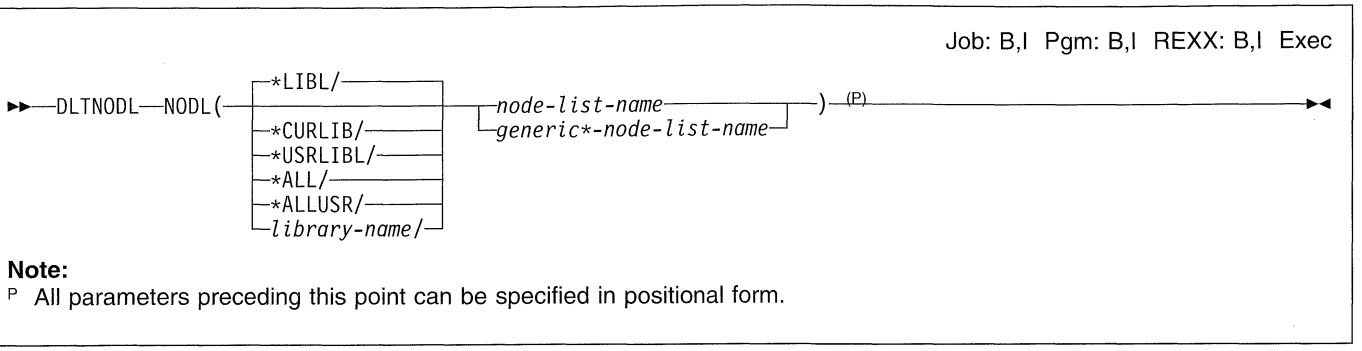
is run only by NETUSER1, a member of the NETUSER1 group, or a user with security officer authority.

Example 3: Deleting All Members

```
DLTNETF  FILE(OLDINFO) MBR(*ALL) NBR(*ALL)
```

This command deletes all available members for file OLDINFO, including all duplicated named members.

DLTNODL (Delete Node List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Node List (DLTNODL) command deletes a node list from a specified library.

Required Parameters

NODL

Specifies the qualified name of the node list being deleted.

The name of the node list can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

node-list-name: Specify the name of the node list being deleted.

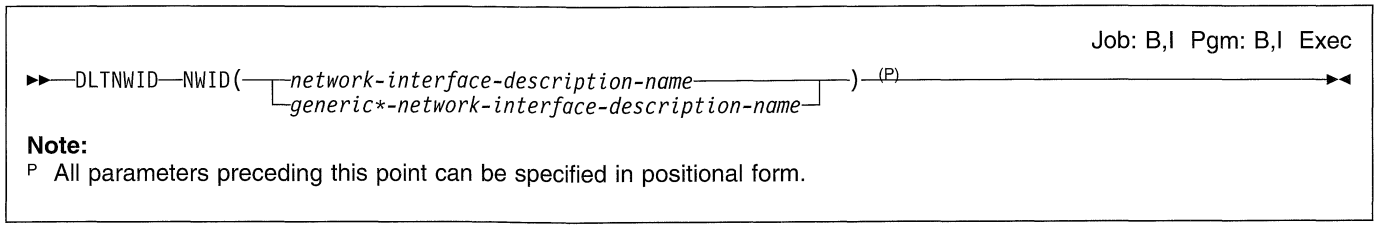
generic-node-list-name:* Specify the generic name of the node list being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTNODL   NODL(MYLIB/NODL*)
```

This command deletes all node lists in library MYLIB whose names begin with NODL.

DLTNWID (Delete Network Interface Description) Command



Purpose

The Delete Network Interface Description (DLTNWID) command deletes specified network interface descriptions. The network interface identified in the network interface description must be varied off before this command is issued.

Lines attached to the network interface and their associated controllers and devices must be varied off before this command is issued.

The line descriptions are detached when the command is started; the detached line descriptions cannot be used until they are associated with another network interface description. If a new network interface description is created, the detached line descriptions can be specified on the command that creates the network interface description. If the detached line descriptions must be attached to existing network interface descriptions, the detached line descriptions must be deleted and created again (or copied), and the new network interface names are given in the NWI (for non-switched attachments) or SWTNWILIST (for switched attachments) parameters of the line description command.

Required Parameters

NWID

Specifies the names of the network interface descriptions being deleted.

network-interface-description-name: Specify the name of the network interface description being deleted.

generic-network-description-name:* Specify the generic name of the network description. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

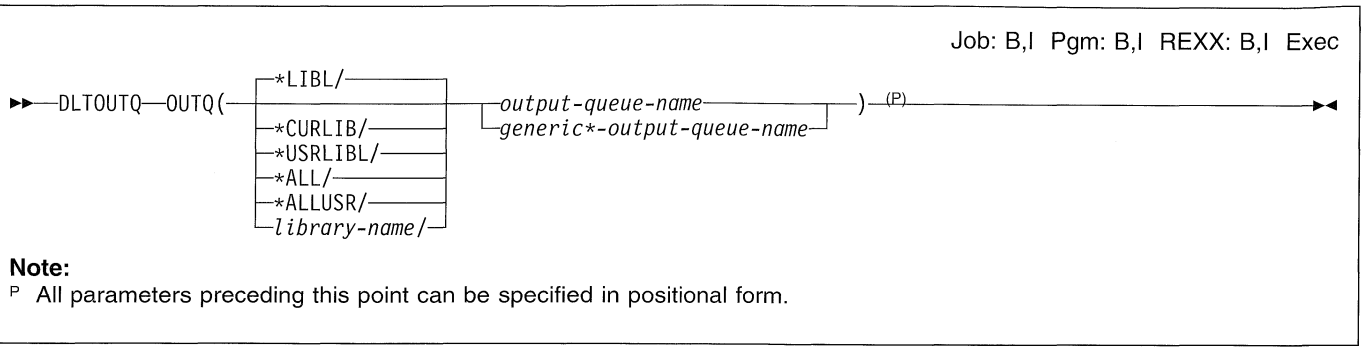
Example

```
DLTNWID   NWID(ISDNNET)
```

This command deletes the network interface description of the network interface named ISDNET from the system.

If the deleted network interface description has any line descriptions associated with it, they are detached and a message containing those line description names is sent to the system operator. The detached line descriptions are then associated with a new network interface description if their names are specified on the command that creates the network interface description.

DLTOUTQ (Delete Output Queue) Command



Note:

| (P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Output Queue (DLTOUTQ) command deletes the specified output queue or queues from the system.

Restrictions:

1. The output queue being deleted cannot contain any entries: the output for each file must be printed, deleted, or moved to a different output queue.
2. A subsystem cannot be active.
3. The queue cannot be in use by a spooling writer.
4. The queue cannot be deleted if it has been created by the system for a specific printer.

Required Parameters

OUTQ

Specifies the qualified name of the output queue.

The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

output-queue-name: Specify the name of the output queue being deleted.

generic-output-queue-name:* Specify the generic name of the output queue being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

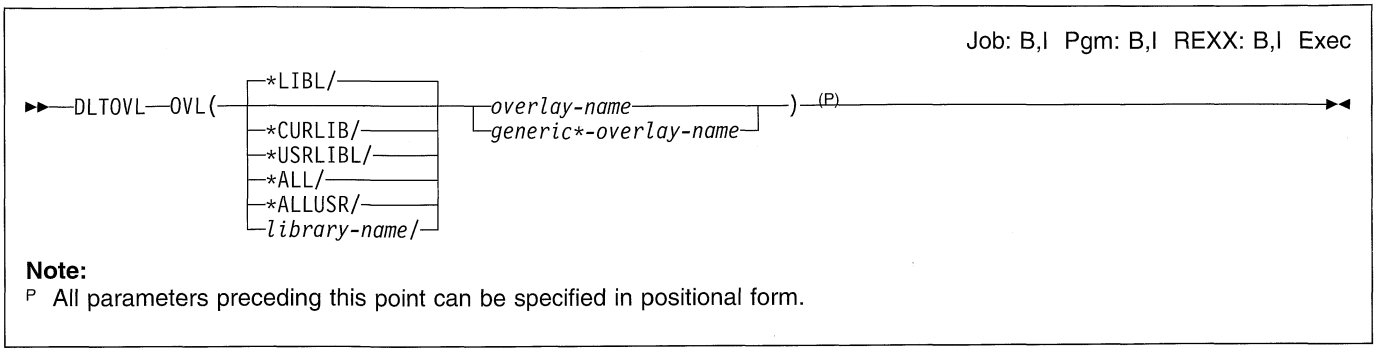
The libraries searched for the output queues being deleted depend on the library qualifier that is specified or assumed.

Example

```
DLTOUTQ  OUTQ(PUNCH2)
```

This command deletes the output queue PUNCH2 from the system.

DLTOVL (Delete Overlay) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Overlay (DLTOVL) command deletes a specified overlay from the specified library. If the overlay is found, it is deleted. If the specified overlay is not found, a message is sent to the user stating that it cannot be found.

Required Parameters

OVL

Specifies the qualified name of the overlay being deleted.

| The name of the overlay can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

overlay-name: Specify the name of the overlay being deleted.

generic-overlay-name:* Specify the generic name of the overlay being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting Overlays in All Libraries

```
DLTOVL    OVL(*ALL/OVL1)
```

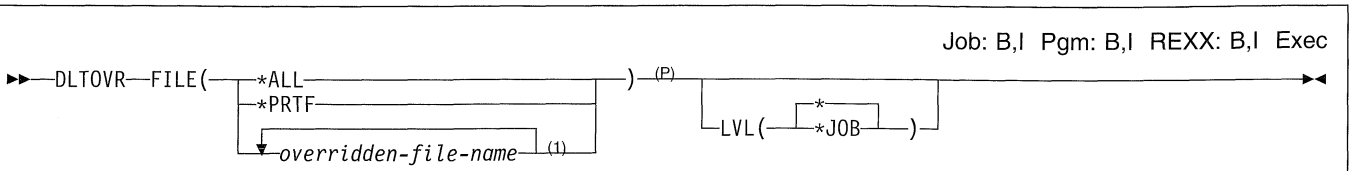
This command deletes all the overlays named OVL1 in all the libraries, including all IBM-supplied libraries.

Example 2: Deleting Overlays in Current Library

```
DLTOVL    OVL(*CURLIB/OV*)
```

This command deletes all the overlays that begin with OV in the current library.

DLTOVR (Delete Override) Command



Notes:
1 A maximum of 50 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Override (DLTOVR) command deletes one or more file overrides (including message file overrides) that were previously specified in a call level. That is, for each overridden file named in the DLTOVR command, the override specified in the same call level as the DLTOVR command is deleted. When the command is specified interactively or outside a program in a batch job, the file overrides for the call level are deleted; when the command is used in a CL program, the file overrides for that program call level are deleted. A file override is the result of an override file command.

The DLTOVR command can delete all the file overrides for all the files in the same call level or the file overrides for specified files in the same call level. Only the file overrides in the call level in which the command is specified are deleted. For example, if an override command is specified in one program in a routing step, and then another program is called that also contains override commands, a DLTOVR command specified in the second program can delete only overrides that occur in that program. The DLTOVR command has no effect on the override command that was specified before the program was called. The deleted file overrides have no effect on subsequent uses of the file.

Required Parameter

FILE

Specifies the names of the overridden files in the call level whose file overrides are being deleted. One or more overridden files can be specified by name, or all files can be specified.

***ALL:** All of the file overrides that exist in the call level where this command is specified are deleted.

***PRTF:** The *PRTF file override that exists in the call level where this command is specified is deleted.

overridden-file-name: Specify the names of one or more overridden files for which the overrides in the call level are deleted.

Optional Parameter

LVL

Specifies the call level of the file overrides deleted. There is a one-to-one correspondence between the call stack entries shown on the call stack from the Work with Job (WRKJOB) command and the call level for that call stack entry.

The first call stack entry name on the call stack (at the top of the list) is the call program or procedure at call level one. The second call stack entry name is the program or procedure at call level two. The last call stack entry name is the program or procedure at the highest call level for the job.

*: The call level of the file overrides to be deleted is the call level of the program that called the DLTOVR command processing program. If the DLTOVR command is called through QCMDEXC, the call level is the same level as that of the caller of QCMDEXC.

***JOB:** The call levels for the file overrides that are scoped to the job are deleted.

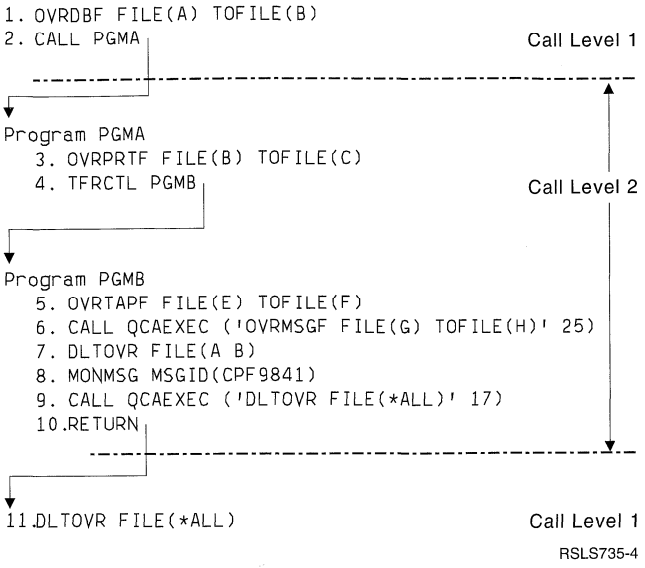
Examples

Example 1

```
1. OVRDBF   FILE(A)  TOFILE(B)
2. OVRPRTF  FILE(C)  TOFILE(D)
3. OVRTAPF  FILE(E)  TOFILE(F)
     •
     •
     •
4. DLTOVR   FILE(A C)
5. DLTOVR   FILE(*ALL)
```

If the first three override commands had been specified earlier in the call level, the files B, D, and F would override files A, C, and E. The fourth command deletes only the file overrides that affect files A and C. The last command deletes all the file overrides that exist in the call level, which in this case is the command overriding file E, the third command.

Example 2



This example assumes that commands 1, 2, and 11 are specified in the same call level, call level number 1. The rest of the commands are in call level 2.

Command 1 causes an override in call level 1 from file A to file B. Command 2 calls PGMA and creates another call level, call level 2.

In program PGMA, command 3 causes an override in call level 2 from file B to file C. Command 4 transfers control from PGMA to PGMB in the same call level, call level 2.

Unlike the CALL command, the TFRCTL command does not create a new call level.

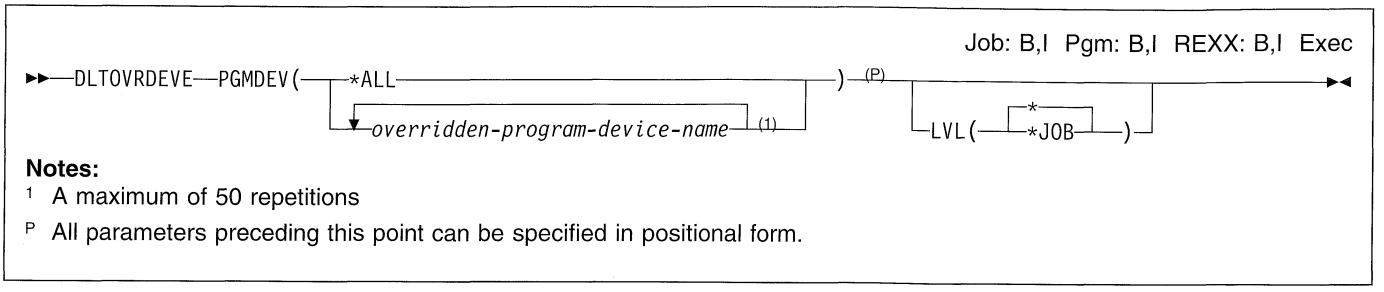
In program PGMB, command 5 causes an override in call level 2 from file E to file F. Command 6 calls QCAEXEC and causes an override from file G to file H. When it is called, QCAEXEC runs as though it is just another command in PGMB, rather than running as a called program. That is, QCAEXEC runs in the same call level (call level 2); it does not create another call level (call level 3).

Command 7 deletes any overrides affecting files A and B in call level 2. In this case, the override specified by command 3 is deleted, but the override specified by command 1 is not. Because an override for file A is not found in call level 2, the escape message CPF9841 (override not found) is sent to PGMB. To prevent a function check, a Monitor Message (MONMSG) command is needed after the DLTOVR command. In this example, command 8 monitors for CPF9841, but specifies no action to be taken if the message is sent. Therefore, when CPF9841 is received, it is monitored and ignored by command 8, and control is passed to the next command in the program.

Command 9 deletes all remaining overrides in call level 2. The overrides specified by commands 5 and 6 are deleted, but the override specified by command 1 is not.

Command 10 causes a return to call level 1, and call level 2 is ended. If any overrides had been specified in call level 2 that were not deleted by DLTOVR commands, they are deleted when call level 2 ends. Command 11 causes all remaining overrides in call level 1 to be deleted; the override specified by command 1 is deleted.

DLTOVRDEVE (Delete Override Device Entry) Command



Purpose

The Delete Override Device Entry (DLTOVRDEVE) command deletes one or more program device overrides that were previously specified in a call level.

For each overridden program device named in the DLTOVRDEVE command, the override specified in the same call level as the DLTOVRDEVE command is deleted. When the command is entered interactively or outside a program in a batch job, the program device overrides for the call level are deleted; when the command is used in a CL program, the program device overrides for that program call level are deleted. A program device override is the result of an override program device command Override Intersystem Communications Function Device Entry (OVRICFDEVE).

The DLTOVRDEVE command can delete all the program device overrides in the same call level, or it can delete a specific program device override in the same call level. Only the call level in which the command is entered has its program device overrides deleted. For example, if an override command is entered in one program in a routing step, and then another program is called that also contains override commands, a DLTOVRDEVE command entered in the second program can delete only overrides that occurred in that program. The DLTOVRDEVE command has no effect on the override command that was entered before the program was called. The deleted program device override has no effect on subsequent uses of the program device.

Required Parameter

PGMDEV

Specifies program devices whose overrides can be specified and deleted.

***ALL:** All program device overrides that exist in the call level where this command is entered are deleted.

overridden-program-device-name: Specify the names of one or more overridden program devices for which the overrides in the call level are deleted.

Optional Parameter

LVL

Specifies the call level of the program device overrides to be deleted. There is a one-to-one correspondence between the call stack entries shown on the call stack from the Work with Job (WRKJOB) command and the call level for that call stack entry.

The first call stack entry name on the call stack (at the top of the list) is the call program or procedure at call level one. The second call stack entry name is the program or procedure at call level two. The last call stack entry name is the program or procedure at the highest call level for the job.

*: The call level of the program device overrides to be deleted is the call level of the program that called the DLTOVRDEVE command processing program. If the DLTOVRDEVE command is called through QCMDEXC, the call level is the same level as that of the caller of QCMDEXC.

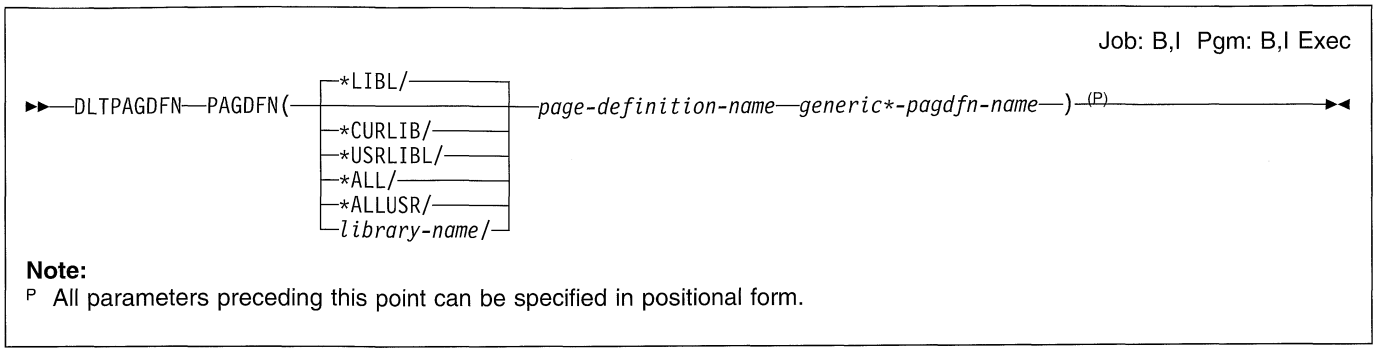
***JOB:** The call levels for the program device overrides that are scoped to the job are deleted.

Example

```
DLTOVRDEVE    PGMDEV(*ALL)
```

This command deletes all the program device overrides that exist in the call level.

DLTPAGDFN (Delete Page Definition) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Page Definition (DLTPAGDFN) command deletes a page definition from the specified library. If the page definition is not found, a message is sent to the user stating that the page segment could not be found.

Restriction: The user must have *OBJEXIST authority to delete an object.

Required Parameters

PAGDFN

Specifies the qualified name of the page definition to be deleted. The user can specify a particular page definition or a generic page definition, and either one can be qualified by a library name.

The name of the page definition can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

page-definition-name: Specify the name of the page definition to be deleted.

generic-page-definition-name:* Specify the generic name of the page definition. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting a Page Definition from the Current Library

```
DLTPAGDFN   PAGDFN(*CURLIB/P1DFLT)
```

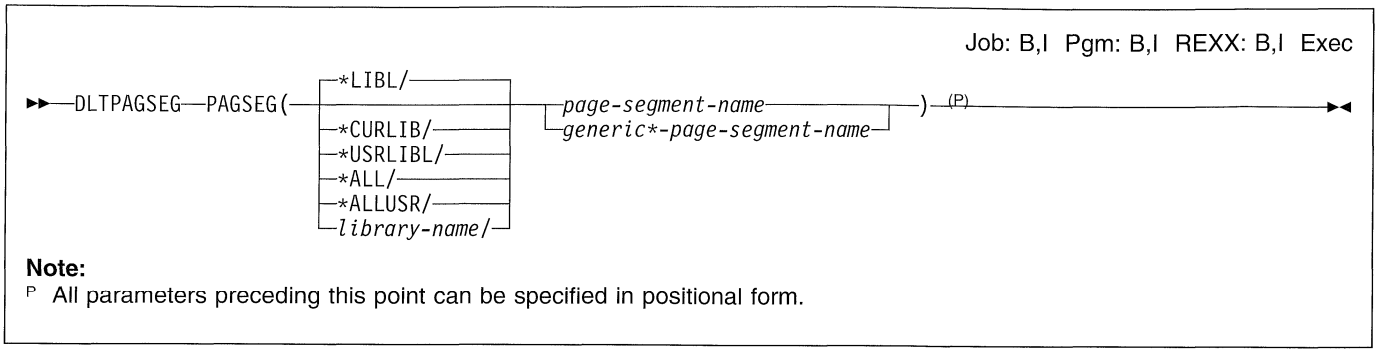
This command deletes page definition P1DFLT if it is found in the current library.

Example 2: Deleting Page Definitions

```
DLTPAGDFN   PAGDFN(*LIBL/P1*)
```

This command deletes all page definitions that begin with P1 in the jobs library list.

DLTPAGSEG (Delete Page Segment) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Page Segment (DLTPAGSEG) command deletes a specified page segment. If the page segment is found, it is deleted. If the page segment is not found, a message is sent to the user stating that it cannot be found.

Required Parameters

PAGSEG

Specifies the qualified name of the page segment being deleted.

The name of the page segment can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

page-segment-name: Specify the name of the page segment being deleted.

generic-page-segment-name:* Specify the generic name of the page segment being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Examples

Example 1: Deleting a Page Segment in the Library List

```
DLTPAGSEG  PAGSEG(*LIBL/PAGSEG1)
```

This command deletes the page segment named PAGSEG1 if it is in the library list. Only the first occurrence is deleted if there is more than one library containing PAGSEG1 in the library list.

Example 2: Deleting a Page Segment in All Libraries

```
DLTPAGSEG  PAGSEG(*ALL/PAGSEG1)
```

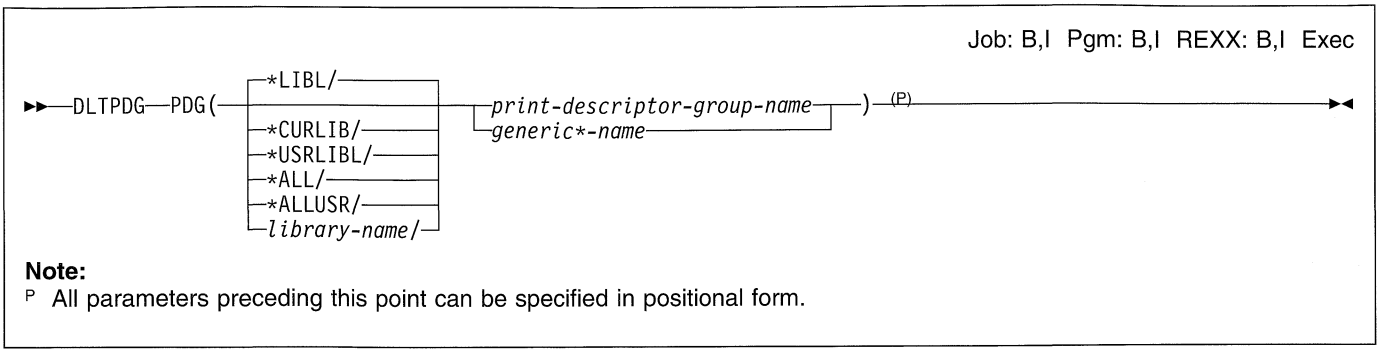
This command deletes all the page segments named PAGSEG1, including any IBM supplied page segments with that name.

Example 3: Deleting Page Segments in Current Library

```
DLTPAGSEG  PAGSEG(*CURLIB/PG*)
```

This command deletes all the page segments whose names begin with PG that are in the job's current library.

DLTPDG (Delete Print Descriptor Group) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Print Descriptor Group (DLTPDG) command deletes a space in which information about a print descriptor group and its associated print descriptor names is stored.

Restriction: The user of this command must have *OBJEXIST authority to the print descriptor group.

Required Parameters

PDG

Specifies the qualified name of the print descriptor group being deleted.

| The name of the print descriptor group can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

print-descriptor-group-name: Specify the name of the print descriptor group to be deleted.

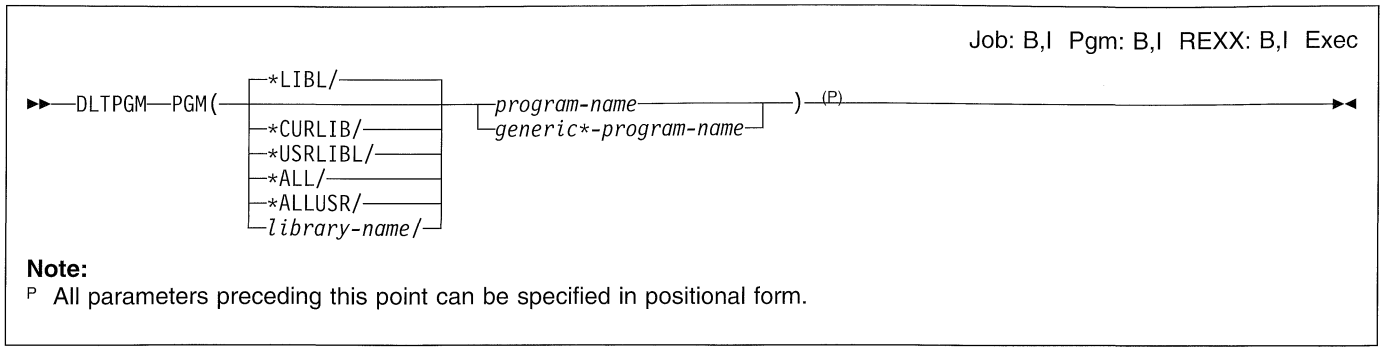
generic-name:* Specify the generic name of the print descriptor group being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTPDG  PDG(P3820*)
```

This command deletes all print descriptor groups that start with P3820*.

DLTPGM (Delete Program) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Program (DLTPGM) command deletes a compiled program or group of programs. If the program is currently being run, the program processing is abnormally ended when this command is issued unless a specific Allocate Object (ALCOBJ) command is in effect in another job. Any high-level language (HLL) or CL program can be deleted.

Restrictions:

1. The user of this command must have *ALL authority to the program, and *READ authority to the library where it is stored.
2. If the program being deleted is currently being debugged, a function check occurs if an implicit reference is made to the deleted program (for example, if a Change Variable (CHGVAR) command specifies PGM(*DFTPGM)). To prevent function checks, use the Remove Program (RMVPGM) command to remove the program from the debugging session before deleting it. If the program is recompiled while the user is in debug mode, remove the program from debug mode (using the RMVPGM command), delete it from the system (using the Delete Program (DLTPGM) command), change and recompile the program, and add the new version of the program to debug mode (using the Add Program (ADDPGM) command).

Required Parameters

PGM

Specifies the qualified name of the programs being deleted. A specific program or a generic program can be specified; either type optionally can be qualified by specifying a library name.

| The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program being deleted.

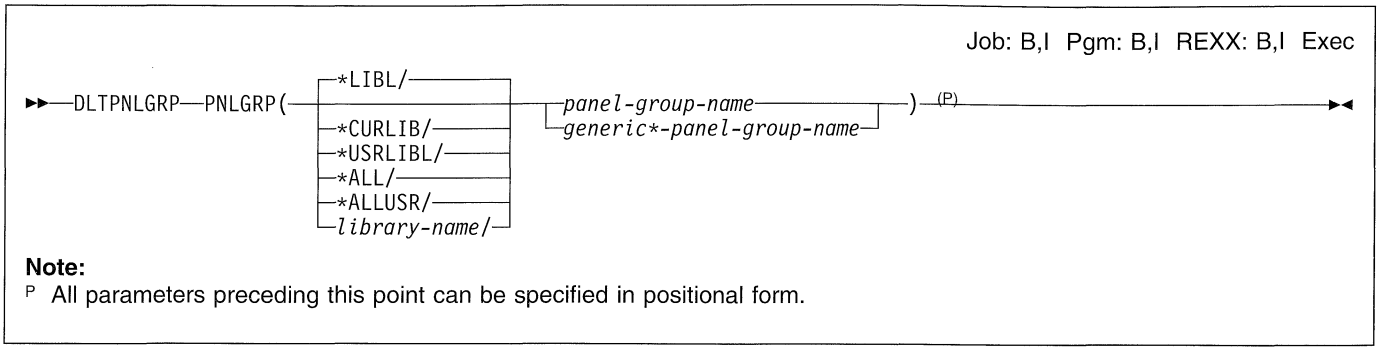
generic-program-name:* Specify the generic name of the | program being deleted. A generic name is a character string | of one or more characters followed by an asterisk (*); for | example, ABC*. The asterisk substitutes for any valid char- | acters. A generic name specifies all objects with names that | begin with the generic prefix for which the user has authority. | If an asterisk is not included with the generic (prefix) name, | the system assumes it to be the complete object name. If | the complete object name is specified, and multiple libraries | are searched, multiple objects can be deleted only if *ALL or | *ALLUSR library values can be specified for the name. For | more information on the use of generic functions, refer to | "Rules for Specifying Names."

Example

```
DLTPGM  PGM(LIB1/PROG1)
```

This command deletes the program PROG1 from the library LIB1 if the user has the proper authority for the program and library.

DLTPNLGRP (Delete Panel Group) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete Panel Group (DLTPNLGRP) command removes the specified panel group from the system.

Restriction: The user of this command must have *OBJEXIST authority for the panel group and *USE authority for the library that contains the panel group.

Required Parameters

PNLGRP

Specifies the qualified name of the panel group being deleted. A particular panel group or a generic panel group can be specified, and either type optionally can be qualified by a library name.

The name of the panel group can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

panel-group-name: Specify the name of the panel group being deleted.

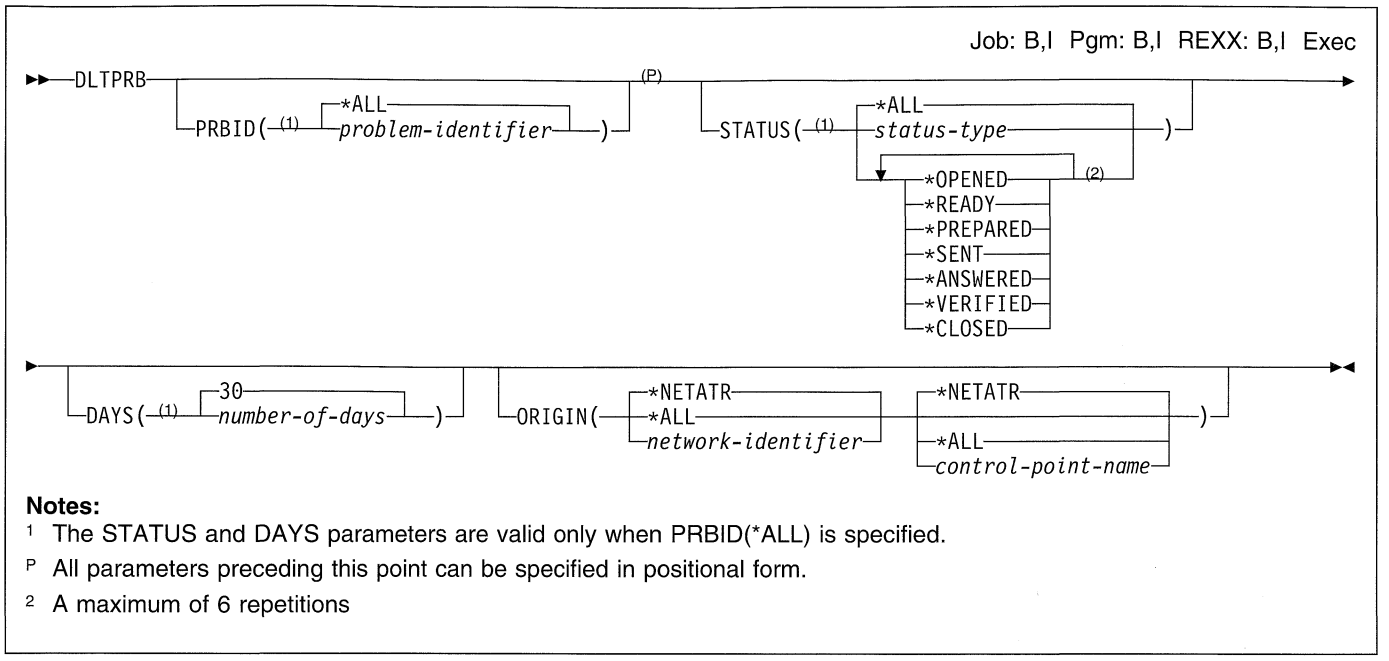
generic-panel-group-name:* Specify the generic name of the panel group being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTPNLGRP  PNLGRP(*CURLIB/AR*)
```

This command deletes all panel groups with names that contain the AR prefix (for example, ARINV, ARREC, ARPERS) from the library listed as the current library.

DLTPRB (Delete Problem) Command



Notes:
1 The STATUS and DAYS parameters are valid only when PRBID(*ALL) is specified.
P All parameters preceding this point can be specified in positional form.
2 A maximum of 6 repetitions

Purpose

The user, typically a system operator or user responsible for handling system problems, uses the Delete Problem (DLTPRB) command to maintain the size of the problem log by deleting one or more entries. More information on controlling problem log size is in the *Operator's Guide*.

Restriction: This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.

Optional Parameters

PRBID

Specifies the problem identifier of the problem being deleted from the problem log. Problems with different system origins can have the same identifier. This parameter can be used with the ORIGIN parameter to select a single problem for deletion from a particular system origin.

***ALL:** All problems are deleted.

problem-identifier: Specify the 10 character problem identifier of the problem being deleted. If a problem identifier is specified, the STATUS and DAYS parameters are ignored.

STATUS

Specifies the status of the problem log entries. The seven status types are as follows:

*OPENED	The problem is in OPENED status. The problem has been identified and a problem record has been created.
*READY	The problem is in READY status. Problem analysis information has been added to the problem record.
*PREPARED	The problem is in PREPARED status. The problem has been prepared for reporting.
*SENT	The problem is in SENT status. The problem has been sent to a service provider, but no answer has been returned.
*ANSWERED	The problem is in ANSWERED status. An answer was returned by the service provider or added by an operator on this system.
*VERIFIED	The problem is in VERIFIED status. The problem has been resolved and the system operator has verified that the problem has been corrected.
*CLOSED	The problem is closed.

Note: This parameter is valid only when PRBID(*ALL) is specified.

***ALL:** All problems that match the user-defined values on this command and any current status types are deleted.

status-type: Specify one of the seven status types. Up to six status types can be specified.

DAYS

Specifies that all problems older than the value specified on this parameter are deleted. Problems with a status type other than CLOSED are deleted the specified number of days after the OPENED date. Problems with a CLOSED status type are deleted after the number of days following the CLOSED date.

This parameter is valid only when PRBID(*ALL) is specified.

30: The problem is deleted after 30 days.

number-of-days: Specify the number of days that can elapse before a problem is deleted. Valid values range from 0 through 999 days. If this value is smaller than the system value, QPRBHLDITV, the system value is used.

ORIGIN

Specifies whether only problem log entries originating on the specified nodes are deleted.

Element 1: Network Identifier

***NETATR:** Only entries originating on systems with the same network identifier as the one defined in the network attributes are deleted.

***ALL:** All entries are deleted regardless of the user's system origin network identifier.

network-identifier: Specify a network identifier. Only entries originating on systems with the specified network identifier are deleted.

Element 2: Control Point Name

***NETATR:** Only entries originating on systems with the same control point name as the one defined in the network attributes are deleted.

***ALL:** All entries originating on systems with the network identifier are deleted.

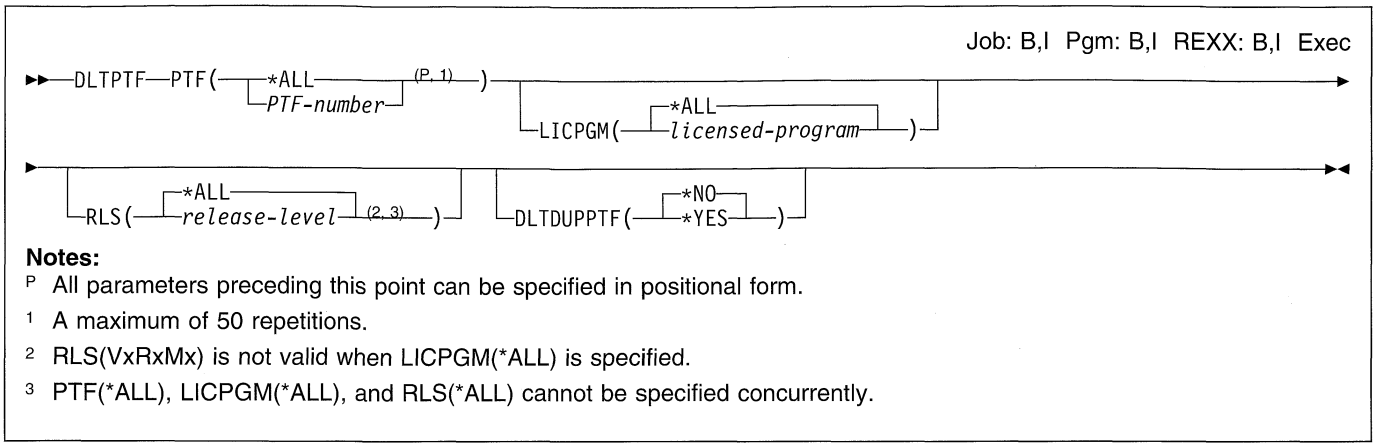
control-point-name: Specify a control point name. Only entries originating on systems with the specified control point name are deleted.

Example

```
DLTPRB  STATUS(*OPENED *READY *SENT) DAYS(15)
```

This command deletes all entries in the OPENED, READY, or SENT status that were added to the problem log prior to 15 days ago.

DLTPTF (Delete PTF) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 50 repetitions.

2 RLS(VxRxMx) is not valid when LICPGM(*ALL) is specified.

3 PTF(*ALL), LICPGM(*ALL), and RLS(*ALL) cannot be specified concurrently.

Purpose

The Delete Program Temporary Fix (DLTPTF) command is used to delete program temporary fix (PTF) save files, their associated cover letters, and the records of PTFs that have been ordered.

Required Parameters

PTF

Specifies the PTF identification number of the PTF to be deleted.

***ALL:** All PTFs are deleted.

PTF-number: Specify the identification number of the PTF to be deleted.

LICPGM

Specifies the 7-character identifier of the licensed program for which the PTFs are to be deleted.

***ALL:** All PTFs are deleted.

licensed-program: Specify the 7-character licensed program identifier.

RLS

Specifies the release level of the PTFs being deleted.

***ALL:** All PTFs are deleted.

release-level: Specify the release level in the format VxRxMx, where Vx is the version number, Rx is the release number, and Mx is the modification number.

DLTDUPPTF

Specifies whether duplicate PTF numbers are deleted.

***NO:** Duplicate PTF numbers are not deleted.

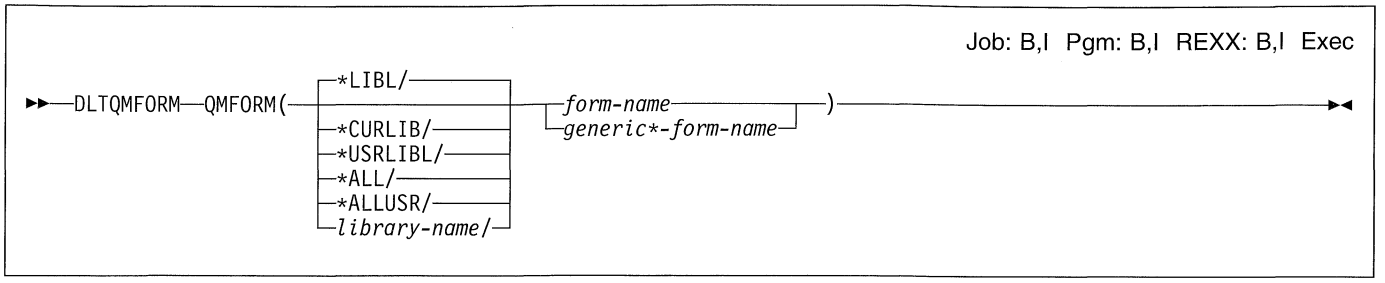
***YES:** Duplicate PTF numbers are deleted.

Example

```
DLTPTF  PTF(MYPTF)
```

This command deletes the save file of the PTF named MYPTF from the QGPL library. Any members in the cover letter file for the PTF named MYPTF are also deleted.

DLTQMFORM (Delete Query Management Form) Command



Purpose

The Delete Query Management Form (DLTQMFORM) command allows the user to delete an existing query management form from a library.

Required Parameters

QMFORM

Specifies the name of the query management form being deleted.

The name of the query management form can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

form-name: Specify the name of the form being deleted.

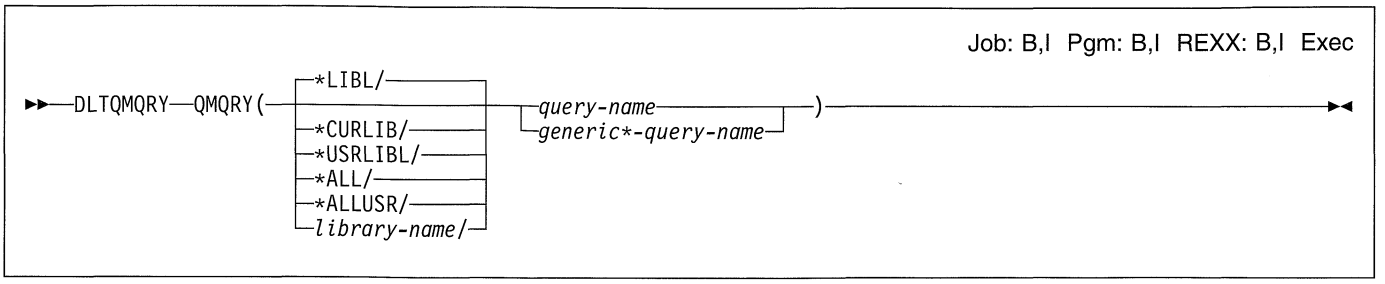
generic-form-name:* Specify the generic name of the form being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, all forms with names that begin with the generic name, and for which the user has authority, are deleted from the specified library or library list. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete form name.

Example

```
DLTQMFORM  QMQRY(RPTLIB/FORM1)
```

This command deletes query management form FORM1 from library RPTLIB.

DLTQMQRY (Delete Query Management Query) Command



Purpose

The Delete Query Management Query (DLTQMQRY) command deletes an existing query management query from a library.

Required Parameters

QMQRY

Specifies the name of the query management query being deleted.

The name of the query management query can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

query-name: Specify the name of the query being deleted.

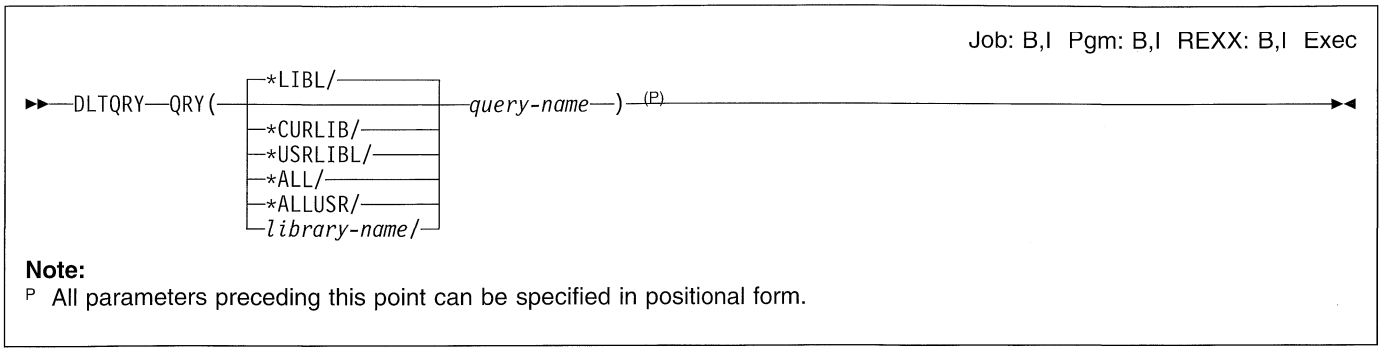
generic-query-name:* Specify the generic name of the query being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, all queries with names that begin with the generic name, and for which the user has authority, are deleted from the specified library or library list. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete query name.

Example

```
DLTQMQRY  QMQRY(RPTLIB/QUERY1)
```

This command deletes query management query QUERY1 from library RPTLIB.

DLTQRY (Delete Query) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Query (DLTQRY) command deletes an existing query definition from a library.

Required Parameters

QRY

Specifies the qualified name of the query definition being deleted.

The name of the query definition can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

query-name: Specify the name of the query definition being deleted.

Examples

Example 1: Deleting a Query Definition

```
DLTQRY  QRY(LIB1/MYDEF)
```

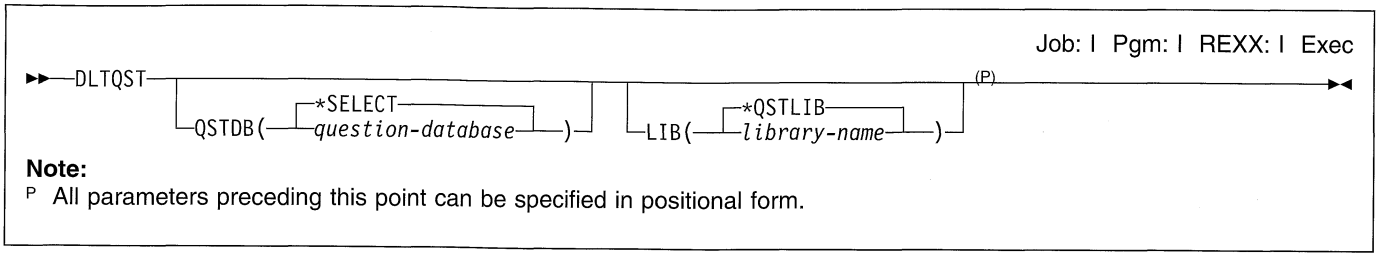
This command deletes query definition MYDEF from library LIB1.

Example 2: Deleting All Query Definitions

```
DLTQRY  QRY(*ALL/MYDEF)
```

This command deletes all the query definitions named MYDEF on the system.

DLTQST (Delete Question) Command



Purpose

The Delete Question (DLTQST) command allows the user to search a local Question-and-Answer (Q & A) database to find one or more questions to delete.

More information on deleting questions is in the *Q & A Database Coordinator's Guide*.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. A user must have authority to the command and be a Q & A coordinator for any Q & A database referred to by the command.
3. The user cannot delete candidate or conversational questions with this command.
4. This command is interactive only.

Optional Parameters

QSTDB

Specifies the Q & A database from which to delete questions.

***SELECT:** The user is asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database: Specify the name of the Q & A database from which to delete questions.

LIB

Specifies the name of the library that contains the Q & A database.

***QSTLIB:** The library containing the specified Q & A database is searched. If *SELECT is specified on the QSTDB parameter, any Q & A database in any library for which the user is authorized can be selected.

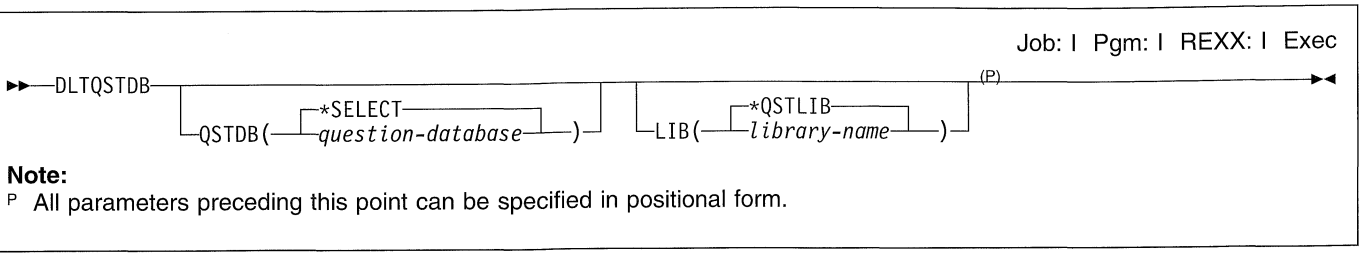
library-name: Specify the name of the library to be searched. If *SELECT is specified on the QSTDB parameter, any database in the library for which the user is authorized can be selected.

Example

```
DLTQST
```

This command shows the Specify Search Variables display.

DLTQSTDB (Delete Question-and-Answer Database) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Question-and-Answer Database (DLTQSTDB) command allows an authorized user to delete a Question-and-Answer (Q & A) database from the system. More information is in the *Q & A Database Coordinator's Guide*.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. A user must have authority to the command and be a Q & A coordinator for any Q & A database referred to by the command.
3. This command is interactive only.

Optional Parameters

QSTDB
Specifies the Q & A database being deleted from the system.

***SELECT:** The user is asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database: Specify the name of the Q & A database being deleted from the system.

LIB
Specifies the name of the library that contains the Q & A database being deleted.

***QSTLIB:** The library containing the specified Q & A database is searched. If *SELECT is specified on the QSTDB parameter, any Q & A database in any library for which the user is authorized can be selected.

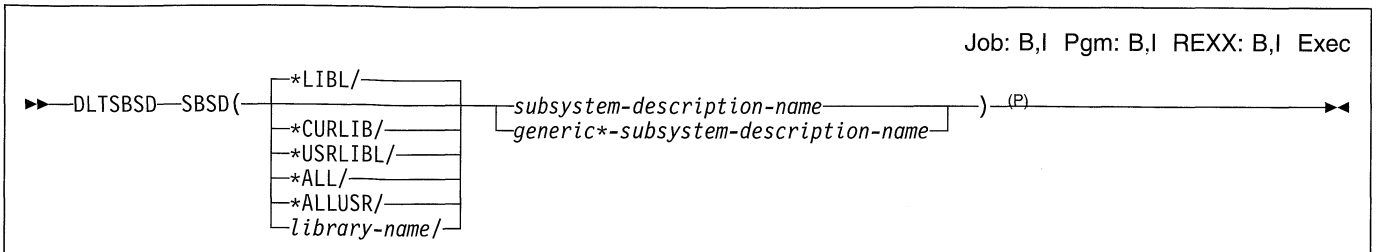
library-name: Specify the name of the library to be searched. If *SELECT is specified on the QSTDB parameter, any database in the library for which the user is authorized can be selected.

Example

```
DLTQSTDB
```

This command shows the Confirm Delete Q & A Database display. If more than one Q and A database is available for selection, the Select Q and A Database display is shown first.

DLTSBSD (Delete Subsystem Description) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Subsystem Description (DLTSBSD) command deletes the specified subsystem descriptions (including any work entries or routing entries that were added to them) from the system. Job queues assigned to this subsystem by the Add Job Queue Entry (ADDJOBQE) command are not deleted. The associated subsystem must be inactive before it can be deleted.

Restrictions:

1. This command cannot be run if an active subsystem is associated with this subsystem description.
2. The user must have object operational and object existence authority for the subsystem description being deleted, and read authority for the libraries.

Required Parameters

SBSD

Specifies the name of the subsystem descriptions being deleted. A specific subsystem description or a generic subsystem description can be specified. Either type can be qualified by a library name. Depending on the library qualifier that is specified or assumed, the following libraries are searched for the subsystem descriptions being deleted:

| The name of the subsystem description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

subsystem-description-name: Specify the name of the subsystem description being deleted.

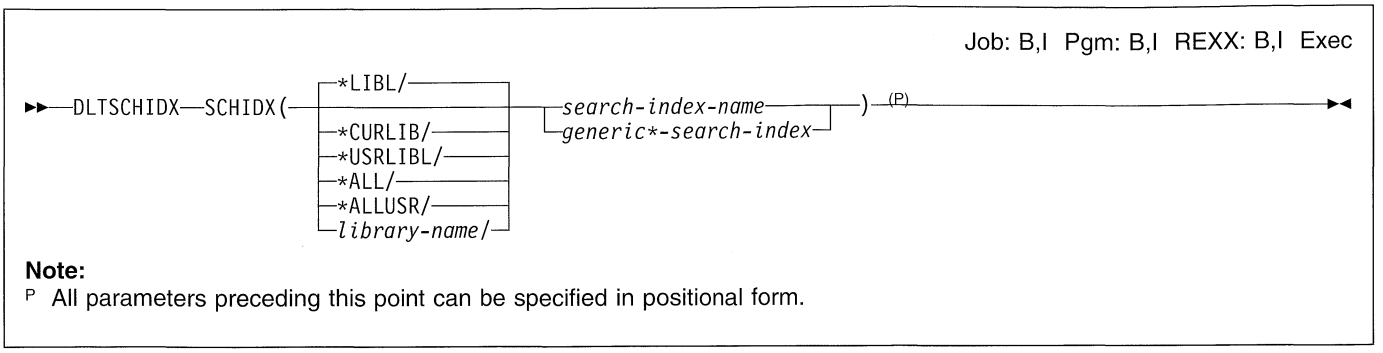
generic-subsystem-description-name:* Specify the generic name of the subsystem description being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTSBSD  SBSD(LIB1/BAKER)
```

This command deletes the inactive subsystem description called BAKER from library LIB1.

DLTSCHIDX (Delete Search Index) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Search Index command removes the specified search index from the system.

Restriction: The user of this command must have *OBJEXIST (object existence) authority for the search index.

Required Parameters

SCHIDX

Specifies the qualified name of the search index being deleted. A particular search index or a generic search index can be specified, and either type can be optionally qualified by a library name.

The libraries that are searched for the indexes being deleted depend on the library qualifier that is specified or assumed.

The name of the search index can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

search-index-name: Specify the name of the search index being deleted.

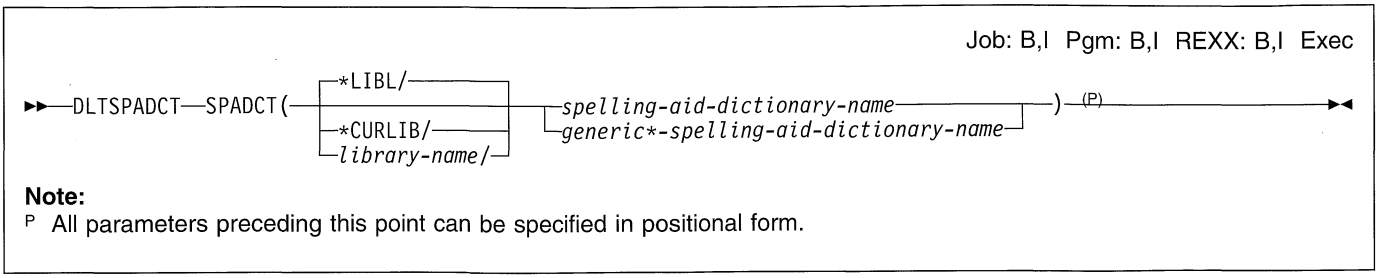
generic-search-index:* Specifies the generic name of the search index being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTSCHIDX   SCHIDX(*CURLIB/AR*)
```

This command deletes all search indexes named with the AR prefix (for example, ARINV, ARREC, ARPERS) from the library listed as the current library.

DLTSPADCT (Delete Spelling Aid Dictionary) Command



Purpose

The Delete Spelling Aid Dictionary (DLTSPADCT) command deletes the specified spelling aid dictionary from the system.

More information on spelling aid dictionaries is in the *Using OfficeVision/400* Word Processing*.

Restriction: The user must have object existence authority or *ALL authority for the spelling aid dictionary being deleted.

Required Parameters

SPADCT

Specifies the qualified names of the spelling aid dictionaries being deleted. A specific spelling aid dictionary or generic spelling aid dictionary can be specified; either type can be qualified by a library name.

The name of the spelling aid dictionary can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

spelling-aid-dictionary-name: Specify the name of the spelling aid dictionary being deleted.

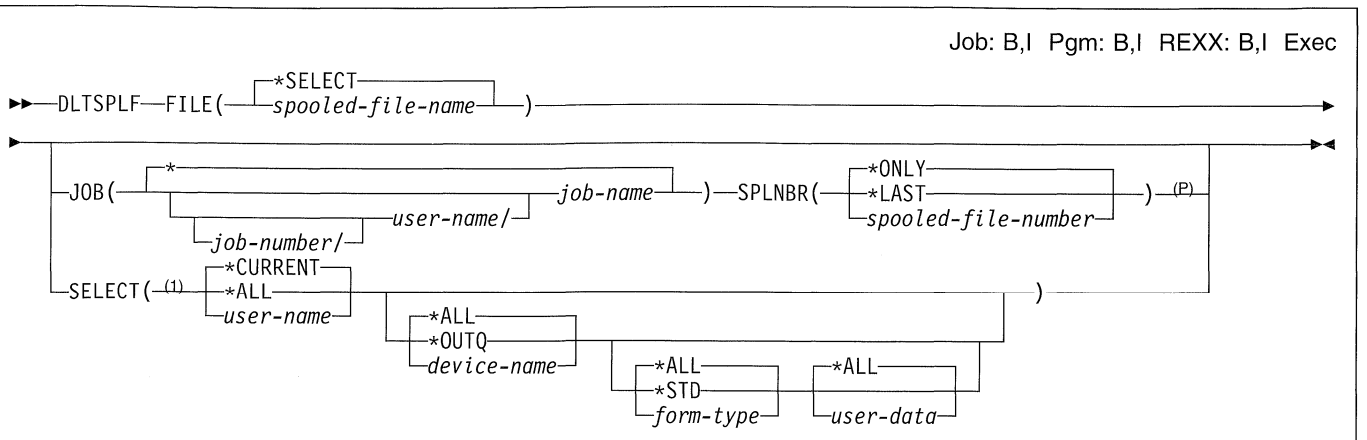
generic-spelling-aid-dictionary-name:* Specify the generic name of the spelling aid dictionary being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTSPADCT  SPADCT(MYLIB/LANGUAGE)
```

This command deletes the spelling aid dictionary named LANGUAGE in library MYLIB.

DLTSPLF (Delete Spooled File) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 *SELECT only allowed FILE(*SELECT)

Purpose

The Delete Spooled File (DLTSPLF) command is used to delete the specified spooled file from the output queue. If the spooled file is currently being produced on a device, it is immediately stopped and deleted. Output that has not been written is lost.

For information on deleting multiple spooled files of a job, see the "Additional Considerations" section of the Work with Job (WRKJOB) command description. For information on deleting all spooled files of a job, see the ENDJOB (End Job) command description.

Required Parameters

FILE

Specifies the name of the spooled file being deleted from the output queue.

***SELECT:** All spooled files that meet the selection requirements specified in the SELECT keyword are deleted from the system. This value is mutually exclusive with the JOB and SPLNBR parameters. Specifying *SELECT causes the JOB and SPLNBR keywords to be ignored.

spooled-file-name: Specify the name of the spooled file being deleted from the system.

Optional Parameters

JOB

Specifies the name of the job that produced (or is producing) the spooled file being deleted from the output queue.

A job identifier is the special value * or a qualified name with up to three elements. For example:

```
*
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** The job that issued this command is the job that produced the file being deleted.

job-name: Specify the name of the job that produced the file being deleted. If no job qualifier is given, all of the jobs currently in the system are searched for the simple job name.

user-name: Specify the name of the user of the job that produced the file being deleted.

job-number: Specify the number of the job that produced the file being deleted.

SPLNBR

Specifies the number of the job's spooled file being deleted from the output queue. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ONLY:** One spooled file from the job has the specified file name. The number of the spooled file is not necessary. If *ONLY is specified and more than one spooled file has the specified file name, a message is sent.

***LAST:** The spooled file with the highest number and the specified file name is used.

spooled-file-number: Specify the number of the spooled file with the specified file name that is being deleted.

SELECT

Specifies which group of files is selected for deletion. Files can be selected based on user device, form type,

and user data. Only files that meet each of the requirements are selected for deletion.

Element 1: User Names

***CURRENT:** Only files created by the user running this command are selected for deletion.

***ALL:** Files created by all users are selected for deletion.

user-name: Only files created by the specified user name are selected for deletion.

Element 2: Device Names

***ALL:** Files queued for any device are selected for deletion.

***OUTQ:** All files that are not queued for a device are selected for deletion. These files are on the output queue that are associated with printers.

device-name: Only files queued for the specified device name are selected for deletion.

Element 3: Form Types

***ALL:** Files for all form types are selected for deletion.

***STD:** Only files that specify the standard form type are selected.

form-type: Only files with the specified form type are selected for deletion.

Element 4: User Data Tags

***ALL:** Files with any user data tag specified are selected for deletion.

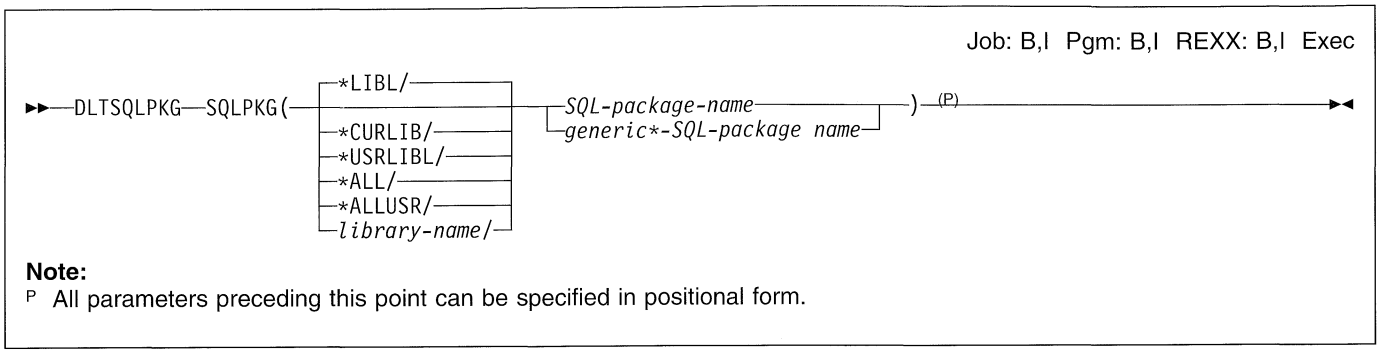
user-data: Only files with the specified user data tag are selected for deletion.

Example

```
DLTSPLF  FILE(WEEKLY)  JOB(000146/SMITH/PAYROLL5)
```

This command deletes the spooled file named WEEKLY (of job PAYROLL5, for user SMITH, job number 000146) from the output queue. Spooled files with different names produced by the job named PAYROLL5 are not affected by this command. If the job produced more than one file named WEEKLY, no file is deleted because SPLNBR(*ONLY) is assumed.

DLTSQLPKG (Delete Structured Query Language Package) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Structured Query Language Package (DLTSQLPKG) command is used to delete one or more SQL packages.

DLTSQLPKG is a local command and must be used on the AS/400 system where the SQL package being deleted is located.

To delete an SQL package on a remote system that is also an AS/400 system, use the Submit Remote Command (SBMRMTCMD) command to run the DLTSQLPKG command on the remote system.

The user can do the following to delete an SQL package from a remote system that is not an AS/400 system:

- Use interactive SQL to run the CONNECT and DROP PACKAGE operations.
- Sign on the remote system and use a command local to that system.
- Create and run an SQL program that contains a DROP PACKAGE SQL statement.

Restriction: The user must have *OBJEXIST and object management authorities for the SQL package and *READ authority for the library where it is located.

Required Parameters

SQLPKG

Specifies the qualified name of the SQL package being deleted. A specific or generic SQL package name can be specified.

The name of the SQL package can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

SQL-package-name: Specify the name of the SQL package being deleted.

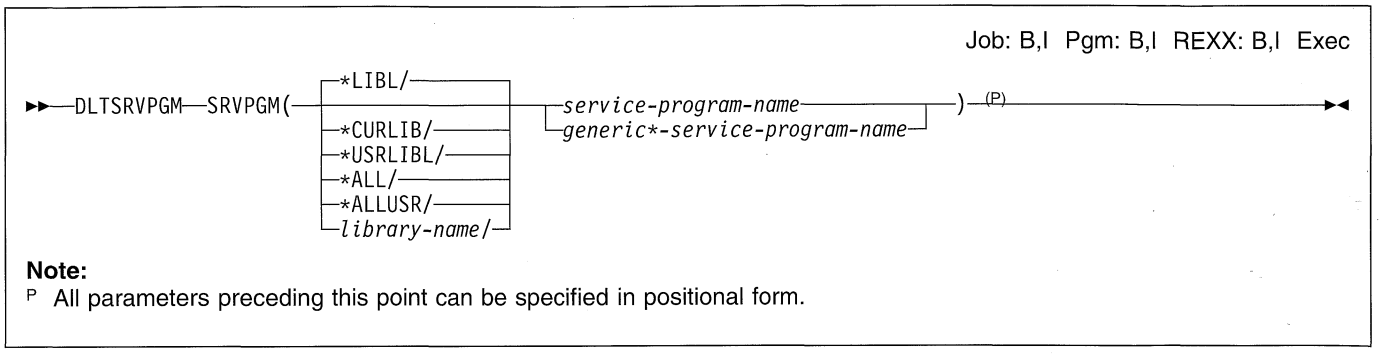
generic-SQL-package-name:* Specify the generic name of the SQL package to be deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, all SQL packages with names that begin with the generic name, and for which the user has authority, are deleted. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete SQL package name.

Example

```
DLTSQLPKG  SQLPKG(JONES)
```

This command deletes the SQL package JONES.

DLTSRVPGM (Delete Service Program) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Service Program (DLTSRVPGM) command deletes a bound service program or group of service programs from a set of modules and binding directories.

Restrictions:

1. You must have *ALL authority to the service program.
2. You must have *READ authority to the library from which the service program is being deleted.

Required Parameters

SRVPGM

Specifies the qualified name of the service program being deleted. A specific service program or a generic service program can be specified.

The name of the service program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

service-program-name: Specify the name of the service program being deleted.

generic-service-program-name:* Specify the generic name of the service program being deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, then all service programs with names that begin with the generic name, and for which the user has authority, are deleted. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete service program name.

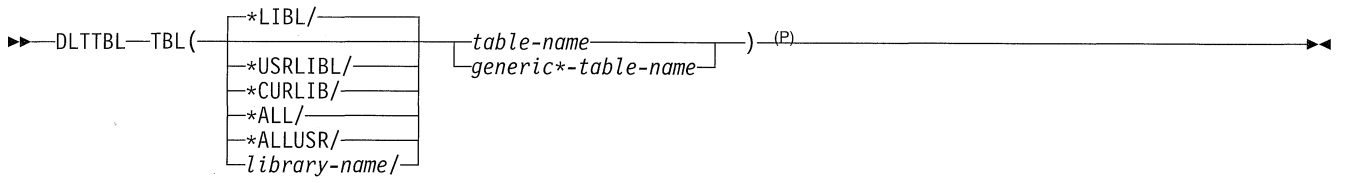
Example

```
DLTSRVPGM   SRVPGM(XYZ/S*)
```

This command deletes all service programs in library XYZ that begin with the letter S.

DLTTBL (Delete Table) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Table (DLTTBL) command deletes specified tables.

Required Parameters

TBL

Specifies the qualified name of the tables being deleted. A specific table or a generic table can be specified; either type can be optionally qualified by a library name. If no library qualifier is given, *LIBL is used to find the table.

The name of the table can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

table-name: Specify the name of the table being deleted.

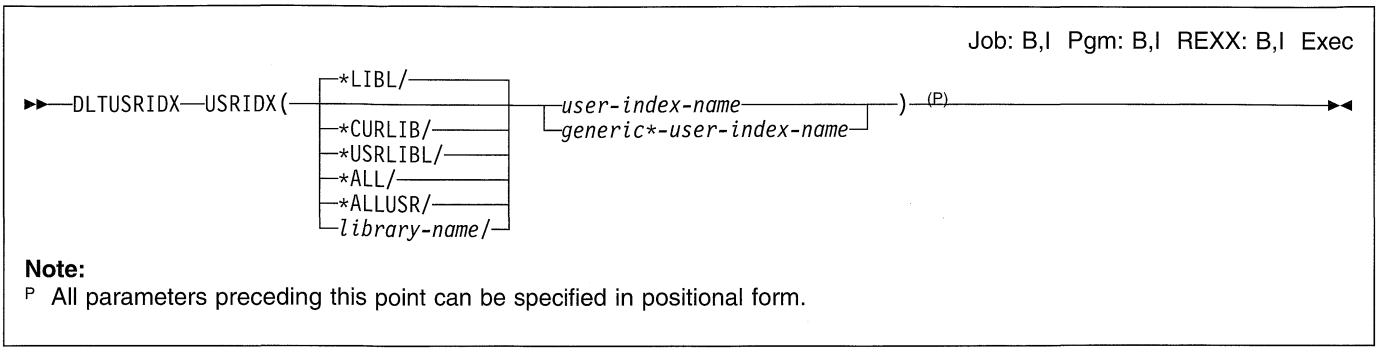
generic-table-name:* Specify the generic name of the table. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTTBL    TBL(SCRAMTBL)
```

This command deletes the table named SCRAMTBL from the system.

DLTUSRIDX (Delete User Index) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete User Index (DLTUSRIDX) command deletes a user index from the system. The user who enters this command must have *OBJEXIST authority and *USE authority for the user index being deleted.

Required Parameters

USRIDX

Specifies the name of the user index that is to be deleted.

The name of the user index can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

user-index-name: Specify the name of the user index that is to be deleted.

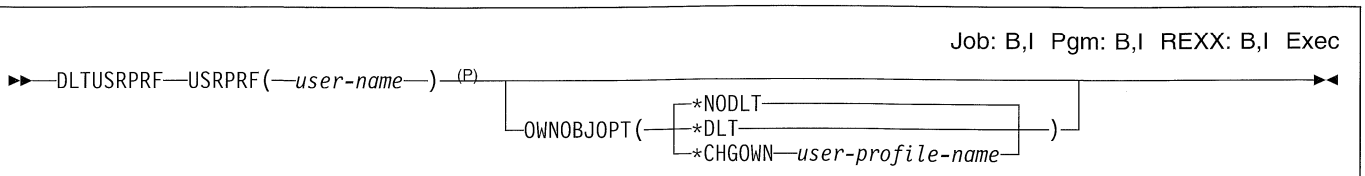
generic-user-index-name:* Specify the generic name of the user index. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTUSRIDX  USRIDX(AS400_BEST/USRIDXTEST)
```

This command deletes the user index named USRIDXTEST in the library named AS400_BEST from the system.

DLTUSRPRF (Delete User Profile) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Delete User Profile (DLTUSRPRF) command deletes a user profile from the system. The user of this command must have *SECADM special authority, which can be obtained through the program adopt operation, and *OBJEXIST authority and *USE authority for the user profile being deleted.

The message queue associated with this user profile is automatically deleted if it is that of the owner of the message queue.

If a user profile is damaged by system failure, it can be deleted by using the Delete User Profile (DLTUSRPRF) command and re-created by using the Create User Profile (CRTUSRPRF) command. After a user profile is re-created, the owned objects can be transferred back to it. Also, authorities that were granted to the damaged profile must be granted again to the new user profile by using the Grant Object Authority (GRTOBJAUT) command.

If several user profiles have been damaged, the saved user profiles can be restored to the system by using the Restore User Profile (RSTUSRPRF) and Restore Authority (RSTAUT) commands.

Restrictions:

1. The user must have use and object existence authority to the user profile.
2. The user must have existence, use, and delete authorities to delete a message queue associated with and owned by the user profile.
3. The user profile cannot be deleted if a user is currently running under the profile, or if it owns any objects and OWNOBJOPT(*NODLT) is specified. All objects in the user profile must first either be transferred to new owners by using the Change Object Owner (CHGOBJOWN) command or be deleted from the system. This can also be accomplished by specifying OWNOBJOPT(*DLT) to delete the objects or OWNOBJOPT(*CHGOWN user-profile-name) to change the ownership. Authority granted to the user does not have to be specifically revoked by the Revoke Object Authority (RVKOBJAUT) command; it is automatically revoked when the user profile is deleted.
4. To delete any object, the user must have object existence authority for the object.
5. This command calls the system distribution directory support to delete the user from the directory and to delete the user from distribution lists, if necessary. The system distribution directory support uses journaling and commitment control for the system distribution directory files (QUSURSYS/QAOS*). Commitment control must be inactive when this function is requested. If commitment control is active when this function is requested, the journal must be QUSRSYS/QAOSDIAJRN.

Required Parameter

USRPRF

Specifies the name of the user profile to be deleted.

Note: The following IBM-supplied user profiles are not valid on this parameter and cannot be deleted:

QDBSHR	QRJE
QDFTOWN	QSNADS
QDOC	QSPL
QDSNX	QSPLJOB
QFNC	QSRV
QGATE	QSRVBAS
QLPAUTO	QSYS
QLPINSTALL	QTSTRQS

Optional Parameter

OWNOBJOPT

Specifies the type of operations being done on the owned objects of the user profile being deleted.

***NODLT:** The owned objects for the user profile do not change, and the user profile is not deleted if the user owns any objects other than the message queue associated with the user profile.

***DLT:** The objects owned by the user profile are deleted. If the deletion of the objects is successful, the user's enrollment information is removed from OfficeVision/400.

Element 1: Change Ownership of Objects

***CHGOWN:** The owned objects for the user profile have ownership transferred to the specified user profile. The user profile is deleted if the transfer of all owned objects is successful.

When *CHGOWN is specified, a user profile name must be specified for the new user profile. The new user profile owns all objects owned by the user profile specified by the USRPRF parameter.

Element 2: User Profile of New Owner

user-profile-name: Specify the name of the user profile.

Additional Considerations

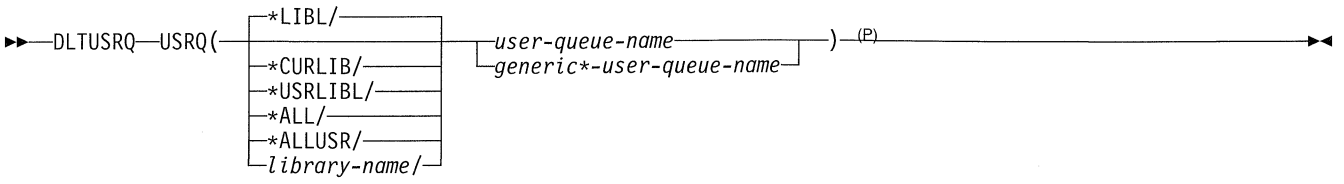
1. Owned object types *USRPRF and *RCT are not deleted from the system. Ownership of these object types is transferred to user profile QDFTOWN.
2. Owned objects of the type *PRDDFN are not deleted. Ownership is transferred to the user profile QSYS.
3. In addition to the above restrictions, all restrictions that apply to DLTLIB also apply to this parameter. For example, if an object is in use, it cannot be deleted. Or, if a physical file has an associated logical file owned by another user, the physical file cannot be deleted.
4. The user profile of a user enrolled in OfficeVision/400 cannot be deleted.
5. Owned objects of type *LIB are not deleted from the system if the library contains objects owned by another user profile. The ownership of the library is transferred to the system user profile, QDFTOWN.

Example

```
DLTUSRPRF  USRPRF(JJADAMS)
```

This command deletes the user profile named JJADAMS from the system if no objects are owned by the user profile and no user is currently running under it.

DLTUSRQ (Delete User Queue) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Delete User Queue (DLTUSRQ) command deletes a user queue from the system. The user who enters this command must have *OBJEXIST authority and *USE authority for the user queue being deleted.

Required Parameters

USRQ

Specifies the name of the user queue that is to be deleted.

The name of the user queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

user-queue-name: Specify the name of the user queue that is to be deleted.

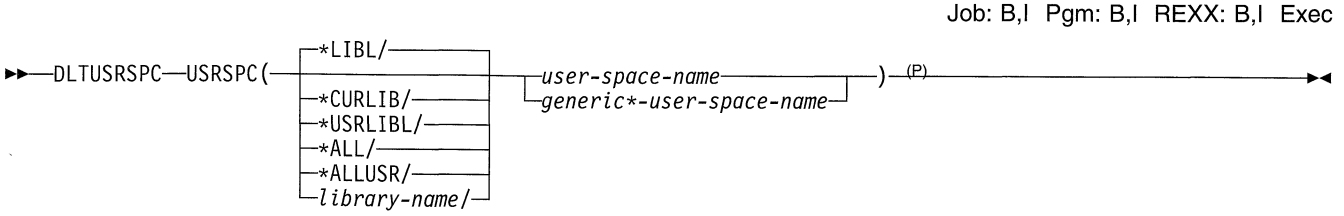
generic-user-queue-name:* Specify the generic name of the user queue. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTUSRQ  USRQ(AS400_BEST/USRQSAMPLE)
```

This command deletes the user queue named USRQSAMPLE in the library named AS400_BEST from the system.

DLTUSRSPC (Delete User Space) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Delete User Space (DLTUSRSPC) command deletes a user space from the system. The user who enters this command must have *OBJEXIST authority and *USE authority for the user space being deleted.

Required Parameters

USRSPC

Specifies the name of the user space that is to be deleted.

The name of the user space can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

user-space-name: Specify the name of the user space that is to be deleted.

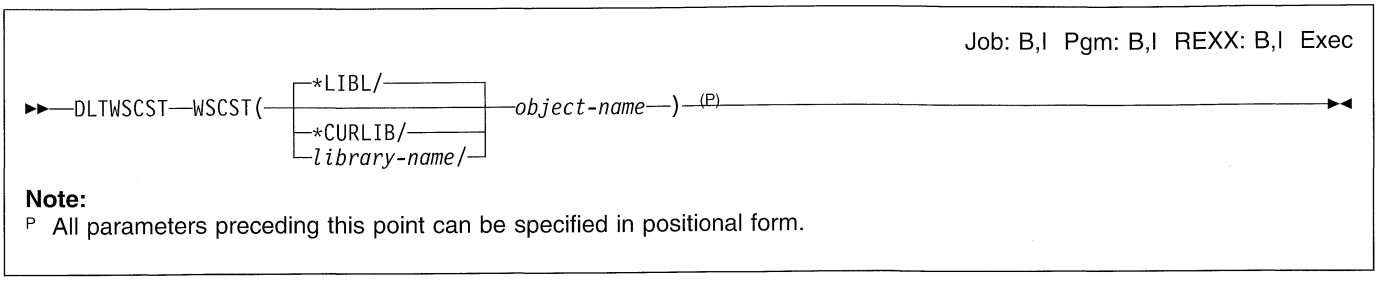
generic-user-space-name:* Specify the generic name of the user space. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be deleted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Example

```
DLTUSRSPC  USRSPC(AS400_BEST/USRSPCTEST)
```

This command deletes the user space named USRSPCTEST in the library named AS400_BEST from the system.

DLTWSCST (Delete Work Station Customizing Object) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Delete Work Station Customizing Object (DLTWSCST) command allows the user to delete a work station customizing object.

Required Parameters

WSCST

Specifies the qualified name of a work station customizing object to be deleted.

| The name of the work station customizing object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

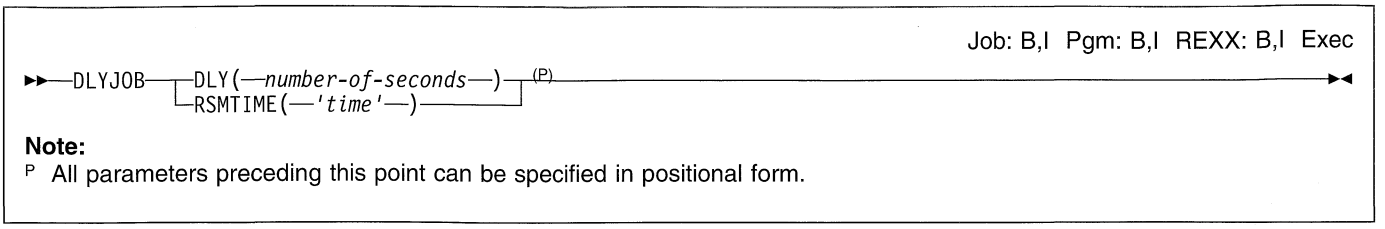
object-name: Specify the name of the object that is deleted.

Example

```
DLTWSCST  WSCST(MYLIB/PERSONNEL)
```

This command deletes the work station customizing object named PERSONNEL.

DLYJOB (Delay Job) Command



Purpose

The Delay Job (DLYJOB) command causes the current job to wait for a specified number of seconds, or until a specified time of day, before running resumes.

Required Parameters

DLY

Specifies the number of seconds to delay the job. The DLY keyword allows up to 999999 seconds (approximately 11 days, 14 hours). This parameter and the RSMTIME parameter are mutually exclusive.

RSMTIME

Specifies the time of day the job resumes running. When specifying a resume time, the date is implied by the time specified. If the resume time is later than the current time of day, the date is assumed to be the current date. If the resume time is earlier than the current time of day, tomorrow's date is assumed (the delay lasts until that time tomorrow). This parameter and the DLY parameter are mutually exclusive.

Examples

Example 1: Delaying a Job for 15 Seconds

```
DLYJOB  DLY(15)
```

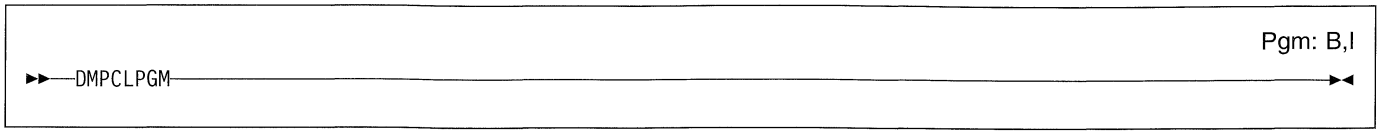
This command delays running of the job for 15 seconds.

Example 2: Delaying a Job Until Midnight

```
DLYJOB  RSMTIME('000000')
```

This command delays running of the job until midnight tonight.

DMPCLPGM (Dump CL Program) Command



Purpose

The Dump CL Program (DMPCLPGM) command dumps all variables (declared in the CL program in which the command is processed) and all messages on the program's message queue to a spooled printer file (QPPGMDMP). This command is valid only in a CL program; after the program is dumped, it continues processing.

Restriction: The user of this command must have read authority for the program.

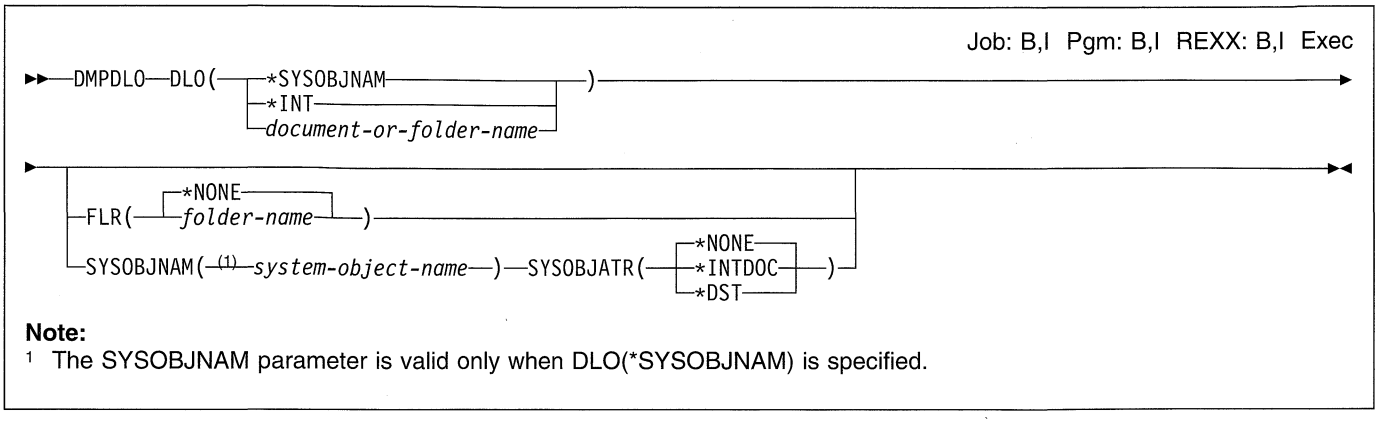
There are no parameters for this command.

Example

```
PGM
DCL . . .
DCL . . .
MONMSG  MSGID(CPF9999)  EXEC(GOTO DUMP)
  •
  •
  •
RETURN
DUMP:  DMPCLPGM
ENDPGM
```

This CL program monitors for the function check message CPF9999. If a function check occurs in the program, control is passed to the command at label DUMP. This causes a dump of the program's message queue and causes the program's variables to be printed. This dump can be used to determine the cause of the function check.

DMPDLO (Dump Document Library Object) Command



Note:
1 The SYSOBJNAM parameter is valid only when DLO(*SYSOBJNAM) is specified.

Purpose

The Dump Document Library Object (DMPDLO) command is used primarily for problem analysis. It copies the contents and/or attributes of folders, documents, or internal document library system objects to a spooled printer file named QPSRVDMP. If the printed output is not spooled, and the printer is not available, the printer file (QPSRVDMP) is overridden.

Restrictions:

1. This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.
2. The user must have read authority to a document or folder to dump it.
3. The user must have *ALLOBJ authority to dump internal system objects.

Required Parameters

DLO

Specifies the document library object being dumped.

***SYSOBJNAM:** A system object name is used to identify the document or folder being dumped. This value must be used to dump an internal or distribution document, or a document that is not in a folder.

***INT:** Internal document library system objects are dumped.

document-or-folder-name: Specify the name of the document or folder being dumped.

Optional Parameters

FLR

Specifies the name of the folder that contains the document.

Note: A folder name can be entered in this parameter only if a folder or document name is entered in the DLO parameter.

***NONE:** The object is not in a folder.

folder-name: Specify the name of the folder that contains the folder or document being dumped.

SYSOBJNAM

Specifies the system object name. This parameter is valid only when DLO(*SYSOBJNAM) or DOCL(*SYSOBJNAM) is specified. A full ten characters must be specified.

SYSOBJATR

Specifies attributes of the object being dumped. A value other than *NONE applies only to documents, and can be entered in this parameter only if *SYSOBJNAM is specified on the DLO parameter.

***NONE:** No attributes are specified for the object.

***INTDOC:** The object being dumped is an internal document.

***DST:** The object being dumped is a distribution document.

Examples

Example 1: Dumping a Document

```
DMPDLO  DLO(KAREN)  FLR(PEGGY)
```

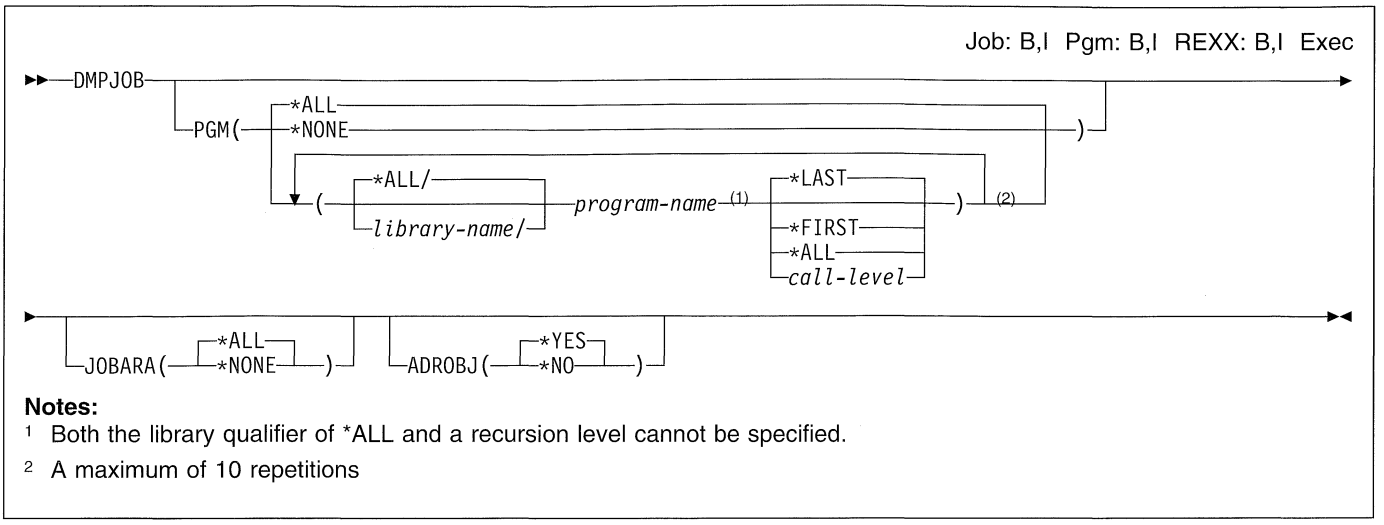
This command dumps a document or a folder named KAREN which is located in the folder named PEGGY.

Example 2: Specifying a System Object Name

```
DMPDLO  DLO(*SYSOBJNAM) SYSOBJNAM(BHZM052634)
```

This command dumps the document library object identified by the system object name BHZM052634.

DMPJOB (Dump Job) Command



Notes:
1 Both the library qualifier of *ALL and a recursion level cannot be specified.
2 A maximum of 10 repetitions

Purpose

The Dump Job (DMPJOB) command dumps the basic data structures, or specific calls of the current job or of the job being serviced as a result of the Start Service Job (STRSRVJOB) command. The information is dumped to a spooled printer file (QPSRVDMP) to be printed. If the user had specified SPOOL(*NO) on either the CHGPRTF command or the OVRPRTF command, then the output is not spooled but printed directly; and, if the printer is not available, then this command overrides the print job and spools the output. When the user specifies SPOOL(*NO) on one of the two commands above, the user must specify QPSRVDMP as the printer file. The dump includes formatted information about the specified programs, and dumps of specified OS/400 system objects and system objects associated with the job.

Restriction: This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.

Optional Parameters

PGM

Specifies which program to dump. Either a single value or a list of values (10 maximum) can be specified.

***ALL:** All programs on the call stack are dumped.

***NONE:** No programs are dumped. Only the lists of called and activated programs are dumped.

Element 1: Program Name

The name of the dumped program can be qualified by one of the following library values:

***ALL:** All libraries on the system are searched. If *ALL is specified, a call level cannot be specified.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the called program to dump. Up to 10 characters for the program name can be specified.

Element 2: Program Call Levels

***LAST:** The last call with the specified name is dumped.

***FIRST:** The first call with the specified name is dumped.

***ALL:** All calls with the specified name are dumped.

call-level: Specify the level of call for a program with multiple calls in the stack. If *ALL is specified for the library name, the call level cannot be specified.

JOBARA

Specifies whether the job structure areas of the process are dumped. Job structure areas consist of the following:

- Work Control Block
- Library Search List
- Job Temporary Library
- Job Local Data Area
- Spool Control Block
- Data Management Communications Queue
- Service Communication Object
- Process Definition Template
- Process Lock List
- Machine Interface (MI) Response Queue

***ALL:** The job structure areas are dumped.

***NONE:** The job structure areas are not dumped.

ADROBJ

Specifies whether objects addressed from the program storage of a program being dumped are also dumped. If

*NONE is specified for the PGM parameter, no addressed objects are dumped.

***YES:** The addressed objects are dumped.

***NO:** The addressed objects are not dumped.

Examples

Example 1: Dumping Programs

```
DMPJOB   PGM((QGPL/UPDATE *FIRST)  (PAYROLL/MASTER *ALL))
  JOBARA(*ALL)  ADROBJ(*NO)
```

This command dumps the first occurrence of QGPL/UPDATE in the call stack and all occurrences of PAYROLL/MASTER. The job structure areas are dumped.

Example 2: Dumping Entire Job Structure

```
DMPJOB
```

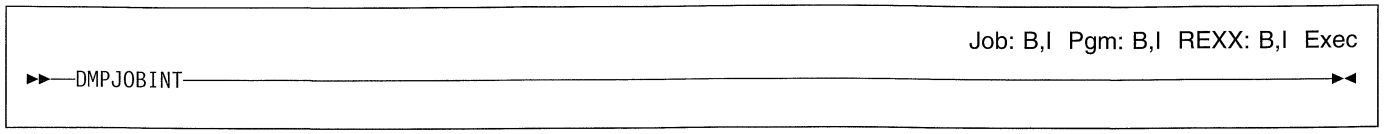
This command dumps the entire job structure.

Example 3: Dumping Lists of Called and Activated Programs

```
DMPJOB   PGM(*NONE)  JOBARA(*NONE)
```

This command dumps the lists of programs called and activated.

DMPJOBINT (Dump Job Internal) Command



Purpose

The Dump Job Internal (DMPJOBINT) command dumps the machine internal data related to the machine process of the current job or the job being serviced as a result of the Start Service Job (STRSRVJOB) command. This data is for use by IBM service representatives. When the internal data is dumped, a dump identifier is sent in a message to the user who sent the DMPJOBINT command. The Print Internal Data (PRTINTDTA) command can be used to print the dump output.

Restriction: This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.

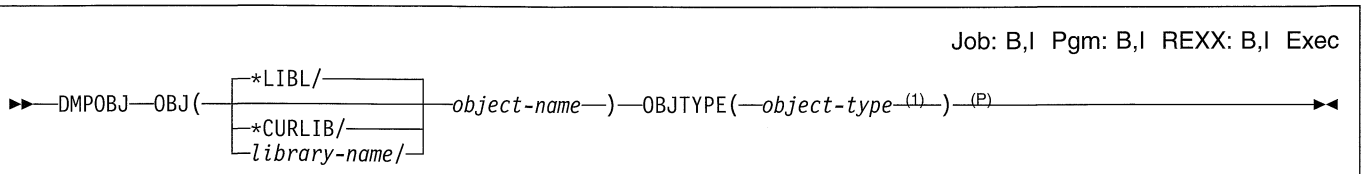
There are no parameters for this command.

Example

```
DMPJOBINT
```

This command dumps, for the job in which the command is entered, the machine internal data associated with the job. A message with the dump identifier is sent to the user entering the command.

DMPOBJ (Dump Object) Command



Notes:
1 A list of the valid OS/400 object types for this command is in Appendix A.
P All parameters preceding this point can be specified in positional form.

Purpose

The Dump Object (DMPOBJ) command dumps the contents and/or attributes of the specified OS/400 system object to a spooled printer file named QPSRVDMP. Whether the contents and/or attributes can be dumped depends upon the object type. If the user had specified SPOOL(*NO) on either the CHGPRTF command or the OVRPRTF command, then the output is not spooled but printed directly; and, if the printer is not available, then this command overrides the print job and spools the output. When the user specifies SPOOL(*NO) on one of the two commands above, the user must specify QPSRVDMP as the printer file. Any library or an OS/400 system object that is stored in a library can be dumped, but only one object can be specified at a time on this command.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. The following user profiles have private authorities to use the command:
   - QPGMR
   - QSYSOPR
   - QSRV
   - QSRVBAS

Required Parameters

OBJ

Specifies the qualified name of the OS/400 system object being dumped.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

object-name: Specify the name of the object being dumped. Only the OS/400 system objects that are stored in libraries can be dumped. Refer to the OBJTYPE parameter for the valid types of objects.

OBJTYPE

Specifies the object type of the OS/400 system object being dumped. Any one of the OS/400 system object types can be specified. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Examples

Example 1: Dumping File Contents

```
DMPOBJ   OBJ(ORDENT/ORDERIN)  OBJTYPE(*FILE)
```

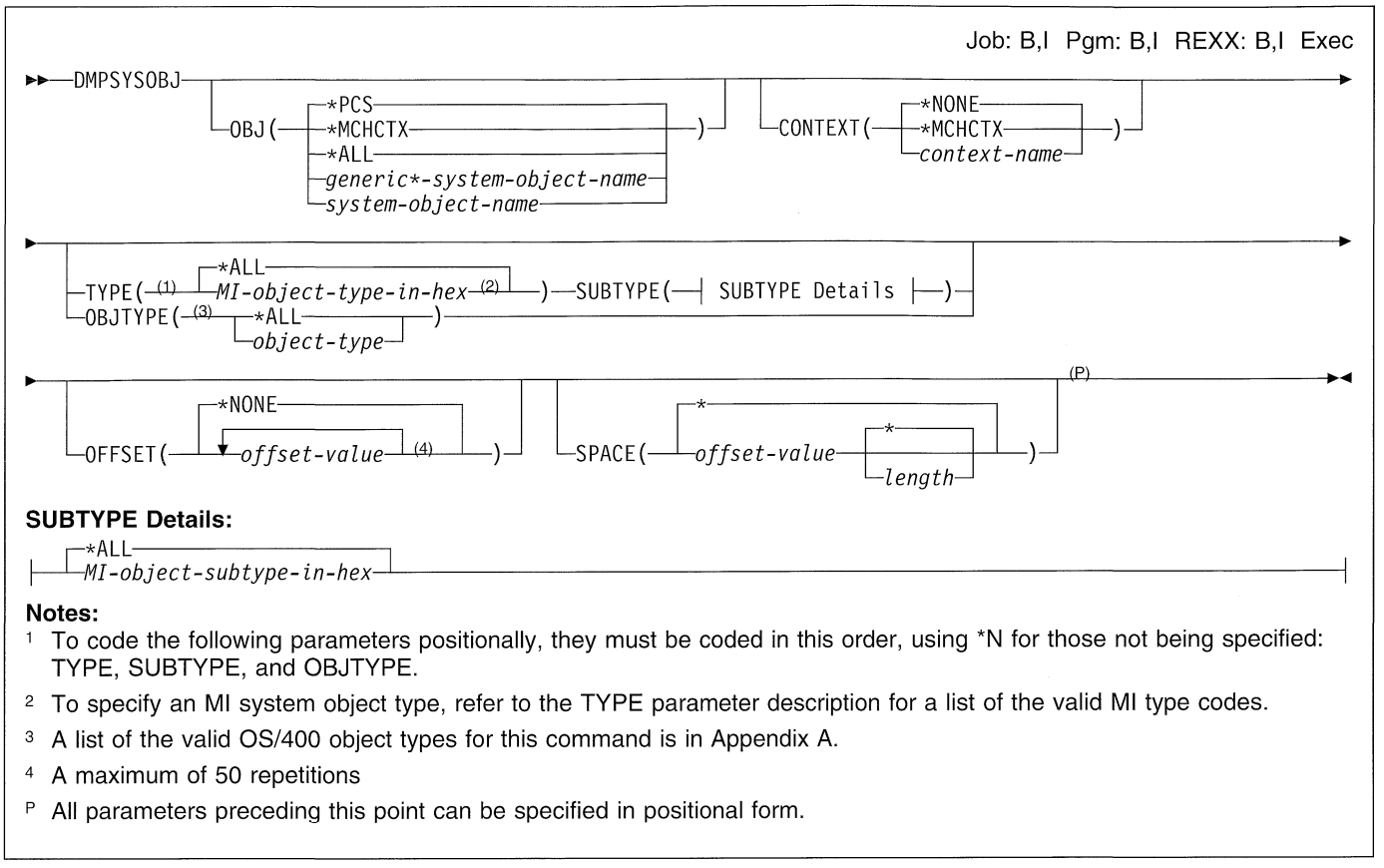
This command dumps the contents of the file named ORDERIN that is stored in the ORDENT library.

Example 2: Dumping a Program

```
DMPOBJ   OBJ(MYPROG)  OBJTYPE(*PGM)
```

This command dumps the first copy of the program MYPROG that is found in the library list. The dump is spooled to the printer output file QPSRVDMP.

DMPSYSOBJ (Dump System Object) Command



Notes:

1 To code the following parameters positionally, they must be coded in this order, using *N for those not being specified: TYPE, SUBTYPE, and OBJTYPE.

2 To specify an MI system object type, refer to the TYPE parameter description for a list of the valid MI type codes.

3 A list of the valid OS/400 object types for this command is in Appendix A.

4 A maximum of 50 repetitions

P All parameters preceding this point can be specified in positional form.

Purpose

The Dump System Object (DMPSYSOBJ) command is used primarily for problem analysis. It dumps the contents and/or attributes of machine interface (MI) system objects to a spooled printer file named QPSRVDMP. If the user had specified SPOOL(*NO) on either the CHGPRTF command or the OVRPRTF command, then the output is not spooled but printed directly; and, if the printer is not available, then this command overrides the print job and spools the output. When the user specifies SPOOL(*NO) on one of the two commands above, the user must specify QPSRVDMP as the printer file. Any MI object that is stored in any library (context) or that is addressable through an object can be dumped. A specific object, a generic group, or all of the MI objects in a context can be specified. The dump operation can also be limited to objects of a specified type and, optionally, of a specified subtype.

Restrictions:

1. This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.
2. The user must have use authority to the object and read authority to the program and the user profile.

Optional Parameters

OBJ

Specifies which of the MI system objects are being dumped. The name of a specific object, the generic name of a group of objects, the process control space of the job, the machine context, or all of the MI objects in a context can be specified. If a library name is specified, the library is dumped, but not the objects in it.

If OBJ(QTEMP) is specified along with either OBJTYPE(*LIB) or TYPE(04) SUBTYPE(01), the temporary job context associated with the job that this command is entered from, or the job being serviced as a result of the Start Service Job (STRSRVJOB) command, is dumped. In either case, the CONTEXT parameter is ignored.

***PCS:** The process control space of the current job or that of the job being serviced as a result of the Start Service Job (STRSRVJOB) command is dumped. OBJ(*PCS) can be used with the OFFSET and SPACE parameters to dump objects in the job structure. If OBJ(*PCS) is specified, the CONTEXT, TYPE, SUBTYPE, and OBJTYPE parameters are ignored.

***MCHCTX:** The machine context (which contains a list of the objects in the context) is dumped. If OBJ(*MCHCTX) is specified, all the other parameters in this command are ignored.

***ALL:** All the MI system objects in the specified context are dumped if they match the requirements specified in TYPE and SUBTYPE (for MI objects), or OBJTYPE (for the OS/400 system objects).

generic-system-object-name:* Specify the OS/400 system or MI generic object name that identifies the group of MI system objects to be dumped. An MI object name can have up to 30 characters in it. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be dumped only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

system-object-name: Specify the name of the OS/400 system or MI object that is to be dumped. Up to 30 characters can be entered. If more than one object has the same name, all objects having that name and matching the attributes specified by the CONTEXT parameter, and either the TYPE and SUBTYPE parameters or the OBJTYPE parameter are dumped. If a specific object is being dumped, the CONTEXT, TYPE, and SUBTYPE parameters or the CONTEXT and OBJTYPE parameters should be specified.

CONTEXT

Specifies which context or library the objects to be dumped are found in.

***NONE:** The object specified by the OBJ parameter is not in any context. *NONE is valid only if *PCS or *MCHCTX is specified or assumed for the OBJ parameter, or if OBJ(QTEMP) is specified along with either OBJTYPE(*LIB) or TYPE(04) SUBTYPE(01).

***MCHCTX:** The objects to be dumped are in the machine context. The following OS/400 system object types, whose MI system object names are given in parentheses, can reside *only* in the *machine* context: library (context), user profile, device (logical unit) description, network interface description, line (network) description, and control unit (controller) description. These types are included in the table given in the TYPE parameter description. *MCHCTX is valid only if one of these five object types is dumped.

context-name: Specify the name of the context containing the objects being dumped. The name of a library, such as QGPL or QTEMP, can be specified. If QTEMP is specified, the objects to be dumped are in the temporary job context associated with the job that this command is entered from or the job being serviced as a result of the Start Service Job (STRSRVJOB) command.

TYPE

Specifies the type of MI objects to be dumped.

***ALL:** All MI object types in the specified context that have the specified name (if used) are dumped.

MI-object-type-in-hex: Specify the hexadecimal value that specifies the type of MI system objects to be dumped. The following table shows the MI system objects and their hexadecimal type codes. The value must be specified with both characters, but it does not have to be enclosed in apostrophes.

MI Type MI System Object	Code
Access group	01
Program	02
Module	03
Context (library)[1]	04
Byte string space	06
Journal space	07
User profile[1]	08
Journal port	09
Queue	0A
Data space	0B
Data space index	0C
Cursor	0D
Index	0E
Commit block	0F
Logical unit (device) description[1]	10
Network (line) description[1]	11
Controller (control unit) description[1]	12
Dump space	13
Class of Service space	14
Mode Description space	15
Network interface description[1]	16
Connection list	17
Queue space	18
Space	19
Process control space	1A
Authorization list space	1B
Dictionary space	1C
Machine context	81

[1] If this object is specified for TYPE, then CONTEXT(*MCHCTX) must also be specified.

SUBTYPE

Specifies the subtype of the specified MI objects to be dumped, or specifies that all subtypes are being dumped.

***ALL:** All the subtypes of the specified MI objects are dumped.

MI-object-subtype-in-hex: Specify the specific subtype of the MI system objects to be dumped. Valid values range from 00 through FF. However, the subtype specified must be for an MI object actually in the specified context. If TYPE(*ALL) is specified, a specific subtype *cannot* be specified.

OBJTYPE

Specifies the object type of the OS/400 system objects to have their associated MI system objects dumped. If

this parameter is specified, the TYPE or SUBTYPE parameters cannot be specified. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ALL:** The specified MI objects of all OS/400 system object types are dumped.

object-type: Specify the specific OS/400 system object type whose associated MI system objects are to be dumped.

OFFSET

Specifies a list of values to use as offsets to indirectly address a single object that is being dumped. The values must be positive hexadecimal values or zeros that, when added to a pointer, result in valid addresses. If an offset of zero is added to a system pointer, the result is a space pointer to the start of the space associated with the object that is addressed by the system pointer. (In this discussion, the associated space of a space object is the space itself.)

Note: The OFFSET and SPACE parameters cannot be specified if *ALL or a generic object name is specified for the OBJ, TYPE, SUBTYPE, or OBJTYPE parameters.

***NONE:** No offset is specified. The object located through the context is dumped.

offset-value: Specify the list of offsets to pointers to use to address the object or space to be dumped. The values specified in this parameter are used as follows:

1. The first offset is added to a space pointer that points to the associated space of the object located through the context. The result is a space pointer that points to a location further into the space.
   a. If only one offset value is specified in this parameter, step 2 is not done and the dump operation, as indicated by the rest of the parameters in the command, is taken.
   b. If more than one offset is specified in this parameter, step 2 is repeated for each additional offset given.
2. Regarding the location pointed to by the space pointer produced in the previous step:
   a. If the location does not contain another pointer, the command is ended, an error message is sent to the user, and no dump operation is done.
   b. If the location contains a space pointer, the (next) offset is added to it. The result is another space pointer that points to either the same space or a different space or associated space.
   c. If the location contains a system pointer, the associated space pointer is set from the system pointer, and the (next) offset is added to the space pointer. The result is a space pointer that points to a location in the associated space of the object addressed by the system pointer.

The result of step 2b or 2c is a space pointer that is used to perform step 2 again if there is another offset. If the last offset has been used, the final result is a location contained in a space pointer that is used as follows:

- If the resulting location contains a system pointer and the SPACE parameter is not specified, the system object pointed to by the system pointer is dumped. If the SPACE parameter is specified, the SPACE specification determines the portion of the system object that is dumped.
- If the resulting location contains a space pointer and the SPACE parameter is not specified, the portion of the space that starts at the location pointed to by the space pointer is dumped. If the SPACE parameter is specified, the SPACE specification determines the portion of the space to be dumped.

The following chart shows the offsets into the process control space (PCS) at which there are pointers to the components of a job structure. If one of these offsets is specified, OBJ(*PCS) must be specified or assumed.

Object (Descriptive Name and Abbreviation)	Object Name	Offset
Data management communications queue (DMCQ)	QDMCQ	20
Job temporary context (QTEMP)	QTEMP	40
MI response queue (MIRQ)	QMIRQ	80
Process definition template (PDT)	PDT	60
Process access group (PAG)	PAG	60 100
Spooling control block (SCB)	QSPSCB	200
Work control block table (WCBT)	QWCBT	10

SPACE

Specifies the area of a space or associated space to be dumped. The space is pointed to by the final pointer determined by the OFFSET parameter. If the OFFSET parameter is not specified, the final pointer is a system pointer to the specified object in the context. See *Note* in the OFFSET parameter description.

Element 1: Offset Value

***:** If the final pointer is a system pointer, the object pointed to by that pointer is dumped. If the final pointer is a space pointer, the portion of the space that starts at the location pointed to by that pointer is dumped.

offset-value: Specify the value to add to the final pointer to point to the beginning of the area to dump. The value specified must be a positive hexadecimal value, ranging from 00000000 through 00FFFFFF, and, when added to the final pointer, results in a valid address.

Element 2: Length

***:** The rest of the space pointed to as a result of the offset value is being dumped.

length: Specify a positive hexadecimal value ranging from 00000001 through 00FFFFFF that specifies the length of the area to be dumped. If the length specified is greater than the actual length of the space, only the actual space available is dumped.

Examples

Example 1: Dumping Indexes

```
DMPSYSOBJ   CONTEXT(QTEMP)  TYPE(0E)
```

This command dumps the contents and attributes of all the indexes in the temporary job context to a spooled file for printing. MI indexes are identified by the type code 0E.

Example 2: Dumping a Device Description

```
DMPSYSOBJ   OBJ(WS1)  CONTEXT(*MCHCTX)  OBJTYPE(*DEVD)
```

This command dumps the device description for work station WS1, which is stored in the machine context.

Example 3: Dumping Process Control Space

```
DMPSYSOBJ   OBJ(*PCS)  SPACE(0  2A0)
```

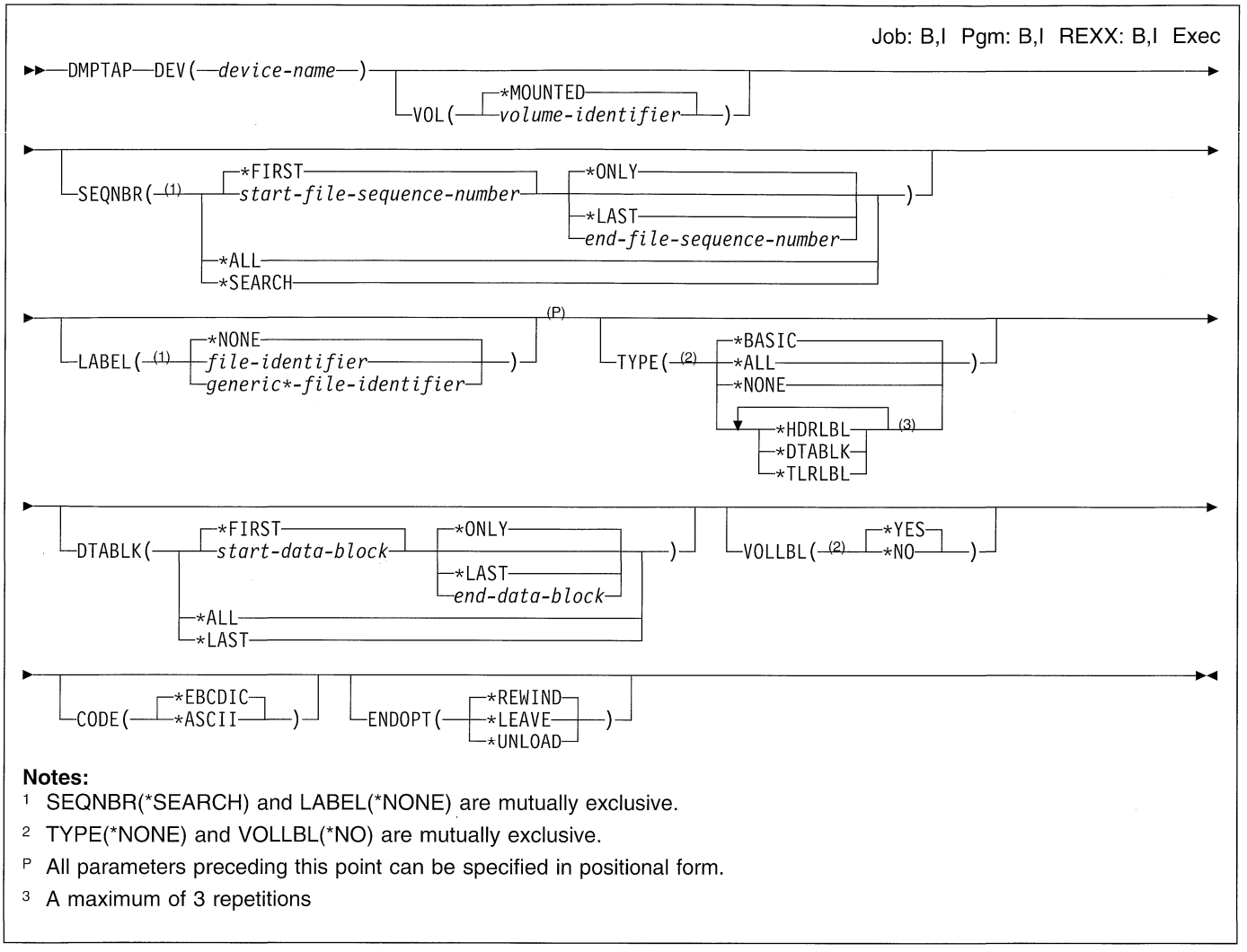
This command dumps the work control block from the space associated with the process control space for the job.

Example 4: Specifying Offset Values

```
DMPSYSOBJ   OBJ(*PCS)  OFFSET(60 E0 10 10)  SPACE(0 20)
```

This command dumps the second call entry of the process automatic storage area (offset 60 E0) for a length of 32 bytes (SPACE(0 20)). If the third call level is dumped, OFFSET(60 E0 10 10 10) is specified.

DMPTAP (Dump Tape) Command



Notes:

1 SEQNBR(*SEARCH) and LABEL(*NONE) are mutually exclusive.

2 TYPE(*NONE) and VOLLBL(*NO) are mutually exclusive.

| P All parameters preceding this point can be specified in positional form.

3 A maximum of 3 repetitions

Purpose

The DMPTAP (Dump Tape) command dumps label information, data blocks, or both from standard labeled tapes or tapes with no labels to a spooled printer file named QPTAPDMP. This command allows the user to dump one or more data files from the tape volume, writing the information to a printer file.

The tape volume being dumped must be on the specified device. After the DMPTAP command is entered, as much of the tape as necessary is read before the requested information is printed.

Data files on secured tapes can be dumped by the security officer only; any user can dump label information on secured tapes.

When the default values for the parameters of the DMPTAP command are used, the tape label areas and a minimal amount of data from the first file are printed. This command can help determine the record format of a data file on a tape with no label, or it can determine the exact contents of all label information for a labeled data file.

Required Parameters

DEV

Specifies the name of the tape device in which the volume being dumped is placed. The volume may or may not be labeled. Specify the name of the tape device.

Optional Parameters

VOL

Specifies one or more volume identifiers used by the file.
| More information on this parameter is in Appendix A,
| "Expanded Parameter Descriptions."

***MOUNTED:** The volume on the specified device is dumped. The volume may or may not be labeled. Note

that VOL(*MOUNTED) and LABEL(*NONE) must be specified to dump a volume that is not labeled.

volume-identifier: Specify the identifier of the labeled volume being dumped. This value can be specified only for dumping a labeled volume. If the tape on the specified device has a different volume identifier than the one specified in this parameter, or if it is not labeled, an error message is sent to the user of the DMPTAP command and the tape is not dumped.

SEQNBR

Specifies the range of sequence numbers for the data files that are dumped. If SEQNBR(*ALL) is specified, then all data files on the tape are dumped. If a range of sequence numbers is specified, then only data files in that range of data file sequence numbers are dumped. The data files dumped may be further restricted by using the LABEL parameter.

The sequence number for a labeled tape data file is stored on labels ahead of and following the data in the file. For a volume that is not labeled, the data file sequence number is determined by the number of tape markers from the beginning of the tape. This parameter can be specified as a list of two values (Elements 1 and 2) or as a single value (*ALL or *SEARCH).

Element 1: First File to Dump

***FIRST:** The range of data files being dumped begins with the first file on the tape volume, regardless of its sequence number.

start-file-sequence-number: The range of data files being dumped begins with the data file with the specified sequence number. Specify a number that is less than or equal to the end-file-sequence-number value.

Element 2: Last File to Dump

***ONLY:** Only a single data file (specified by the start-file-sequence-number) is dumped.

***LAST:** The range of data files being dumped begins with the start-file-sequence-number data file and ends with the last data file on the end of the reel.

end-file-sequence-number: The range of data files being dumped ends with the specified sequence number data file. Specify a number that is greater than or equal to the start-file-sequence number.

Other Single Values

***ALL:** All data files on the volume on the specified device are dumped.

***SEARCH:** The volume that is on the device is searched for a data file with an identifier that matches the LABEL parameter value; when a match is found, the data file is dumped. If VOLLBL(*NO) is specified and the last operation on the device specified ENDOPT(*LEAVE) (the tape is positioned at the location where the last operation ended), the file search begins with the first data file beyond the current tape position. If VOLLBL(*YES) is specified or ENDOPT(*LEAVE) was not used for the last operation (or if the tape was manually rewound since an ENDOPT(*LEAVE) operation), the search begins with the first data file on the volume. SEQNBR(*SEARCH) is not valid when LABEL(*NONE) is specified, and cannot be used to dump a tape volume that is not labeled.

LABEL

Specifies the identifier of the specific data files that are dumped. The file identifier for a tape data file is stored on labels ahead of and following the data in the file.

***NONE:** All data files on the volume in the specified SEQNBR range are dumped. Note that VOL(*MOUNTED) and LABEL(*NONE) *must* be used to dump a tape volume that is not labeled.

file-identifier: Specify the data file identifier (17 alphanumeric characters maximum) of the data files being dumped. The system compares the LABEL identifier with the data file identifier on the labels of each file in the range specified by the SEQNBR parameter. All data files with an identifier that matches the LABEL identifier are dumped; any data file with an identifier that does not match the LABEL identifier is not dumped.

generic-file-identifier:* Specify a character string for a generic label identifier (17 alphanumeric characters maximum), which contains at least one character followed by an asterisk (*). Any tape file that has a file identifier with the same prefix as the generic data-file-identifier is dumped. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be dumped only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

TYPE

Specifies the type of information that is being dumped. The dump may consist of the data file header labels or trailer labels (for a labeled tape volume only), data blocks from the data portion of the file, or all three types of information. If a tape volume is on the device that is not labeled, only *BASIC, *ALL, or *DTABLK can be specified or an error message is sent to the user of the DMPTAP command and the volume is not dumped.

***BASIC:** For a standard-labeled volume, the dump includes header labels and the data blocks specified by the DTABLK parameter. For a volume that is not labeled, only the data blocks (DTABLK parameter) are dumped.

***ALL:** For a standard-labeled volume, the dump includes header labels, trailer labels, and data blocks.

For a volume that is not labeled, TYPE(*ALL) dumps only data blocks (since there are no labels).

***NONE:** No data file is dumped. If TYPE(*NONE) is specified, the tape volume being dumped must be labeled, and VOLLBL(*NO) cannot be specified, or an error message is sent to the user of the DMPTAP command.

***HDRLBL:** The data file header labels are dumped. Header labels immediately ahead of the data in the file that they apply to. All header labels for the specified data files are dumped, including user-specified header labels (if any exist). TYPE(*HDRLBL) is not valid for volumes that are not labeled.

***DTABLK:** One or more data blocks from the file data are dumped. The blocks within the data file that are dumped are specified by the DTABLK parameter.

***TLRLBL:** All data file trailer labels are dumped. Trailer labels immediately follow the data in the file to which they apply. All the trailer labels for the specified data files are dumped, including user-specified trailer labels (if any exist). TYPE(*TLRLBL) is not valid for volumes that are not labeled.

DTABLK

Specifies which data blocks are dumped. This parameter is used to limit the amount of tape file data dumped to the printer. If neither TYPE(*BASIC) nor TYPE(*ALL) is specified and the TYPE parameter value does not include *DTABLK, this parameter is ignored.

This parameter can be specified as a list of two values (Elements 1 and 2) or as a single value (*ALL or *LAST).

Element 1: First Block to Dump

***FIRST:** The data blocks being dumped begins with the first block in the data file.

start-data-block: Specify the number of the first data block within each file that is dumped. If this number is greater than the number specified for the end-data-block portion of the DTABLK parameter, an error message is sent to the user who requested the dump, and the tape is not dumped. If the start-data-block value is larger than the actual number of data blocks in the data file, then the last data block in the file is dumped (with no error messages).

Element 2: Last Block to Dump

***ONLY:** Only the data block specified by the first part of the DTABLK parameter is dumped.

***LAST:** The range of data blocks that are dumped starts with the data block specified by the start-data-block value and goes to the last block in the file.

end-data-block: Specify the number of the last data block to dump within each file. If this number is less than the number specified for the start-data-block part of the DTABLK parameter, an error message is sent to the user who requested the dump, and the tape is not dumped. If the end-data-block value is larger than the actual number of data blocks in the data file, then all blocks from the start-block number to the end of the file are dumped (with no error messages).

Other Single Values

***ALL:** All data blocks in the specified data files on this volume are dumped. If a data file is continued from another volume or continues onto another volume, only the part of the data file that is stored on this volume is dumped.

***LAST:** Only the last data block in the data file is dumped.

VOLLBL

Specifies whether volume labels are dumped. This parameter is ignored for volumes that are not labeled.

***YES:** All volume labels (including user-specified labels) are dumped.

***NO:** No volume labels are dumped; the volume printout does, however, include the volume identifier of a labeled volume and other basic information for any dumped tape.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The tape contains data in the EBCDIC character code. The dump output contains the hexadecimal value and the EBCDIC character equivalent of each data byte.

***ASCII:** The tape contains data in the ASCII character code. The dump output contains the hexadecimal value and the ASCII character equivalent of each data byte.

ENDOPT

Specifies whether the tape is rewound only or rewound and unloaded after the operation ends.

***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE:** The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

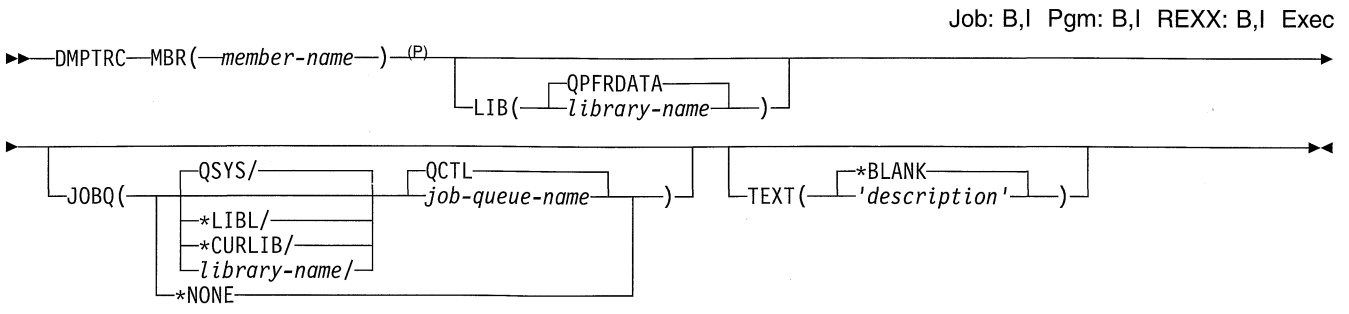
***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

Example

```
DMPTAP  DEV(QTAPE2)  SEQNBR(5)  TYPE(*DTABLK)
  DTABLK(3 7)
```

This command dumps information from the tape volume that is on device QTAPE2. Data blocks 3 through 7 within the data file specified by sequence number 5 are dumped to a printer file.

DMPTRC (Dump Trace) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Dump Trace (DMPTRC) command copies data from the vertical microcode (VMC) trace table to a database file. The user can optionally run the job interactively or submit it as a batch job.

Required Parameters

MBR

Specifies the member name of the database file where the trace table data is dumped. If STRPFRMON was used to collect the trace data, the same name that was specified on the Start Performance Monitor (STRPFRMON) command should be used. If Performance Tools/400 is installed, this is a requirement for the reporting function of the Print Transaction Report (PRTTNSRPT) command.

Optional Parameters

LIB

Specifies the library where the database file for trace data is located. If the file is not found in the specified library, the system automatically creates it in that library.

| The name of the database file can be qualified by one of the following library values:

QPFRDATA: The data is located in the IBM-supplied performance data library, QPFRDATA.

library-name: Specify the name of the library to be searched.

JOBQ

Specifies the qualified name of the job queue on which | this job is placed. The name of the job queue can be qualified by one of the following library values:

QSYS: The IBM-supplied system library, QSYS, is used to locate the job queue.

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCTL: The IBM-supplied controlling subsystem QCTL is used.

job-queue-name: Specify the name of the job queue.

***NONE:** No job is submitted. The DMPTRC request runs interactively.

TEXT

Specifies text that briefly describes the database member. More information on this parameter is in | Appendix A, "Expanded Parameter Descriptions."

***BLANK:** Text is not specified.

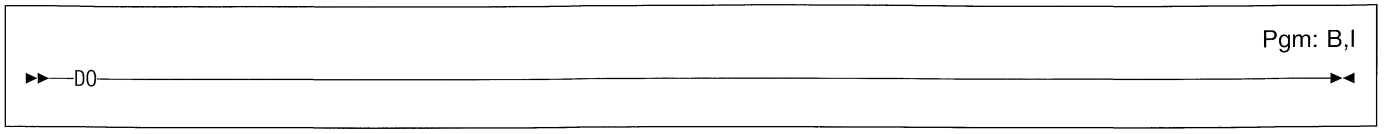
'description': Specify no more than 50 characters of text, enclosed in apostrophes.

Example

```
DMPTRC  MBR(TUESAM)
```

This command causes existing VMC trace data to be written to the member TUESAM in library QPFRDATA. The file used is QAPMDMPT. The request is submitted to the job queue QCTL in library QSYS. It runs as a batch job.

DO (Do) Command



Purpose

The Do (DO) command allows the user to group commands within a CL program. It is used with the ENDDO command to identify a group of commands that are run together as a group. Usually, the DO command specifies the starting of a group of commands that are run as a result of a decision made by the processing of an IF command. However, the DO command does not have to be associated with an IF command. When used with an IF command, the DO command can be either the true part of the decision (that is, the value of the THEN parameter of the IF command), or the false part of a decision (on the ELSE command). Every Do group must be ended by the ENDDO command. Do groups can be nested within other Do groups, but each group must have an ENDDO command to end its level of nesting.

Restriction: This command is valid only within a CL program. Up to 10 levels of Do groups can be nested within each other.

There are no parameters for this command.

Examples

Example 1: Processing a Group of Commands

```
DO
   •
   •   (group of CL commands)
   •
ENDDO
```

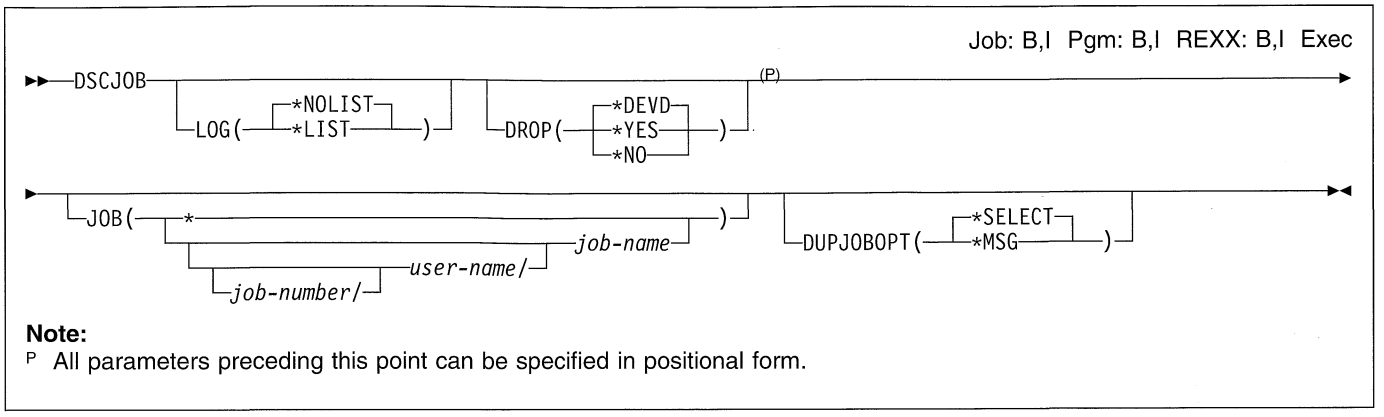
The commands between the DO and ENDDO commands are processed once, as a group of commands.

Example 2: Processing a Group of Commands

```
If &SWITCH DO
   •
   •   (group of CL commands)
   •
ENDDO
```

The commands between the DO and ENDDO commands are processed if the value in the logical variable &SWITCH is '1'. If &SWITCH is not '1', then control passes immediately to the next command following the ENDDO command.

DSCJOB (Disconnect Job) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Disconnect Job (DSCJOB) command allows the interactive user to disconnect all the jobs at the work station and return to the sign-on display.

Restrictions:

1. A job being disconnected must be an interactive job.
2. A job which is being held cannot be disconnected.
3. A pass-through job cannot be disconnected unless the user has used the system request to return the source system from the pass-through target.
4. A user must have *JOBCTL authority to disconnect a job.
5. A job cannot be disconnected if PC organizer is active.

Optional Parameters

LOG

Specifies whether the job log for this interactive job is deleted or is included in the job's spooled output for printing. This entry takes precedence over the LOG parameter value specified for the job. This parameter only has meaning if the disconnected job is canceled due to exceeding the disconnect time interval. The time interval is defined by the system value QDSCJOBITV.

***NOLIST:** An intermediate representation of the program listing is not created.

***LIST:** The job log is spooled for printing with the job's spooled output.

DROP

Specifies whether the switched line attached to the work station is disconnected (dropped) if no other work stations on the same line are signed on. This parameter is ignored if the work station is attached to a non-switched line.

***DEVD:** The value specified on the DROP parameter of the work station's device description is assumed.

***YES:** The switched line is disconnected when the job is disconnected if no other work stations are signed on the line.

***NO:** The switched line is not disconnected when the job is disconnected.

JOB

Specifies the name of a job being disconnected from a work station. The job or jobs that are disconnected from a work station can be reconnected when the user again signs on the same work station.

A job identifier is a qualified name with up to three elements. For example:

```
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: The user must have *JOBCTL authority to specify the name of an active or interactive job.

job-name: Specify the name of the job that is being disconnected. If no job qualifier is given, all jobs currently in the system are searched for the name of the job. If duplicates of the specified name are found, a qualified job name must be specified. The qualified job name includes the user name and/or the job number.

user-name: Specify the name of the user who owns the job. Specifying the user as a qualifier is necessary only if a duplicate job name exists for different users. If a duplicate job name exists for the same user, the job must be qualified with the job number.

job-number: Specify the job number. Specifying the job number as a qualifier is necessary only if a duplicate job name exists for the same user.

DUPJOBOPT

Specifies the action taken when duplicate jobs are found by this command.

***SELECT:** The selection display is shown when duplicate jobs are found during an interactive session. Otherwise, a message is issued.

***MSG:** A message is issued when duplicate jobs are found.

Examples

Example 1: Disconnecting All Interactive Jobs

```
DSCJOB
```

This command enables the user of the work station to disconnect all the interactive jobs associated with the work station. The switched line is dropped only if that is specified in the work station device description of this work station and if no other work station on this line is active. If the job is disconnected when the disconnect interval in the QDSCJOBITV system value is reached, the job is ended and the job log is not included with the job's spooled output.

Example 2: Disconnecting Job Without Releasing Switched Line

```
DSCJOB  LOG(*LIST)  DROP(*NO)
```

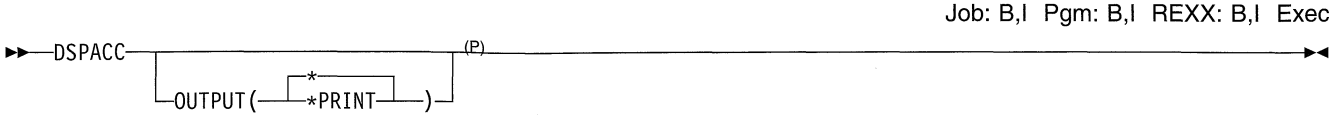
This command disconnects the interactive job, but the switched line is not released. If the job is ended due to the QDSCJOBITV system value, the job log is included with the job's spooled output.

Example 3: Deleting Information in Job Log

```
DSCJOB  LOG(*NOLIST)  DROP(*DEVD)  JOB(123497/DEPT1/DSP04)
```

This command disconnects the interactive job 123497/DEPT1/DSP04 and any other jobs on that work station, for example, secondary jobs or group jobs. If the job is disconnected when the disconnect interval in the QDSCJOBITV system value is reached, the job is ended and the job log is not included with the job's spooled output. The work station device description is checked to determine whether the switched line is disconnected.

DSPACC (Display Access Code) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Access Code (DSPACC) command shows the access codes currently defined in the system. The display shows both the access code number and the descriptive text associated with the access code. The entries on the display are shown in numeric order from the lowest number to the highest number.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

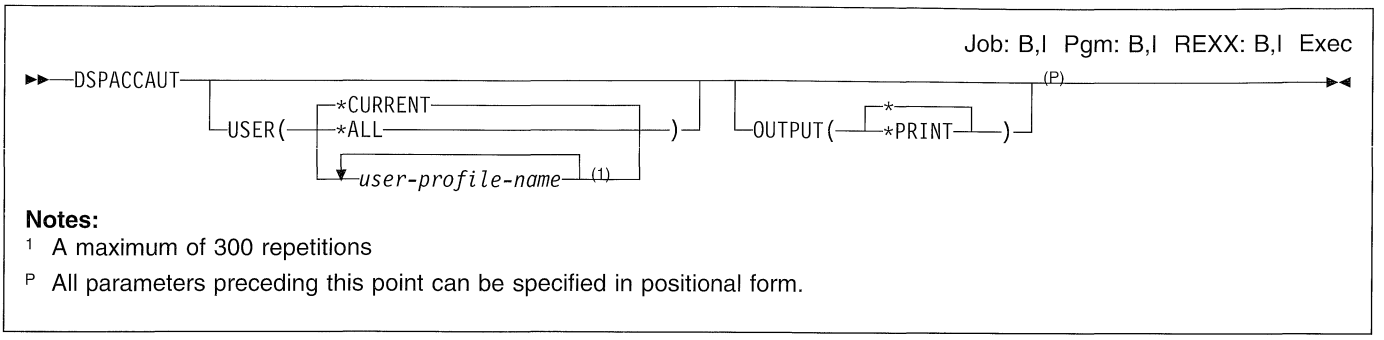
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPACC
```

This command, if entered interactively, displays all access codes currently on the system.

DSPACCAUT (Display Access Code Authority) Command



Notes:
1 A maximum of 300 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Access Code Authority (DSPACCAUT) command shows the access codes to which a user or group of users currently have authority. The display shows a list of user profile names, as well as the access codes to which each user is authorized. The access codes are in numeric sequence for each user.

Optional Parameters

USER

Specifies the user profile name for which authorized access codes are displayed.

***CURRENT:** The user profile under which the current job is running is used.

***ALL:** All user profile names for users in the system distribution directory with access code authority are displayed, as well as the access codes to which those users have authority.

user-profile-name: Specify the name of the user profile for which the access codes are displayed. Up to 300 user profile names can be specified.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

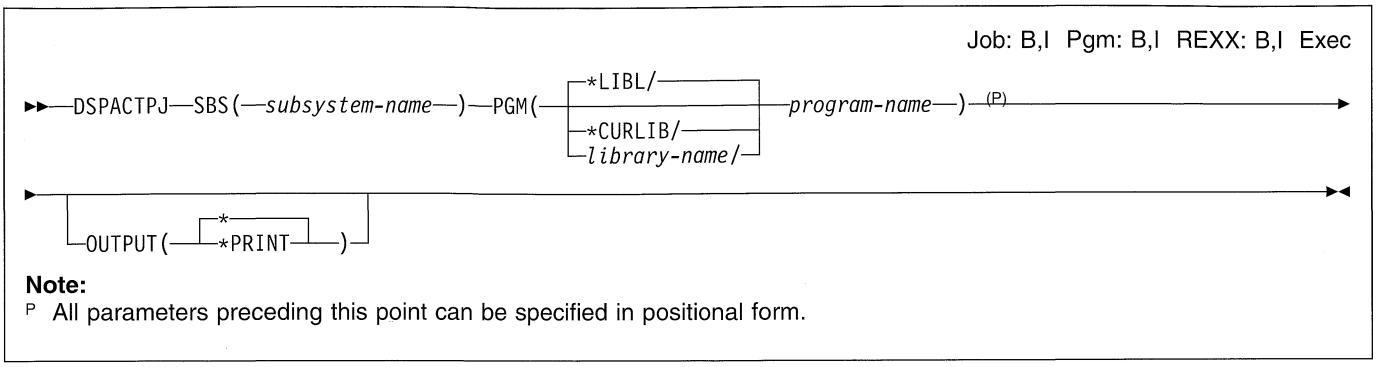
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPACCAUT  USER(*CURRENT)  OUTPUT(*PRINT)
```

This command prints all access codes to which the current user is authorized.

DSPACTPJ (Display Active Prestart Jobs) Command



Note:

| (P) All parameters preceding this point can be specified in positional form.

Purpose

The Display Active Prestart Jobs (DSPACTPJ) command displays statistics and performance information for active prestart jobs associated with a prestart job entry in an active subsystem. All information on the display is collected from the time the prestart job entry is started (either at subsystem startup or when the STRPJ command is run for this entry), or from the time the reset key is pressed.

Restriction: This command is restricted to a user with *USE authority.

Required Parameters

SBS

Specifies the name of the active subsystem that contains the active prestart jobs.

PGM

Specifies the qualified name of the program in the active prestart job entry.

| The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program in the active prestart jobs.

Optional Parameters

OUTPUT

Specifies whether the output from the command is being shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Job Information

```
DSPACTPJ  SBS(PJSBS)  PGM(QGPL/PGM1)
```

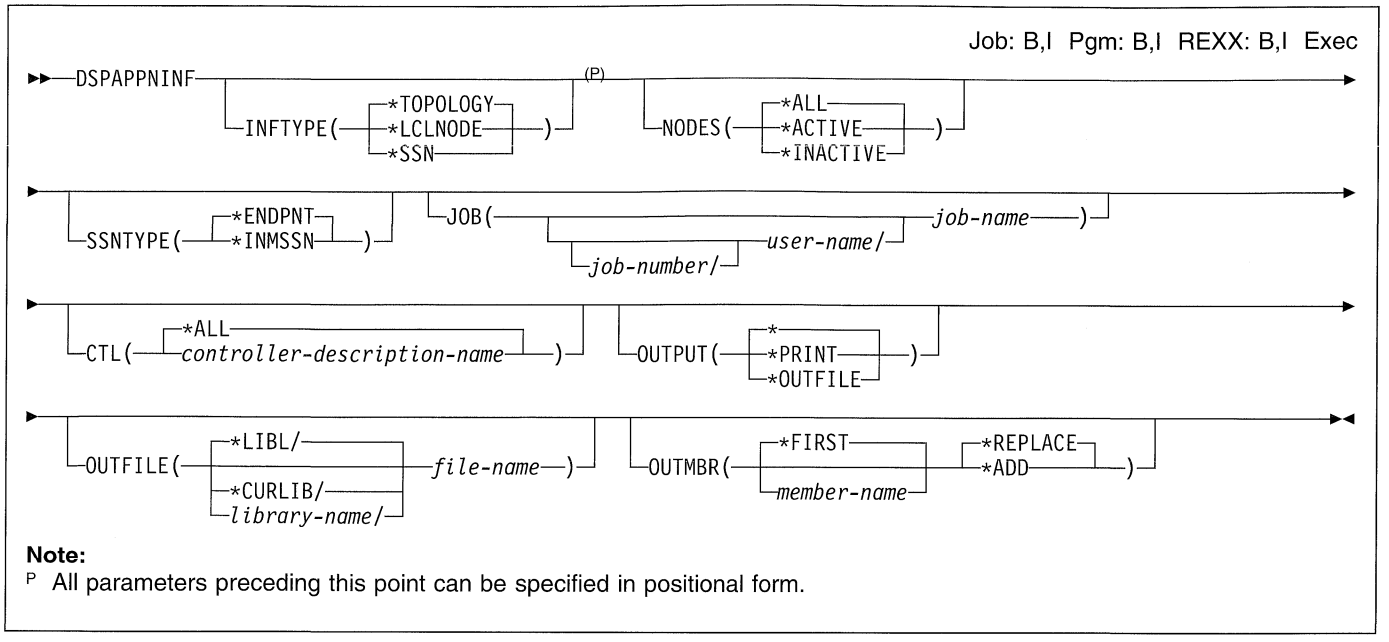
This command displays information for the prestart job entry in subsystem PJSBS with program PGM1 in the QGPL library.

Example 2: Printing Job Information

```
DSPACTPJ  SBS(PJSBS)  PGM(QGPL/PGM2)  OUTPUT(*PRINT)
```

This command prints active prestart job information for the prestart job entry in the active subsystem PJSBS with program PGM2 in the QGPL library.

DSPAPPNINF (Display APPN Information) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display APPN Information (DSPAPPNINF) command provides the user with Advanced Peer-to-Peer Networking (APPN) network information which is used to assist in problem analysis. The user specifies one basic type of network information being shown, printed, or stored in an output file: information about the network topology, the local directory, or session information. More information on APPN networking is in found in the *APPN Guide*.

Optional Parameters

INFTYPE

Specifies the type of information that is shown or printed.

***TOPOLOGY:** The contents of the topology database is shown or printed. This consists of the set of all nodes in the network, their characteristics, and the set of link destination nodes along with the characteristics of the connecting links for each node.

***LCLNODE:** The contents of the local directory is shown or printed. This consists of the local node and the names of all remote control points for which information exists in the local directory.

***SSN:** Information about intermediate sessions or sessions that have the local node as a session end point is shown or printed.

NODES

Specifies which set of nodes will be displayed. This parameter is only valid when INFTYPE(*TOPOLOGY) is specified.

***ALL:** Information about all nodes in the network is printed or displayed.

***ACTIVE:** Information about all network nodes that have at least one active link is printed or displayed.

***INACTIVE:** Information about all network nodes that have no active links is printed or displayed.

SSNTYPE

Specifies which type of session information is shown. This parameter is valid only when INFTYPE(*SSN) is specified.

***ENDPNT:** Information about sessions for which the local node is a session end point is shown or printed.

***INMSSN:** Information about intermediate sessions being routed through the local node is shown or printed.

JOB

Specifies the name of the job for which session information is shown or printed. If OUTPUT(*) is specified and no job name is specified, a list of APPN job names run since the most recent initial program load (IPL) of the system is shown. A job name can be selected from that list. If OUTPUT(*PRINT) or OUTPUT(*OUTFILE) is specified without a job name, the session information for all APPN jobs is printed or stored in a specified output file.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
job-name
user-name/job-name
job-number/user-name/job-name
```

This parameter is valid only if SSNTYPE(*ENDPNT) is specified.

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

CTL

Specifies the controller description name for which intermediate session information is displayed or printed. This parameter is valid only if SSNTYPE(*INMSSN) is specified.

***ALL:** Intermediate session information for all controllers will be displayed or printed.

controller-description-name: Specify the name of the controller description for which intermediate information will be displayed or printed.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** The requested data is shown on the display station.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the output file that is used to store the requested APPN information. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file that is used to store the requested APPN information.

OUTMBR

Specifies the name of the database file member to which the output is directed. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

Element 1: Member to Receive Output

***FIRST:** The requested APPN information is stored in the first file member.

member-name: Specify the name of the output file member used to store the requested APPN information. The member name can be from 1 to 10 characters in length.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Printing a List

```
DSPAPPNINF  INFTYPE(*TOPOLOGY)
  NODES(*ALL)  OUTPUT(*PRINT)
```

This command prints the list of all nodes currently existing in the APPN network, the set of links destination nodes associated with each node, and the link characteristics for each link.

Example 2: Showing a List of PCIDs

```
DSPAPPNINF  INFTYPE(*SSN)  SSNTYPE(*ENDPNT)
  JOB(APPNJOB/USERPROF/000001)  OUTPUT(*)
```

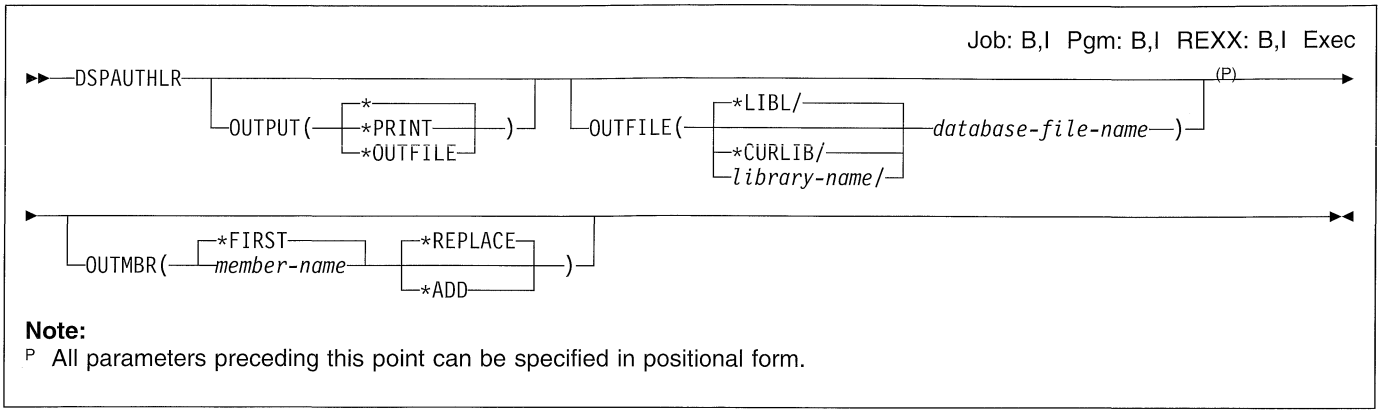
This command shows a list of procedure correlation session identifiers (PCIDs) associated with the job name APPNJOB/USERPROF/000001. From this list, the user can specify an option to show additional information about a session.

Example 3: Storing Contents of a Directory

```
DSPAPPNINF  INFTYPE(*LCLNODE)  OUTPUT(*OUTFILE)
  OUTFILE(USERLIB/APPNFILE)  OUTMBR(*FIRST, *REPLACE)
```

This command stores the contents of the local directory in the first member of an output file named USERLIB/APPNFILE. If information already exists in this member, the new information replaces the existing information.

DSPAUTHLR (Display Authority Holder) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Authority Holder (DSPAUTHLR) command shows a list of authority holders. The list shows the name of the object that the authority holder secures, the name of the library where the object is found, the object type, and the owner of the object.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the name of the database file to which the output of the display is directed. If this file does not exist, this command creates a database file in the specified library. If the file is created by this function, the text is 'Outfile for DSPAUTHLR' and the public authority is *EXCLUDE.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member already exists, the user has the option to add new records to the end of the existing member or clear the member and then add the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it. If the member already exists, the user has the option to add new records to the end of the existing member or clear the member and then add the new records.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

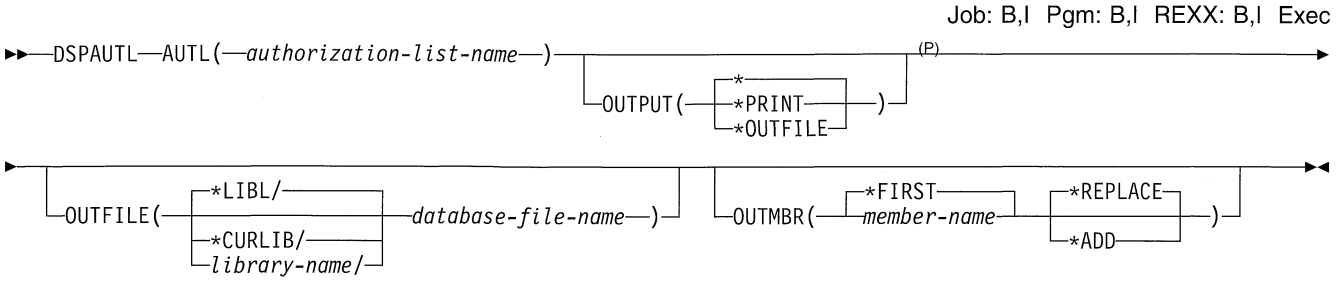
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPAUTHLR   OUTPUT(*PRINT)
```

This command sends the display of the authority holder list to the printer.

DSPAUTL (Display Authorization List) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Authorization List (DSPAUTL) command shows the list of users (and their levels of authority) that make up the authorization list.

The owner of the authorization list, and any user who is not specified as *EXCLUDE on the authority list, can view the whole list of users and their authorities when they display an authorization list. The authority to see the list may be from a user's group profile or may have been adopted.

When an authorization list is first shown, the specific authorities shown are determined by the level of detailed information specified on the USROPT parameter in the user profile.

When the DSPAUTL command is used to display an authorization list, the user specifies the name of the authorization list, whether the authorization list should be shown on the display or sent to a printer, or (optionally) whether the output should be sent to an OUTFILE.

Required Parameter

AUTL

Specifies the name of the authorization list to be shown.

Optional Parameters

OUTPUT

Specifies whether the output is sent to the display, to a printer file, or to a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the name of the database file to which the output of the display is directed. If the OUTFILE does not exist, this command creates a database file in the specified library. If the file is created by this function, the descriptive text is 'Outfile for DSPAUTL' and the public authority is *EXCLUDE. The database format (QSYDSAUT) of the output file is the same as that used in the IBM-supplied database file QAOBJAUT.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member already exists, the option exists to add new records to the end of the existing outfile member or clear the member and add the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it. If the member already exists, the option exists to add new records to the end of the existing outfile member or clear the member and add the new records.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

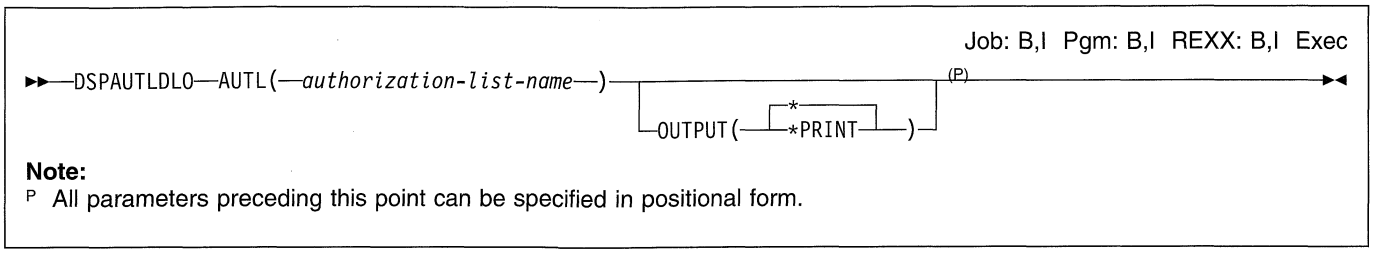
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPAUTL   AUTL(DEPT48X)  OUTPUT(*PRINT)
```

This command sends the display of the authorization list to the printer.

DSPAUTLDLO (Display Authorization List Document Library Objects) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Authorization List Document Library Objects (DSPAUTLDLO) command shows the list of documents and folders whose access security is specified by the authorization list specified in the AUTL parameter.

Restrictions:

1. If a user on the list has authority other than *EXCLUDE, or is not on the list and public authority is something other than *EXCLUDE, that user is authorized to display the documents and folders.
2. If the user is not authorized to the document or folder because access authorities are private, the document library object is marked "not authorized" in the text field.

Required Parameters

AUTL

Specifies the name of the authorization list whose list of documents and folders is displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the spooled output of the job.

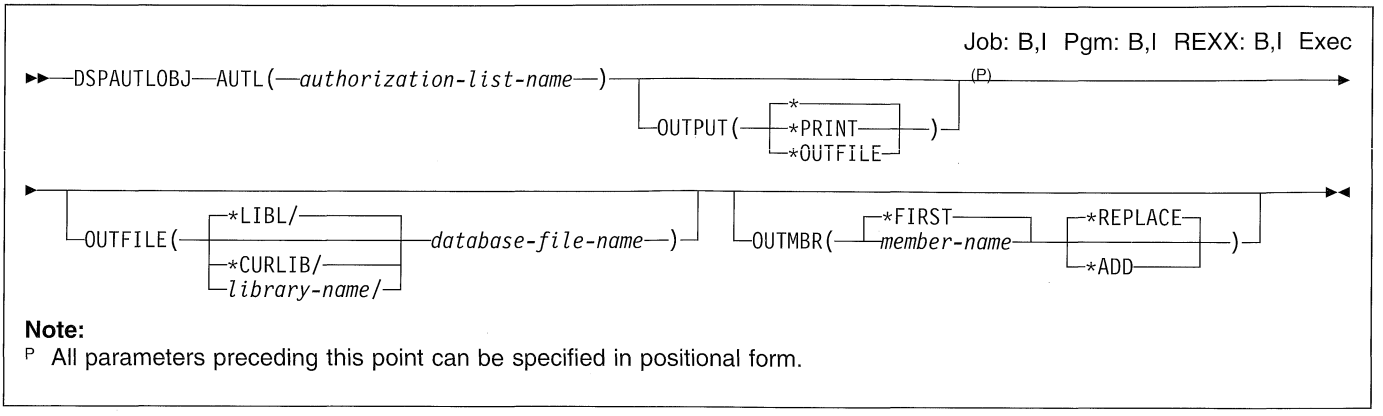
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPAUTLDLO  AUTL(PAYROLL)  OUTPUT(*PRINT)
```

This command sends the display output for the authorization list named PAYROLL to a printer.

DSPAUTLOBJ (Display Authorization List Objects) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Authorization List Objects (DSPAUTLOBJ) command shows the list of objects that are secured by the authorization list specified in the AUTL parameter. If a user is on the list with authority other than *EXCLUDE, or is not on the list and public authority is something other than *EXCLUDE, the user is authorized to display the objects. If the user is not authorized to an object because of private authorities, the object is marked *not authorized* in the text field.

Required Parameter

AUTL

Specifies the name of the authorization list whose list of objects is shown.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file.

| More information on this parameter is in Appendix A, | "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file to which the output of the display is directed. If this file does not exist, this command creates a database file in the specified library. If the file is created by this function, the text is 'Outfile for DSPAUTLOBJ' and the public authority is *EXCLUDE.

| The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member already exists, the user has the option to add new records to the end of the existing member or to clear the existing member and then add the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

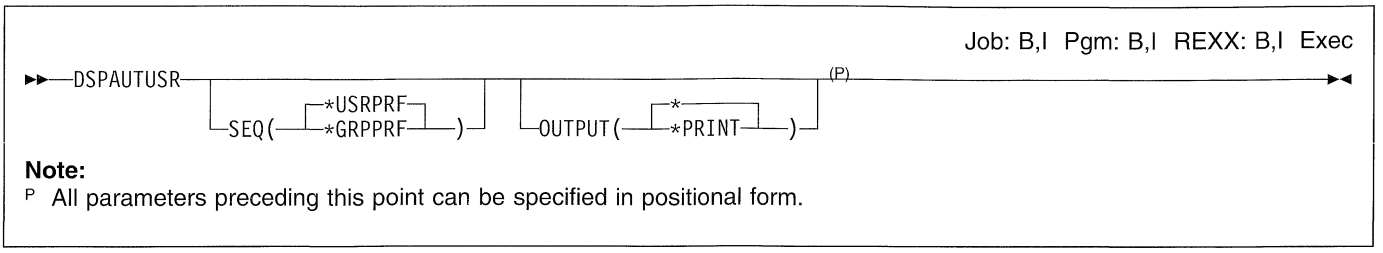
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPAUTLOBJ  AUTL(PAYROLL)  OUTFILE(*LIBL/PAYROLL)
   OUTMBR(DARL *REPLACE)
```

This command places the output in the database file PAYROLL, member name DARL. If member DARL already exists, the system clears it and adds the new records.

DSPAUTUSR (Display Authorized Users) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Authorized Users (DSPAUTUSR) command displays or prints the names of the authorized system users, in alphabetic order. The following information is provided for each user: the group profile of which the user is a member, the most recent password change date, whether the user profile has a password, and the text of the user profile.

Note: While this command is searching for user profile information to display, another job cannot change user profiles (for example, with the Change User Profile (CHGUSRPRF) command).

Restriction: The list of system users contains only the names of the user profiles to which the user of this command has at least read (*READ) authority.

Optional Parameters

SEQ

Specifies whether the list of system users is in alphabetic sequence by user profile name or by group profile name.

***USRPRF:** The list is in alphabetic sequence by user profile name.

***GRPPRF:** The list is in alphabetic sequence by group profile name. The members of each group are listed in alphabetical order by user profile name.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Authorized Users and Group Profile Names

```
DSPAUTUSR
```

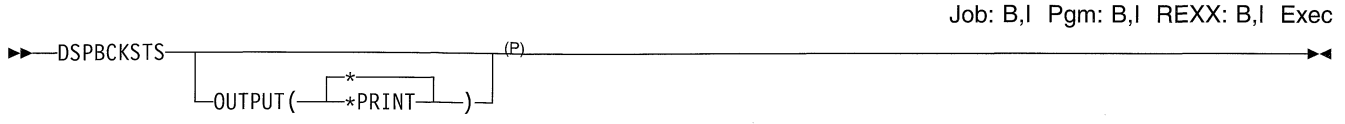
This command shows the list of authorized users and their group profile names. The list is in alphabetic sequence by user profile name. OUTPUT(*) is also assumed; the list is shown on the display or printed depending on whether the command was submitted at a work station or as part of the batch input stream.

Example 2: Printing Output

```
DSPAUTUSR  SEQ(*GRPPRF)  OUTPUT(*PRINT)
```

This command causes authorized system user profile names and their group profile names to be printed. The output is printed in alphabetic sequence by group profile name.

DSPBCKSTS (Display Backup Status) Command



Note:
P All parameters preceding this point can be specified positionally.

Purpose

The Display Backup Status (DSPBCKSTS) command allows the user to display information about the tape sets used for backup and what was saved on each of them. Only information about backups performed using the backup options is displayed.

Optional Parameters

OUTPUT
Specifies whether the backup status is displayed or printed.

*: The output is displayed if requested by an interactive job, or printed with the job's spooled output if requested by a batch job.

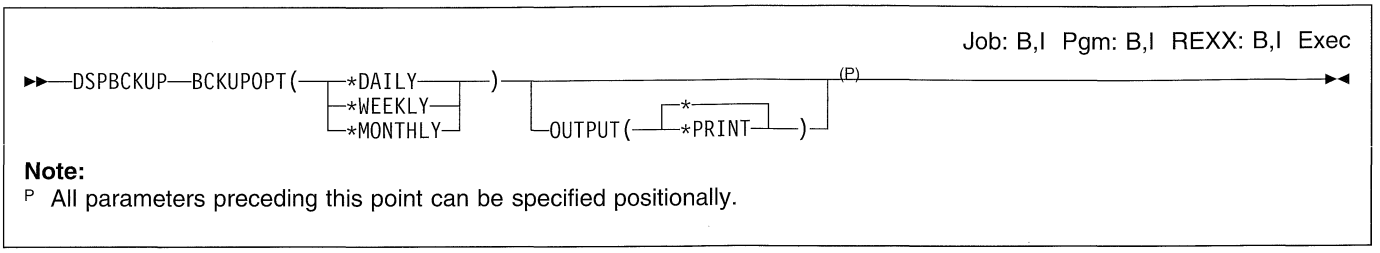
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPBCKST  OUTPUT(*PRINT)
```

This command prints the backup status.

DSPBCKUP (Display Backup Options) Command



Note:
P All parameters preceding this point can be specified positionally.

Purpose

The Display Backup Options (DSPBCKUP) command allows the user to display the options in one of the predefined backups. More information on backup is in the *Advanced Backup and Recovery Guide*.

Required Parameters

BCKUPOPT
Specifies the backup options to be displayed.

***DAILY:** The daily backup options are displayed.

***WEEKLY:** The weekly backup options are displayed.

***MONTHLY:** The monthly backup options are displayed.

Optional Parameters

OUTPUT
Specifies whether the backup options are displayed or printed.

***:** The output is displayed if requested by an interactive job, or printed with the job's spooled output if requested by a batch job.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Backup Options

```
DSPBCKUP  BCKUPOPT(*DAILY)
```

This command displays the daily backup options.

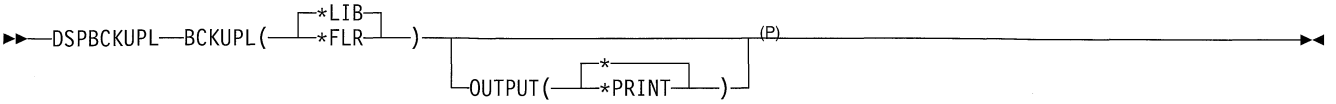
Example 2: Printing Backup Options

```
DSPBCKUP  BCKUPOPT(*MONTHLY)  OUTPUT(*PRINT)
```

This command prints the monthly backup options.

DSPBCKUPL (Display Backup List) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:
P All parameters preceding this point can be specified positionally.

Purpose

The Display Backup List (DSPBCKUPL) command allows the user to view libraries and folders for backup. More information on backup is in the *Advanced Backup and Recovery Guide*.

Required Parameters

BCKUPL
Specifies the backup list to display.

***LIB:** The library backup list is displayed.

***FLR:** The folder backup list is displayed.

Optional Parameters

OUTPUT
Specifies whether the backup list is displayed or printed.

***:** The output is displayed if requested by an interactive job, or printed with the job's spooled output if requested by a batch job.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying a Library Backup List

```
DSPBCKUPL  BCKUPL(*LIB)
```

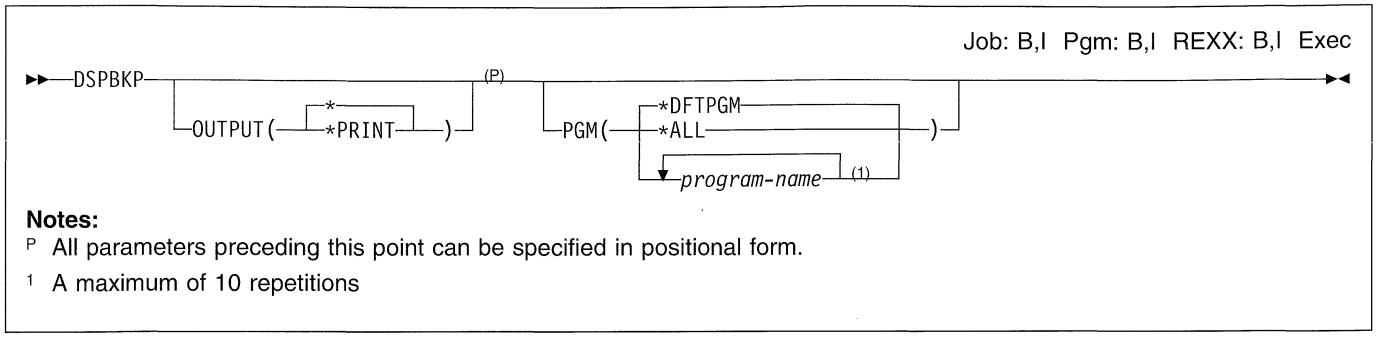
This command displays the library backup list.

Example 2: Printing a Folder Backup List

```
DSPBCKUPL  BCKUPL(*FLR)  OUTPUT(*PRINT)
```

This command prints the folder backup list.

DSPBKP (Display Breakpoints) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 10 repetitions

Purpose

The Display Breakpoints (DSPBKP) command shows the locations of all the breakpoints currently set in the specified programs running in the debug mode. The breakpoints and the names of the program variables associated with each breakpoint are shown.

Restrictions:

1. This command is valid only in the debug mode. To start debug mode, refer to the STRDBG (Start Debug) command.
2. This command cannot be used to display breakpoints in a bound program.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting display station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

PGM

Specifies which programs in the debug mode have their breakpoint locations and associated program variables shown.

***DFTPGM:** The breakpoint locations in the default program are shown.

***ALL:** The breakpoint locations in the programs currently in the debug mode are shown.

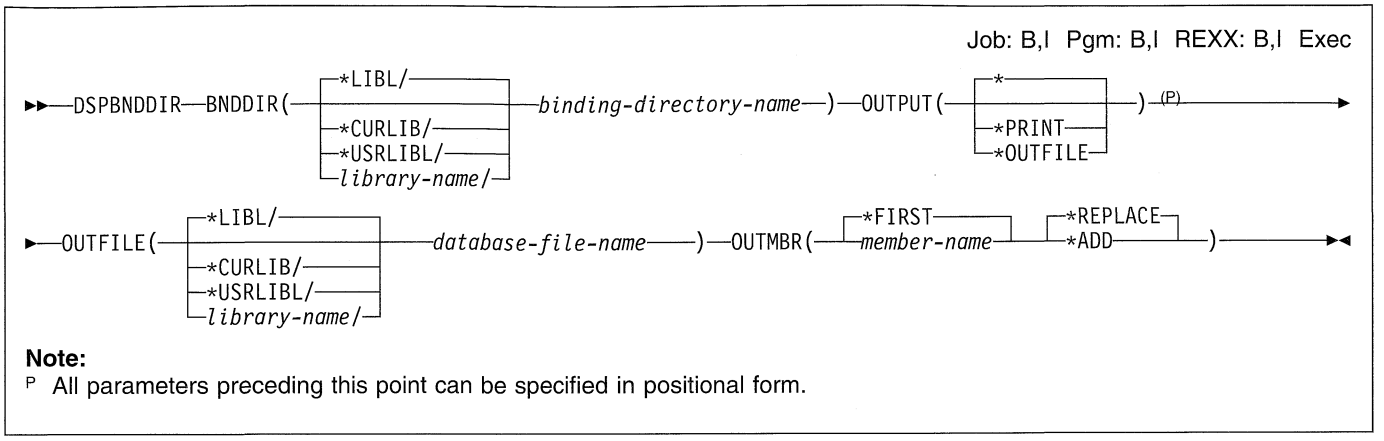
program-name: Specify the names of from 1 through 10 programs already in the debug mode whose breakpoint locations are to be shown.

Example

```
DSPBKP
```

Assuming that program MYPROG is the default program in an interactive debug session, this command shows all of the breakpoint locations that are currently set in MYPROG. The names of the program variables associated with each breakpoint are also shown.

DSPBNDDIR (Display Binding Directory) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Binding Directory (DSPBNDDIR) command displays the contents of a binding directory.

Restrictions:

1. You must have *USE authority to the library where the binding directory is located.
2. You must have object operational and *READ authority to the binding directory.

Required Parameters

BNDDIR
Specifies the binding directory to be displayed.

The name of the binding directory can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

binding-directory-name: Specify the name of the binding directory to be displayed.

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE
Specifies the qualified name of the database file to which the output of the display is directed. If the OUTFILE does not exist, this command creates a database file in the specified library. The public authority of the file is the same as the create authority specified for the library in which the file is created. The record format of the output file is the same as that used in the IBM-supplied database file. The IBM-supplied database file and record format follows:

File	QABNDBND
Record Format	QBNDSPBD

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display.

OUTMBR
Specifies the name of the database file member to which the output of the display is directed. If a member already exists, and *REPLACE is specified, the system clears it and adds the new records. If the member does not exist and a member name is not specified, the system creates a member with the name of the file spec-

ified in the OUTFILE parameter. If a member name is specified, but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

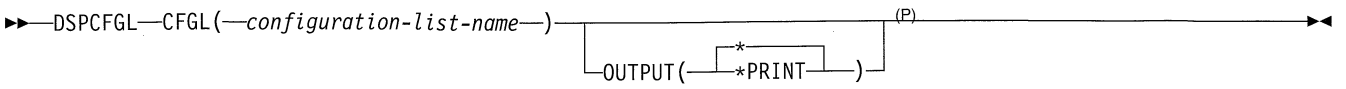
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPBNDDIR   BNDDIR(STORE)
```

This command displays a binding directory named STORE.

DSPCFGL (Display Configuration List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Configuration List (DSPCFGL) command displays the entries of a configuration list. Output is directed to a display or a spooled printer file as indicated by the OUTPUT parameter and job type.

Required Parameters

CFGL
Specifies the name of the configuration list being displayed.

Optional Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

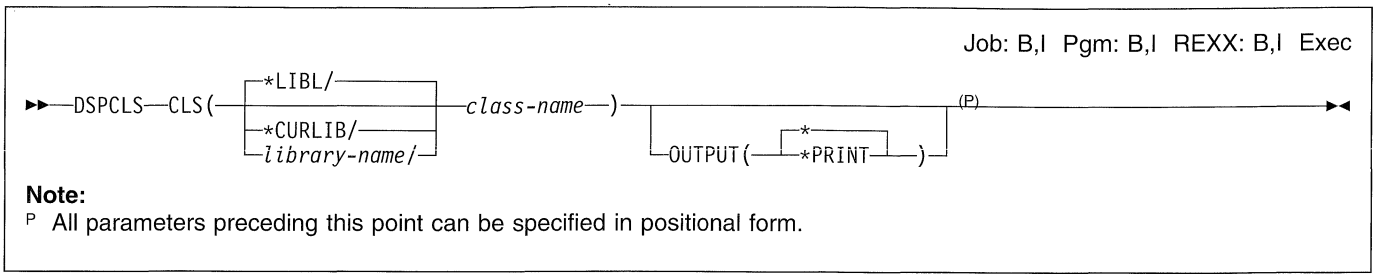
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCFGL  CFGL(CONFIG01)
```

This command displays the configuration list named CONFIG01.

DSPCLS (Display Class) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Class (DSPCLS) command is used during system set-up when work is divided on the system or when the user wants to know the run attributes such as job priority and time slice of a class.

Restriction: The user, typically a system operator or system administrator, must have object operational authority for the class before its attributes can be displayed.

Required Parameters

CLS

Specifies the qualified name of the class that has its attributes displayed. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

The name of the class can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

class-name: Specify the name of the class that has its attributes displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

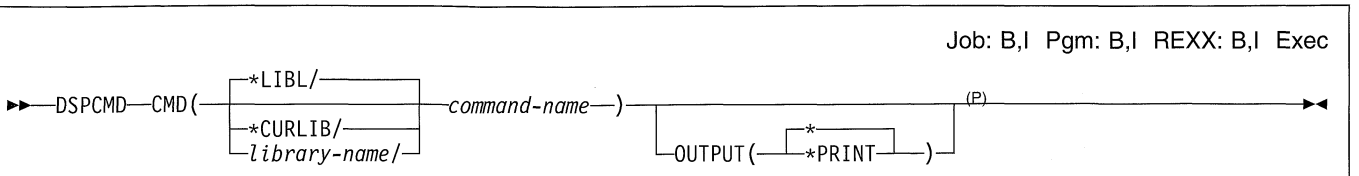
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCLS  CLS(CLASS1)  OUTPUT(*PRINT)
```

This command directs the attributes of class CLASS1 to the job's output spooling queue to be printed.

DSPCMD (Display Command) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Command (DSPCMD) command shows some of the values that were specified for parameters in the Create Command (CRTCMD) command.

Required Parameters

CMD

Specifies the qualified name of the user-defined or IBM-supplied command.

The name of the command can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

command-name: Specify the name of the command values being displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

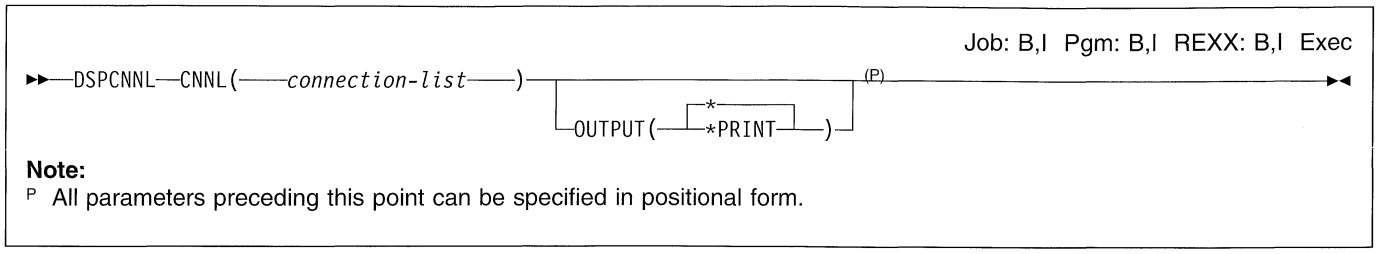
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCMD    CMD(PAYROLL)
```

This command shows all current user-assigned parameter values for the user-defined command PAYROLL at the display station.

DSPCNNL (Display Connection List) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Connection List (DSPCNNL) command displays a connection list and its entries.

Required Parameters

CNNL
Specifies the name of the connection list.

Optional Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).

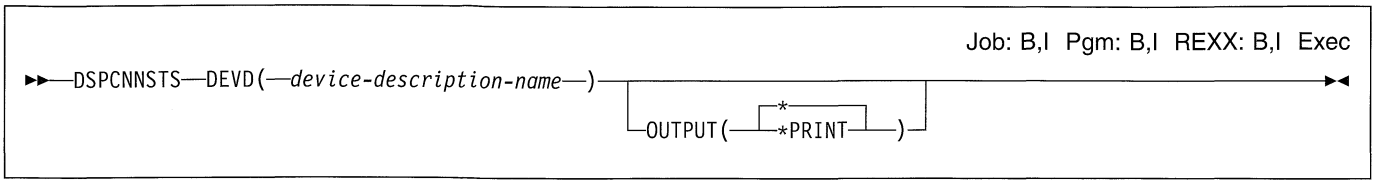
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCNNL   CNNL(MYCNNL)
```

This command displays information about the connection list named MYCNNL. The information is displayed on the work station from which the command was submitted. If the command is entered from a batch job, the output from the display is printed with the job's spooled output. All entries associated with the connection list are displayed.

DSPCNNSTS (Display Connection Status) Command



Purpose

The Display Connection Status (DSPCNNSTS) command shows current information about connection-oriented protocols used by, and all acceptable inbound routing data specified for, network devices. If one or more connections are active, the connection characteristics are shown for each.

Restrictions:

1. The user must have operational authority to the device specified on the DEVD parameter.
2. This command is valid for all network devices, but connection-oriented status is provided only for devices with a link type of X.25.

Required Parameters

DEVD

Specifies the name of the network device description whose active connections are being displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

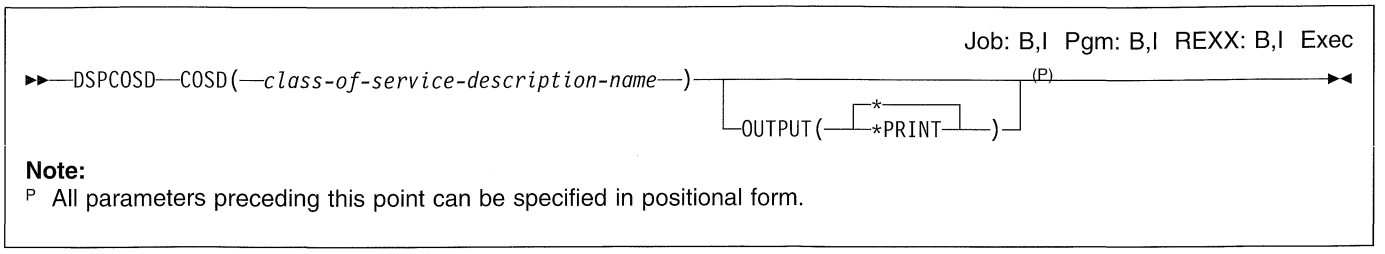
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCNNSTS   DEVD(PRTR48X)  OUTPUT(*PRINT)
```

This command prints the status of all active connections for device PRTR48X.

DSPCOSD (Display Class-of-Service Description) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Class-of-Service Description (DSPCOSD) command displays a class-of-service description. Output is directed to a display or a spooled printer file as indicated by the OUTPUT parameter and job type.

Required Parameters

COSD

Specifies the name of the class-of-service description.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

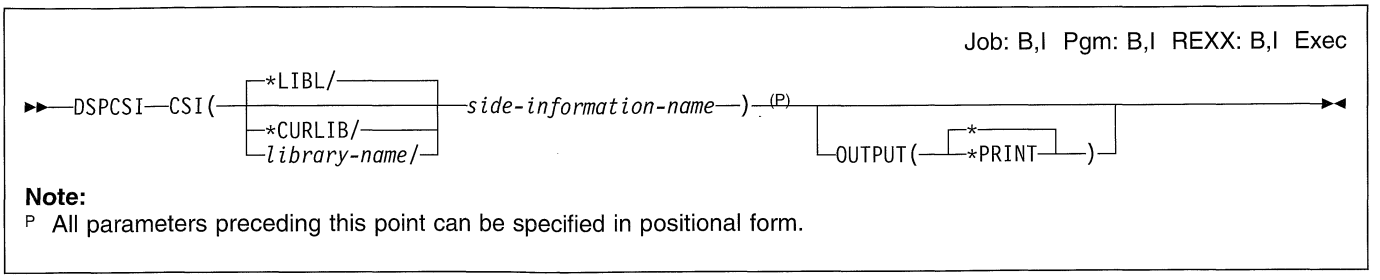
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCOSD  COSD(COSD1)  OUTPUT(*PRINT)
```

This command prints the class-of-service description COSD1. The information is displayed on the work station from which the command was submitted (unless *PRINT was not specified, in which case the information is sent to a spooled printer file associated with the user's job). If the command is entered from a batch job, the output from the display is printed with the job's spooled output on a printer.

DSPCSI (Display Communications Side Information) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Communications Side Information (DSPCSI) command is used to display or print the specified side information object.

Required Parameters

CSI

Specifies the name of the side information object to be displayed. An object name must be specified.

The name of the side information object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

side-information-name: Specify the name of the object that contains the desired side information. This is the Common Programming Interface (CPI)-Communications symbolic destination name (sym_dest_name).

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Locating an Object

```
DSPCSI CSI(SIDEOBJ)
```

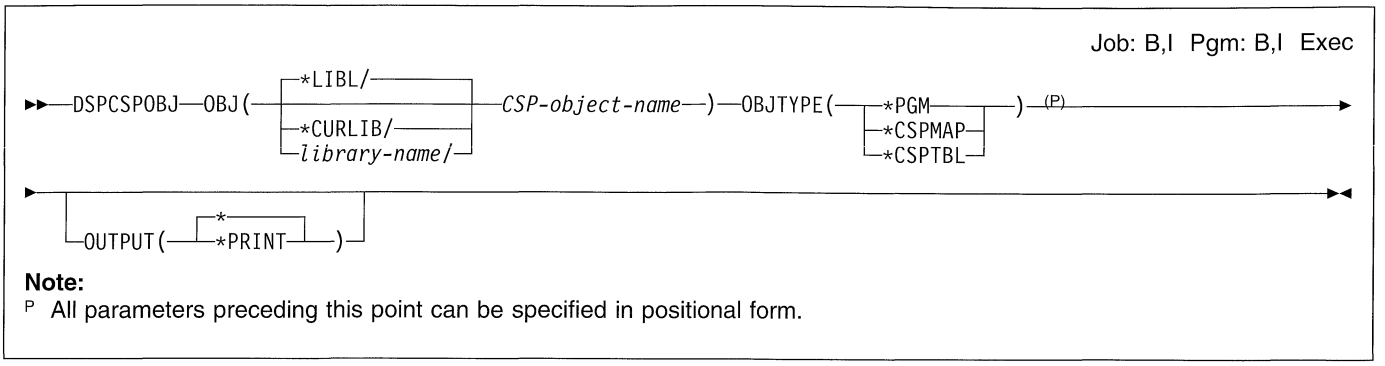
This command locates the first side information object named SIDEOBJ in the library list and displays the side information.

Example 2: Printing Side Information

```
DSPCSI CSI(QGPL/SIDEOBJ) OUTPUT(*PRINT)
```

This command prints the side information contained in the object SIDEOBJ in library QGPL with the job's spooled output.

DSPCSPOBJ (Display CSP/AE Object) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Display CSP/AE Object (DSPCSPOBJ) command displays a Cross System Product/Application Execution (CSP/AE) object in hexadecimal and character representation. Object types valid for this command are *CSPMAP, *CSPTBL, and *PGM (attribute CSPAE only). This command may be used while analyzing problems or while debugging an application.

Required Parameters

OBJ

Specifies the qualified name of the CSP/AE object to display in hexadecimal and character representation.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

CSP-object-name: Specify the name of the object to be displayed.

OBJTYPE

Specifies the type of the CSP/AE object to display.

***PGM:** The CSP/AE program object is displayed.

***CSPMAP:** The CSP/AE map group object is displayed.

***CSPTBL:** The CSP/AE table object is displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

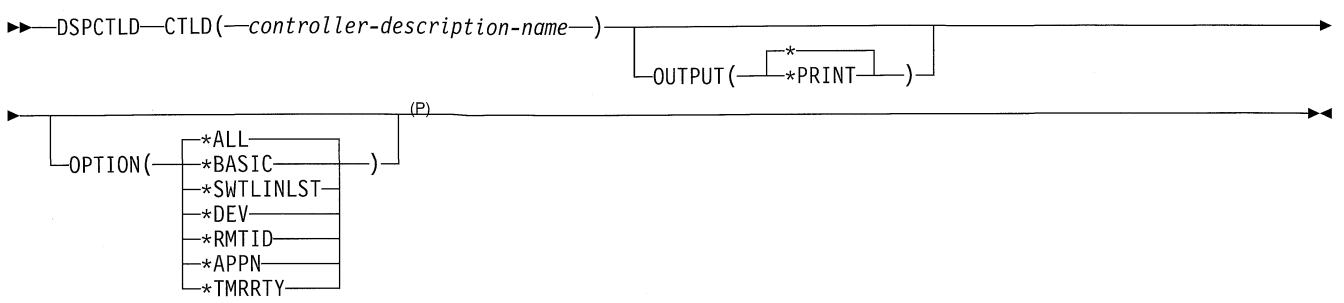
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPCSPOBJ  OBJ(INVNT)  OBJTYPE(*PGM)
```

This command displays the contents of the program object INVNT if found in the user's library list. Output from the command is displayed at the work station.

DSPCTLD (Display Controller Description) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Controller Description (DSPCTLD) command displays a controller description. Output is directed to a display or spooled printer file as indicated by the OUTPUT and OPTION parameters as well as the job type.

Required Parameters

CTLD

Specifies the name of the controller description being displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

OPTION

Specifies the displays to be shown.

***ALL:** All displays that apply to this controller are shown.

***BASIC:** The basic displays that apply to this controller are shown, but additional information displays are not shown.

***SWTLINLST:** The switched line list displays that apply to this switched or switched network backup (SNBU) controller are shown.

***DEV:** The device attachment displays that apply to this controller are shown.

***RMTID:** The remote identifiers for switched binary synchronous communications (BSC) controllers are displayed.

***APPN:** The advanced peer-to-peer networking (APPN) values for this controller are displayed.

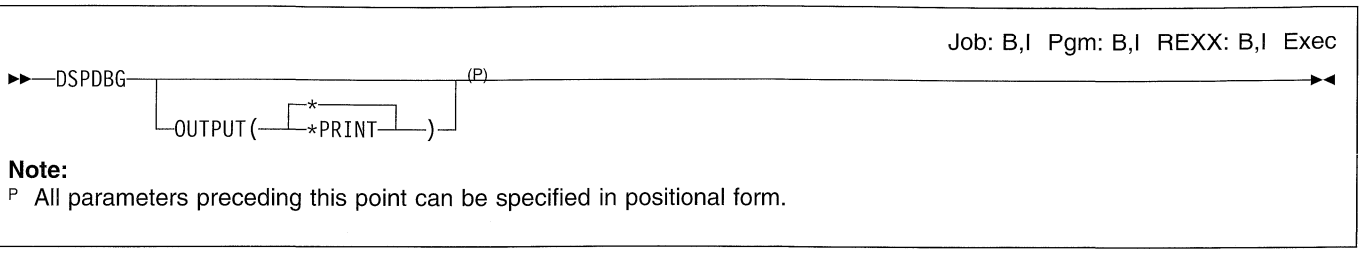
***TMRRTY:** The timer and retry options for this controller are displayed.

Example

```
DSPCTLD   CTLD(CONTROL01)
```

The command displays information about the controller description named CONTROL01. The information is displayed on the work station from which the command was submitted. If the command is entered from a batch job, the output from the display is printed with the job's spooled output.

DSPDBG (Display Debug) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Debug (DSPDBG) command shows the current status of debug mode. It shows the following:

- The call stack, indicating which programs are currently being debugged
- The instruction number of the calling instruction or the instruction number of each breakpoint at which a program is stopped
- The recursion level

Programs that are in debug mode but have not been called are also shown.

Restrictions:

1. This command is valid only in debug mode. To start debug mode, see the Start Debug (STRDBG) command.
2. This command cannot be used if the user is servicing another job, and that job is on a job queue, or is being held, suspended, or ended.
3. This command cannot be used to show the procedures of a bound program on the stack. Use the Display Job (DSPJOB) command to show those procedures.
4. This command cannot be used to show bound programs that are being debugged.

Optional Parameter

OUTPUT

Specifies whether the output from the command is shown at the requesting display station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

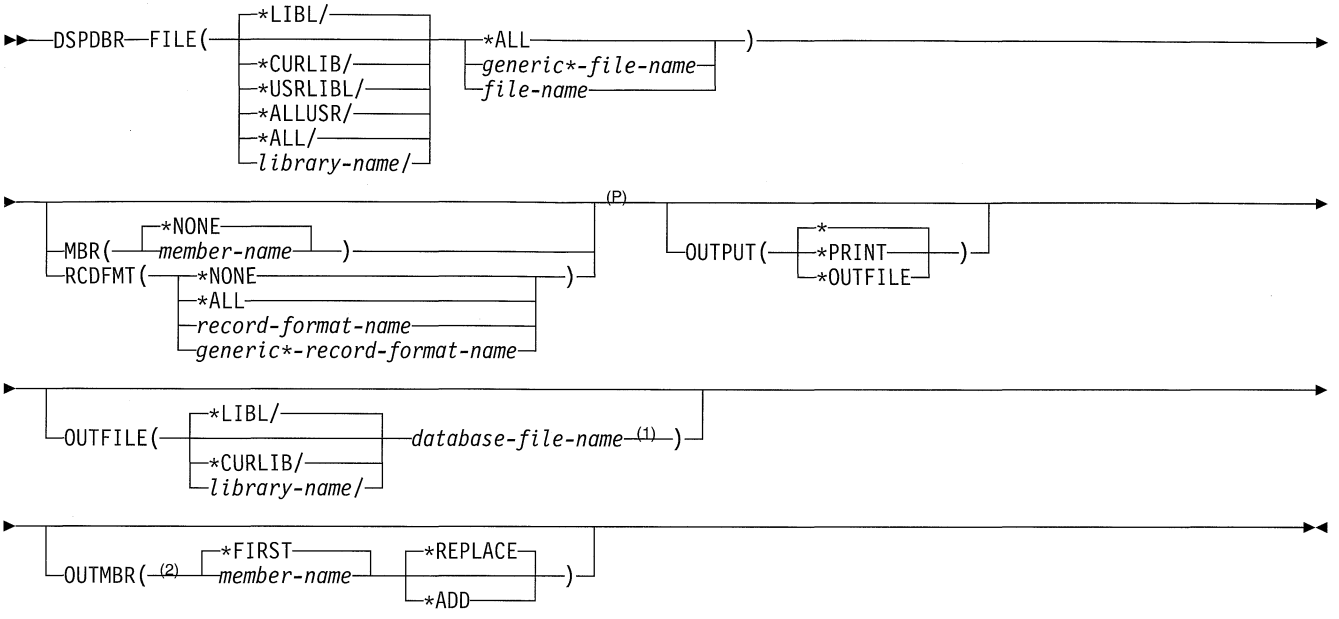
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPDBG
```

If entered interactively, this command shows the current attributes of debug mode at the display station. Also shown are the breakpoints at which any of the programs being debugged are stopped, the recursion levels of the programs that are currently active, and the names of the programs that have not been called.

DSPDBR (Display Database Relations) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 If OUTPUT(*OUTFILE) is specified, a file name must be specified for OUTFILE.

2 If OUTFILE is not specified, OUTBMR must not be specified.

Purpose

The Display Database Relations (DSPDBR) command gives relational information about database files. The information identifies the physical and logical files that are dependent on a specific file, files that use a specific record format, or file members that are dependent on a specific file member. This command is used to display or print the information, or to place the information in a database output file (an OUTFILE). This command does not apply to device files.

If the information is put in a database output file, the record format that is used is named QWHDRDBR. The fields in record format QWHDRDBR are the same as the fields in the IBM-supplied format QWHDRDBR in file QADSPDBR in the library QSYS. The following information is contained in the database output file:

- For each file specified in the command, the database record contains:
  - The name of the specified file, its library name, and the file type of the specified file
  - The name of the record format used for the file, if a name is specified for RCDFMT
  - The information retrieval date(s) for the file information; the latest date contains the most accurate information, if changes have been made to the files
- One of the following is also included in the record:
  - The names of all the files that are dependent on the specified file for access path sharing or data sharing; the names of the libraries containing the files and the type of sharing are also included
  - The names of all the file members that are dependent on the specified member, their library names, and the type of sharing
  - The names of all the files that are dependent on the specified record format, and their library names

Restrictions:

1. To show each file specified, the user must have object operational authority for the file. Also, of the libraries specified by the library qualifier, only the libraries for which the user has *USE authority are searched for the files.
2. To create an OUTFILE, the user must have *USE authority to the Create Physical File (CRTPF) command and *ADD authority to the library. To use an existing OUTFILE, the user must have operational (*OPR) and *ADD authority to the file.
3. The user must have object management and delete authority also, if *REPLACE is specified.

Required Parameters

FILE

Specifies the qualified name of the database file or the generic name of several database files about which relational information is shown. This parameter can also specify that all files in the specified library or libraries (*LIBL/*ALL, for example), or all files in all libraries (*ALL/*ALL), have relational information about them shown. Only the libraries in the specified library qualifier that the user either owns or is authorized to use are searched for the files.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All files in the specified library (or in all libraries identified in the library qualifier to which the user has access) have their relational information shown.

generic-file-name:* Specify the name of the database file or the generic name of several database files in the specified library qualifier that have relational information shown. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

file-name: Specify the full name of the database file that has its relational information shown.

Optional Parameters

MBR

Specifies the name of the member in a database file whose dependent information is shown, printed, or placed in a database file.

***NONE:** No information about the file member's relations is provided. File relations or record format relations are provided.

member-name: Specify the name of the database file member whose relations information is provided. If a member name is specified, either one file name or a generic file name must be specified. If MBR is specified, RCDFMT cannot be specified.

RCDFMT

Specifies the name of the record format of the database file member whose relations information is provided. If RCDFMT is specified, MBR cannot be specified.

***NONE:** No record format relations information is provided.

***ALL:** Relations information about all record formats in the specified files is provided.

record-format-name: Specify the full name of the record format whose relations information is provided.

generic-record-format-name:* Specify the generic name of the record format or several record formats in the specified files whose relations information is provided. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or placed in a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output on a printer.

***OUTFILE:** The output is sent to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database output file where the specified relational information is stored. If the specified file does not exist, the system creates a database file and member in the specified library.

When creating the database output file, the current date, time, and system name must be included. The system name is the name of the source system, not the target system.

If the file is created, the text is "Outfile for DSPDBR" and the public authority is *EXCLUDE.

Note: The outfile format must be the same as QWHDRDBR of system file QADSPDBR in the QSYS library.

The name of the database output file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file where the relational information is stored. If the specified file is not found, a file and a file member are created in the specified library, or in the current library if its name is not qualified. This file can be used when other DSPDBR commands are entered. The IBM-supplied database file, QADSPDBR, cannot be specified.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified in the OUTFILE parameter.

member-name: Specify the name of the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it. If the member exists, the user has the option to add records to the end of the existing member or to clear the existing member before adding the new records.

Element 2: Operation to Perform on Member

***REPLACE:** If a member exists, the system clears it and adds the new records.

***ADD:** If a member exists, the system adds the new records to the end of the existing records.

Examples

In the following examples, assume that there is an interactive environment and that the user of the command is authorized to access all relevant libraries and objects.

Example 1: Displaying Database Relations Information

```
DSPDBR   FILE(LIBRARY1/FILE1)  RCDFMT(FORMAT1)
```

This command shows a list of the names and database relations information for all files that use the FORMAT1 format and are associated with FILE1 in LIBRARY1. Because the environment is interactive, the output is shown at the work station running this command.

Example 2: Displaying Database Relations Information

```
DSPDBR   FILE(LIB1/FILE1)
```

This command shows database relations information for all files that are dependent on FILE1 in LIB1 for data sharing. It is shown at the work station running this command.

Example 3: Displaying Database Relations Information

```
DSPDBR   FILE(LIB1/FILE1)  MBR(MEMBER1)
```

This command shows database relations information for all members that are dependent on MEMBER1 in FILE1 in LIB1 for data sharing or access path sharing. They are shown at the work station running this command.

Additional Considerations

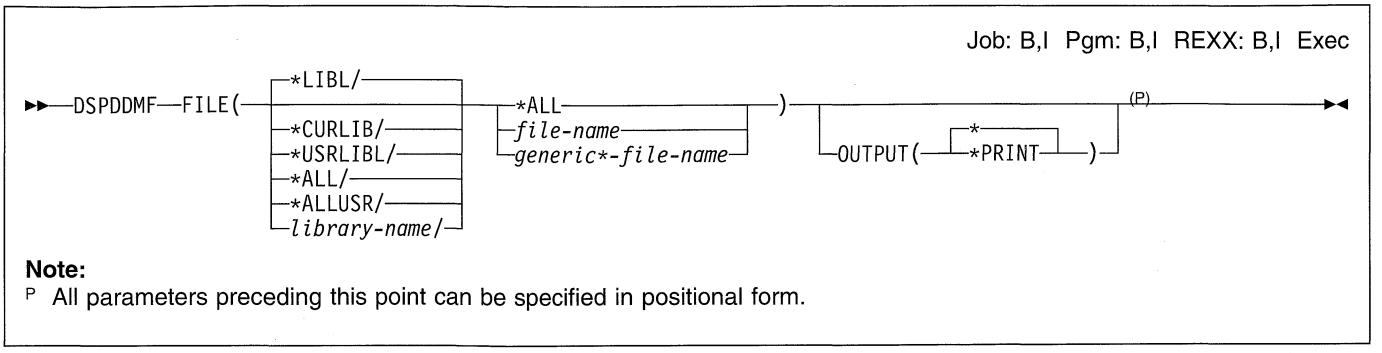
When the DSPDBR command is entered, the database is searched for the specified relationships and a record is generated for each relationship (dependency) that is found. The records are placed in the printer device file named QPDSPDBR. If OUTPUT(*PRINT) is specified on the command, the records are listed on the printer in the following order:

- Header information, which lists the DSPDBR command input values and the files identified in the database that match the command's request.
- The names of the files that are dependent on the files identified by the FILE parameter. A file can be dependent on another file for its access path or its data. This information is listed when neither MBR nor RCDFMT is specified.
- If a member name was specified in the MBR parameter, the names of the file members that are dependent on a member in the file (a member can be dependent on another member for its access path or its data).

- The names of the database files that are dependent on a record format of the specified files, if a record format name was specified in the RCDFMT parameter.

If the DSPDBR command is entered interactively and OUTPUT(*) is specified or assumed, the records in the printer device file are shown instead of printed.

DSPDDMF (Display Distributed Data Management File) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Distributed Data Management File (DSPDDMF) command is used to display or print information, such remote file name and remote system location, about a specified distributed data management (DDM) file.

Required Parameters

FILE

Specifies the qualified name of the DDM file whose description is being changed.

| The name of the DDM file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All files in the library or libraries specified by the library qualifier have their descriptions shown.

file-name: Specify the name of the file whose description is shown.

generic-file-name:* Specify the generic name of the file. | A generic name is a character string of one or more | characters followed by an asterisk (*); for example, | ABC*. The asterisk substitutes for any valid characters. | A generic name specifies all objects with names that | begin with the generic prefix for which the user has | authority. If an asterisk is not included with the generic | (prefix) name, the system assumes it to be the complete | object name. If the complete object name is specified, | and multiple libraries are searched, multiple objects can | be displayed only if *ALL or *ALLUSR library values can | be specified for the name. For more information on the | use of generic functions, refer to "Rules for Specifying | Names."

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter | Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

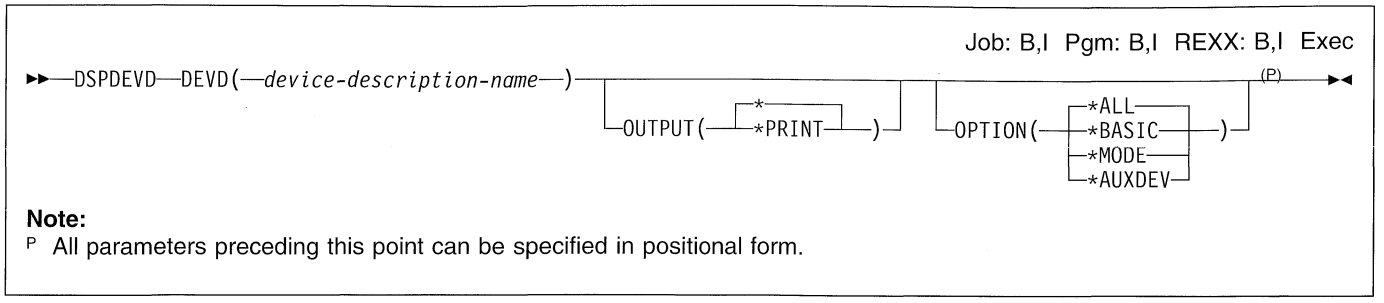
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPDDMF
```

This command shows the Display DDM File display.

DSPDEVD (Display Device Description) Command



Purpose

The Display Device Description (DSPDEVD) command displays a device description. Output is directed to a display or a spooled printer file as indicated by the output and option parameters, and job type.

Required Parameters

DEVD

Specifies the name of the device description being displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

OPTION

Specifies the displays to be shown.

***ALL:** All displays that apply to this device type are shown.

***BASIC:** The basic displays that apply to this device type are shown, but additional information displays are not shown.

***MODE:** The display showing mode attachments to this advanced program-to-program communications (APPC) device is shown.

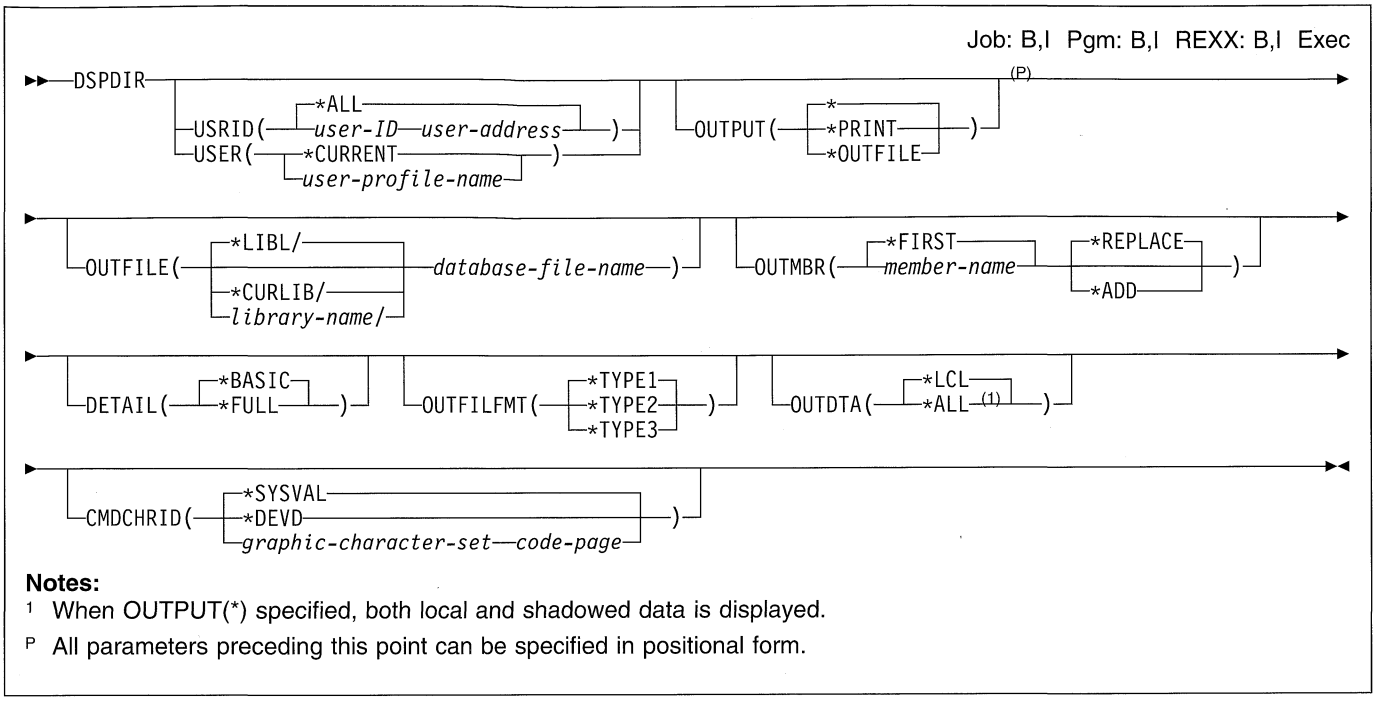
***AUXDEV:** The display showing auxiliary devices for this 5292 model 2 is shown.

Example

```
DSPDEVD  DEVD(WRKSTN01)
```

This command displays information about the device description named WRKSTN01. The information is displayed on the work station from which the command was submitted. If the command is entered from a batch job, the output from the display is printed with the job's spooled output.

DSPDIR (Display Directory) Command



Purpose

The Display Directory (DSPDIR) command is used to display, print, or create a database file for some or all system distribution directory entries. The database file output is displayed, printed, or created based on the following:

- For displayed output, when the USRID or USER parameter applies to more than one directory entry, the system provides a list of entries. When these parameters uniquely identify a directory entry, the system provides the details for the entry.
- For printed or database file output, if the USRID parameter specifies the full directory, the DETAIL parameter determines whether a list of the entries or the full details for each entry is the output. If the USRID or USER parameter specifies a user ID or profile, the full details for that user are sent to the output.

The DSPDIR command does not allow updating the directory. That function is provided interactively, with display support by the Work with Directory (WRKDIR) command, and with the Add Directory Entry (ADDDIRE), Remove Directory Entry (RMVDIRE), Change Directory Entry (CHGDIRE) commands, and Rename Directory Entry (RNMDIRE) commands.

Optional Parameters

USRID

Specifies the user ID and address of the user for whom the directory entries are displayed, printed, or directed to a database file. If the USRID parameter is specified, the USER parameter cannot be specified.

For displayed output, if the user ID and address specifies an entry that has only one description, the directory details for that entry are displayed. If several descriptions are associated with the specified user ID and address, a list of matching user IDs and addresses is displayed from which the user can select a user ID and address to view the details.

For printed output or database file output, the full directory details, including all descriptions for that user ID and address, are printed or sent to a database file. More information on specifying the user ID and address is in the *Distribution Services Network Guide*.

***ALL:** All entries in the system distribution directory are displayed, printed, or directed to a database file. The entries are provided in alphabetical order by user ID and address. If the output is directed to a printed list or output file, the DETAIL parameter specifies whether a list of user IDs, addresses, and descriptions is the output or if the full directory detail is the output.

Element 1: User ID

user-ID: Specify the user ID of the user for whom an existing directory entry is displayed, printed, or directed to a database file.

Element 2: User Address

user-address: Specify the user address of the user for whom an existing directory entry is displayed, printed, or directed to a database file.

USER

Specifies a user profile for which directory entries are displayed, printed, or output to a database file. If the

USER parameter is specified, the USRID parameter cannot be specified.

For displayed output, if the profile specifies an entry that has only one description associated with it, the directory details for that entry are displayed. If multiple descriptions are associated with the specified profile, a list of all the user IDs and addresses matching the profile is displayed. The user can then select a user ID and address to view the details.

***CURRENT:** The user profile under which the current job is running is used.

user-profile-name: Specify a user profile name whose corresponding directory entry details are the output.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. If the command is run as part of a batch job, the output is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file to which the output of the display is directed. If the database file does not exist, this command creates it in the specified library. If no library is specified, the database file is created in the user default library specified in the user profile. If no default library is specified, the database output file is created in the QGPL library.

If the database file is qualified with *LIBL but the system cannot find the file, the database file is created in the user's default library, if it is specified in the user profile. If the default library is not specified, the file is created in the QGPL library.

See the OUTFILFMT parameter for the valid output file types and their names. More information on defining the format of database files as output files is in the *Office Services Concepts and Programmer's Guide*.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display. If the database file is qualified with *LIBL but the system cannot find the file, one is created in the user's default library, if specified. If the default library was not specified, the file is created in the QGPL library.

OUTMBR

Specifies the name of the database file member to which the output is directed. If a member already exists, the system uses the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member with the name of the output file specified on the OUTFILE parameter. If an output file member name is specified, but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

DETAIL

Specifies how much detail is printed or directed to the database file. This parameter is not used when the output is directed to a display (OUTPUT(*)) or when a specific user ID or profile is specified.

***BASIC:** The output is a list of all user IDs, addresses, and descriptions contained in the directory.

***FULL:** The output is the full set of details for the user in the directory. When USRID(*ALL) is specified, the printed list contains a page for each unique user ID and address from the directory, with all of the details for that entry. Multiple descriptions for a user ID and address are included with the details. For a database file, a record is created for each unique user ID and address and description, containing all of the details for each entry. When several descriptions exist for a user ID and address, a full record is the output for each description, and only the description field is different.

When a user ID and address or user profile name is specified on an input parameter, the printed list contains one page of output with all descriptions included with the details for the user ID and address. The output file has a full detail record for each description. Only the description field is different between the records.

OUTFILFMT

Specifies the format of the output file.

***TYPE1:** The format is defined by model output file QAOSDIRO in library QSYS with record format name OSDIRE. This format does not include the new directory fields added since Release 2.0.

***TYPE2:** The output file format is defined by model output files QAOSDIRB and QAOSDIRF in library QSYS.

If DETAIL(*BASIC) is specified, the output file contains only the basic fields, and model outfile QAOSDIRB is used with record format name OSDIRB. If DETAIL(*FULL) is specified, the output file contains all of the detail fields, and model outfile QAOSDIRF is used with record format name OSDIRF.

***TYPE3:** The output file contains all the detail fields and the X.400 originator/recipient (O/R) name. Model outfile QAOSDIRX in library QSYS is used with record format name OSDIRX.

OUTDTA

Specifies the type of data to include as output to the value specified on the OUTPUT parameter. This parameter is not used when the output is directed to a display (OUTPUT(*)) or when a specific user ID or profile is specified.

***LCL:** Locally-defined data is included. Shadowed data is not included.

***ALL:** All directory entry data is included. This includes locally-defined data and shadowed data.

CMDCHRID

Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the command parameters. Valid values range from 1 through 9999.

Element 2: Code Page

code-page: Specify the code page. Valid values range from 1 through 9999.

Examples

Example 1: Showing the Display for One Description

```
DSPDIR  USRID(HURST NEWYORK)
```

This command shows the Display Directory Entry Details display for user ID and address HURST NEWYORK if this user ID has only one description associated with it. If more than one description exists, the Display Directory Entries display is shown with all entries for the user ID and address.

Example 2: Printing Directory Information

```
DSPDIR  USER(JONES)  OUTPUT(*PRINT)
```

This command prints the full directory details for user profile JONES. If there is more than one description in the directory for JONES, the printed output contains each description.

Example 3: Sending Full Details to a Database File

```
DSPDIR  OUTPUT(*OUTFILE)  OUTFILE(MYLIB/DIRLIST)
  OUTMBR(*FIRST *REPLACE)
  DETAIL(*FULL)
```

This command outputs a record to the database file DIRLIST in library MYLIB for each entry in the directory. This record contains the full details for each user. The format of the output file is the one used prior to Release 3.0 because of the default of OUTFILFMT(*TYPE1). If the file does not exist, it is created. The data replaces the data that is in the file, if it exists.

Example 4: Sending Basic Information to a Database File

```
DSPDIR  OUTPUT(*OUTFILE)  OUTFILE(MYLIB/DIRLIST)
  DETAIL(*BASIC)  OUTFILFMT(*TYPE2)
```

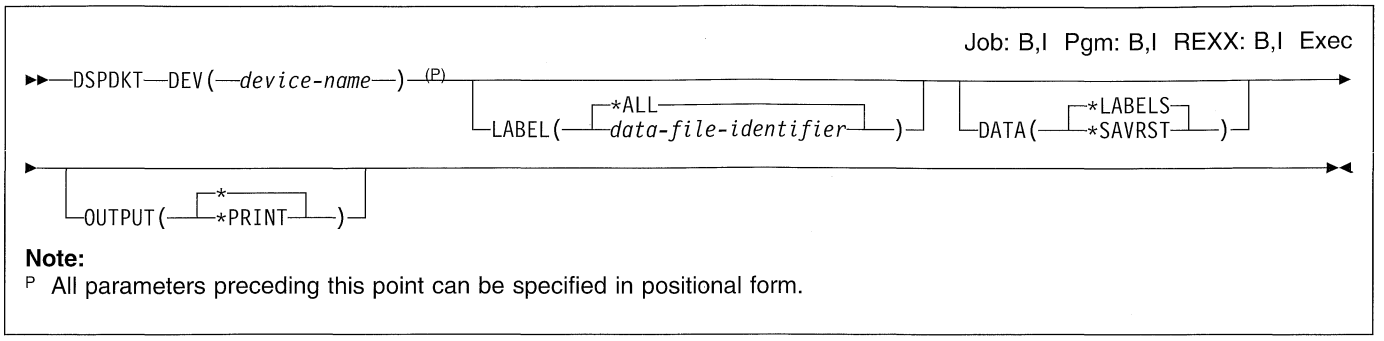
This command outputs a record to the database file named DIRLIST in the MYLIB library for each entry in the directory. The data contains only the basic fields, user ID, address, and description. If the file named DIRLIST in the MYLIB library already exists for releases 1.0 or 2.0 data (see Example 2), an error occurs; otherwise, the file is be created in the Release 3.0 format.

Additional Considerations

Error messages are returned when errors are encountered while processing the DSPDIR command. If the command is entered interactively, error messages are displayed at the work station. When a batch job enters the command, error messages are returned, and the job that runs the command is responsible for processing them.

If the USRID parameter specifies the user ID and address of a user not currently in the directory, an error message is returned. Likewise, if a user profile not contained in the directory is specified on the USER parameter, an error message is returned.

DSPDKT (Display Diskette) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Diskette (DSPDKT) command shows the volume label and data file identifier information that is on the diskette. Information about the objects saved to the diskette can also be shown. The information can be printed or shown on a display device.

If a user specifies DATA(*SAVRST), information is retrieved which includes a description of each object saved to the diskette and summary information about all saved objects.

Note: When showing diskettes with labels that are not IBM standard labels, unpredictable results may occur. To initialize the diskette, run the Initialize Diskette (INZDKT) command with CHECK(*NO) specified.

Required Parameters

DEV

Specifies the name of the device on which the volume being displayed is located.

Optional Parameters

LABEL

Specifies the data file identifier, or all the identifiers, of the data files on the diskette being shown. The data file identifier is stored in a label in the volume label area of the diskette, and it specifies the identifier of the file that exists on the diskette.

***ALL:** All data file identifiers on the diskette specified by the DEV parameter are shown.

data-file-identifier: Specify the data file identifier (17 alphanumeric characters maximum) of the data file that is shown. (For data files in the basic, H-, or I-format, the identifier can contain up to 8 characters.)

DATA

Specifies the type of information that is shown.

***LABELS:** Volume and data file identifiers are shown.

***SAVRST:** Shows the save or restore summary information about the command and each saved object.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output on a printer. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

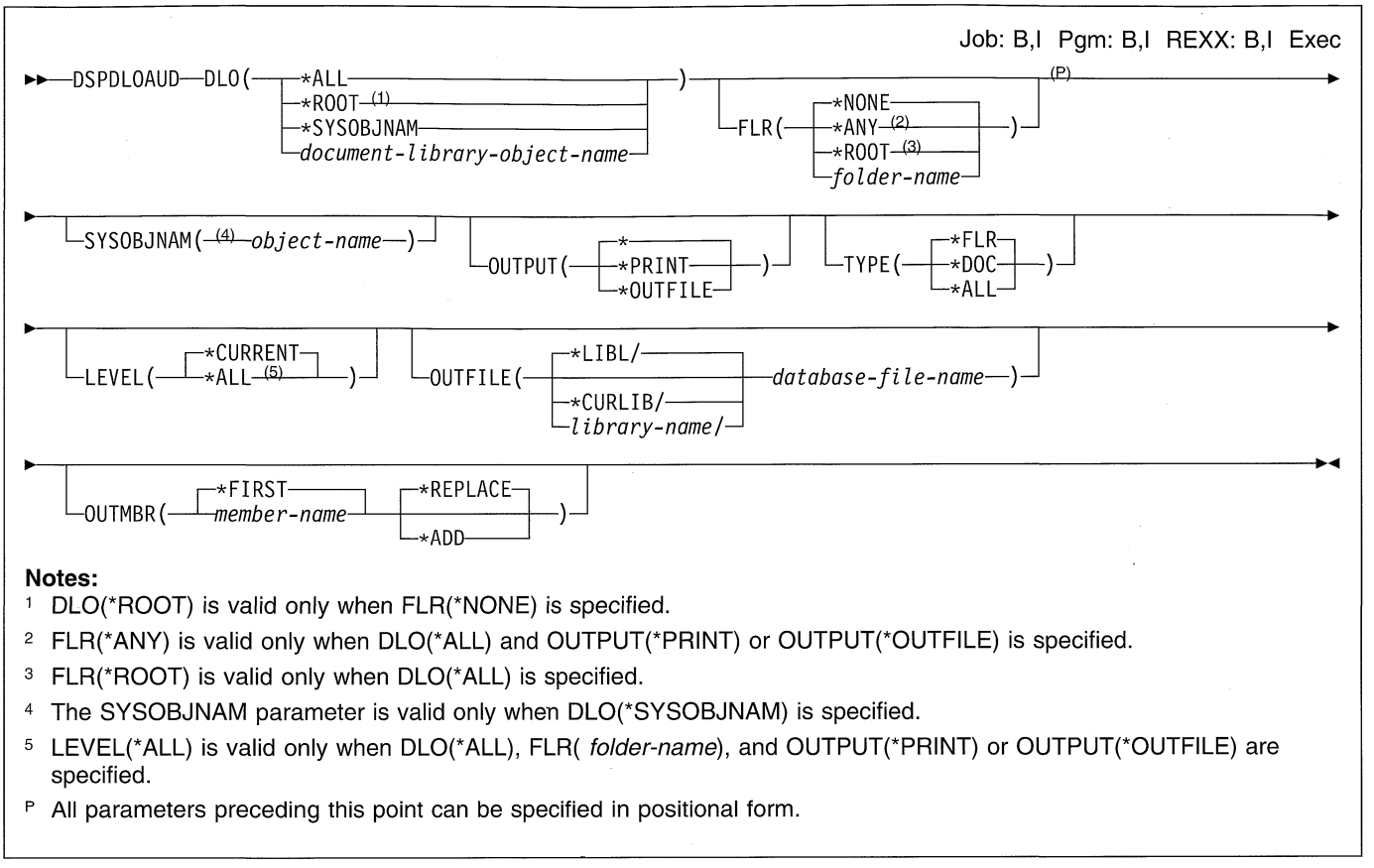
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPDKT   DEV(DKT01)
```

This command shows the volume label and all the data file identifiers for the diskette in device DKT01.

DSPDLOAUD (Display Document Library Object Audit) Command



Notes:

1 DLO(*ROOT) is valid only when FLR(*NONE) is specified.

2 FLR(*ANY) is valid only when DLO(*ALL) and OUTPUT(*PRINT) or OUTPUT(*OUTFILE) is specified.

3 FLR(*ROOT) is valid only when DLO(*ALL) is specified.

4 The SYSOBJNAM parameter is valid only when DLO(*SYSOBJNAM) is specified.

5 LEVEL(*ALL) is valid only when DLO(*ALL), FLR(*folder-name*), and OUTPUT(*PRINT) or OUTPUT(*OUTFILE) are specified.

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Document Library Object Audit (DSPDLOAUD) command allows the user to display the auditing level of a document or folder.

Restrictions:

1. You must have at least *USE authority to the document or folder or have *ALLOBJ or *AUDIT special authority to display the auditing level of the document or folder.
2. You must be enrolled in the system distribution directory unless you have *ALLOBJ or *AUDIT authority.

Required Parameters

DLO

Specifies the name of the document or folder whose auditing value is displayed.

***ALL:** The auditing levels for all DLOs in the specified folder are displayed.

***ROOT:** The auditing level for the *ROOT level folder is displayed. The *ROOT level folder contains the default auditing level for all new first-level folders.

***SYSOBJNAM:** The auditing level for the document or folder with the system object name specified on the SYSOBJNAM parameter is displayed.

document-library-object-name: Specify the document library object for which the auditing level is displayed.

Optional Parameters

FLR

Specifies the folder containing the document library object whose auditing level is displayed.

***NONE:** The document or folder is not contained in a folder.

***ANY:** The auditing levels of all DLOs on the system are displayed.

***ROOT:** The auditing level of all first-level folders is displayed.

folder-name: Specify the name of the folder that contains the document library object for which the auditing level is displayed.

SYSOBJNAM

Specifies the system object name. This parameter is valid only when DLO(*SYSOBJNAM) is specified. Ten characters must be specified.

OUTPUT

Specifies whether the output from this command is displayed, printed, or directed to a database file. More

information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

TYPE

Specifies whether the documents or folders contained in the folder specified on the FLR parameter are displayed. This parameter is valid when FLR(*folder-name*) is specified and is defaulted in all other cases. When DLO(*ALL) FLR(*ANY) is specified, TYPE(*ALL) is used. When DLO(*ALL) FLR(*ROOT) is specified, TYPE(*FLR) is used. When DLO(*ALL) FLR(*NONE) is specified, TYPE(*DOC) is used. When a single object is specified on the command, the TYPE parameter is ignored.

***FLR:** The folders contained in the specified folder are displayed.

***DOC:** The documents contained in the specified folder are displayed.

***ALL:** The documents and folders contained in the specified folder are displayed.

LEVEL

Specifies whether documents and folders at nested levels in the specified folder are displayed.

***CURRENT:** Only the documents and folders at the current level are included in the output.

***ALL:** The documents and folders at all levels are included in the output.

OUTFILE

Specifies the name of the database file to which the output of the command is directed. If the output file does not exist, the command creates a database file in the specified library. If the command creates the file, the text associated with the file is "Output file for DSPDLOAUD". The public authority is the same as the create authority specified for the library in which the file is created.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file to which the output from the command is directed.

OUTMBR

Specifies the name of the database file member to which the output from the command is directed. If the member already exists, the system uses the value specified on the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member using the name of the output file specified on the OUTFILE parameter. If an output file member name is specified but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member in the database file is used.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Displaying an Auditing Level

```
DSPDLOAUD  DLO(MYDOC)  FLR(MYFLR)
```

This command displays the auditing level of document MYDOC in folder MYFLR.

Example 2: Printing Auditing Levels

```
DSPDLOAUD  DLO(*ALL)  FLR(*ROOT)  OUTPUT(*PRINT)
```

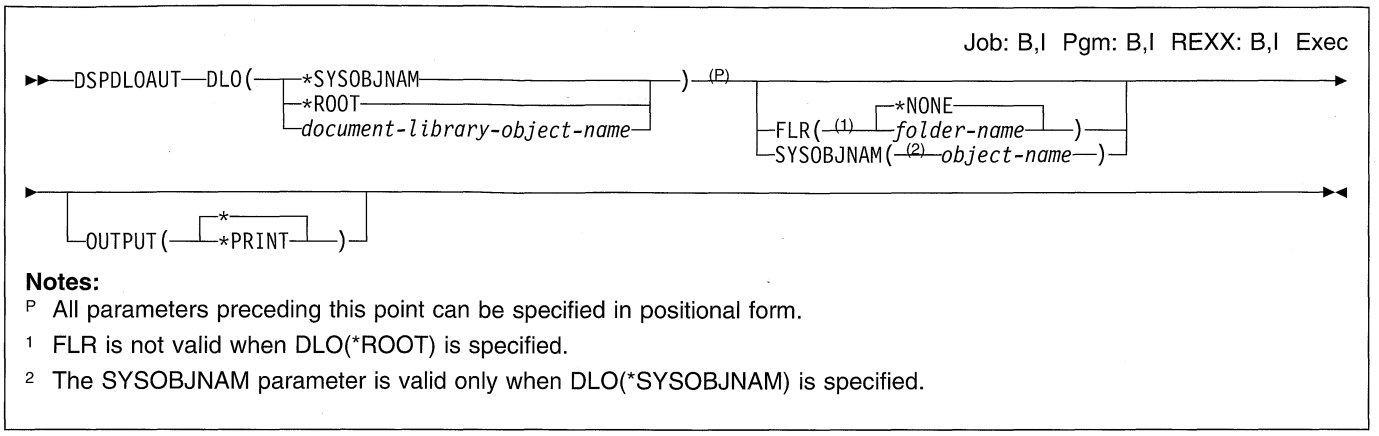
This command prints the auditing levels of all first-level folders on the system.

Example 3: Listing all Auditing Levels

```
DSPDLOAUD  DLO(*ALL)  FLR(*ANY)
  OUTPUT(*OUTFILE)  OUTFILE(MYLIB/MYFILE)
```

This command lists the auditing levels of all DLOs on the system in the output file MYFILE in library MYLIB.

DSPDLOAUT (Display Document Library Object Authority) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 FLR is not valid when DLO(*ROOT) is specified.

2 The SYSOBJNAM parameter is valid only when DLO(*SYSOBJNAM) is specified.

Purpose

The Display Document Library Object Authority (DSPDLOAUT) command shows the list of authorized users of an object and their assigned authorities.

The following information is displayed for the specified document or folder:

- The name of the document library object
- The name of the authorization list securing the document or folder (if there is one)
- Personal status of the document or folder
- Specific user authority for the document or folder
- The authority given to the users with no specific authority for the document or folder
- If a function key is pressed, access codes attached to the document library object are displayed.

Restrictions:

1. A user must have *USE authority to the document or folder to display that user's authority information about the document or folder.
2. If the user has *ALL or *ALLOBJ authority to the document or library object, the user sees all authority information.
3. The user must have *ALLOBJ special authority to display the *ROOT folder public authority.

Required Parameters

DLO

Specifies the name of the document or folder to be displayed.

***SYSOBJNAM:** The system object names specified in the SYSOBJNAM parameter are displayed.

***ROOT:** The public authority value of the *ROOT folder is displayed.

document-library-object-name: Specify the user-assigned name of the document or folder to be displayed. Up to 12 characters can be specified.

Optional Parameters

FLR

Specifies the name of the folder that contains the document.

***NONE:** A folder name is not specified. If DLO(name) is specified and the object is located in a folder, FLR(*NONE) cannot be specified.

folder-name: Specify the name of the folder that contains the object. The name can consist of a series of folder names if the folder that contains the object is located in another folder. Up to 63 characters can be specified.

SYSOBJNAM

Specifies the system object name. A full 10 characters must be specified. This parameter is valid only when DLO(*SYSOBJNAM) is specified.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

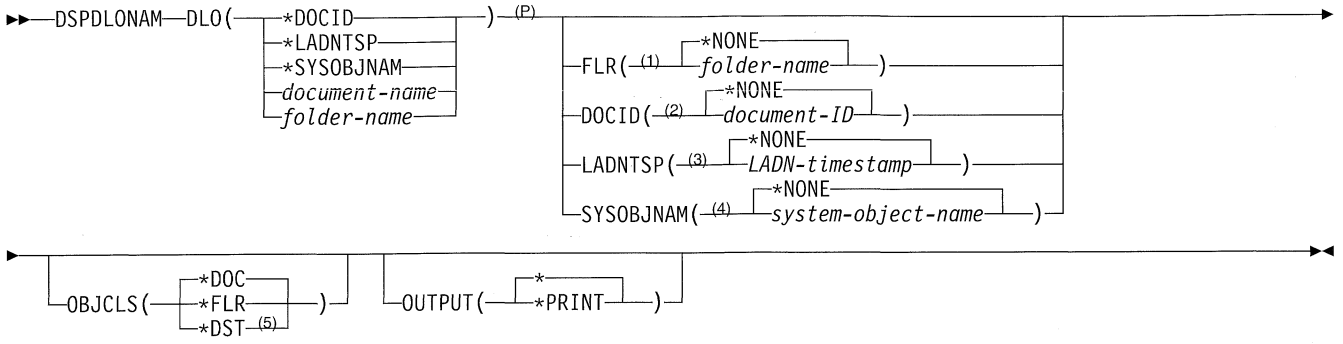
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPDLOAUT  DLO(DOCA)  FLR(MYFLR)  OUTPUT(*PRINT)
```

This command prints for DOCA in folder MYFLR a list of all authorized users and their authorities if the user of this command has *ALL authority to DOCA in folder MYFLR. This command prints only the user's authorities if the user of this command has *USE authority. Access codes are also shown.

DSPDLONAM (Display Document Library Object Name) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 FLR(*NONE) must be specified if *DOCID, *LADNTSP, or *SYSOBJNAM is specified on the DLO parameter.

2 Valid only if DLO(*DOCID) is specified.

3 Valid only if DLO(*LADNTSP) is specified.

4 Valid only if DLO(*SYSOBJNAM) is specified.

5 Valid only if DLO(*LADNTSP) or DLO(*SYSOBJNAM) is specified.

Purpose

The Display Document Library Object Name (DSPDLONAM) command shows a list of names for a folder, filed document, or distribution document.

Restrictions:

1. A user must have *USE authority to the filed document or folder to display the various forms of the name.
2. Users must have *ALLOBJ authority to display the various forms of the name for a distribution document.

Required Parameters

DLO

Specifies the name of the document or folder for which the various forms of the name are displayed or printed.

***DOCID:** The library-assigned name specified on the DOCID parameter is used to identify the document or folder.

***LADNTSP:** The timestamp from the library-assigned document name (LADN) specified on the LADNTSP parameter is used to identify the document or folder.

***SYSOBJNAM:** The system object name specified on the SYSOBJNAM parameter is used to identify the document or folder.

document-name: Specify the user-assigned name of the document.

folder-name: Specify the user-assigned name of the folder.

Optional Parameters

FLR

Specifies the name of the folder where the object specified on the DLO parameter is located.

***NONE:** The name of the folder that contains the object is not specified, the object is not contained in a folder, or the object is specified using the DOCID, LADNTSP, or SYSOBJNAM parameter.

folder-name: Specify the name of the folder that contains the object.

Note: FLR(*NONE) must be specified if the object is a first-level folder.

DOCID

Specifies the library-assigned name of the document or folder.

***NONE:** The object is not identified using its library-assigned name.

document-ID: Specify the library-assigned name of the document or folder object. Library-assigned names are 16 to 24 characters in length in the format YYYYMMDDHHMNSSHSSNSNSNSN, where:

YYYY = year
MM = month
DD = day
HH = hour
MN = minute
SS = second
HS = hundredths of a second

SNSNSNSN = system name (from 1 through 8 characters in length or omitted)

LADNTSP
Specifies the LADN timestamp of the document or folder.

***NONE:** The object is not identified using its LADN timestamp.

LADN-timestamp: Specify the LADN timestamp of the document or folder. The LADN timestamp is 16 hexadecimal characters in length in the format YYYYMMDDHHMNSSHS, where:

YYYY = year
MM = month
DD = day
HH = hour
MN = minute
SS = second
HS = hundredths of a second

SYSOBJNAM
Specifies the system object name.

***NONE:** The object is not identified using its system object name.

system-object-name: Specify the 10-character system object name of the document or folder.

OBJCLS
Specifies the class of the object to locate.

***DOC:** The object is a filed document.

***FLR:** The object is a folder.

***DST:** The object is a distribution document.

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

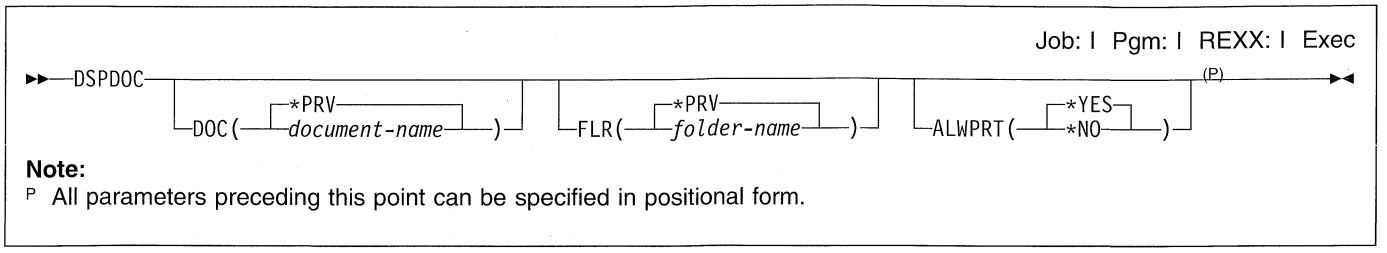
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPDLONAM   DLO(MYDOC)  FLR(MYFLR)  OBJCLS(*DOC)
```

This command finds the document MYDOC in folder MYFLR and displays all forms of the document's name. If the job is running in batch mode, the information is printed and not displayed.

DSPDOC (Display Document) Command



Purpose

The Display Document (DSPDOC) command allows the user to display a document using the word processing function of OfficeVision/400. More information on displaying documents is in the *Using OfficeVision/400* Word Processing*.

Optional Parameters

DOC

Specifies the name of the document to be displayed.

***PRV:** The name used in the previous session is used.

document-name: Specify the name of the document that is displayed.

FLR

Specifies the name of the folder that contains the document.

***PRV:** The name used in the previous session is used.

folder-name: Specify the name of the folder that contains the document to be displayed.

ALWPRT

Specifies whether the user is able to print a document while viewing it.

***YES:** The user can print a document while viewing it.

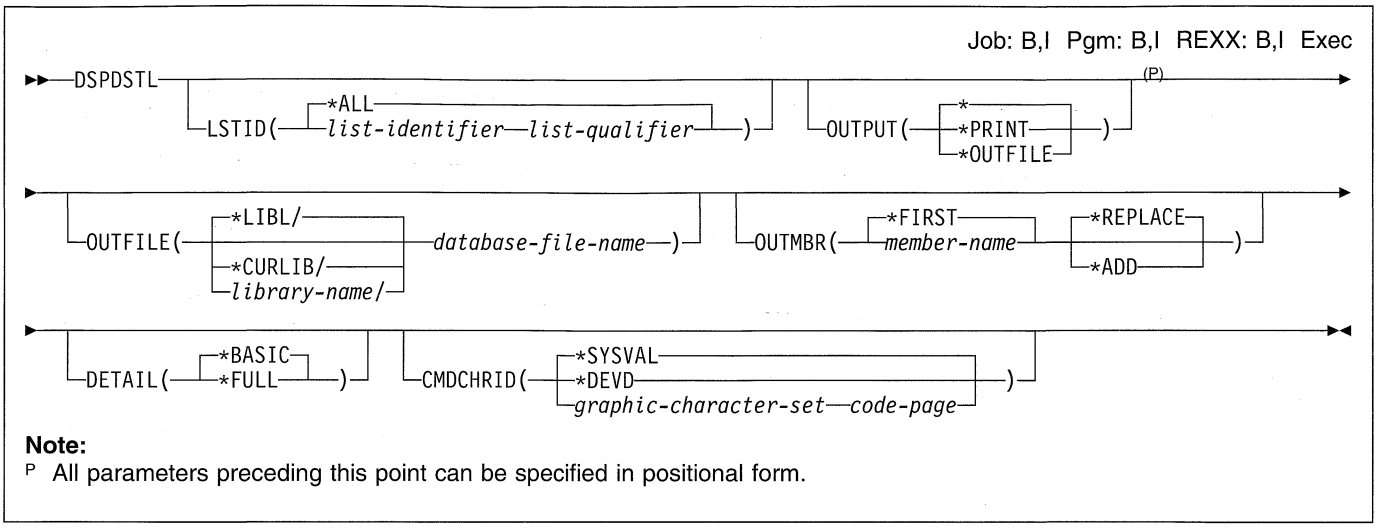
***NO:** The user cannot print a document while viewing it.

Example

```
DSPDOC   DOC(MYDOC)   FLR(MYFLR)
```

This command displays the document MYDOC in folder MYFLR.

DSPDSTL (Display Distribution List) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Distribution List (DSPDSTL) command is used to display, print, or create a database output file for distribution lists contained in the distribution directory as follows:

- For displayed output, a list of distribution lists is displayed. When a specific list identification (ID) has been specified in the LSTID parameter, the displayed list contains only the specified distribution list.
- For printed or database file output, if the LSTID parameter specifies *ALL, the DETAIL parameter determines whether a list of distribution lists or a list of entries in each distribution list is the output. When the LSTID parameter specifies a specific list, a list of entries in the list is output.

No distribution lists can be created or deleted, nor can updates be made to existing lists from this command. These functions are provided interactively, with display support by the Work with Distribution List (WRKDSTL) command, and with the Create Distribution List (CRTDSTL), Delete Distribution List (DLTDSTL), Add Distribution List (ADDDSTLE), and Remove Distribution List (RMVDSTLE) commands.

Optional Parameters

LSTID

Specifies which distribution lists to display, print, or output to a database file.

***ALL:** All distribution lists in the system distribution directory are included in the output. The entries are provided in alphabetical order by list ID. If output is to a printed list or outfile, the DETAIL parameter determines whether a list of distribution lists or the entries in each list is the output.

Element 1: List Identifier

list-identifier: Specify the list identifier (ID) of a distribution list.

Element 2: List Qualifier

list-qualifier: Specify the list qualifier of the distribution list.

Note: The distribution list identifier/list qualifier is in two parts, separated by at least one space. If any lowercase characters are specified, the system changes them to uppercase.

The naming rules for the two-part list ID are identical to the rules for the user ID and address. A complete description of these rules is in the *Distribution Services Network Guide*.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. If the command is run as part of a batch job, the output is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the name of the database file to which the output of the display is directed. If the database file does not exist, this command creates it in the specified library. If no library is specified, the database file is created in the user default library specified in the user profile. If no default library was specified, the database file is created in the QGPL library. If the file is created,

the text is "OUTFILE for DSPDSTL" and the public authority is *EXCLUDE.

More information on defining the format of database output files is in the *Office Services Concepts and Programmer's Guide*.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display. If the database file is qualified with *LIBL but the system cannot find the file, one is created in the user's default library, if it is specified. If the default library was not specified, the file is created in the QGPL library.

OUTMBR

Specifies the name of the database file member to which the output is directed. If a member already exists, the system uses the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member with the name of the output file specified on the OUTFILE parameter. If an output file member name is specified, but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

DETAIL

Specifies how much detail is printed or written to the database file. This parameter is not used when the output is displayed (OUTPUT(*)) or when a specific list ID is specified in the LSTID parameter.

Anytime LSTID(*ALL) is specified, the DETAIL parameter is used to determine whether a list of all distribution lists or a list of all entries in every distribution list is the output.

Model output file QAOSDSTO in library QSYS is used.

***BASIC:** The output is a list of all distribution lists contained in the directory.

***FULL:** The output contains all of the entries in every distribution list contained in the directory.

CMDCHRID

Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameters from the display device description where this command was entered. This option is supported only when the command is entered from an interactive job. If this option is specified in a batch job, an error message is returned.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the command parameters. Valid values range from 1 through 9999.

Element 2: Code Page

code-page: Specify the code page. Valid values range from 1 through 9999.

Examples

Example 1: Displaying a List

```
DSPDSTL  LSTID(*ALL)  OUTPUT(*)
```

This command displays a list of all distribution lists in the directory. Specifying DSPDSTL without parameters would result in the same action.

Example 2: Printing a List

```
DSPDSTL  LSTID(DEPT48K DISTLIST)  OUTPUT(*PRINT)
```

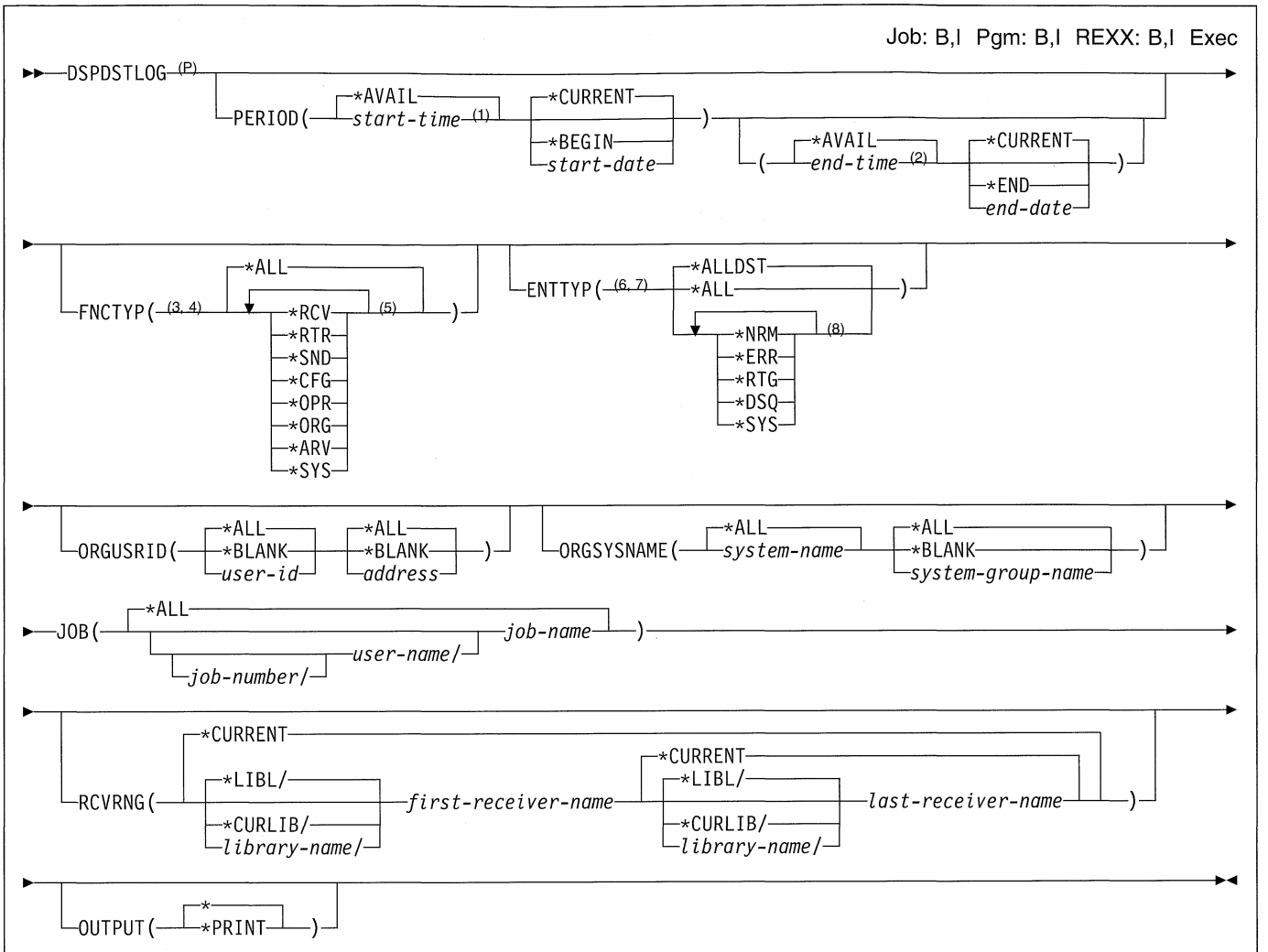
This command prints a list of all entries in the distribution list DEPT48K DISTLIST. The detail parameter is not used when printing for a specific list.

Example 3: Directing Output to a Database File

```
DSPDSTL  OUTPUT(*OUTFILE)  OUTFILE(ALLLISTS)
  OUTMBR(*FIRST *REPLACE)  DETAIL(*BASIC)
```

This command outputs one record for each distribution list contained in the directory to the database file ALLLISTS. If this file is not found in the library list, it is created in the QGPL library, since no library is specified.

DSPDSTLOG (Display Distribution Log) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 If *BEGIN is specified as the start-date, then start-time is ignored.

2 If *END is specified as the end-date, then end-time is ignored.

3 If FNCTYP(*CFG) is specified, then ENTTYP(*ALLDST), or (*ALL), or (*RTG), or (*DSQ) must be specified.

4 If FNCTYP(*SYS) is specified, then ENTTYP(*ALL) or (*SYS) must be specified.

5 A maximum of eight repetitions.

6 If ENTTYP(*RTG) or (*DSQ) is specified, then FNCTYP(*ALL) or FNCTYP(*CFG) must be specified.

7 If ENTTYP(*SYS) is specified, then FNCTYP(*ALL) or FNCTYP(*SYS) must be specified.

8 A maximum of 5 repetitions

Purpose

The Display Distribution Log (DSPDSTLOG) command provides a convenient interface to the Systems Network Architecture distribution services (SNADS) log. The SNADS log (the QSNADS journal) contains entries that track SNADS operations that have been performed on the system. These operations include sending, receiving, and routing distributions and configuration changes.

This command provides a wide range of selection criteria which allows easy access to the desired information. The selection criteria includes time period, function type (such as router and sender/receiver), entry type (such as normal, error, and configuration), and job name. Any number of selection parameters can be entered, and the resulting output is cumulative, based on all parameters entered.

Both displayed and printed output can be produced in conjunction with the input parameter specified. If the user requests output that can be shown, a summary of all entries matching the selection criteria are shown. The summary display includes basic information, such as function type, entry type, date-time of logging, job name, and originating user ID. From the summary display, any of the entries can be chosen to view the details of that log entry. The detail display gives more specific information on that particular log entry. If the user requests printed output, a detail printout for each log entry in the selection is produced.

By specifying the appropriate selection criteria, the user can determine the following types of information:

- Configuration changes that were made, when and by whom they were made for all tables or for specific tables
- Configuration changes correlated with error entries or distributions routed and/or sent
- List error entries for all functions or for specific functions (such as routing errors)
- All distributions received, routed, or sent during a selected time interval
- Activity for a specific sender or receiver job

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. This command cannot be used to show information on the 12 by 80 size work station (*DS2) or on the console (*DS1).

Optional Parameters

PERIOD

Specifies the period of time for which the logged data is shown. The following values can be coded in this parameter, which contains two lists of two elements each.

Element 1: Starting Time

One of the following is used to specify the starting time at which or after which the data must have been logged. Entries logged before the specified time and date are not shown.

***AVAIL:** The logged data that is available for the specified starting *date* is shown.

start-time: Specify the starting time for the specified starting date that indicates the logged data that is shown. The time is specified in 24-hour format with or without a time separator as follows:

- With a time separator, specify a string of 5 or 8 digits where the time separator separates the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command fails.
- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where **hh** = hours, **mm** = minutes, and **ss** = seconds. Valid values for **hh** range from 00 through 23. Valid values for **mm** and **ss** range from 00 through 59.

Element 2: Starting Date

One of the following is used to specify the starting date on which or after which the data must have been logged. Entries logged before the specified date are not shown.

***CURRENT:** The logged data for the current day and between the specified starting and ending times (if specified) is shown.

***BEGIN:** The logged data from the beginning of the log is shown. If *BEGIN is specified, then any time value other than *AVAIL for start-time is ignored.

start-date: Specify the date displayed. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

Element 3: Ending Time

One of the following is used to specify the ending time before which the data must have been logged.

***AVAIL:** The logged data that is available for the specified ending date is shown.

end-time: Specify the ending time for the specified ending date that determines the logged data that is shown.

Element 4: Ending Date

One of the following is used to specify the ending date before which or on which the data must have been logged.

***CURRENT:** The logged data for the current day and between the specified starting and ending times (if specified) is shown.

***END:** The last day on which data was logged is shown. If PERIOD(*END) is specified, a time value other than *AVAIL for end-time is ignored.

end-date: Specify the ending date for which logged data is shown. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

FNCTYP

Specifies the SNADS function that was being performed when SNADS log entries were made. If the default value *ALL is not specified, a maximum of eight functions may be specified.

***ALL:** All SNADS functions that made log entries are used.

***RCV:** The SNADS receiver function is used.

***RTR:** The SNADS router function is used.

***SND:** The SNADS sender function is used.

***CFG:** The SNADS configuration function is used. If FNCTYP(*CFG) is specified, the ORGUSRID and ORGSYSNAME parameters cannot be specified.

***OPR:** The SNADS operation function is used.

***ORG:** The SNADS originator function is used.

***ARV:** The SNADS arrival function is used.

***SYS:** The SNADS system function is used.

ENTTYP

Specifies the type of log entries shown. If *ALL or *ALLDST are not specified, a maximum of five entry types can be specified.

***ALLDST:** All log entry types are shown except *SYS.

***ALL:** All log entry types are shown.

***NRM:** Normal (log entries without errors) log entries are shown.

***ERR:** Error log entries are shown.

***RTG:** Routing table log entries or secondary node ID log entries are shown.

***DSQ:** Distribution queue log entries are shown.

***SYS:** The QSNADS journal entries are shown.

ORGUSRID

Specifies the originating user ID and address of logged distributions. This parameter allows the user to display or print only those logged entries made because of distributions originated by a user with the specified ID or address.

The user ID and address are translated to the character set and code page '697 500' using the job's coded character set identifier (CCSID) before being used to select log entries.

Element 1: Originating User ID

***ALL:** All log entries are shown regardless of the originating user ID.

***BLANK:** Only log entries with an originating user ID of all blanks (as in the case of SNADS status distributions) are shown.

originating-user-ID: Specify the originating user ID.

Element 2: Originating User Address

***ALL:** All log entries are shown regardless of the originating address.

***BLANK:** Only log entries with an originating address of all blanks (as in the case of SNADS status distributions) are shown.

originating-user-address: Specify the originating address.

ORGSYSNAME

Specifies the name of the originating system and group of logged distributions. This parameter allows the user to display or print only those logged entries made because of distributions that originated from the specified system or group.

AS/400, System/38, and System/36 systems do not specify a system group. Specify *BLANK to show the originating system group name of distributions from these systems. This may not be true of distributions originating from other SNADS implementations.

The name and group are translated to the character set and code page '697 500' using the job's coded character set identifier (CCSID) before being used to select log entries.

Element 1: Originating System Name

***ALL:** All log entries are shown regardless of the originating system.

originating-system-name: Specify the name of the originating system.

Element 2: Originating System Group Name

***ALL:** All log entries are shown regardless of the originating system group name.

***BLANK:** Only log entries with an originating address of all blanks are shown.

originating-system-group-name: Specify the originating system group name.

JOB

Specifies the name of the SNADS job that made the log entry. This parameter allows the user to display or print only those entries logged by the specified job. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

A job identifier is a special value or a qualified name with up to three elements. For example:

```
*ALL
job-name
user-name/job-name
job-number/user-name/job-name
```

***ALL:** Entries are shown regardless of the job that logged them.

job-name: Specify the name of the job being disconnected. If no job qualifier is given, all jobs currently in the system are searched for the name of the job. If duplicates of the specified name are found, a qualified job name must be specified.

user-name: Specify the name of the user who owns the job being disconnected. Specifying the user as a qualifier is necessary only if a duplicate job name exists for different users. If a duplicate job name exists for the same user, the job must be qualified with the job number.

job-number: Specify the number of the job being disconnected. Specifying the job number as a qualifier is only necessary if a duplicate job name exists for the same user.

RCVRNG

Specifies the qualified names of the journal receivers that contain the SNADS logs. This parameter displays entries from SNADS logs that are kept in journal receivers that are no longer active.

Note: The maximum number of receivers that are used in a range of receivers is 256. If more than 256 receivers are specified, an error message is sent and no changes are applied.

***CURRENT:** Only the currently attached receiver is used in applying the journal entries. Application of journal entries continues for all journal receivers in the chain, beginning with the receiver specified by the first parameter value through the currently attached journal receiver.

Element 1: First Receiver Name

The name of the journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

first-receiver-name: Specify the name of the journal receiver used as the first (oldest) receiver.

Element 2: Last Receiver Name

***CURRENT:** The currently attached receiver is the last (newest) receiver used in applying the journal entries.

The name of the journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

last-receiver-name: Specify the qualified name of the journal receiver used as the last (newest) receiver with journal entries applied. If the end of the receiver chain is reached before finding a receiver with this name, the operation is not done and an escape message is sent.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Example

Example 1: Printing a Distribution Log

```
DSPDSTLOG  OUTPUT(*PRINT)
```

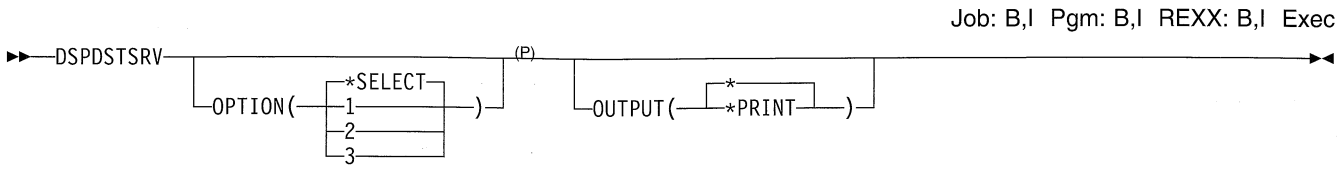
This command directs the distribution log information to the job's output spooling queue to be printed. If OUTPUT(*) is specified, and the command was entered from a work station, the information about the distribution log is shown at the work station.

Example 2: Getting a SNADS Distribution Log

```
DSPDSTLOG  ENTTYPE(*SYS)
```

This command directs the SNADS distribution log entries to be shown on the work station display for an interactive job, or printed with the job's spooled output for a batch job.

DSPDSTSRV (Display Distribution Services) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Distribution Services (DSPDSTSRV) command shows or prints the distribution queues table, routing table, or secondary system name table defined for the local system. A detailed description of the Systems Network Architecture service (SNADS) network is in the *Distribution Services Network Guide*.

Note: This command does not allow changes to the SNADS network. Changes to the network can be made using the Configure Distribution Services (CFGDSTSRV) command.

Restriction: Messages that report errors about system names or distribution queues may show or print different characters than the user entered because of internal system transformations.

The internal value for a system name or distribution queue may differ from the characters shown by the DSPDSTSRV command, depending on the language used for the work station.

Optional Parameters

OPTION

Specifies an option from the distribution services menu display that bypasses the initial menu and goes directly to the indicated table. The distribution queues table, routing table, or secondary system name table can be specified without showing the distribution services menu display. If the output is printed, the information for options 1, 2, and 3 is printed.

***SELECT:** The option is selected from the distribution services menu display.

1: The distribution queues table, which describes the devices and send times for communicating with adjacent systems, is shown.

2: The routing table entries, which describe the destination systems to which the system can route, are shown.

3: The secondary system name entries, which list all names by which the local system is known, are shown.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

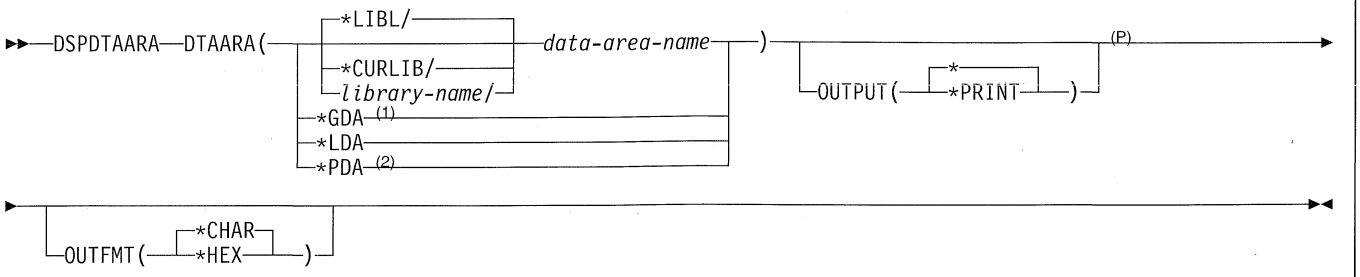
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPDSTSRV  OUTPUT(*PRINT)
```

This command prints the current SNADS configuration status.

DSPDTAARA (Display Data Area) Command



Notes:
1 This value is valid only if this is a group job.
2 This value is valid only if this is a prestart job.
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Data Area (DSPDTAARA) command displays the attributes and value of the specified data area. The following attributes are displayed: the type and length of the data area, the library where the data area is located (there is no library associated with a local data area, the group data area, or the program initialization parameter data area), and the text describing the data area.

If the job is a group job, the data area specified may be the group data area (*GDA). This data area is automatically associated with the group and it cannot be read from jobs outside the group. The length of this character data area is 512 bytes. More information about group jobs is in the *Work Management Guide*.

The local data area (*LDA) is a character data area 1024 bytes long. It is automatically associated with the job and cannot be accessed from another job.

If the job is a prestart job, the data area specified may contain program initialization parameter data (*PDA). This data area is automatically associated with the prestart job and is inaccessible from other jobs. The length of this character data area is 2000 bytes. More information about prestart jobs is in the *Work Management Guide*.

Restriction: To use this command, the user must have object operational authority for the data area and read authority for the library. No specific authority is required for the local data area or the group data area.

Required Parameters

DTAARA

Specifies the name and library of the data area whose attributes and value are to be displayed.

The name of the data area can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

data-area-name: Specify the name of the data area to be displayed.

***GDA:** The group data area is displayed if this is a group job.

***LDA:** The local data area is displayed.

***PDA:** The program initialization parameter data area is displayed if this is a prestart job.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

OUTFMT

Specifies the format in which the objects are shown.

***CHAR:** The output is displayed in character format.

***HEX:** The output is displayed in both hexadecimal format and character format, unless the display is shown on the system console, in which case only the

hexadecimal format is displayed. This value is valid only for character data areas.

Examples

Example 1: Displaying Output in Hexadecimal and Character Format

```
DSPDTAARA  DTAARA(HEXDATA)  OUTPUT(*)  OUTFMT(*HEX)
```

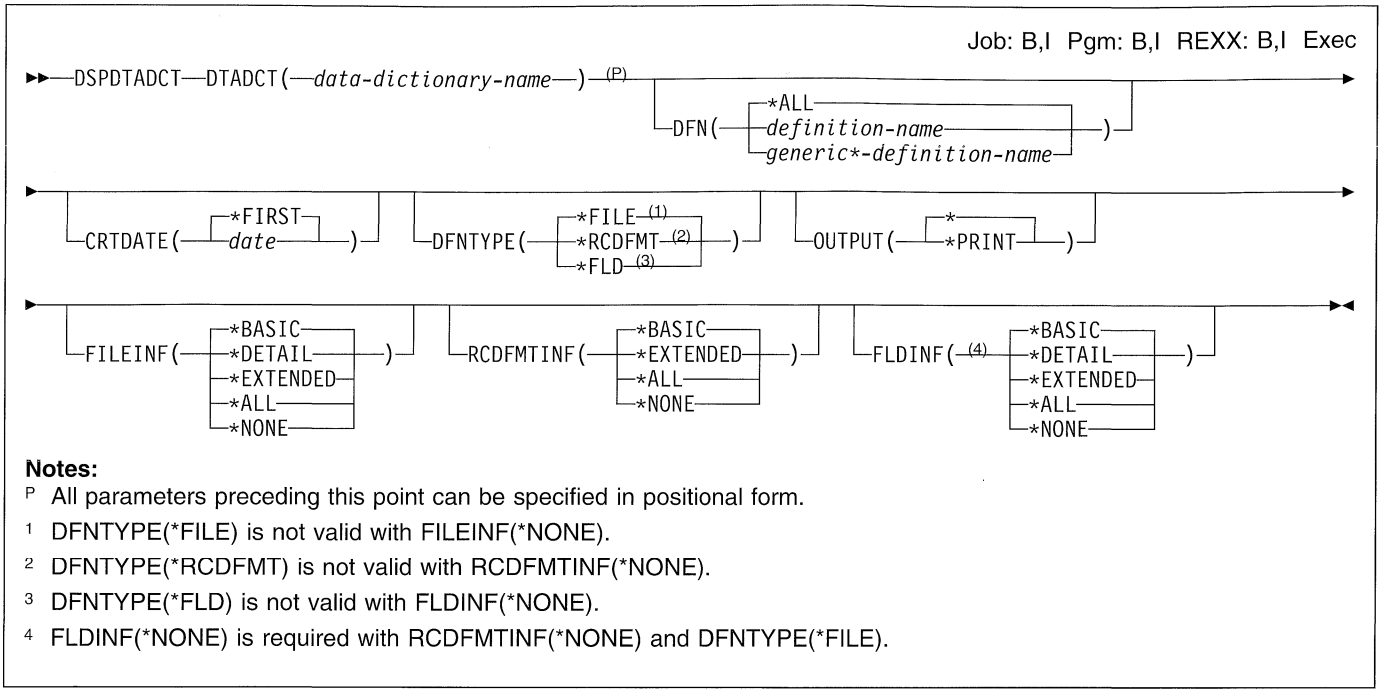
The value and attributes of data area HEXDATA are displayed if the user has proper authority. Both character and hexadecimal representations are shown.

Example 2: Displaying Output in Character Format

```
DSPDTAARA  DTAARA(TIME)  OUTPUT(*)
```

The value and attributes of the data area TIME are displayed if the user has the proper authority. The library list is used to find the data area.

DSPDTADCT (Display Data Dictionary) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 DFNTYPE(*FILE) is not valid with FILEINF(*NONE).

2 DFNTYPE(*RCDFMT) is not valid with RCDFMTINF(*NONE).

3 DFNTYPE(*FLD) is not valid with FLDINF(*NONE).

4 FLDINF(*NONE) is required with RCDFMTINF(*NONE) and DFNTYPE(*FILE).

Purpose

The Display Data Dictionary (DSPDTADCT) command displays or prints the contents of a field definition, record format definition, or file definition in a data dictionary.

Required Parameters

DTADCT

Specifies the name of the dictionary that contains the definition to be displayed or printed.

Optional Parameters

DFN

Specifies the name of the definition to be displayed or printed.

***ALL:** All definitions of the type selected by DFNTYPE in the dictionary are displayed or printed.

definition-name: Specify the name of the definition to be displayed or printed.

generic-definition-name:* Specify the generic name of the definition. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

CRTDATE

Specifies the creation date of the object.

***FIRST:** The first definition created with this DFN name is printed or displayed. If several definitions with the same name were created on the same date and a specific DFN name is specified, only the first one created on that date is displayed or printed. If a generic name is specified, the CRTDATE is ignored and all definitions equal to the DFN parameter are printed or displayed.

date: Specify the creation date of the definition to display or print. The date must be specified in the format used by the job.

DFNTYPE

Specifies the type of definition to display or print.

***FILE:** The file definitions are displayed or printed.

***RCDFMT:** The record format definitions are displayed or printed.

***FLD:** The field definitions are displayed or printed.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

FILEINF

Specifies the level of detail used in displaying or printing the file definition information.

***BASIC:** The following basic information is displayed or printed: the definition name, the definition type, the dictionary in which the definition is located, the date the definition was created, the user ID of the person who created it, the date it was last changed, the user ID of the person who made the last change, the definition's text, the number of format definitions used by this file definition, and the type of file.

***DETAIL:** The basic file definition information, plus any key field information, is displayed or printed.

***EXTENDED:** The detailed file definition information, plus any long comment about the file definition, is displayed or printed.

***ALL:** The extended file definition information, plus a list of files that use this definition, is displayed or printed.

***NONE:** No file definition information is printed or displayed.

RCDFMTINF

Specifies the level of detail used in printing or displaying the contents of the record format information.

***BASIC:** The following basic information is displayed or printed: the definition name, the definition type, the dictionary where the definition is located, the date the definition was created, the user ID of the person who created it, the date it was last changed, the user ID of the person who made the last change, the definition's text, the number of field definitions used by this format definition, and the record length.

***EXTENDED:** The basic file definition information, plus any long comment about the format definition, is displayed or printed.

***ALL:** The extended file definition information, any record ID codes, and a list of file definitions and files that use this format definition are displayed or printed.

***NONE:** No record format information is printed or displayed.

FLDINF

Specifies the level of detail used in printing or displaying the field definition information.

***BASIC:** The following basic field definition information is displayed or printed: the definition name, the definition type, the dictionary where the definition is located, the definition's text, the data type, field length, buffer length, and column headings. If record format information is also requested, the basic information also includes the buffer position and field usage.

***DETAIL:** The basic field definition information plus the date the definition was created, the user ID of the person who created it, the date it was last changed, the user ID of the person who made the last change, the keyboard shift, the alias name, and editing information is displayed or printed.

***EXTENDED:** The detailed field definition information, plus any long comment about the field definition, is displayed or printed.

***ALL:** All extended information, plus a list of definitions and files that use this definition, is displayed or printed.

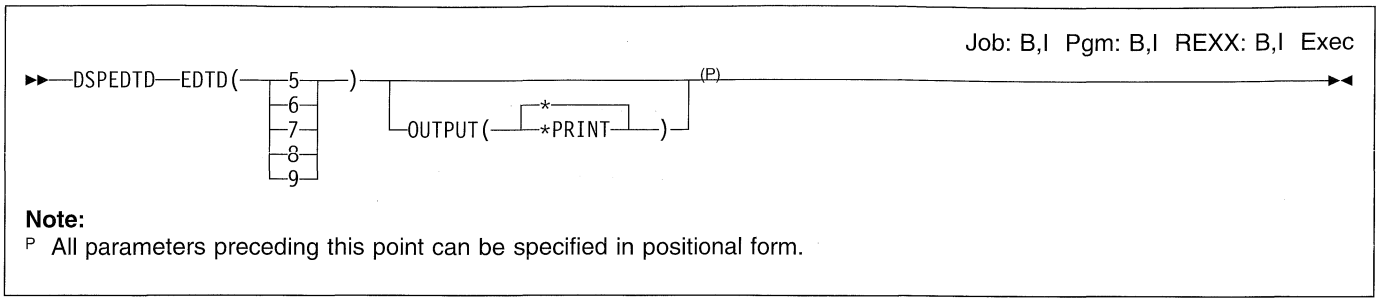
***NONE:** No field information is displayed or printed.

Example

```
DSPDTADCT   DTADCT(MINE)  DFN(*ALL)  DFNTYPE(*FILE)
  OUTPUT(*PRINT)  FILEINF(*BASIC)  RCDFMTINF(*BASIC)
  FLDINF(*EXTENDED)
```

This command prints all file definitions in the data dictionary MINE. Basic information for the file definitions is printed. Basic information for the format definitions in each file definition, and the extended information for the field definitions in each RECORD format definition in the file definitions are printed.

DSPEDTD (Display Edit Description) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Edit Description (DSPEDTD) command shows information about the specified user-defined edit description.

Required Parameters

EDTD
Specifies the single-digit code (5, 6, 7, 8, or 9) that identifies the user-defined edit description to be displayed.

Optional Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

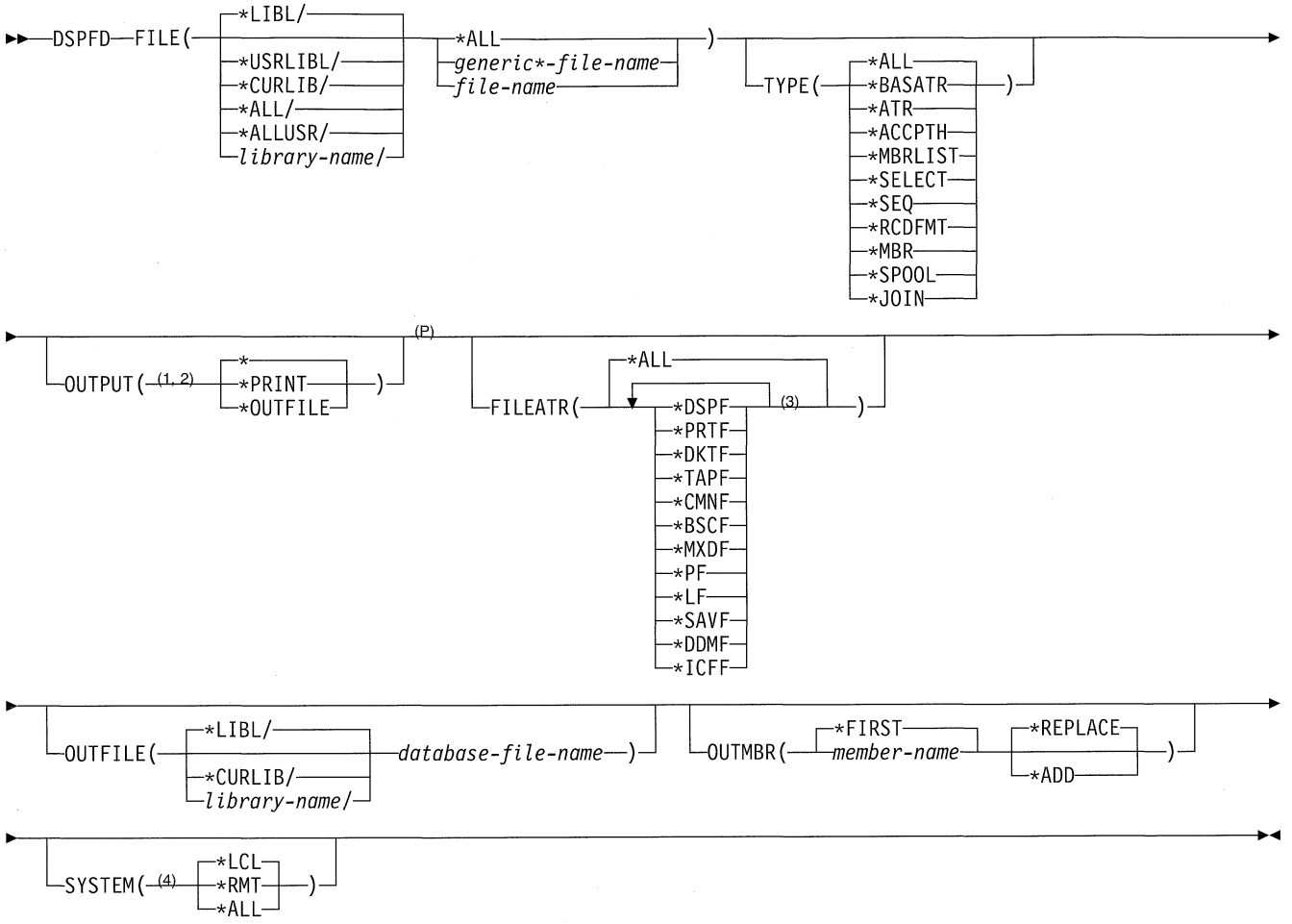
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPEDTD    EDTD(6)
```

This command shows the user-defined edit description 6 on either a printer or a display.

DSPFD (Display File Description) Command



Notes:

1 If OUTPUT(*OUTFILE) is specified, a file name must be specified for the OUTFILE parameter. If TYPE(*BASATR) is specified, OUTPUT(*OUTFILE) must be specified.

2 If OUTPUT(*OUTFILE) is specified, a type other than *ALL must be specified on the TYPE parameter. (Only one type may be specified.) If TYPE(*ATR) is specified and OUTPUT(*OUTFILE) is specified, FILEATR(*ALL) or FILEATR(*MXDF) cannot be specified.

P All parameters preceding this point can be specified in positional form.

3 A maximum of 12 repetitions

4 If SYSTEM(*RMT) is specified for displayed or listed output, the only valid file attributes (FILEATR) are *PF, *LF, or *ALL. If SYSTEM(*RMT) or SYSTEM(*ALL) is specified for outfile processing, the FILEATR parameter must be either *PF or *LF.

Purpose

The Display File Description (DSPFD) command shows one or more types of information retrieved from the file descriptions of one or more database and/or device files. The information is provided for each file that has the specified name and that is found in the libraries named in the specified library qualifier to which the user has access. The information can be shown, printed, or directed to a database output file (OUTFILE). More information on files used by the DSPFD command is in the *Database Guide*.

Restrictions:

1. Before the specified files can be shown, the user must have object operational authority to those files.
2. Of the libraries identified by the library qualifier, only libraries to which the user has read authority are searched for the specified files.

3. If TYPE(*ALL), TYPE(*MBR), or TYPE(*MBRLIST) is specified and if the file is a physical file, the user needs at least one data authority (read, write, update, or delete) to the file to receive information about the members.

Required Parameters

FILE

Specifies the qualified name of the file or the qualified generic name of several files that have their descriptions shown, printed, or directed to a database file. This parameter can specify that all files in the specified library or libraries (*LIBL/*ALL for example), or that all files in all libraries (*ALL/*ALL), have their descriptions provided.

Note: The user must have authority to both the specified libraries and the specified files before the file descriptions are provided.

The name of the file description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** The descriptions of all files in the library or libraries with the specified library qualifier are shown.

generic-file-name:* Specify the generic name of the file or group of files that have their descriptions shown. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

file-name: Specify the full name of the file whose description is shown.

Optional Parameters

TYPE

Specifies the types of file information that are provided. TYPE(*BASATR) is valid only if a file name is specified for the OUTFILE parameter. If a file name is specified for the OUTFILE parameter, TYPE(*ALL) cannot be specified. Only one TYPE can be specified if a file name is specified for the OUTFILE parameter. See the table in the FILEATR parameter description.

***ALL:** All the types of information that apply to the specified file are provided. *ALL cannot be specified or defaulted if a file name is specified for the OUTFILE parameter.

***BASATR:** The basic attribute information common to all files is provided. TYPE(*BASATR) is valid only if OUTPUT(*OUTFILE) is also specified. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDBAS from system file QAFDBASI; if the file does exist, it must have this format.

***ATR:** The attribute information for the specified file is provided. For database files, the attributes include the access path type and the maximum number of members. If the file specified for the OUTFILE parameter does not exist, the file is created with a format unique to the requested file attribute. If the file does exist, it must have this format. Refer to the following FILEATR parameter descriptions for the format used for the OUTFILE.

***ACCPTH:** For physical and logical files only, the access paths of the specified file are provided. For keyed access paths, the composite key description is also provided. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDACP from system file QAFDACCP; if the file does exist, it must have this format.

***MBRLIST:** For physical and logical files only, an alphabetical list that contains the names of all members in the specified file and a brief description of each member is provided. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDML from system file QAFDMBRL; if the file does exist, it must have this format.

***SELECT:** For logical files only, the select/omit attribute is provided. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDDSO from system file QAFDSELO; if the file does exist, it must have this format.

***SEQ:** For physical and logical files only, the collating sequence is provided. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDSEQ from system file QAFDCSEQ; if the file does exist, it must have this format.

***RCDFMT:** The record format names and record format level information for the file, including the record format name(s) and data association information for referenced files are provided. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDFMT from system file QAFDRFMT; if the file does exist, it must have this format.

***MBR:** For physical and logical files only, information about the file members (names, creation dates, sizes, types, and other attributes) in the specified file is provided. Members are in the order in which they were created. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDMBR from system file QAFDMBR; if the file does exist, it must have this format.

***SPOOL:** For device files only, the spooling attributes of the specified diskette or printer file are provided. The spooling attributes include the number of output copies produced, the output priority, and the maximum number of records in the file. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDSPL from system file QAFDSPOL; if the file does exist, it must have this format.

***JOIN:** For join logical files only, the join from-file, the join to-file, and the fields that are involved in the join are provided. If the file specified for the OUTFILE parameter does not exist, the file is created with the format QWHFDJN from system file QAFDJOIN; if the file does exist, it must have this format.

OUTPUT

Specifies whether the output from the command is provided at the requesting work station, printed, or placed in a database output file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Note: If OUTPUT(*OUTFILE) is specified, a type other than *ALL must be specified for the TYPE parameter (only one type may be specified). If TYPE(*ATR) is specified while OUTPUT(*OUTFILE) is specified, FILEATR(*ALL) or FILEATR(*MXDF) cannot be specified.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

FILEATR

Specifies the type of file whose attributes are shown. If SYSTEM(*RMT) is specified for displayed or printed output, the only valid file attributes are *PF, *LF, and *ALL. If SYSTEM(*RMT) or SYSTEM(*ALL) is specified for outfile processing, the only valid file attributes are *PF, *LF, and *ALL. See the table at the end of this parameter description.

***ALL:** The attributes of all files are shown. If a value is specified for the OUTFILE parameter, and TYPE(*ATR) is specified, then FILEATR cannot be *ALL.

***DSPF:** The attributes of display files are provided. This includes both *DSPF and *DSPF38. If a value is specified for the OUTFILE parameter and TYPE(*ATR) is specified, display files or mixed files with display device entries are shown. If a name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDDSP from system file QAFDDSP; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***PRTF:** The attributes of printer files are provided. This includes both *PRTF and *PRTF38. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDPRT from system file QAFDPRT; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***DKTF:** The attributes of diskette files are provided. This includes both *DKTF and *DKTF38. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDDKT from system file QAFDDKT; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***TAPF:** The attributes of tape files are provided. This includes both *TAPF and *TAPF38. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDTAP from system file QAFDTAP; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***CMNF:** The attributes of communications files are provided. This includes both *CMNF and *CMNF38. If a value is specified for the OUTFILE parameter and TYPE(*ATR) is specified, only communication files or mixed files with communications device entries are shown. If a name specified for the OUTFILE parameter

is for a file that does not exist, the file is created with the format QWHFDCMN from system file QAFDCMN; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***BSCF:** The attributes of BSC communications files are provided. This includes both *BSCF and *BSCF38. If a value is specified for the OUTFILE parameter and TYPE(*ATR) is specified, only BSC communication files or mixed files with BSC communications device entries are shown. If a name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDBSC from system file QAFDBSC; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***MXDF:** The attributes of mixed files are provided. If a value is specified for the OUTFILE parameter, and TYPE(*ATR) is specified, then FILEATR cannot be *MXDF. This file attribute is not valid if SYSTEM(*RMT) is specified.

***PF:** The attributes of physical files are provided. This includes both *PF and *PF38. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDPHY from system file QAFDPHY; if the file exists, it must have this format.

***LF:** The attributes of logical files are provided. This includes both *LF and *LF38. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDLGL from system file QAFDLGL; if the file exists, it must have this format.

***SAVF:** The attributes of save files are provided. This includes both *SAVF and *SAVF38. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDSAV from system file QAFDSAV; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***DDMF:** The attributes of the Distributed Data Management (DDM) files are provided. This includes both *DDMF and *DDMF38. If TYPE(*ATR) is specified and the name for the OUTFILE is for a file that does not exist, the file is created with the format QWHFDDDM from system file QAFDDDM; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

***ICFF:** The attributes of ICF files are provided. If TYPE(*ATR) is specified and the name specified for the OUTFILE parameter is for a file that does not exist, the file is created with the format QWHFDICF from system file QAFDICF; if the file exists, it must have this format. This file attribute is not valid if SYSTEM(*RMT) is specified.

Command Parameters		Output File and Record Format	
TYPE	FILEATR	OUTFILE	FORMAT
*BASATR	Note 1	QAFDBASI	QWHFDBAS
*ATR	*BSCF	QAFDBSC	QWHFDBSC
*ATR	*CMNF	QAFDCMN	QWHFDCMN
*ATR	*DSPF	QAFDDSP	QWHFDDSP
*ATR	*PRTF	QAFDPRT	QWHFDPRT
*ATR	*DKTF	QAFDDKT	QWHFDDKT
*ATR	*TAPF	QAFDTAP	QWHFDTAP
*ATR	*PF	QAFDPHY	QWHFDPHY
*ATR	*LF	QAFDLGL	QWHFDLGL
*ATR	*ICFF	QAFDICF	QWHFDICF
*ATR	*SAVF	QAFDSAV	QWHFDSAV
*ATR	*DDMF	QAFDDDM	QWHFDDDM
*ACCPTH	Note 2	QAFDACCP	QWHFDACP
*SELECT	Note 3	QAFDSELO	QWHFDDSO
*SEQ	Note 2	QAFDCSEQ	QWHFDSEQ
*MBR	Note 2	QAFDMBR	QWHFDMBR
*SPOOL	Note 4	QAFDSPOL	QWHFDSPL
*RCDFMT	Note 1	QAFDRFMT	QWHFDFMT
*MBRLIST	Note 2	QAFDMBRL	QWHFDML
*JOIN	Note 5	QAFDJOIN	QWHFDJN

The user must specify the FILEATR parameter only when TYPE(*ATR) is used. The default of FILEATR(*ALL) is valid on any of the other TYPE values as long as the user is requesting file types that match the appropriate request.

Notes:

1. Any entry or multiple entries can be made in the FILEATR parameter.
2. Only for physical and logical files
3. Only for logical files
4. Only for device files
5. Only for join logical files

OUTFILE

Specifies the qualified name of the database output file to which the output of the display is directed.

When creating the database output file, the current date, time, and system name must be included. The system name is the name of the source system, not the target system. If the file entered as the OUTFILE does not exist, the system creates it in the specified library, or in the *CURLIB library when *LIBL is specified as the library name. If the file exists and has the correct format, the system uses it. If the file is created, the text is 'Outfile for DSPFD' and the public authority is *EXCLUDE.

Note: The outfile format must correspond to that of the TYPE and FILEATR parameter descriptions. See the table at the end of the FILEATR parameter description.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the command. If no file by that name is found, a file by that name is created and stored either in the specified library, or, if not qualified, in the current library. The file can be used again by other DSPFD commands that request the same types of information. The IBM-supplied model database files cannot be specified.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the user has the option to add records to the end of the existing member or to clear the existing member before adding the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

SYSTEM

Specifies whether the information provided is about files on the local system (*LCL), on the remote system or systems (*RMT), or on both the local and the remote systems (*ALL). For performance considerations, unless explicitly looking for information about files on systems other than the local system, the default value of *LCL must not be changed.

***LCL:** The information provided is about local files only.

***RMT:** The information provided is about remote files that have been named on the RMTFILE parameter of successfully issued CRTDDMF commands. To view this information, specify the name of the distributed data management (DDM) file, not the name of the remote file. Only information returned by the remote system is shown.

***ALL:** The information provided is about files on both the local and the remote systems. For DDM files, the displays first show information about the local DDM file and then, if available, information about the remote file that was named on the RMTFILE parameter of the CRTDDMF command.

Examples

In the following examples, assume that the commands are entered at a display work station and that the user of the command is authorized to access all relevant libraries and objects.

Example 1: Displaying Definition of a File

```
DSPFD   FILE(*ALL/FILE1)
```

This command shows the definition of FILE1 as defined in all libraries authorized for the user on the local system. The information is displayed at the work station running the command.

Example 2: Displaying Attributes of Local DDM File

```
DSPFD   FILE(LIBRARY1/FILE1)
```

This command shows the definition of FILE1 as defined in LIBRARY1. If FILE1 is a Distributed Data Management (DDM) file, only the attributes of the local DDM file are shown at the work station running the command.

Example 3: Displaying Definition of All Files

```
DSPFD   FILE(*ALL/*ALL) TYPE(*ALL) SYSTEM(*ALL)
```

This command shows the definition of all the files in all libraries authorized for the user on both the local and on all remote systems. For DDM files, the displays first show information about the local DDM file and then, if available, information about the remote file that is named on the RMTFILE parameter of that CRTDDMF command. The information is shown at the work station running the command.

Additional Considerations

When the DSPFD command is entered, the specified libraries are searched for the file or files specified by the FILE parameter; then a group of records that give file-level information about each file is created. The records are placed in the printer device file named QPDSPFD. If OUTPUT(*PRINT) is specified on the command, the records are listed on the printer in the following order:

1. On the first line, the leftmost fields show system identifier numbers. The right side shows the date and time the job was run and the page number.
2. The file, library, type of information, file attributes, and processor are shown (page 1 only).
3. The specific file, library, and type of file (to which the following information applies) are shown.

4. Header information that identifies the type of information being presented is shown, followed by the requested attribute information.

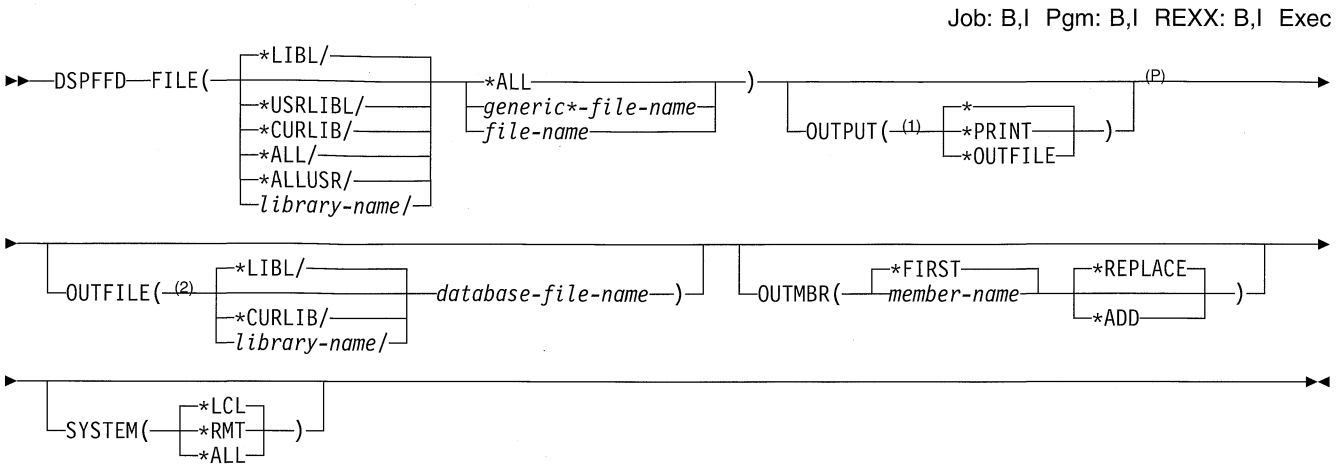
If the attributes of more than one file are shown or printed, the attributes of the next file are shown only after all of the requested attributes for the first file are shown. For printed output, a separate page is used for each type of information that is printed, as indicated by the TYPE parameter.

For displayed or printed output, if no detailed information exists for the specified file type, an appropriate indication is given under the section heading for that type of display. For example, if a logical file is shown and TYPE(*MBR *MBRLIST) was specified, but no members have yet been defined for the file, the message *(no members in file)* is shown or printed under the section headings for the member attributes display and for the member list display.

For output to a display, the first four lines of each display contain information about the spooled file and fields used for display spooled file functions.

Numeric values in select/omit field descriptions are shown without decimal points. See the description of the fields (by using the DSPFFD command) to determine the decimal positions.

DSPFFD (Display File Field Description) Command



Notes:
1 If OUTPUT(*OUTFILE) is specified, a file name must be specified for OUTFILE.
P All parameters preceding this point can be specified in positional form.
2 If OUTFILE is not specified, OUTMBR must not be specified.

Purpose

The Display File Field Description (DSPFFD) command shows, prints, or places in a database file field-level information for one or more files in a specific library or all the libraries to which the user has access.

If the information is put in a database file, the database file will have a record format named QWHDRFFD. The fields in record format QWHDRFFD are the same as the fields in the IBM-supplied format QWHDRFFD in file QADSPFFD in the library QSYS. The following information is contained in the database file:

- For each file specified in the command, the database record contains:
  - The name of the file, the name of the library containing the file, the file type, and file member
  - The file creation date and the number of record formats in the file
  - The name of the record format used by the file, the format level identifier, the format text description, the format record length, and the number of fields in the format
  - The information retrieval date and time
- For each field in the record format, the record also contains the following, if applicable:
  - The field name and external field name
  - The type and length of the field
  - For fields referencing other fields, the name of the referenced file, record format, and field; if any attributes of the referenced field were changed, the attribute type is given
  - The edit code, edit word, and column headings associated with the field
  - A signal of whether validity checking is performed on the field
  - The validity check message identifier, the message file, and the library
  - The use of the field

To create an OUTFILE, the user must have object operational authority to the Create Physical File (CRTPF) command and add authority to the library. To use an existing OUTFILE, the user must have object operational and add authority to the file. The user must also have object management and delete authority if *REPLACE is specified on the OUTMBR parameter.

Restrictions:

1. Before users can display each file specified, they must have object operational authority for the file.
2. Also, of the libraries specified by the library qualifier, only the libraries for which the user has read authority are searched.

Required Parameters

FILE

Specifies the qualified name of the file or the generic name of several files that have their field-level descriptions shown. This parameter specifies that all files in the specified library or libraries (*LIBL/*ALL for example), or that all files in all libraries (*ALL/*ALL), have their field-level descriptions shown.

Note: The user must have authority to both the files and the libraries before this information is shown.

The name of the file field description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All files in the library or libraries specified have their field-level information shown.

generic-file-name:* Specify the generic name of the file or files that have field-level information shown. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

file-name: Specify the name of the file whose field-level information is shown.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or placed in a database outfile. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file where the displayed information is stored.

When creating the database output file, the current date, time, and system name must be included. The system name is the name of the source system, not the target system. If the specified output file does not exist, then the system creates a database file and member. If the file is created, the text is *OUTFILE for DSPFFD* and the public authority is *EXCLUDE.

Note: The outfile format must be the same as QWHDRFFD of system file QADSPFFD. For more information on the outfile format, refer to *Database Guide*.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file where the field level information is written. If no file is found by that name, a file by that name is created and stored in the specified library, or in *CURLIB if the file name is not qualified. This file can be used again when other DSPFFD commands are entered. The IBM-supplied database file QADSPFFD cannot be specified.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the user has the option to add records to

the end of the existing member or to clear the existing member and then add the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

SYSTEM

Specifies whether the information returned by the DSPFFD command is about files on the local system (*LCL), about files on the remote system or systems (*RMT), or about files on both local and remote systems (*ALL). For performance considerations, unless explicitly looking for information about files on systems other than the local system, the default value of *LCL should not be changed.

***LCL:** The information provided is about local files only.

***RMT:** The information provided is about remote files that have been named on the RMTFILE parameter of successfully run CRTDDMF commands. To look at this information, specify the name of the distributed data management file, not the name of the remote file. Only the information returned by the remote system is shown.

***ALL:** The information provided is about files on both the local and the remote systems. For DDM files, the displays show only information about the remote file that was named on the RMTFILE parameter of the CRTDDMF command.

Examples

Example 1: Displaying Information About a File on the Local System

```
DSPFFD   FILE(LIB1/FILE2)
```

This command shows the field-level information about file FILE2 in LIB1 on the local system. The information is shown at the work station where the command was entered.

Example 2: Displaying Information About Files on the Local and Remote Systems

```
DSPFFD   FILE(*ALL/*ALL)  SYSTEM(*ALL)
```

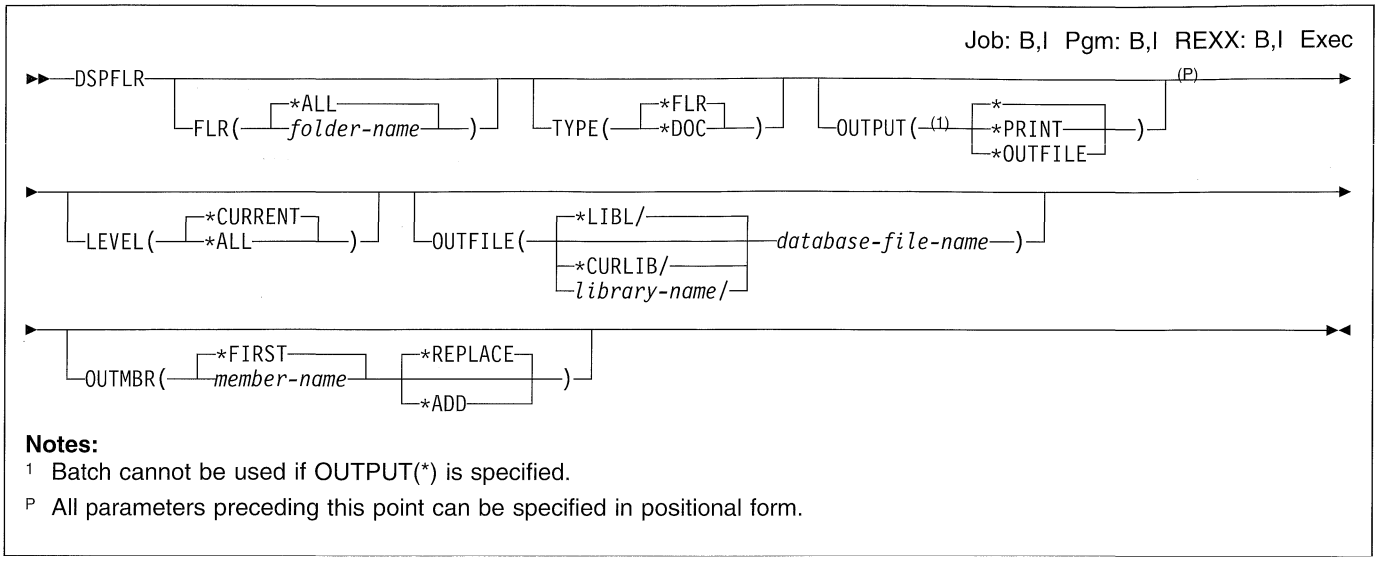
This command shows the field-level information of all files in all libraries authorized to the user on the local system and on all remote systems. For distributed data management files, the display shows only information about the remote file that is named on the RMTFILE parameter of the CRTDDMF command. The information is shown at the work station where the command was entered.

Example 3: Directing Output to a Database File

```
DSPFFD   FILE(QGPL/FLDREF)  OUTPUT(*OUTFILE)
  OUTFILE(QGPL/FLDREFX)
```

This command puts the field-level information for the file FLDREF in the QGPL library on the local system into a database file named FLDREFX in the general purpose library, QGPL. The FLDREFX file in the QGPL library can then be processed by a program.

DSPFLR (Display Folder) Command



Notes:
1 Batch cannot be used if OUTPUT(*) is specified.
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Folder (DSPFLR) command allows the user to display or print a list of folders and documents or to create an out file that contains the list of folders or documents.

More information on displaying or printing lists of folders and documents is in the *Using OfficeVision/400* Word Processing*.

Optional Parameters

FLR

Specifies the name of the folders or members on the list.

***ALL:** All the folders that the user has authority to see are on the list.

folder-name: Specify the name of the folder whose contents is listed.

TYPE

Specifies whether documents or folders are listed.

***FLR:** The list contains folders.

***DOC:** The list contains documents.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

LEVEL

Specifies whether the list includes folders nested within folders or only folders at the current level. This parameter is not allowed when OUTPUT(*) is specified. For documents, *ALL cannot be specified.

***CURRENT:** Only folders at the current level are listed.

***ALL:** Nested folders within the folders specified are listed.

OUTFILE

Specifies the qualified name of the database file where the output of the display is directed. If the OUTFILE does not exist, this command creates a database file in the specified library. If the file is created by this function, the text for the database file is 'OUTFILE created by DSPFLR', and the authority given to all users is *EXCLUDE.

More information on defining the format of database files (OUTFILES) is in the *Office Services Concepts and Programmer's Guide*.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specifies the qualified name of the database file that receives the output of the display.

OUTMBR

Specifies the name of the database file member to which the output is directed. If a member already exists, the system uses the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member with the name of the output file specified on the OUTFILE parameter. If an output file member name is specified, but the member does not exist, the system creates it.

Element 1: New Member

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Existing Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Displaying or Printing Output

```
DSPFLR  FLR(GENERAL)  TYPE(*FLR)  OUTPUT(*)
```

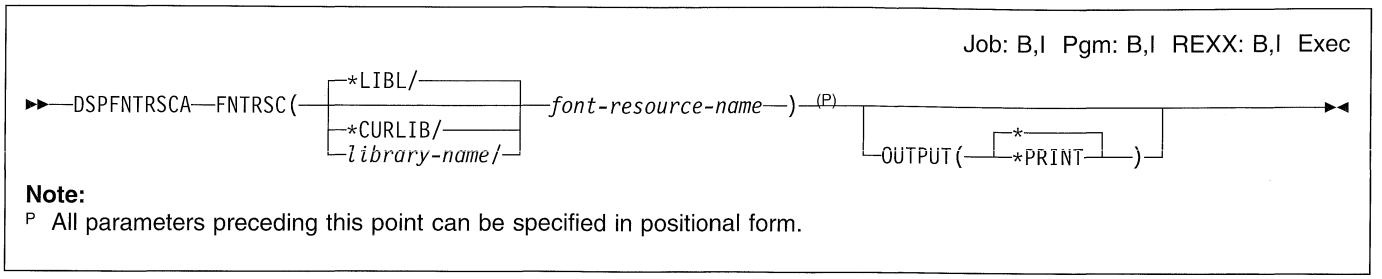
This command shows a list of folders for the folder, GENERAL, at the requesting work station (if requested by an interactive job) or the output is printed with the job's spooled output (if requested by a batch job).

Example 2: Directing Output to a Database File

```
DSPFLR   FLR(*ALL)  TYPE(*FLR)  OUTPUT(*OUTFILE)
  LEVEL(*ALL)  OUTFILE(MYLIB/MYFILE)
  OUTMBR(MYMBR *REPLACE)
```

This command shows all the folders that the user has authority to view. The output is directed to MYMBR, in MYFILE located in MYLIB. If member MYMBR already exists, MYMBR is cleared and then the new records are added.

DSPFNTRSCA (Display Font Resource Attributes) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Font Resource Attributes (DSPFNTRSCA) command shows the following for the specified font resource:

- Object attribute
- Picture element density for font character sets
- Descriptive text for the font resource

Required Parameters

FNTRSC

Specifies the qualified name of the font resource whose attributes are displayed.

The name of the font resource can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

font-resource-name: Specify the name of the font resource whose attributes are displayed.

Optional Parameters

OUTPUT

Specifies whether the output is directed to the display station screen or to a printer. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

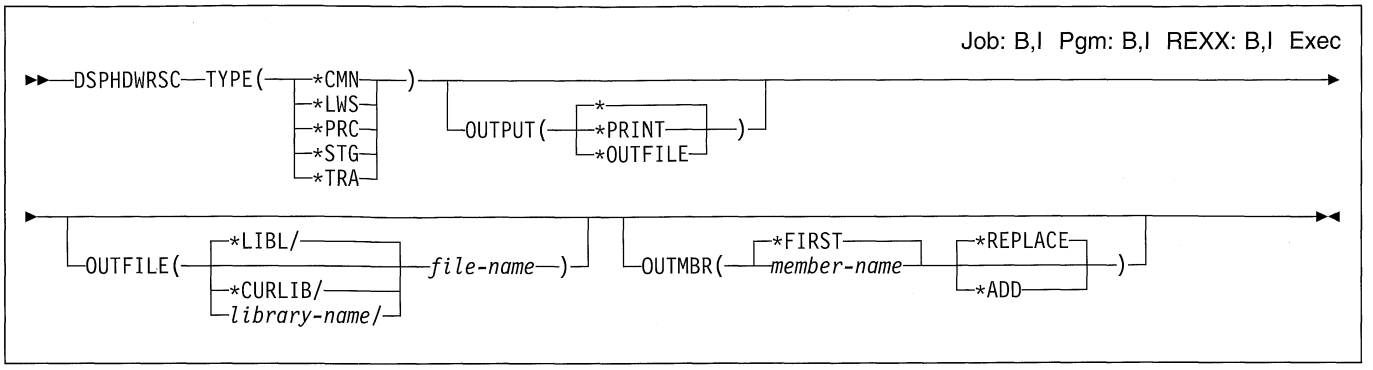
***PRINT:** The output is printed with the job's spooled output.

Example.

```
DSPFNTRSCA  FNTRSC(SHALIMAR/X0A0557C)
```

This command displays the attributes associated with the font resource X0A0557C in library SHALIMAR.

DSPHDWRSC (Display Hardware Resources) Command



Purpose

The Display Hardware Resources (DSPHDWRSC) command is used to display, print, or direct to an output file the various types of System Resource Management (SRM) information.

Required Parameter

TYPE

Specifies the type of information that is to be displayed, printed, or directed to an output file.

***CMN:** Displays, prints, or directs to an output file the contents of each SRM communications resource record. This information consists of resource name, resource type, serial number, location, status, address, configuration description, and resource description.

***LWS:** Displays, prints, or directs to an output file the local work station resource information. This information consists of resource name, resource type, serial number, location, status, address, configuration description, and resource description.

***PRC:** Displays, prints, or directs to an output file the processor resource information. This information consists of resource name, resource type, serial number, location, status, and resource description.

***STG:** Displays, prints, or directs to an output file the storage devices resource information. This information consists of resource name, resource type, serial number, location, status, address, configuration description, and resource description.

***TRA:** Displays, prints, or directs to an output file the TR LAN adapters resource information. This information consists of the LAN controller adapter address (12 hex digits) and user-assigned adapter name and adapter description.

For information to be accessible from this command, the user must specify an adapter name and description to an existing LAN controller address through the Work with TRLAN Adapters (WRKTRA) command or must create a new adapter entry.

Optional Parameters

OUTPUT

Specifies whether SRM data is displayed, printed, or directed to an output file.

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The requested data is printed with the job's spooled output.

***OUTFILE:** The requested data is directed to an output database file.

OUTFILE

Specifies the qualified name of the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library.

If a new file is created, the system uses one of the following physical files as a model, depending on the resource type specified on the TYPE parameter.

Resource Type	Physical File Model	Format name
*CMN	QARZDCMN	QRZDCMN
*LWS	QARZDLWS	QRZDLWS
*PRC	QARZDPRC	QRZDPRC
*STG	QARZDSTG	QRZDSTG
*TRA	QARZDTRA	QRZDTRA

Field level information for any of the above files can be obtained using the Display File Field Description (DSPFFD) command and specifying the physical file in library QSYS. The information for the following fields is not shown with the DSPFFD command:

Table 43 (Page 1 of 2). Status of Device Types

Field	Field values and description
Record Format	0 = *CMN communications resource 1 = *LWS local work station 2 = *PRC processor type 3 = *STG storage resource 4 = *TRA token-ring LAN

Table 43 (Page 2 of 2). Status of Device Types

Field	Field values and description
Resource Level	For *CMN type records: 1 = Input/output processor 2 = Input/output adapter 3 = Communications port For *LWS or *STG type records: 1 = Input/output processor 2 = Controller 3 = Device For *PRC type records: 0 = Processor unit 1 = Processor/main storage 2 = Next level main storage
Resource Description	00 = Communications processor 01 = Communications adapter 02 = Communications port 03 = Work station controller 04 = Printer device 05 = Work station device 06 = ASCII work station controller 07 = Work station printer 08 = Main card enclosure 09 = Bus extender 10 = System processor 11 = Service processor 12 = Main storage card 13 = Storage controller 14 = Tape controller 15 = Disk storage controller 16 = Diskette controller 17 = Tape unit 18 = Disk unit 19 = Diskette unit 20 = Tape input/output processor 21 = Work station processor 22 = System control panel 23 = Next level main storage card
Resource Status	0 = Not detected 1 = Inoperative 2 = Not ready 3 = Operational
Resource Extended Description (only applies to the QARZDCMN physical file model for *CMN)	If resource description is 00 - Communications processor: 01 = Comm processor 02 = Combined function IOP If resource description is 01 - Communications adapter: 01 = Comm adapter 02 = LAN adapter 03 = ISDN adapter 04 = Fax adapter If resource description is 02 - Communications port: 01 = Comm port 02 = ISDN port 03 = Token-ring port 04 = Autocall port 05 = Ethernet port
Resource Extended Description (continued)	06 = V.24 port 07 = X.21 port 08 = V.35 port 09 = V.24 enhanced port 10 = V.36 port 11 = FDDI port 12 = SDDI port 13 = Fax port

This parameter is valid only if OUTPUT(*OUTFILE) is specified. Only one type of resource information: *CMN, *LWS, *PRC, *STG, or *TRA can be directed to one output file.

The possible library values are:

***LIBL:** The library list is used to locate the output file.

***CURLIB:** The current library for the job is used to locate the output file. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library where the output file is located.

file-name: Specify the name of the file where the requested information is to be located.

OUTMBR

Specifies the member name to receive the requested SRM data. This parameter is valid only if OUTPUT (*OUTFILE) is specified.

Element 1: Member to Receive Output

***FIRST:** The first member name receives the requested SRM data.

member-name: Specify the name of the member to receive the requested SRM data. Valid values range from 1 through 10 characters.

Element 2: Operation to Perform on Member

***REPLACE:** The new data replaces the existing data.

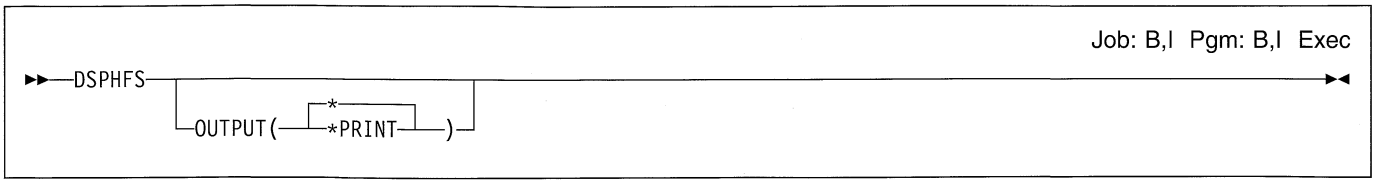
***ADD:** The new data is added to the end of the existing data.

Example

```
DSPHDWRSC TYPE(*STG) OUTPUT(*OUTFILE) OUTFILE(STG)
```

This command places SRM information for storage resources in the output file STG. If output file STG is located in the library list, the records of its first member are replaced by the new records. If output file STG is not located in the library list, it is created in the current library with first member STG. This member contains the new records.

DSPHFS (Display Hierarchical File Systems) Command



Purpose

The Display Hierarchical File Systems (DSPHFS) command displays a list of registered file systems. The information on the list includes file system names, levels, and 50 character descriptions that are supplied by the file system at registration time.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: The output is sent to the display.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Historical File System Information

```
DSPHFS  OUTPUT(*)
```

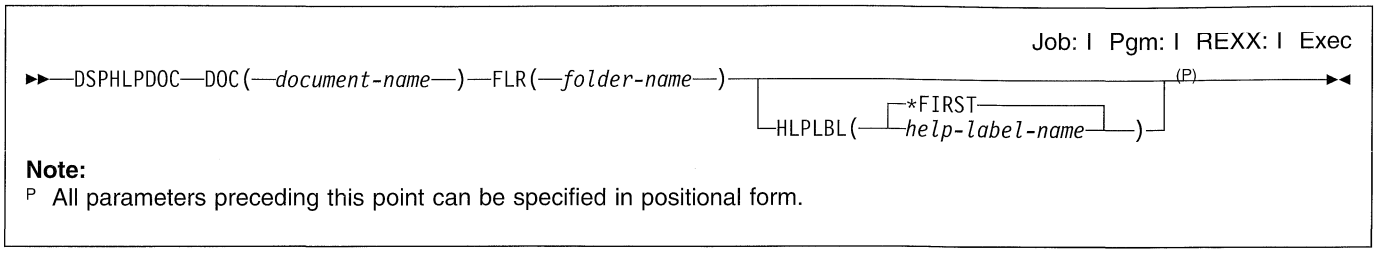
This command sends the output to the display.

Example 2: Printing Historical File System Information

```
DSPHFS  OUTPUT(PRINT)
```

This command sends the output to the printer file.

DSPHLPDOC (Display Help Document) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Help Document (DSPHLPDOC) command displays a document in its final form. The document is created using the word processing function of OfficeVision/400. More information on help documents is in the *Using OfficeVision/400* Word Processing*.

Required Parameters

DOC
Specifies the name of the document to display. Specify the name of the document. The document must be a resolved document.

FLR
Specifies the name of the folder that contains the document.

Optional Parameters

HLPLBL
Specifies where in the document the display is started. The user can use the page keys or other positioning requests to look for the information.

***FIRST:** The document is displayed at the start of the first page.

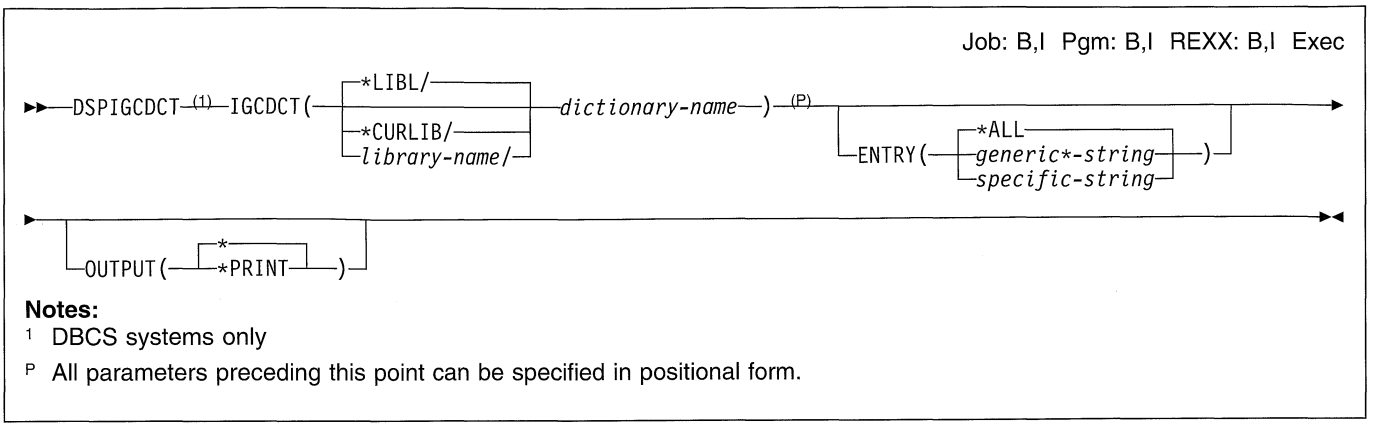
help-label-name: Specify the name of the label that is used as the starting point for displaying the document. This only indicates where to start viewing the document.

Example

```
DSPHLPDOC  DOC(KDOC) FLR(NFLR) HLPLBL(*FIRST)
```

This command uses the document KDOC in folder NFLR as the help document.

DSPIGCDCT (Display DBCS Conversion Dictionary) Command



Purpose

The Display DBCS Conversion Dictionary (DSPIGCDCT) command displays or prints the alphanumeric entries in the specified DBCS conversion dictionary and their related words. The system refers to the DBCS conversion dictionary when doing DBCS conversion.

Note: Use of the conversion function is not recommended for Chinese and Korean double-byte character sets (DBCS).

Required Parameters

IGCDCT

Specifies the qualified name of the DBCS conversion dictionary to be shown or printed. If you do not specify a library name, the first dictionary found when searching your library list is shown.

The name of the dictionary can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

dictionary-name: Specify the name of the dictionary that will be shown or printed.

Optional Parameters

ENTRY

Specifies the alphanumeric entries to be shown or printed with their related words.

***ALL:** The system displays or prints all of the dictionary entries and their related words.

generic-string:* The system displays or prints all of the dictionary entries that start with the specified string and their related words. The string cannot be longer than 12 characters.

A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

specific-string: Specify an alphanumeric entry (up to 12 characters) and its related words that are shown or printed.

OUTPUT

Specifies whether the output from this command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Entries

```
DSPIGCDCT   IGCDCT(DBCSLIB/QUSRIGCDCT)  OUTPUT(*)
```

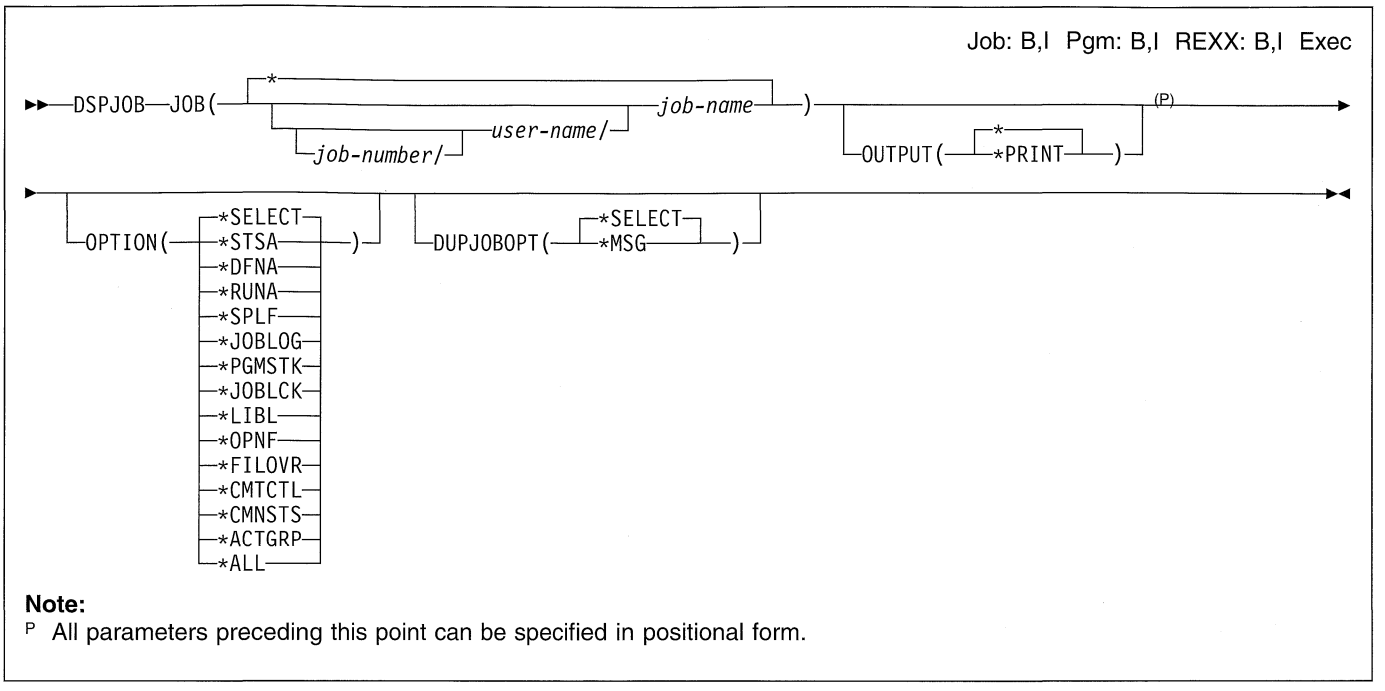
This command displays at a work station all entries in the DBCS conversion dictionary named QUSRIGCDCT, that is stored in the library DBCSLIB.

Example 2: Printing Entries

```
DSPIGCDCT   IGCDCT(DBCSLIB/QUSRIGCDCT)  ENTRY('?')
OUTPUT(*PRINT)
```

This command prints the entry ? and its related words from the DBCS conversion dictionary named QUSRIGCDCT, which is stored in the library DBCSLIB.

DSPJOB (Display Job) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Job (DSPJOB) command displays, for the specified user job, any of the following information:

- Job status attributes
- Job definition attributes
- Job run attributes
- Spooled file information
- Job log information
- Call stack information
- Job lock information
- Library list information
- Open file information
- File override information
- Commitment control status
- Communications status
- Activation group information

The information for the following options can be shown only when the job is active: job run attributes, call stack information, job lock information, library list information, job log information, open file information, file override information, commitment control status, and communication status. The following options can be found whether the user's job is on the job queue, on an output queue, or active in the system: job status attributes, job definition attributes, and spooled file information. However, the job is not considered to be in the system until all of its input has been completely read in; only then is an entry placed on the job queue.

Restrictions:

1. Unless a user has special job control (*JOBCTL) authority, or unless another job has the same user name specified, only the user's job can be displayed.
2. Activation group information for a job cannot be shown if the job is being held when this command is run.

Optional Parameters

JOB

Specifies the name of the user job whose information is to be displayed. If no job qualifier is given, a search for the simple job name is made of all of the jobs currently in the system. If duplicates of the specified name are found, a list of messages that contains the qualified job names of all duplicates is displayed.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
*
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: The job whose information is displayed is the job from which this display command is entered.

job-name: Specify the name of the job whose information is to be displayed.

user-name: Specify the name of the user of the job whose information is to be displayed.

job-number: Specify the number of the job whose information is to be displayed.

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. If the output is to go to a printer, the job log information is not included. To print the job log information, use the Display Job Log (DSPJOBLOG) command. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

OPTION

Specifies the information to be shown on the display station (or printed as specified on the OUTPUT parameter). Only one value may be selected.

***SELECT:** Shows a list of options that allows the user to select the displayed job information.

***STSA:** The status attributes associated with the job are displayed.

***DFNA:** The definition attributes associated with the job are displayed.

***RUNA:** The run attributes associated with the job are displayed.

***SPLF:** The spooled files associated with the job are displayed.

***JOBLOG:** The job log associated with the job is displayed. The job log is printed or displayed depending on what is specified for the OUTPUT parameter. This value is not shown when either OPTION(*SELECT) or OPTION(*ALL) is specified.

***PGMSTK:** The call stack associated with the job is displayed.

***JOBLCK:** The job locks associated with the job are displayed.

***LIBL:** The library list associated with the job is displayed.

***OPNF:** The open files associated with the job are displayed.

***FILOVR:** The file overrides associated with the job are displayed.

***CMTCTL:** The commit control status of the job is displayed.

***CMNSTS:** The communications status of the job is displayed.

***ACTGRP:** The activation groups associated with the job are displayed.

***ALL:** All information associated with the job is displayed.

DUPJOBOPT

Specifies the action taken when duplicate jobs are found by this command.

***SELECT:** The selection display is shown when duplicate jobs are found during an interactive session. Otherwise, a message is issued.

***MSG:** A message is issued when duplicate jobs are found.

Examples

Example 1: Printing the Spooled Output

```
DSPJOB  JOB(SMITH/PAYROLL)  OUTPUT(*PRINT)
```

This command directs the information for the job named PAYROLL submitted by the user named SMITH to the job's output spooling queue for printing.

Example 2: Displaying the Job's Spooled Output

```
DSPJOB  OPTION(*SPLF)
```

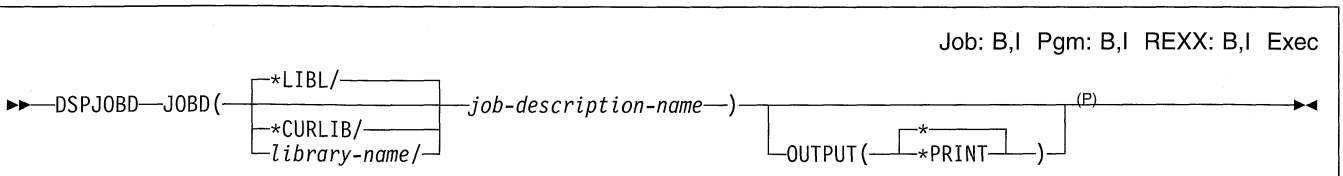
This command displays the spooled output for the current job.

Example 3: Displaying All of the Job's Information

```
DSPJOB  OPTION(*ALL)
```

This command displays all of the information for the current job.

DSPJOBD (Display Job Description) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Job Description (DSPJOBD) command displays the contents of the specified job description.

Restriction: The user must have object operational authority for the job description.

Required Parameters

JOBD

Specifies the qualified name of the job description to be displayed.

| The name of the job description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

job-description-name: Specify the name of the job description to be displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output on a printer. More information on
| this parameter is in Appendix A, "Expanded Parameter
| Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

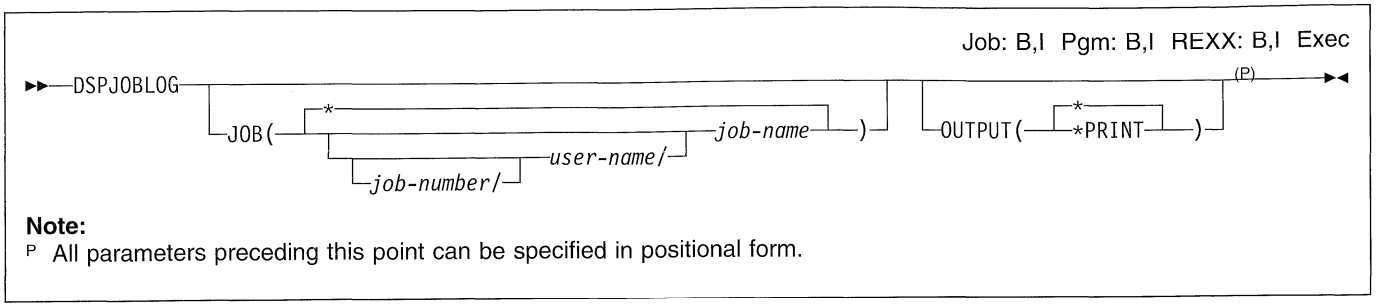
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPJOBD  JOBD(MYLIB/SPECIAL)
```

This command displays the job description named SPECIAL that is stored in the library MYLIB.

DSPJOBLOG (Display Job Log) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Job Log (DSPJOBLOG) command shows commands and related messages for a job whose job log has not been written. This command also shows commands for a job on a job queue that has not started processing.

This command can be used to monitor the progress of a job.

Note: This function can also be accessed through the Display Job (DSPJOB) and Work with a Job (WRKJOB) commands.

Restrictions:

1. To display a job log, the user must have special job control authority (*JOBCTL), or the job must have the same user name as the person running this command.
2. To display a job that is running for the security officer or a member of the security officer group, the user also must have *ALLOBJ special authority.

Optional Parameters

JOB

Specifies the name of the job whose job log is shown. If no job qualifier is given, all jobs currently in the system are searched for the simple job name. If duplicates of the specified name are found, a list of messages containing the qualified job names of all duplicate names is shown.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
*
job-name
user-name/job-name
job-number/user-name/job-name
```

*: The job in which this command is issued is the job whose job log is shown.

job-name: Specify the name of the job whose job log is shown.

user-name: Specify the name of the user of the job whose job log is shown.

job-number: Specify the number of the job whose job log is shown.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

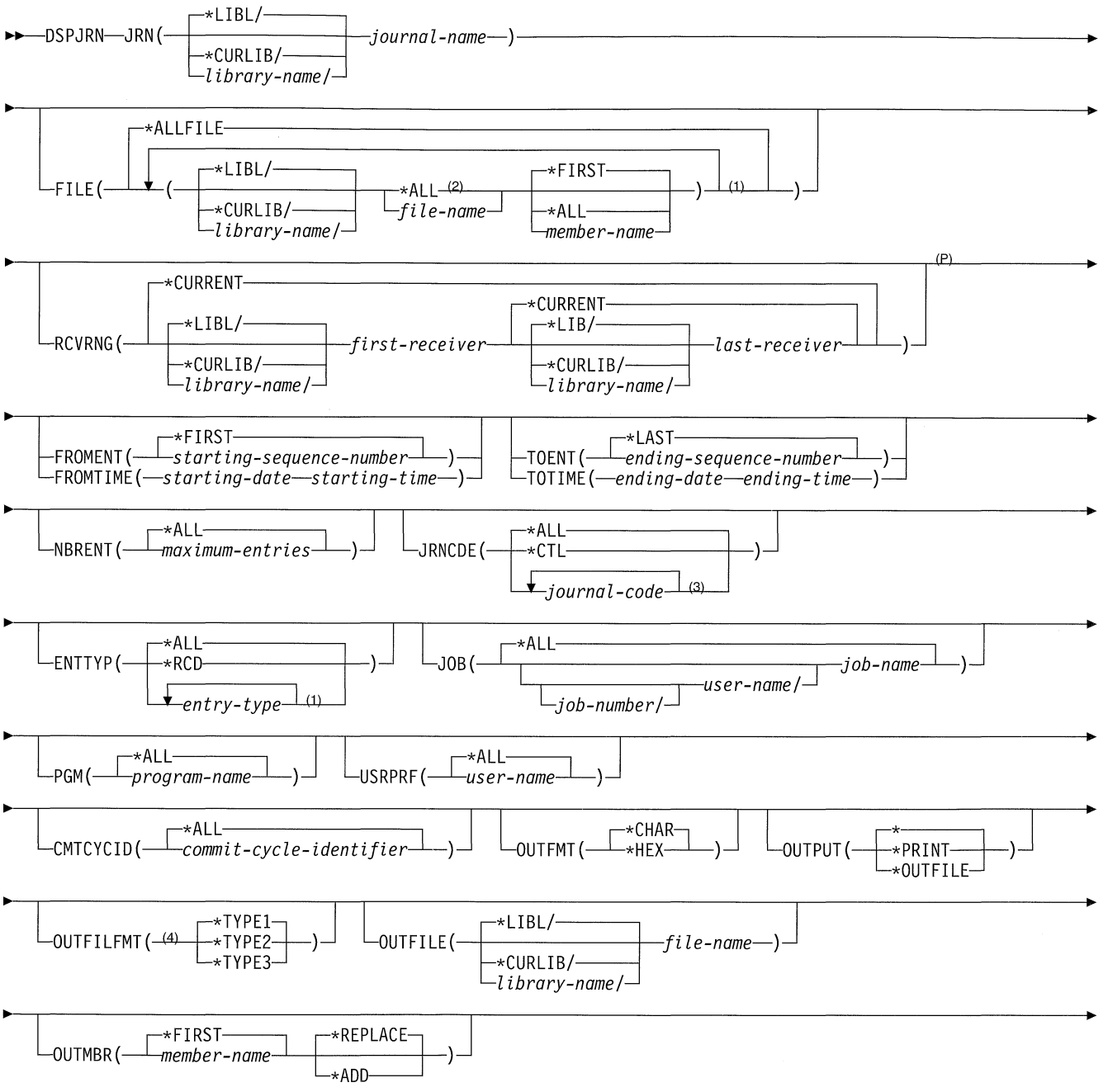
Example

```
DSPJOBLOG   JOB(ANDERSON/PAYROLL)  OUTPUT(*PRINT)
```

This command produces a job log for job PAYROLL for user ANDERSON.

DSPJRN (Display Journal) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Notes:

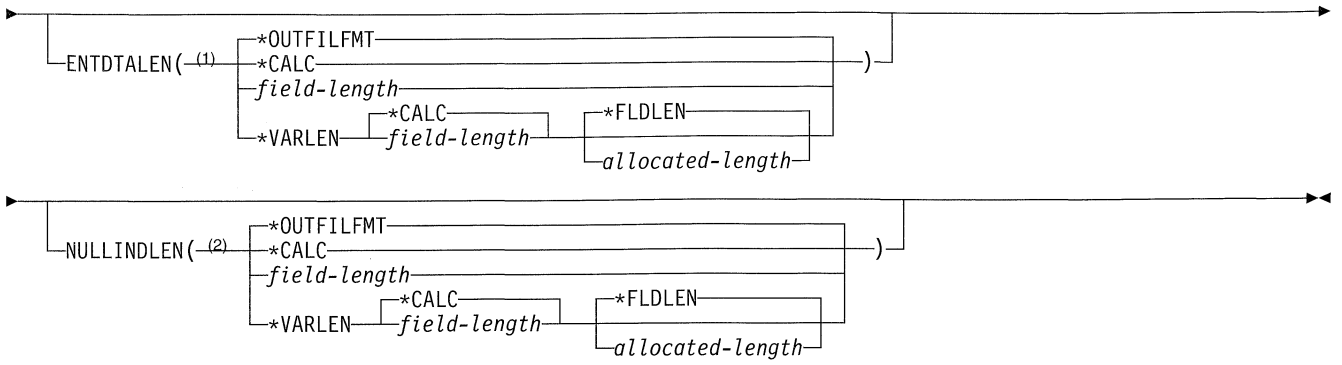
1 A maximum of 50 repetitions

2 If *ALL is specified instead of file-name, the format must be library-name/*ALL.

P All parameters preceding this point can be specified in positional form.

3 A maximum of 10 repetitions.

4 The OUTFILFMT parameter can be specified only if OUTPUT(*OUTFILE) is specified.



Notes:
1 The ENTDTALEN parameter is valid only if OUTPUT(*OUTFILE) is specified.
2 The NULLINDLEN parameter can be specified only if OUTFILEFMT(*TYPE3) is specified.

Purpose

The Display Journal (DSPJRN) command allows the user to change journal entries (contained in one or more receivers) into a suitable form for output. Output can be shown, printed with the job's spooled printer output, or written to a database file. If the database file exists, records may either replace or be added to the current data in the indicated file member. The system creates the specified database file and member if they do not exist. Database files created by the system have a standard format. A warning message is sent and the records are truncated if any of the entries are longer than the specified maximum record length of the output files.

The contents of selected entries in the journal receivers may be converted for output and shown, printed, or directed to a database output file. It is also possible to selectively limit the entries that are displayed. If no journal entries satisfy the selection or limitation criteria, an escape message is sent indicating that fact.

Gaps may exist in the sequence numbers of the entries converted. These occur because some of the journal entries represent internal system information. These entries are not converted.

It is possible to show journal entries whose journal sequence numbers are reset in the chain of receivers being specified.

If journal receivers are attached and detached in pairs, the system initially tries to use the first of the receivers in the pair (the first of the two shown in the WRKJRNA receiver directory display). If the first of the pair cannot be used (for example, because it is not found or is damaged), the system tries to use the second receiver of the pair. If both receivers cannot be used, the command ends.

When the output of the command is directed to the requesting work station, basic information on the journal entries is shown. The next sequential set of entries can be shown by pressing the Page key. If TOENT(*LAST) is specified on the command, the ending receiver is *CURRENT, and the last journal entries in the journal are shown, the Page Down key shows any new journal entries added to the journal since the last time the Page Down key was pressed. From the basic display, an option can be selected to show information in detail for any journal entry being shown.

If the output is printed with the job's spooled printer output, all of the information that would be shown is printed.

If the output is either shown at the requesting work station or printed with the job's spooled printer output, a hexadecimal representation of the entry-specific data can be requested.

Restrictions:

1. The file specified for the database output file must not be journaled to the same journal. More information on database output file record formats is in the *Database Guide*.
2. If the sequence number is reset in the range of the receivers specified, the first occurrence of FROMENT or TOENT is used, if they are specified.
3. If more than one selection value is specified, a journal entry must satisfy all of the selection values being generated. Selection values are FILE, JOB, PGM, ENTTYP, CMTCYCID, and JRNCDE. If none are specified, all entries are converted as specified in the values given for the FROMENT, TOENT, FROMTIME, TOTIME, and NBRENT parameters.

Required Parameter

JRN

Specifies the qualified name of the journal from which the journal entries are retrieved for conversion and output.

The name of the journal can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

journal-name: Specify the name of the journal.

Optional Parameters

FILE

Specifies up to 50 qualified file names whose journal entries are converted for output. The FILE parameter also specifies the name of the file member whose journal entries are to be converted for output.

If the specified physical file member currently exists on the system, all journal entries in the specified receiver range for that physical file member are converted for output. If the specified physical file member does not exist on the system, journal entries in the specified receiver range for any physical file member of that name that previously existed on the system are converted for output.

The single value *ALLFILE can be specified on the list of two values.

***ALLFILE:** For all files, all journal entries on the specified journal receivers are converted.

Element 1: File Name

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***ALL:** Journal entries in all physical files in the specified library (the library name must be specified) whose changes are currently journaled are converted for output. If *ALL is specified and the user does not have the required authority to all of the files, an error occurs, and the command ends.

file-name: Specify the name of the physical database file whose journaled changes are being converted for output.

Element 2: Member Name

***FIRST:** Entries for the first member in the file are converted for output.

***ALL:** Entries for currently existing members of the file are converted for output.

member-name: Specify the name of the file member whose entries are converted for output.

If *ALL is specified for the file-name value, this member name is used for all applicable files in the library. For example, if library-name/*ALL *FIRST is specified on the FILE parameter, the journal entries of the first members of all applicable files in the specified library are converted for output.

Note: If more than the maximum number of members is identified (4096 members if a list of files is specified), an error occurs and no entries are converted for output. This restriction is ignored if *ALLFILE is specified.

If the specified physical file does not exist on the system, specify either *ALL or a specific file member name.

RCVRNG

Specifies the first (beginning) and last (ending) journal receivers (the receiver range) that contain the journal entries being converted for output. The system starts with the first journal receiver (as specified by the first value) and proceeds through the receiver chain until the last receiver (as specified by the last value) is processed. If dual receivers are used at any time, the first of the receivers is used when chaining through the receivers. The Work with Journal Attributes (WRKJRNA) command can be used to show the order of the receivers in the receiver chain. If a problem (such as damaged or not found receivers) is found in the receiver chain prior to starting the conversion, the system tries to use the second of the dual receivers. If the second of the receivers is damaged or not found, the command ends.

***CURRENT:** The journal entries in the currently attached receiver are converted for output.

Element 1: First Receiver

The name of the first journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

first-receiver: Specify the name of the first journal receiver whose entries are converted for output.

Element 2: Last Receiver

***CURRENT:** Conversion of journal entries continues for all journal receivers in the chain that begins with the receiver specified by the first parameter value through the currently attached journal receiver.

The name of the second journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

last-receiver: Specify the name of the last journal receiver whose entries are to be converted for output. If the end of the receiver chain is reached before a receiver with this name is found, an error occurs, and no journal entries are converted.

Note: If the maximum number of receivers (256) in the range is surpassed, an error occurs and no journal entries are converted.

FROMENT

Specifies the first journal entry to be considered for conversion for output.

***FIRST:** The first journal entry in the specified journal receiver range is the first entry to be converted.

starting-sequence-number: Specify the sequence number of the first entry to be converted.

FROMTIME

Specifies the date and time of the first journal entry to be converted for output.

The first journal entry with the specified date and time or the next later journal entry is the starting point for conversion of journal entries. The format of the date must be defined by the job attributes DATFMT and, if separators are used, DATSEP. The time is specified in 24-hour format and can be specified with or without a time separator:

- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where hh = hours, mm = minutes, and ss = seconds. Hours, minutes, and seconds must each be exactly 2 digits (use leading zeros if necessary).
- With a time separator, specify a string of 5 or 8 digits where the time separator specified for your job is used to separate the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command will fail.

TOENT

Specifies the last journal entry to be converted for output.

***LAST:** The last journal entry in the specified journal receiver chain is the final entry to be converted.

ending-sequence-number: Specify the sequence number of the final journal entry being converted for output.

TOTIME

Specifies the date and time of the last entry to be converted for output. The journal entry with the specified date and time or the latest earlier journal entry is the ending point for conversion of journal entries. See the FROMTIME parameter for a description of time formats.

NBRENT

Specifies the total number of journal entries to be converted for output.

***ALL:** All journal entries that satisfy the selection values in the specified journal receivers are converted.

maximum-entries: Specify the maximum number of journal entries to be converted. If the specified journal entry assigned by the TOENT or TOTIME parameter is reached before the value specified for NBRENT is met, the command ends normally.

JRNCDE

Specifies whether the entries converted for output are limited to the journal entries that contain the specified journal code.

***ALL:** The conversion of journal entries is not limited to entries with a particular journal code.

***CTL:** Only journal entries written to control the journal functions are converted (journal codes = 'J' and 'F').

journal-code: Specify the journal code that limits the journal entries to be converted. Only journal entries that contain the specified journal codes are converted. A list of journal codes that can be specified is in the *Advanced Backup and Recovery Guide*.

ENTTYP

Specifies whether the journal entries to be converted for output are limited to the journal entries of the specified types.

***ALL:** The conversion of journal entries is not limited to entries of a particular type.

***RCD:** Only entries that have an entry type for record level operations are converted (these are entry types BR, DL, DR, PT, PX, UB, UP, and UR).

entry-type: Specify the entry type that limits the number of journal entries to be converted. Up to 50 valid entry types can be specified. Only journal entries of the specified entry type are to be converted. A list of valid entry types is in the *Advanced Backup and Recovery Guide*.

JOB

Specifies whether the journal entries to be converted for output are limited to those for a specified job. If the fully qualified name of the job is not specified, all of the journal entries that contain the simple job name are to be converted.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
*ALL
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ALL:** The conversion is not limited to entries for a particular job.

job-name: Specify the name of the job whose journaled changes are converted.

user-name: Specify the name of the user of the job whose journaled changes are to be converted.

job-number: Specify the number of the job whose journaled changes are to be converted.

PGM

Specifies whether the journal entries to be converted for output are limited to the journal entries for a specified program.

***ALL:** The conversion is not limited to entries for a specified program.

program-name: Specify the name of the program whose journaled changes are to be converted. Only journaled changes for this program are converted.

USRPRF

Specifies whether the journal entries to be converted are limited to those for a specified user profile name. The user profile name is the user profile under which the job is run that deposited the journal entries.

***ALL:** The conversion is not limited to entries for a specified user profile.

user-name: Specify the name of the user profile whose journaled changes are to be converted. Only journaled changes for this user profile are to be converted.

CMTCYCID

Specifies whether the journal entries to be converted for output are limited to the journal entries made during a specified commit cycle. Only journal entries that contain the commit cycle identifier are converted.

***ALL:** Conversion of journal entries is not limited to entries for a specified commit cycle.

commit-cycle-identifier: Specify the identifier for the commit cycle whose journaled changes are to be converted. Only journaled changes for this commit cycle are to be converted.

OUTFMT

Specifies the format in which the objects are shown.

Note: This parameter is ignored if OUTPUT(*OUTFILE) is specified.

***CHAR:** The entry-specific data portion of the journal entry is shown in character format.

***HEX:** The entry-specific data portion of the journal entry is shown in hexadecimal format.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled printer output, or sent to the database file specified on the OUTFILE parameter. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output. If TOENT(*LAST) is specified on the command, and the last journal entries in the journal are shown, pressing the Page Down key shows any new journal entries added to the journal since the last time the Page Down key was pressed.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILFMT

Specifies the format of the journal entries written to the specified OUTFILE. This parameter can be specified only if the value *OUTFILE is specified on the OUTPUT parameter.

***TYPE1:** The converted journal entries do not include the user profile, the null value indicators, and the system name fields. The format of the information in each journal entry is shown in the following table.

*Table 44 (Page 1 of 2). *TYPE1 Journal Entry Format*

	Length	From	To
Entry Length	5	1	5
Sequence Number	10	6	15
Journal Code	1	16	16
Journal Entry Type	2	17	18
Date	6	19	24
Time	6	25	30
Job Name	10	31	40
User Name	10	41	50
Job Number	6	51	56
Program Name	10	57	66
Object Name	10	67	76
Object Library	10	77	86
Member Name	10	87	96
Count/RRN	10	97	106
Flag	1	107	107
Commit Cycle ID	10	108	117
Reserved	8	118	125
Entry Specific Data	100[1]	126	225

*Table 44 (Page 2 of 2). *TYPE1 Journal Entry Format*

	Length	From	To

Note:

1 This is the field length in the system-supplied database file, QADSPJRN. However, a length of up to 32641 bytes can be specified for this field using the ENTDTALEN parameter.

***TYPE2:** The converted entries include the information returned when OUTFILFMT(*TYPE1) is specified, the name of the user profile for the job that generated the displayed journal entries, and the name of the system on which the output records were generated. The format of the information in each journal entry is shown in the following table.

*Table 45. *TYPE2 Journal Entry Format*

	Length	From	To
Entry Length	5	1	5
Sequence Number	10	6	15
Journal Code	1	16	16
Journal Entry Type	2	17	18
Date	6	19	24
Time	6	25	30
Job Name	10	31	40
User Name	10	41	50
Job Number	6	51	56
Program Name	10	57	66
Object Name	10	67	76
Object Library	10	77	86
Member Name	10	87	96
Count/RRN	10	97	106
Flag	1	107	107
Commit Cycle ID	10	108	117
User Profile	10	118	127
System Name	8	128	135
Reserved	20	136	155
Entry Specific Data	100[1]	156	255

Note:

1 This is the field length in the system-supplied database file, QADSPJR2. However, a length of up to 32611 bytes can be specified for this field using the ENTDTALEN parameter.

***TYPE3:** The converted entries include the information returned when OUTFILFMT(*TYPE2) is specified, and a field for the null value indicators in the converted journal entries. The format of the information in each converted journal entry is shown in Table 46.

*Table 46. OUTFILFMT(*TYPE3) Journal Entry Format*

Field Name	Length	From	To
Entry Length	5	1	5
Sequence Number	10	6	15
Journal Code	1	16	16
Journal Entry Type	2	17	18
Timestamp[1]	26	19	44
Job Name	10	45	54
User Name	10	55	64
Job Number	6	65	70
Program Name	10	71	80
Object Name	10	81	90
Object Library	10	91	100
Member Name	10	101	110
Count/RRN	10	111	120
Flag	1	121	121
Commit Cycle ID	10	122	131
User Profile	10	132	141
System Name	8	142	149
Reserved	20	150	169
Null Value Indicators	52[2]	170	221
Entry-Specific Data	102[3]	222	323

Notes:

1 The date and time of the journal entry is in 26-byte SAA timestamp format instead of separate date and time fields as in *TYPE1 and *TYPE2.

2 This is a 50-character variable-length field in the system-supplied database file QADSPJR3. However, a length of up to 8000 characters can be specified for this field using the NULLINDLEN parameter.

3 This is a 100-character variable-length field in the system-supplied database file QADSPJR3. However, a length of up to 32596 characters can be specified for this field using the ENTDTALEN parameter if it is a fixed-length field and the null value indicators field is also a fixed-length field.

OUTFILE

Specifies the qualified name of the physical database file to which the output is directed. If the output file already exists, the system attempts to use it. Records may replace or be added to the current data in the file member. If no records are written to the database file (because of the specified selection values), and *REPLACE is specified on the OUTMBR parameter, records are cleared from the existing database file. If the output file does not exist, the system creates a database physical file with the name specified in the OUTFILE parameter in the identified library, even if no records are written (because of the specified selection values). If a new file is created and OUTFILFMT(*TYPE1) is specified, the system uses

QADSPJRN in QSYS with a format name of QJORDJE as a model. If a new file is created and OUTFILFMT(*TYPE2) is specified, the system uses QADSPJR2 in QSYS with a format name of QJORDJE2 as a model. If a new file is created and OUTFILFMT(*TYPE3) is specified, the system uses QADSPJR3 in QSYS with a format name of QJORDJE3 as a model. The format supplied defines the normal output fields for the common fields in each journal entry. A member is created for the file with the name specified in the OUTMBR parameter if it does not exist.

The name of the output file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the output file that receives the journal entries.

OUTMBR

Specifies the name of the database file member to which the output is directed. If a member already exists, the system uses the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member with the name of the output file specified on the OUTFILE parameter. If an output file member name is specified, but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member of the file specified by the OUTFILE parameter receives the journal entries. If the OUTFILE is created and *FIRST is specified, the created member has the same name as the created file. If OUTMBR is not specified, *FIRST is assumed.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

ENTDTALEN

Specifies the maximum field length of the entry-specific data portion of the journal entry when the output file is created by the system. This field contains part of the variable portion of the journal entries (such as the after image of records for update journal entries). If OUTPUT(*OUTFILE) is not specified, or if the output file exists, then this parameter is ignored.

The range of allowable values for the combination of ENTDTALEN and NULLINDLEN parameters is given in Table 47.

SINGLE VALUES

For this parameter, a single value or a single list of three elements can be specified.

***OUTFILFMT:** The length of the entry-specific data field in the output file depends on the value specified on the OUTFILFMT parameter. The attributes of this field are the same as those of the entry specific data field in the corresponding system-supplied model output file.

If *TYPE1 or *TYPE2 is specified on the OUTFILFMT parameter the entry specific data field in the output database file will be a fixed length field 100 characters in length.

If OUTFILFMT(*TYPE3) is specified, the entry specific data field in the output file is a variable-length field with a maximum field length of 100 characters and an allocated length of 100 characters. The buffer is 2 bytes longer than the maximum field length to include the length portion of the variable-length field.

***CALC:** The system calculates the length of the entry-specific data field to accommodate the longest entry-specific data of all journal entries in the specified receiver range. In this case, the entry-specific data field is a fixed-length character field. The minimum length of the field is 114 characters. If the length calculated by the system causes the record format length to exceed the maximum record length, a message is sent and the entry-specific data field is truncated. See Table 47 for the possible maximum record lengths of the various data types that can be specified for the ENTDTALEN and NULLINDLEN parameters.

field-length: Specify the number of characters for the field length. In this case, the field is a fixed-length character field. The maximum value prevents the total record length of the output file from being greater than the length allowed for physical database files. A nonzero value is required by the system to ensure that the field exists when the output file is created.

See Table 47 for the allowable field lengths of the various data types that can be specified for the ENTDTALEN and NULLINDLEN parameters.

LIST OF ELEMENTS

To specify a variable-length field, specify *VARLEN for the first element and length values for the last two elements (or let the system use the default values).

Element 1: Variable-Length Field

***VARLEN:** The entry-specific data field is a variable-length field. This value can be specified only when OUTFILFMT(*TYPE3) is specified.

Element 2: Maximum Field Length

The maximum length of the variable-length, entry-specific data field.

***CALC:** The system calculates the maximum length of the entry-specific data field to accommodate the longest entry-specific data of all journal entries in the specified receiver range. The minimum length of this field is 114 bytes. The corresponding buffer length is 116 bytes: 114 bytes of data and 2 bytes for the length portion of the variable-length field.

field-length: Specify the number of characters for the maximum field length. See Table 47 for the allowable field lengths of the various data types that can be specified for the ENTDTALEN and NULLINDLEN parameters.

Element 3: Allocated Length

The allocated length of the variable-length entry-specific data field.

***FLDLEN:** The allocated length is the same as the maximum length of the field.

allocated-length: Specify the number of characters for the allocated length.

If *CALC is specified for the second element and the system-calculated maximum field length is smaller than the specified allocated length, the allocated length is set to the maximum field length. If the maximum field length is specified, the allocated length cannot exceed the maximum field length.

NULLINDLEN

Specifies the length of the null value indicators field when the output file is created by the system. This parameter can be specified only if OUTFILFMT(*TYPE3) is specified.

Null value indicators are present only in journal entries that contain a record image where the corresponding physical file has null capable fields. There is one null value indicator per field in the physical file. Each indicator is one character in length and can be one of two values:

```
'F0'X = Corresponding field is not null.
'F1'X = Corresponding field is null.
```

The range of allowable values for the combination of ENTDTALEN and NULLINDLEN parameters is given in Table 47.

SINGLE VALUES

For this parameter, a single value or a single list of three elements can be specified.

***OUTFILFMT:** The null value indicators field has the same attributes as the corresponding field in the system-supplied model output file QADSPJR3. It is a variable-length character field with a maximum length of 50 characters and an allocated length of 50 characters. The buffer is 2 bytes longer than the maximum field length to include the length portion of the variable-length field.

***CALC:** The system calculates the length of this field to accommodate the journal entry with the maximum number of null value indicators in the specified receiver range. In this case, the null value indicators field is a fixed-length character field. The minimum length of this field is 1 character to ensure that the field exists when the output file is created. If the length calculated by the system causes the record format length to exceed the maximum record length, a message is sent and the null value indicators field is truncated. See Table 47 for the possible maximum record lengths of the various data types that can be specified for the ENTDTALEN and NULLINDLEN parameters.

field-length: Specify the number of characters for the field length. Valid values range from 1 through 8000. A nonzero value is required by the system to ensure that the field exists when the output file is created. In this case, the null value indicators field is a fixed-length character field.

LIST OF ELEMENTS

To specify a variable-length field, specify *VARLEN for the first element and length values for the last two elements (or let the system use the default values).

Element 1: Variable-Length Field

***VARLEN:** The null value indicators field is a variable-length field.

Element 2: Maximum Field Length

Specifies the maximum length of the variable-length null value indicators field.

***CALC:** The system calculates the length of this field to accommodate the journal entry with the maximum number of null value indicators in the specified receiver range. To ensure that this field exists in the output file that is created, the minimum length of this field is 1 byte. The corresponding buffer length is 3 bytes: 1 byte of data and 2 bytes for the length portion of the variable-length field.

field-length: Specify the maximum length of the null value indicators field. Valid values range from 1 through 8000 characters.

Element 3: Allocated Length

Specifies the allocated length of the variable-length null value indicators field.

***FLDLEN:** The allocated length is the same as the maximum length of the field.

If *CALC is specified for Element 2 and the system-calculated maximum length is smaller than the specified allocated length, the allocated length is set to the maximum field length. If the maximum field length is specified, the allocated length cannot exceed the maximum field length.

allocated-length: Specify the number of characters for the allocated length.

Table 47. Range of Values for ENTDTALEN and NULLINDLEN Parameters

Output File Format	Entry-Specific Data Format	Null Value Indicators Format	Entry Data Length	Null Line Indicator Length	Maximum Record Length
*TYPE1	Fixed	Not applicable	1 through 32641	Not applicable	32766
*TYPE2	Fixed	Not applicable	1 through 32611	Not applicable	32766
*TYPE3	Fixed	Fixed	1 through 32596	1 through 8000	32766
*TYPE3	Fixed	Variable	1 through 32570	1 through 8000	32740
*TYPE3	Variable	Fixed	1 through 32570	1 through 8000	32740
*TYPE3	Variable	Variable	1 through 32568	1 through 8000	32738

Note: When the file contains variable length fields, the maximum record length does not include the 2 bytes per variable length field.

Examples

Example 1: Converting Journal Entries for Display

```
DSPJRN   JRN(MYLIB/JRNLA)
```

When issued at a work station, this command converts and shows the first journal entries in the current journal receiver attached to the journal JRNLA in library MYLIB. Subsequent entries are shown by pressing the Page Down key. When entered from a batch job, the above command prints all converted journal entries in the journal receiver currently attached to journal JRNLA in library MYLIB with the job's spooled printer output. The entry-specific data portion of the journal entries are shown in character format.

Example 2: Converting Journal Entries to an Output File

```
DSPJRN   JRN(MYLIB/JRNLA)  FILE((LIB1/A MBR3)  (LIB1/C)
  (LIB2/*ALL *ALL))  RCVRNG((RCVLIB/RCV27
  RCVLIB/RCV30))  FROMENT(4736)  ENTTYP(UP DL)
  JOB(000666/QPGMR/WORKST01)  PGM(TSTPGMA)
  ENTDTALEN(280)  OUTPUT(*OUTFILE)
  OUTFILE(MYLIB/JRNENTFIL1)
```

This command converts selected journal entries in the journal receiver chain (from receiver RCV27 in library RCVLIB to receiver RCV30 in library RCVLIB) that are attached to JRNLA in library MYLIB and places them in the first member of the database file JRNENTFIL1 in library MYLIB. If the database file does not exist, it is created with a format of QJORDJE. The last field in the format is 280 bytes in length. The journal entry that has a sequence number of 4736 is the first entry written to the output file. Only entries for record updates and deletes made by program TSTPGMA in the job 000666/QPGMR/WORKST01 to member MBR3 of file A in library LIB1, the first member of file C in library LIB1, and all members of all files in library LIB2 are written to the output file.

Example 3: Converting Journal Entries

```
DSPJRN   JRN(SS/J)  FILE(SS1/PF)
  RCVRNG((SS/R1  *CURRENT))  JRNCDE(F)  USRPRF(MAC7)
  OUTFILE(FMTLIB/ENTFILE)  OUTFILFMT(*TYPE2)
```

This command converts selected journal entries in the journal receiver range. The range begins with journal receiver R1 in library SS1 and ends with the journal receiver currently attached to journal J in library SS. The entries are placed in the first member of the database file ENTFILE in library FMTLIB. If the file does not exist, it is created with the QJORDJE2 format. The last field of the format is 100 bytes in length. Only entries with file-level information made by the user MAC7 to the first member of file PF in library SS1 are written to the output file.

Example 4: Converting Journal Entries

```
DSPJRN   JRN(LIBPROD/PRODJRN)  FILE(APPLIB/PFILE)
  OUTFILFMT(*TYPE3)  OUTFILE(JRNLIB/ENTFILE)
  ENTDTALEN(*VARLEN 500 100)  NULLINDLEN(25)
```

This command converts selected journal entries from the journal receiver currently attached to journal PRODJRN in library LIBPROD. The entries are placed in the first member of the database file ENTFILE in library JRNLIB. If the file does not exist, it is created with the QJORDJE3 format. The entry-specific data field is a variable-length field with a maximum field length of 500 characters and an allocated length of 100 characters. The null value indicators field is a fixed-length field of 25 characters. Only entries with file-level information for the first member of file PFILE in library APPLIB are written to the output file.

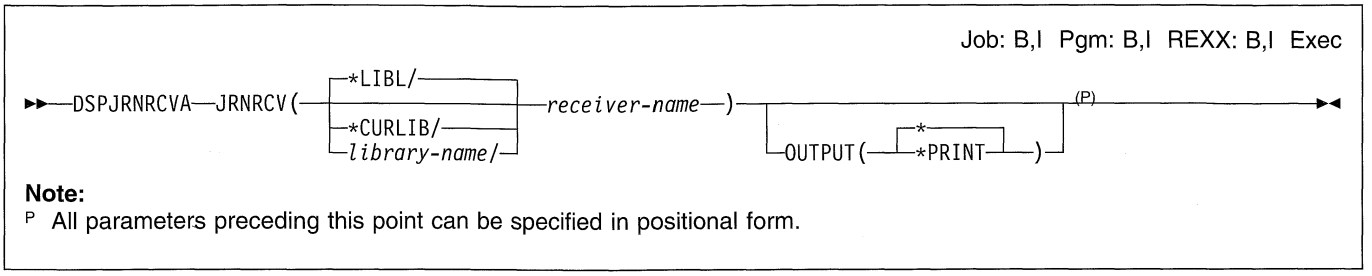
Example 5: Converting Journal Entries

```
DSPJRN   JRN(JRNLIB/JRNA)  FILE(FILLIB/FILEA)
  FROMENT(4736)  ENTTYP(UP DL)
  JOB(000666/QPGMR/WORKST01)  PGM(TSTPGMA)
  ENTDTALEN(280)  OUTPUT(*OUTFILE)
  OUTFILFMT(*TYPE3)  OUTFILE(ENTLIB/ENTFILE)
  ENTDTALEN(*CALC)  NULLINDLEN(*VARLEN *CALC *FLDLEN)
```

This command converts journal entries with file-level information for the first member of file FILEA in library FILLIB from the journal receiver currently attached to journal JRNA in library JRNLIB. The entries are written to the first member of database file ENTFILE in library ENTLIB. If the file does not exist, it is created with the QJORDJE3 format. The system calculates the length of the fixed-length, entry-specific data

field. The system also calculates the length of the variable-length null value indicators field. The allocated length of this field is the same as the maximum field length.

DSPJRNRCVA (Display Journal Receiver Attributes) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Journal Receiver Attributes (DSPJRNRCVA) command shows the creation and current operational attributes of a journal receiver, including the name of the journal the receiver is now attached to, or was last attached to (if the receiver is not currently attached), and the names of the journal receivers that were attached before and after the specified receiver. If a dual journal receiver was used, the name of the dual journal receiver that was attached at the same time as the specified receiver is shown. The information also includes the number of journal entries contained in the journal receiver, the length of the longest entry-specific data, the maximum number of null value indicators in a journal entry, the journal sequence numbers of the first and last entries on the journal receiver, and the date and time that the receiver was attached and detached.

From the display supplied by the command, an option can be selected to display the previous receiver, the next receiver, or the dual receiver.

Required Parameter

JRNRCV

Specifies the qualified name of the journal receiver that is to be displayed.

The name of the journal receiver can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

receiver-name: Specify the name of the journal receiver to be displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

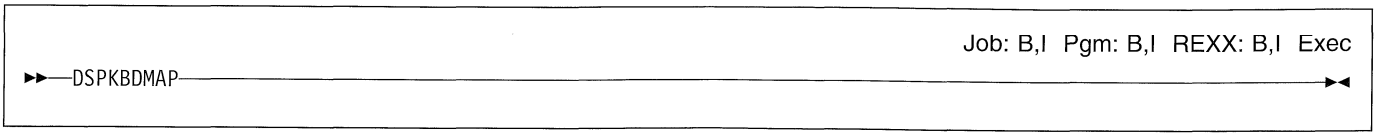
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPJRNRCVA   JRNRCV(MYLIB/JRNRCLA) OUTPUT(*PRINT)
```

This command prints the current operational attribute information of journal receiver JRNRCLA in library MYLIB with the job's spooled printer output.

DSPKBDMAP (Display Keyboard Map) Command



Purpose

The Display Keyboard Map (DSPKBDMAP) command shows the current F-to-PF map for the 3270 work station device from which the command was sent. More information on user-assignable keyboard mapping is in the *New User's Guide*.

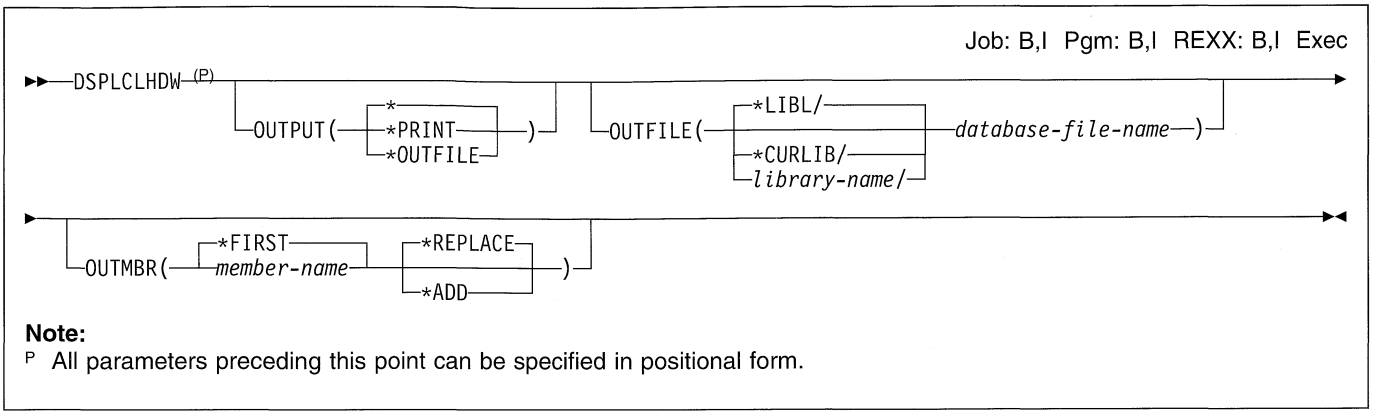
There are no parameters for this command.

Example

```
DSPKBDMAP
```

The DSPKBDMAP command uses the 3270 help screen to display the active PF key mapping for the device the command was entered on.

DSPLCLHDW (Display Local Hardware) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The DSPLCLHDW (Display Local Hardware) command allows the user to display, print, or write the local hardware resources to an output file.

Note: The output for this command may not match the output for the Print Device Address (PRTDEVADR) command.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file to which the output of the command is directed. If the OUTFILE does not exist, a database file is created in the specified library.

If a new file is created, the system uses the physical file QARZHWOF in library QSYS as a model file. This file has a record format name of QARZHDWD. Field level information for this file can be obtained using the Display File Field Description (DSPFFD) command and specifying QARZHWOF as the file name and QSYS as the library. If this function creates the file, the text reads *output file for DSPLCLHDW*. The authority for users is *EXCLUDE, except for users who have specific authority, group authority, or authority from an authorization list. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file to which the output of the command is directed.

OUTMBR

Specifies the name of the database file member to which the output is directed. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the user can add records to the end of the existing member or clear the existing member and add the records at the top of the member.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

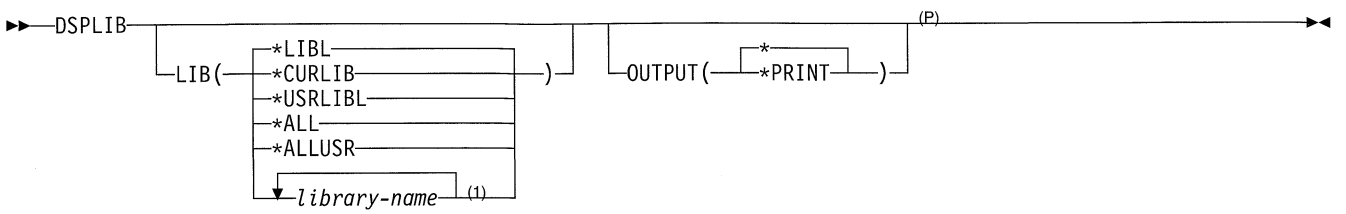
Example

```
DSPLCLHDW  OUTPUT(*OUTFILE)  OUTFILE(*CURLIB/NAMES)
```

DSPLCLHDW

This command sends the output from the command to the first member of the file NAMES in the current library. The output replaces the information in the member.

DSPLIB (Display Library) Command



Notes:
1 A maximum of 15 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Library (DSPLIB) command displays the contents of one or more specified libraries; that is, it displays a list of the names and types of all objects contained in each library, regardless of the authorization on each object. The command can also be used to display a list of libraries from which individual libraries may be selected for a display of their objects. From the display of a library's contents, options can be specified to request displays that show more specific information about the objects in the library.

Restriction: The user must have *READ authority for each library specified to display its contents.

Notes:

1. For printed output, the total-size field of the library includes the size of the objects in the library plus the size of the library object itself.
2. If *USRLIBL, *LIBL, *ALL, or *ALLUSR is specified on the LIB parameter and output is printed, the contents of the libraries is printed. If output is displayed for one of these names, a list of the libraries is displayed from which individual libraries may be selected for further display.

Optional Parameters

LIB

Specifies the names of the libraries that are shown or printed.

The name of the library can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Example

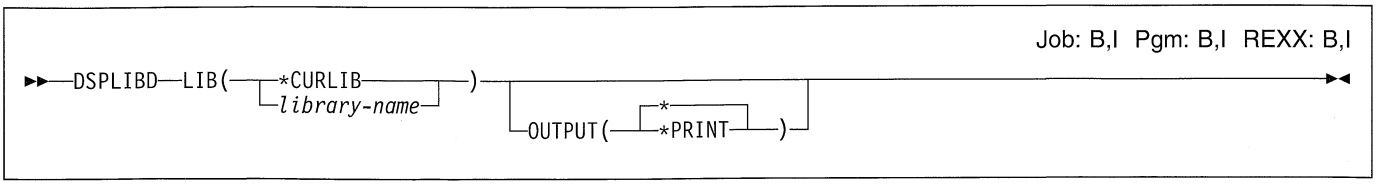
```
DSPLIB   LIB(QGPL)
```

The names, types, and basic descriptions of all the objects located in the QGPL library are either shown on the work station from which the command was submitted, or printed on the system printer if the command was run in a batch job.

Additional Considerations

If more than one library is to be displayed, they are displayed one at a time. The display lists the name and type of each library, the number of objects in the library, and, for printed output, the size of the combined objects. This size also includes the size of the library object in QSYS. If this value is followed by a plus (+) sign, objects in the library are not currently available, are damaged, being locked, or are not authorized. The plus sign indicates that the actual total of all objects is greater than the value displayed.

DSPLIBD (Display Library Description) Command



Purpose

The Display Library Description (DSPLIBD) command is used to display the description of a library. The description includes the type of library, auxiliary storage pool (ASP), create authorities, the create object auditing value, and a text description.

Restriction: The user must have authority to the library that is to be displayed.

Required Parameter

LIB

Specifies the name of the library for which information is being displayed.

The name of the library can be qualified by one of the following library values:

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

Optional Parameter

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

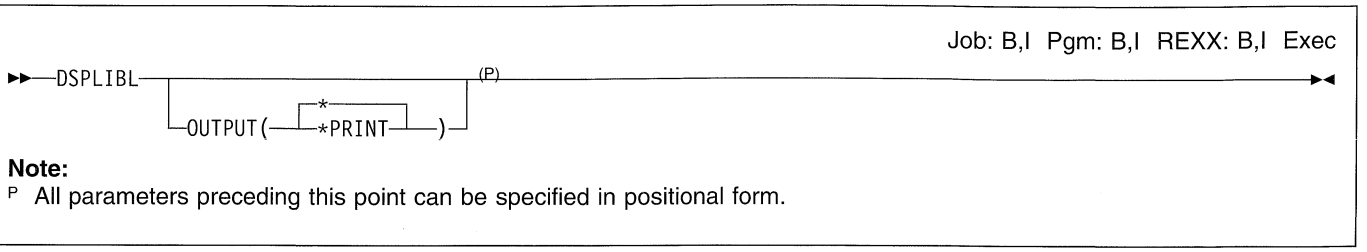
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPLIBD   LIB(QGPL)
```

This command displays type, ASP, create authority value, and text description for library QGPL.

DSPLIBL (Display Library List) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Library List (DSPLIBL) command displays the system portion of the library list, the current library entry of the job's library list, product libraries, and the user portion of the library list.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

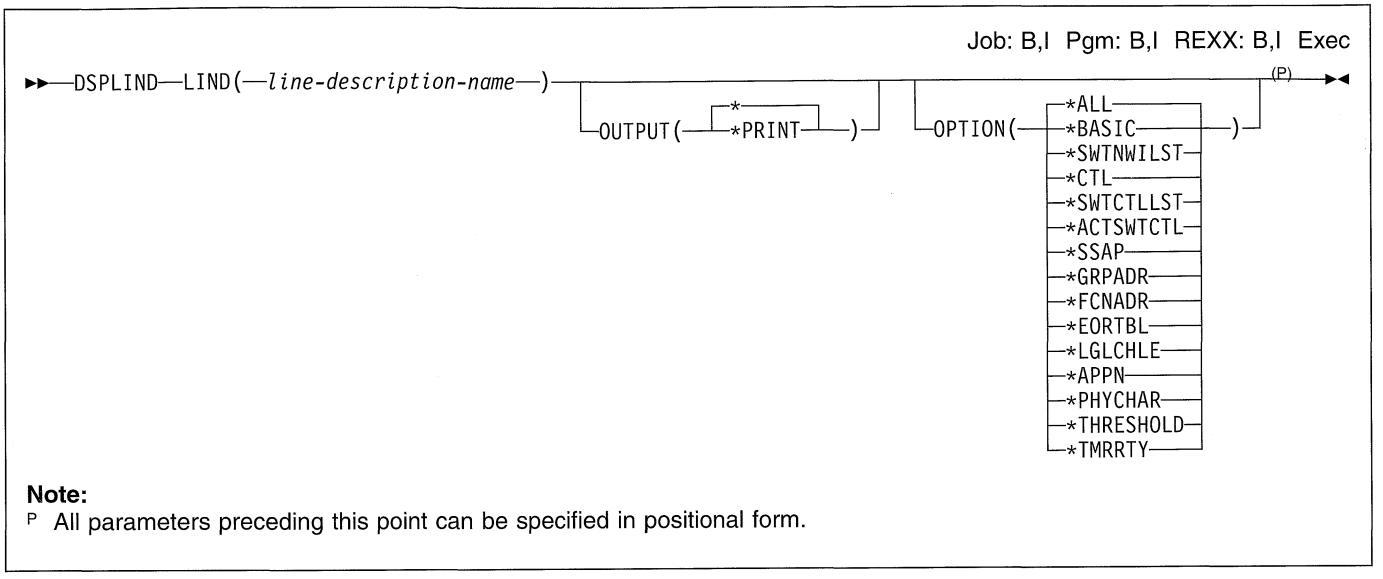
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPLIBL
```

The names, types, and text of the libraries that are in the job's library list are displayed on the user's work station from which the command was submitted, or the list is printed on the system printer, if the command was part of the batch input stream.

DSPLIND (Display Line Description) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Line Description (DSPLIND) command displays a line description. Output is directed to a display or a spooled printer file as indicated by the OUTPUT parameter and job type.

Required Parameters

LIND

Specifies the name of the line description to be displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

OPTION

Specifies the panels to be displayed.

***ALL:** All panels that apply to this specific line type are displayed.

***BASIC:** The basic displays that apply to this line type are shown, but additional information displays are not shown.

***SWTNWILST:** The switched network interface list panels that apply to this line are displayed.

***CTL:** The attached nonswitched controller list panels for this line are displayed.

***SWTCTLLST:** The switched controller list panels that apply to this line are displayed.

***ACTSWTCTL:** The active switched controller list panels that apply to this line are displayed.

***SSAP:** The source service access point panels for this token-ring or Ethernet line are displayed.

***GRPADR:** The group address panels for this Ethernet line are displayed.

***FCNADR:** The functional address panels for this token-ring are displayed.

***EORTBL:** The end-of-record table panels for this asynchronous line are displayed.

***LGLCHLE:** The logical channel entries for this X.25 line are displayed.

***APPN:** The advanced peer-to-peer networking (APPN) values for this line are displayed.

***PHYCHAR:** The information concerning the physical characteristics of an asynchronous line is displayed.

***THRESHOLD:** The information concerning the threshold values of an IDLC line is displayed.

***TMRRTY:** The timer and retry values for this line are displayed.

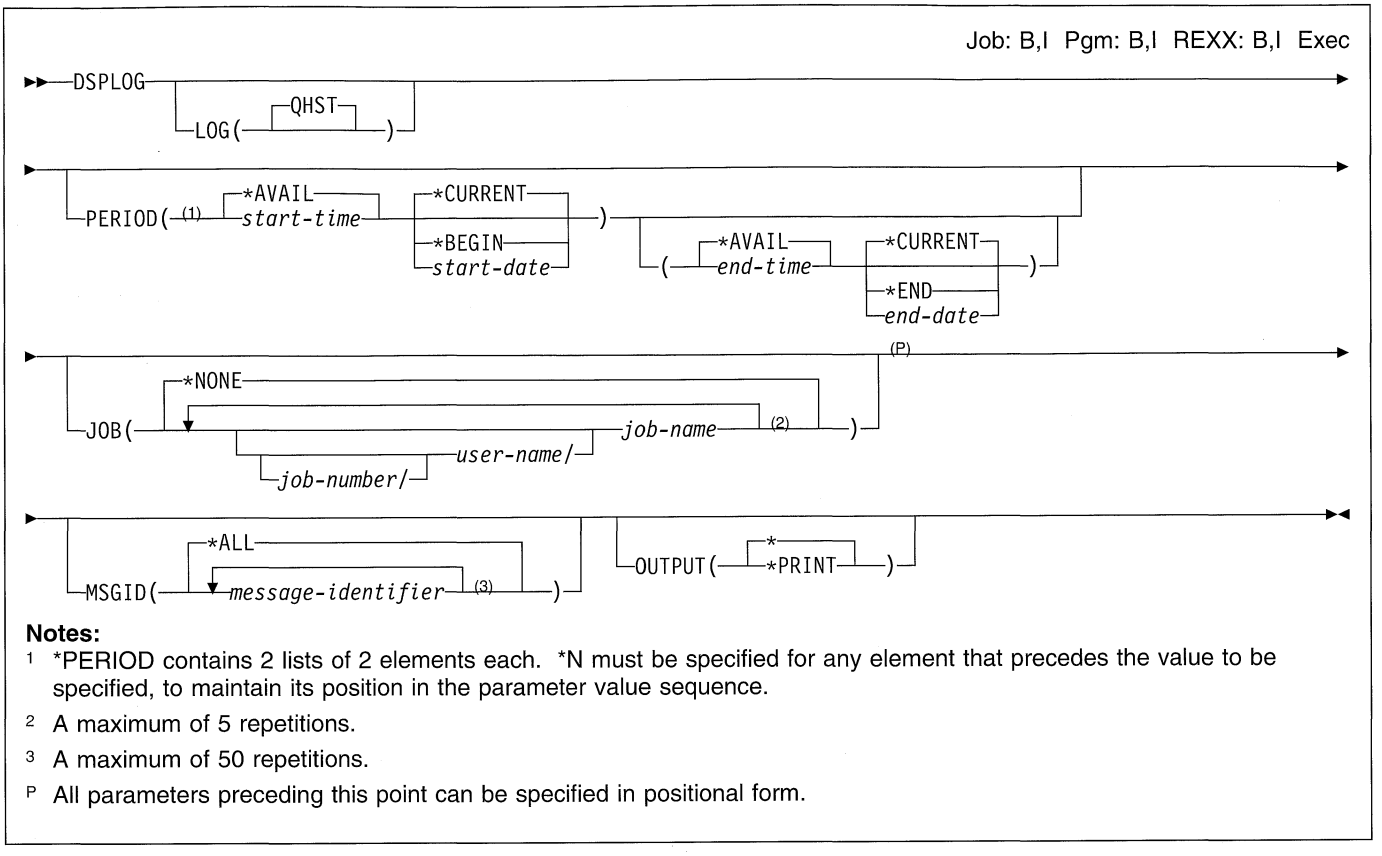
Example

```
DSPLIND LIND(LINE21)
```

This command displays information about the line description named LINE21. The information is shown on the work

station display from which the command was submitted. If the command is entered from a batch job, the output from the display is printed with the job's spooled output.

DSPLOG (Display Log) Command



Notes:

1 *PERIOD contains 2 lists of 2 elements each. *N must be specified for any element that precedes the value to be specified, to maintain its position in the parameter value sequence.

2 A maximum of 5 repetitions.

3 A maximum of 50 repetitions.

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Log (DSPLOG) command shows the system history log (QHST). The history log contains information about the operation of the system and the current system status.

The display contains the messages sent to the log, the date and time the message was sent, and the name of the job that sent it.

Optional Parameters

LOG

Specifies which of the system logs is shown.

QHST: The system history log QHST is shown.

PERIOD

Specifies the time period covered by the logged message data being shown. The values that can be coded for this parameter are specified as two lists with two elements in each list. If PERIOD is not specified, the following values are assumed:

```
PERIOD((*AVAIL *CURRENT) (*AVAIL *CURRENT))
```

Element 1: Starting Time

One of the following is used to specify the starting time at which or after which the data must have been logged. Entries logged before the specified time and date are not shown.

***AVAIL:** The logged data that is available for the specified starting *date* is shown.

start-time: Specify the starting *time* on the specified starting date that indicates the logged data to be shown. The time is specified in 24-hour format with or without a time separator as follows:

- With a time separator, specify a string of 5 or 8 digits where the time separator separates the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command fails.
- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where **hh** = hours, **mm** = minutes, and **ss** = seconds. Valid values for **hh** range from 00 through 23. Valid values for **mm** and **ss** range from 00 through 59.

Element 2: Starting Date

One of the following is used to specify the starting date on which or after which the data must have been logged. Entries logged before the specified date are not shown.

***CURRENT:** The logged data for the current day and between the specified starting and ending times (if specified) is shown.

***BEGIN:** The logged data from the beginning of the log is shown.

start-date: Specify the date shown. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

Element 3: Ending Time

One of the following is used to specify the ending time before which, or at which, the data must have been logged.

***AVAIL:** The logged data that is available for the specified ending *date* is shown. The time can be entered as either 4 or 6 digits (hhmm or hhmmss) where hh = hours, mm = minutes, and ss = seconds. If time separators are used to separate the time values, the character string must be enclosed in apostrophes (`'hh:mm:ss'`). If a time separator other than the one specified for the job is used, the command fails.

Note: The values specified for the ending date and time are ignored if the output is shown. That is, all data in the log that was logged on or after the specified starting date and time can be shown, regardless of the ending date and time specified.

end-time: Specify the ending time for the specified ending date that determines the logged data that is printed.

Element 4: Ending Date

One of the following is used to specify the ending date before which or on which the data must have been logged.

***CURRENT:** The current day is the last day for which logged data is shown.

***END:** The last day on which data was logged is shown. If PERIOD(*END) is specified, a time value other than *AVAIL for end-time is ignored.

end-date: Specify the ending date for which logged data is to be printed. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

Note: If no output is received after you run the DSPLOG command with *PRINT specified, the dates of some message data may be out of sequence. To print the data in this case, specify: `PERIOD((*AVAIL *BEGIN)(*AVAIL *END))`.

JOB

Specifies the names of the jobs (if any) whose messages in the system log are shown. If a job name is not qualified, all jobs by that name in the log will have their messages displayed.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
*NONE
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***NONE:** No job name is used to indicate which messages are shown.

job-name: Specify the names of up to five jobs that are to have their logged messages shown.

user-name: Specify the names of the users of the jobs that are to have their messages displayed.

job-number: Specify the numbers of the jobs that are to have their logged messages displayed.

MSGID

Specifies up to fifty message identifiers (if any) of the logged messages that are shown. These messages are shown only if they were logged in the period of time specified in this command and in the jobs specified, if job names were given in this command.

***ALL:** All logged messages, regardless of their identifiers, are shown if they meet the previous parameters' specifications.

message-identifier: Specify the identifiers of up to fifty messages that are shown. (Refer to the description of the MSGID parameter in the Add Message Description (ADDMSGD) command description for more information on message identifiers.)

To display specific generic types of messages, specify the 3-character code that identifies the message file followed by all zeros. For example, CPF0000 specifies that all CPF messages that meet the specifications of the previous parameters are shown. If an identifier is specified as `pppnn00`, any message beginning with the specified five characters (`pppnn`) can be shown.

OUTPUT

Specifies whether the output from the command is shown at the requesting display station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Logged Messages for Current Date

```
DSPLOG   LOG(QHST)
```

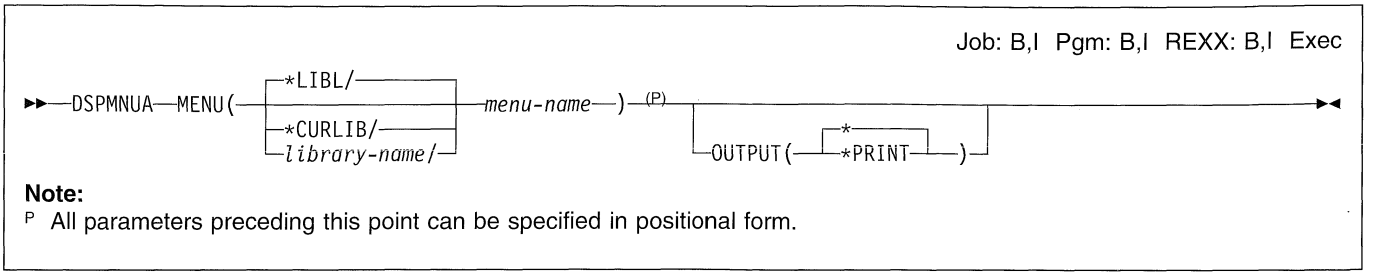
This command shows all the logged messages (and their associated data) that are available in the history log for the current date.

Example 2: Displaying Logged Messages for September 1988

```
DSPLOG   JOB(MYJOB)  PERIOD((*AVAIL 090188)
  (*AVAIL 093088))  MSGID(CPF0000)
```

This command displays all CPF messages, in the history log for MYJOB, that were logged during September 1988.

DSPMNUA (Display Menu Attributes) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Menu Attributes (DSPMNUA) command shows the following:

- Menu type
- Command line length
- Display function keys on the menu
- Program name and library for program menus
- Display file name and library for free-format menus
- Message file name and library for free-format menus
- The type of command line to be shown for display file menus:
  - *LONG
  - *SHORT
  - *NONE (an option line is used)
- Current library
- Product library
- Descriptive text

Restriction: The user must have *USE authority for the menu whose attributes are being shown and *USE authority for the library where the menu is located.

Required Parameters

MENU

Specifies the qualified name of the menu whose attributes are displayed.

The name of the menu can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

menu-name Specify the name of the menu whose attributes are displayed.

Optional Parameters

OUTPUT

Specifies whether the output is directed to the display station screen or to a printer. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

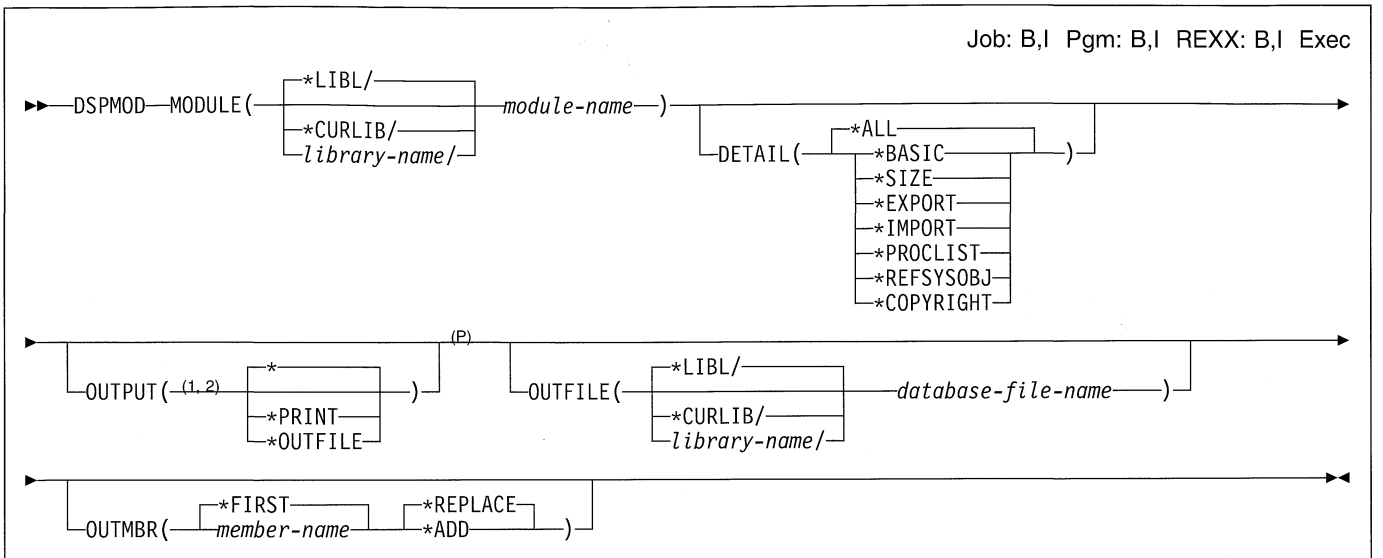
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPMNUA MENU(PAYROLL)
```

This command shows the attributes of a menu named PAYROLL at the display station (if the command is running interactively). The menu is found by searching the library list (*LIBL default value).

DSPMOD (Display Module) Command



Notes:

1 If OUTPUT(*OUTFILE) is specified, a file name must be specified for the OUTFILE parameter.

2 If OUTPUT(*OUTFILE) is specified, a detail other than *ALL must be specified on the DETAIL parameter and only one DETAIL may be specified.

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Module (DSPMOD) command displays information about a module. The display includes information about the compiler, the source from which the module was created, the processing attributes of the module, and the size of the module.

Restriction: To use this command, you must have *USE authority to the module being displayed and *READ authority to the library in which the module is stored.

Required Parameters

MODULE

Specifies the qualified name of the module for which information is displayed.

The name of the module can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

module-name: Specify the name of the module for which information is displayed.

Optional Parameters

DETAIL

Specifies the set of information to be displayed for the module. More than one value can be specified, but *ALL must be specified as a single value. DETAIL(*ALL) is valid only when information is displayed or printed.

***ALL:** All the information applicable to the specified module is displayed or printed.

***BASIC:** The basic module information, module compatibility section, and SQL information is shown.

***SIZE:** The size and size limits for this module are shown.

***EXPORT:** The symbols and type defined in this module that are exported to other modules are shown.

***IMPORT:** The symbols and type defined externally to this module are shown.

***PROCLIST:** A list of the procedure names with their type is shown.

***REFSYSOBJ:** A list of the system objects referred to by this module when the module is bound into a bound program or service program is shown.

***COPYRIGHT:** Copyright information for this module is shown.

OUTPUT

Specifies whether the output from this command is displayed, printed, or directed to a database file. More

information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

Note: DETAIL(*ALL) is not valid when OUTPUT(*OUTFILE) is specified.

OUTFILE

Specifies the qualified name of the database file to which the output of the display is directed. If the OUTFILE does not exist, this command creates a database file in the specified library. If the file is created, the text is "Output file for DSPMOD". The public authority of the file is the same as the create authority specified for the library in which the file is created. The record format of the output file will be the same as that used in the IBM-supplied database file.

The IBM-supplied database files and their record formats are as follows:

Detail	File	Record Format
*BASIC	QABNDMBA	QBNDMBAS
*SIZE	QABNDMSI	QBNDMSIZ
*EXPORT	QABNDMEX	QBNDMEXP
*IMPORT	QABNDMIM	QBNDMIMP
*PROCLIST	QABNDMPR	QBNDMPRO
*REFSYSOBJ	QABNDMRE	QBNDMREF
*COPYRIGHT	QABNDMCO	QBNDMCOP

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display.

OUTMBR

Specifies the name of the database file member to which the output is directed. If a member already exists, the system uses the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member with the name of the output file specified on the OUTFILE parameter. If an output file member name is specified, but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

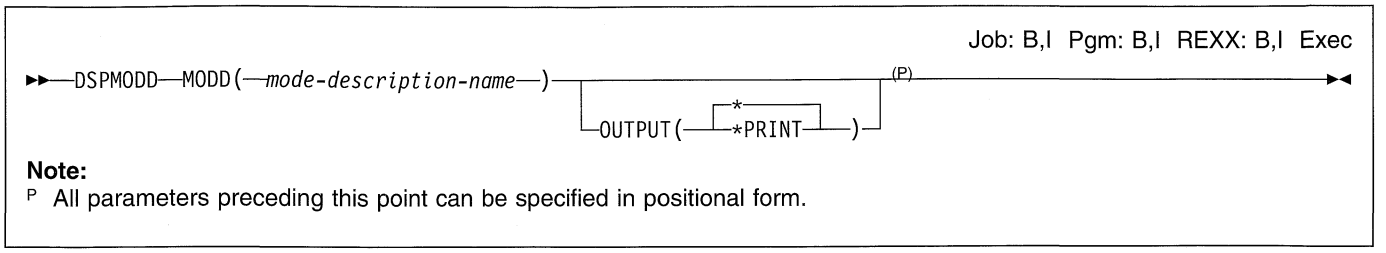
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPMOD   MODULE(MYMOD)
```

This command displays module object MYMOD from the library list.

DSPMODD (Display Mode Description) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Mode Description (DSPMODD) command displays a mode description for advanced-program-to-program communications (APPC) devices. Output is directed to a display or a spooled printer file as indicated on the OUTPUT parameter and job type.

Required Parameters

MODD
Specifies the name of the mode description.

Optional Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

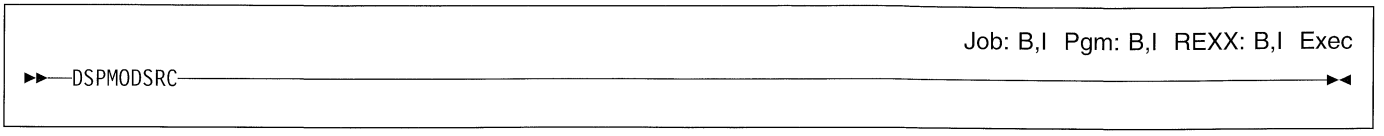
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPMODD  MODD(CICS2)
```

This command displays information about mode description CICS2. If the command is entered from a batch job, the output from the display is printed with the job's spooled output.

DSPMODSRC (Display Module Source) Command



Purpose

The Display Module Source (DSPMODSRC) command allows the user to show the source debug displays for debugging bound programs.

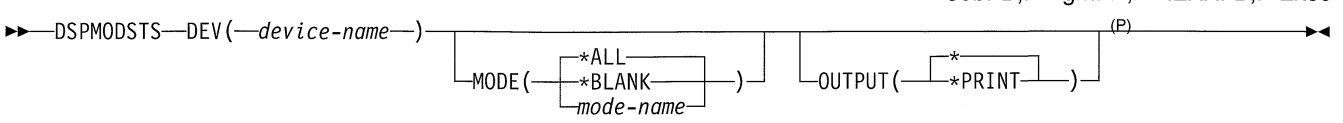
There are no parameters for this command.

Example

```
DSPMODSRC
```

This command shows the source debug displays for programs that are to be debugged.

DSPMODSTS (Display Mode Status) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Mode Status (DSPMODSTS) command displays the status of all modes associated with an APPC device description. The display shows the status of the APPC device, the current number of source, target, and detached conversations in use, and configured and operational session maximum values.

Restriction: This command is valid only for APPC device descriptions.

Required Parameters

DEV

Specifies the name of the APPC device description.

Optional Parameters

MODE

Specifies the name of the mode whose status is being shown.

***ALL:** All the modes used by the specified device are shown.

***BLANK:** The mode name consisting of 8 blank characters is used.

mode-name: Specify the name (up to 8 characters) of the mode whose status is being shown for the specified device. The mode name may contain the characters A through Z, 0 through 9, $, #, and @.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

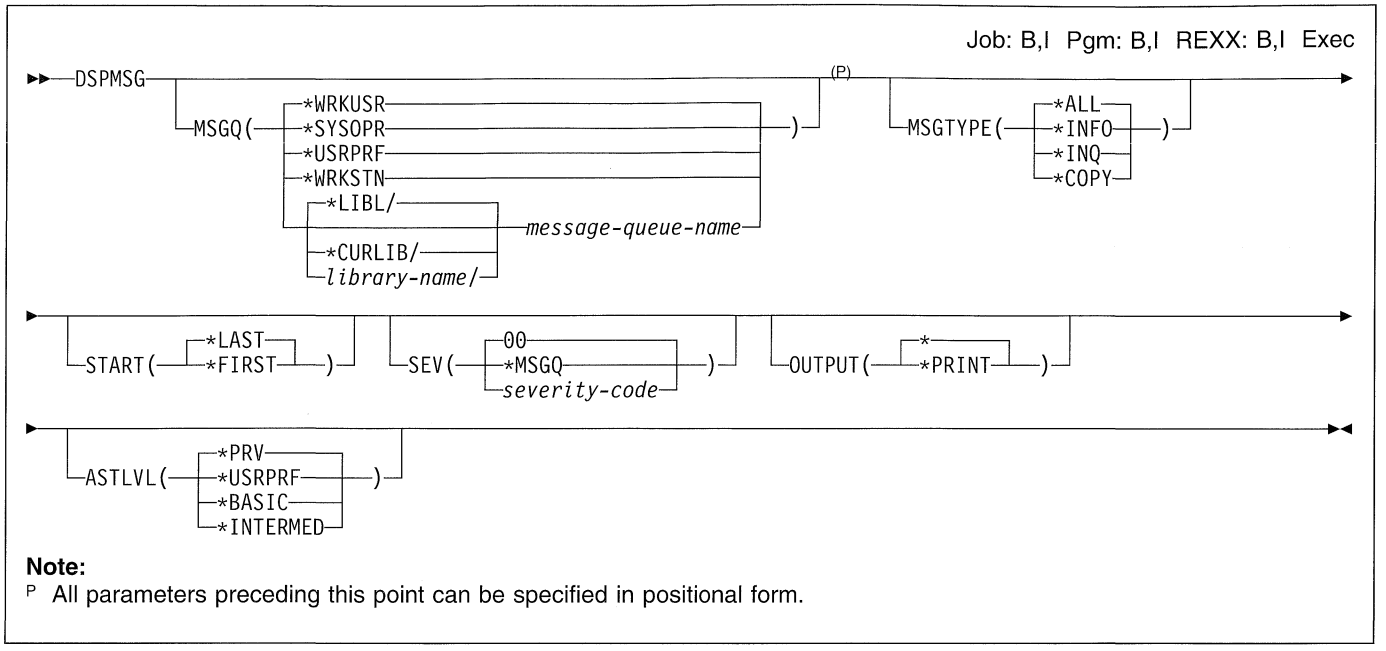
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPMODSTS  DEV(MINN2)  MODE(CICS2)
```

This command shows the status of the mode CICS2 used by the device MINN2. If the command is entered from a batch job, the output from the display is printed with the job's spooled output.

DSPMSG (Display Messages) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Messages (DSPMSG) command is used by the display station user to show the messages received at the specified message queue. If the message queue is not allocated to the job in which this command is entered or to any other job, it is implicitly allocated by this command for the duration of the command. When the messages are shown, options are also shown that allow the user to either remove one or more messages from the queue or to enter a reply to each inquiry message.

Note: Refer to the section entitled "Handling Messages" in the *System Operator's Guide*, SC41-8082 for a description of how to print a single message description or a group of message descriptions.

Restrictions:

1. DSPMSG cannot be specified when another job has already allocated the message queue by specifying the ALCOBJ command operating in the *EXCL lock state.
2. The DSPMSG command allows the user to see, but not remove, messages for the job when the message queue is in the *BREAK or *NOTIFY mode for another job.
3. When the message queue is in the *BREAK mode for another job, the break handling program for the message queue determines whether the user is allowed to respond to inquiry messages. If PGM(*DSPMSG) is specified, the user can reply to inquiry messages, but if PGM(user-program) is specified, no reply to inquiry messages is allowed.
4. If multiple jobs show the same inquiry message, and all jobs reply to the message, only the first reply to the message is valid. Subsequent replies from other jobs cause error messages to be issued.

Optional Parameters

MSGQ

Specifies the qualified name of the message queue from which messages are shown.

Note: The values *WRKUSR, *SYSOPR, and *WRKSTN can be used only when the user is in an interactive job.

***WRKUSR:** Messages are shown from the work station's message queue. After exiting this display, the message queue for the job's user profile is shown. If there are no messages on the work station message queue, only the user profile message queue is shown.

***SYSOPR:** Messages from the system operator message queue (QSYSOPR) are shown.

***USRPRF:** Messages from the current user profile message queue are displayed.

***WRKSTN:** Messages are shown from the work station's own message queue.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

message-queue-name: Specify the name of the message queue from which messages are shown. The

system operator can specify QSYSOPR to show messages on the system operator message queue.

MSGTYPE
Specifies the type of messages in the message queue that are shown.

***ALL:** All messages that are in the message queue are shown.

***INFO:** Only informational messages (those not requiring a reply) are shown.

***INQ:** Only inquiry messages (those requiring a reply) are shown.

***COPY:** Copies of the inquiry messages that were sent to other message queues and still require replies are shown.

START
Specifies whether the newest messages or the oldest messages in the message queue (of those specified by the MSGTYPE and SEV parameters) are shown first. The display station user can roll the display up or down to see other messages if the message list occupies more than one display screen. This parameter is ignored if the output is directed to a spooled printer file.

***LAST:** The last (newest) message on the message queue is shown on the bottom line of the display. The display station user can press the Roll Down key to show older messages.

***FIRST:** The first (oldest) message on the message queue is shown on the top line of the display. The display station user can press the Roll Up key to show newer messages.

SEV
Specifies the severity code of the message. The severity code indicates the severity level of the condition that causes the message to be sent.

00: All messages in the specified message queue are shown.

***MSGQ:** Messages with a severity code greater than or equal to the severity code specified for the message queue are shown.

severity-code: Specify a value, ranging from 00 through 99, that specifies the minimum severity code that a message must have to still be shown. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

OUTPUT
Specifies whether the output from the command is shown at the requesting display station or listed with the job's spooled output on a printer. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output. Immediate messages can be 512 bytes long.

***PRINT:** The output is printed with the job's spooled output.

Note: Immediate and predefined messages are shown truncated to 105 characters when printed.

ASTLVL
Specifies the user interface to display.

***PRV:** The previous user interface used is displayed.

***USRPRF:** The user interface specified in the current user profile is used.

***BASIC:** The Operational Assistant user interface is used. This user interface separates messages into two categories: (1) messages needing a reply and (2) messages not needing a reply. New messages are shown at the top of each message list.

***INTERMED:** The system user interface is used.

Note: The DSPMSG command documentation assumes the *INTERMED user interface is shown. If the *BASIC user interface is shown, see the documentation for the Work with Messages (WRKMSG) command.

Examples

Example 1: Displaying Messages From Work Station Message Queue and User Profile Message Queue

```
DSPMSG
```

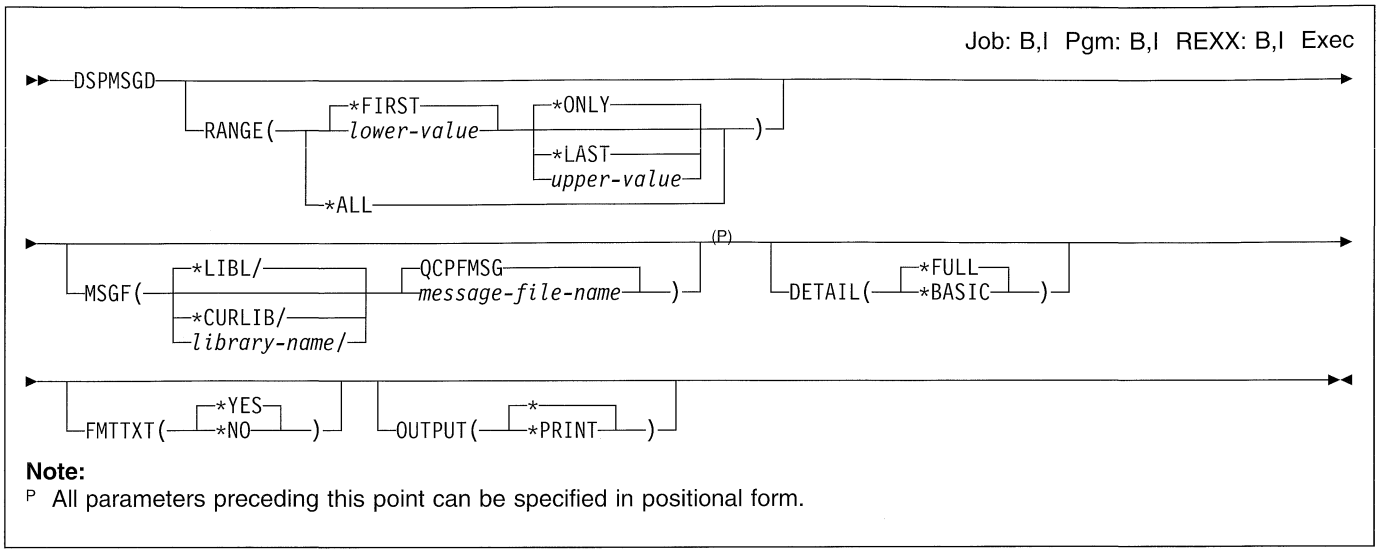
This command displays messages in the requester's work station message queue followed by the user profile message queue. If there are no messages in the work station queue, the user profile message queue is shown immediately.

Example 2: Displaying Informational Messages

```
DSPMSG   MSGQ(SMITH)  MSGTYPE(*INFO)
```

This command displays, at the requester's work station, any informational messages in the message queue named SMITH.

DSPMSGD (Display Message Descriptions) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Message Descriptions (DSPMSGD) command shows detailed information about the messages in a message file. The descriptions of specific messages or a range of messages in one message file can be specified by their identifiers, or all messages in one message file can be specified.

Optional Parameters

RANGE

Specifies the range of message identifiers in the specified message file for which message descriptions are shown.

All message descriptions in the message file specified by the MSGF parameter are shown or printed when *ALL is specified on this parameter.

If the starting message identifier is not specified, the message descriptions begin with the first message in the specified message file. If the ending message identifier is not specified, the default value, *ONLY, is used, and only the description of the starting message identifier is shown or printed.

Note: The ending message identifier is ignored when DETAIL(*BASIC) and OUTPUT(*) (interactive display) are specified.

Element 1: Starting Message Identifier:

The first of the two values in the RANGE parameter specifies the message identifier of the first message description to show or print.

***FIRST:** The first message in the file specified by the MSGF parameter is the first one whose description is shown or printed.

lower-value: Specify the 7-character identifier of the first message description to show or print.

Element 2: Ending Message Identifier:

The second of the two values in the RANGE parameter specifies the message identifier for the ending (last) message description to print. This value is ignored if OUTPUT(*) and DETAIL(*BASIC) is specified.

***ONLY:** Only the description with the message identifier specified as a starting message identifier is shown or printed if *FULL is specified on the DETAIL parameter.

***LAST:** The last message in the file is the last one whose description is shown or printed.

upper-value: Specify the 7-character identifier of the last message description to print.

Other Single Values:

***ALL:** All message descriptions in the message file specified by the MSGF parameter are shown or printed.

MSGF

Specifies the qualified name of the message file from which the message descriptions are taken.

| The name of the message file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QCPFMSG: Message descriptions are taken from the system message file, QCPFMSG.

message-file-name: Specify the name of the message file to use.

DETAIL

Specifies the amount of detail about the message to show or print.

***FULL:** Detailed message descriptions are shown or printed. If OUTPUT(*PRINT) is specified, the detailed descriptions are printed. If OUTPUT(*) is specified, a menu is shown from which the user can select which details to view.

***BASIC:** A list of the specified message identifiers, their severity, and first-level messages are shown or printed. If the user is working from an interactive display, the user can choose to show or print detailed descriptions of messages selected from the list.

FMTTXT

Specifies whether the message text is formatted.

***YES:** The message text is formatted.

***NO:** The message text is unformatted.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying or Printing Descriptions

```
DSPMSGD   RANGE(CPF1100  CPF3600)
  MSGF(QSYS/QCPFMSG)
```

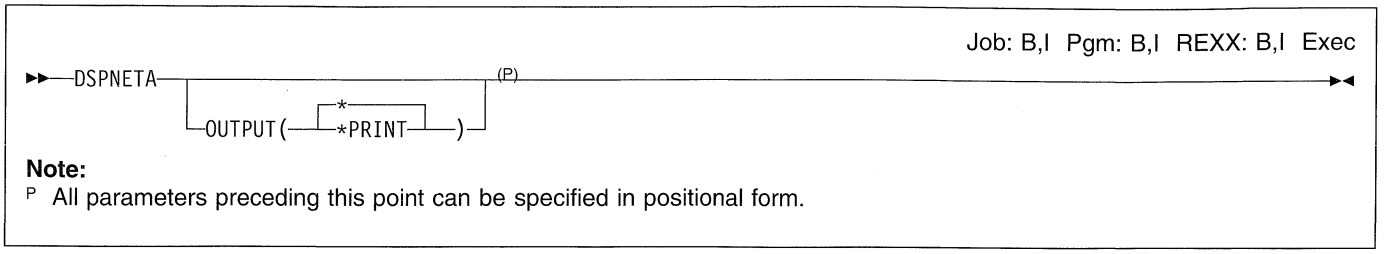
If this command is entered from a display station, the descriptions of the specified messages are shown on the display. If this command is entered from a batch job, the descriptions of the messages are sent to the job's spooled output queue.

Example 2: Printing Message Descriptions

```
DSPMSGD   RANGE(*FIRST  IDU0571)
  MSGF(QIDU/QIDUMSG)  FMTTXT(*NO)
  OUTPUT(*PRINT)
```

This command prints the message descriptions for message identifiers in the file that are in the following range: from the first message in the QIDUMSG message file through the message whose identifier is IDU0571. The message is unformatted.

DSPNETA (Display Network Attributes) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Network Attributes (DSPNETA) command displays the network attributes of the system.

Optional Parameters

OUTPUT

Specifies whether the output of the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

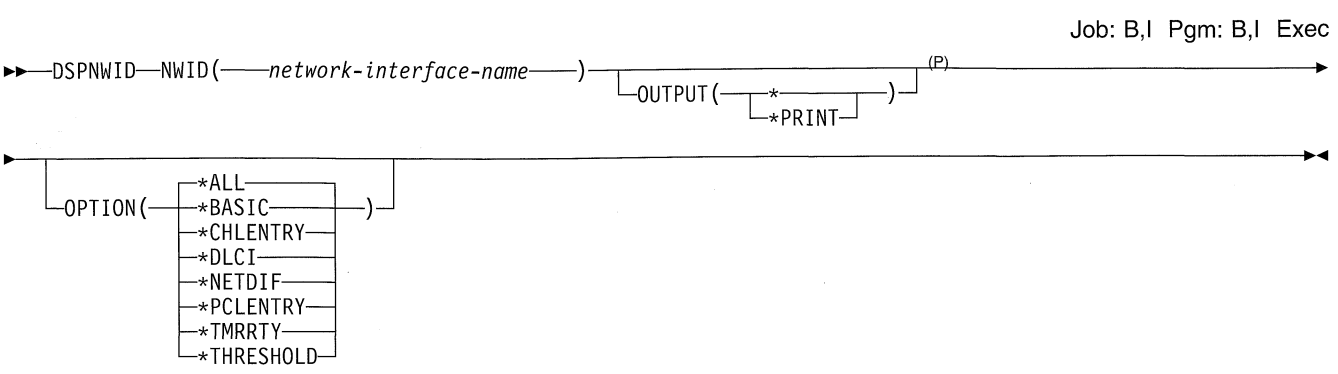
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPNETA  OUTPUT(*)
```

If the job is interactive, this command displays the network attributes of the system at the work station. If the job is batch, the network attributes are printed with the job's spooled output.

DSPNWID (Display Network Interface Description) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Network Interface Description (DSPNWID) command displays a network interface description.

Required Parameters

NWID

Specifies the name of the network interface description to be displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

OPTION

Specifies the panels to be displayed.

***ALL:** All panels that apply to this specific line type are displayed.

***BASIC:** The basic displays that apply to this line type are shown, but additional information displays are not shown.

***CHLENTRY:** The channel entry list panels associated with the network interface description are displayed. This value is valid only when an integrated services digital network (ISDN) is used.

***DLCI:** The data link connection identifier information, including the DLCI numbers, status, and active or attached lines is displayed. This value is valid only when a frame relay network (FR) is used.

***NETDIF:** The values of the network difference parameters in the network interface description are displayed. This value is valid only when ISDN is used.

***PCLENTRY:** The protocol-specific information, including the protocol entries for the network interface description, is displayed. This value is valid only when ISDN is used.

***TMRRTY:** The timer and retry values for this line are displayed.

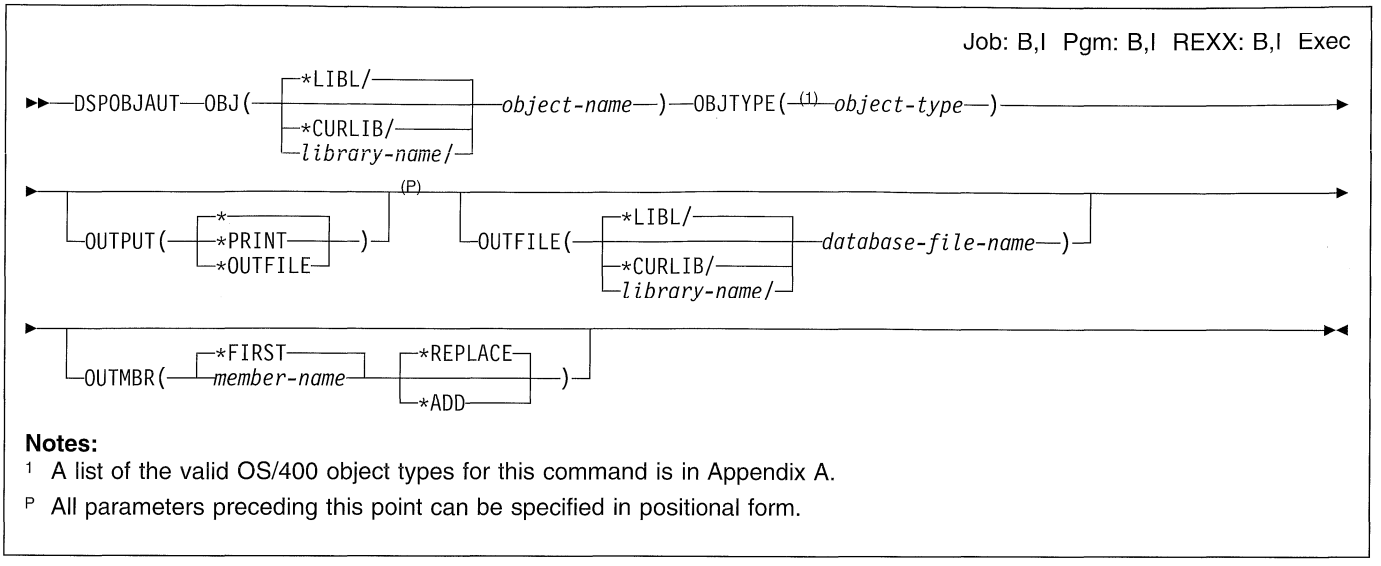
***THRESHOLD:** The values specified on the threshold parameters in the network interface description are displayed. This value is valid only when ISDN is used.

Example

```
DSPNWID   NWID(ISDNNET)
```

This command displays information about the network interface description named ISDNNET. The information is shown an the work station display from which the command was submitted. If the command is entered from a batch job, the output from the display is printed with the job's spooled output.

DSPOBJAUT (Display Object Authority) Command



Notes:
1 A list of the valid OS/400 object types for this command is in Appendix A.
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Object Authority (DSPOBJAUT) command shows the list of authorized users of an object and their authorities for the object. If the object is secured by an authorization list, the name of the authorization list is also shown.

The following are shown for the specified object:

- The object name.
- The name of the library that contains the object.
- The name of the object's owner.
- The object's type.
- The name of the authorization list securing the object.
- A list of all the users who are authorized to use the object.
- The authorities that each user has for the object.

If the user entering the command does not have object management authority for the object, only that user's name and authorities are shown. The names of the other users and their authorities for the object are not shown. If an object does not have an owner name associated with it, no authorities for the object are shown.

Required Parameters

OBJ

Specifies the qualified name of the object for which the authorized users and their authorities are displayed.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

object-name: Specify the name of the object.

OBJTYPE

Specifies the type of object that has its authorized users and authorities shown. Any one of the object types can be specified. For example, to display a list of users authorized to use a file, specify the value *FILE. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file.

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the name of the database file to which the output of the display is directed. If the output file does not exist, this command creates a database file in the specified library. If the file is created by this function,

the text is *'OUTFILE created by DSPOBJAUT'*. The public authority is *EXCLUDE. The database format (QSYDSAUT) of the output file is the same as that used in the IBM-supplied database file QAOBJAUT.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file to which the output is directed.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Displaying Users and Authorities

```
DSPOBJAUT   OBJ(ARLIB/PROG1)  OBJTYPE(*PGM)
```

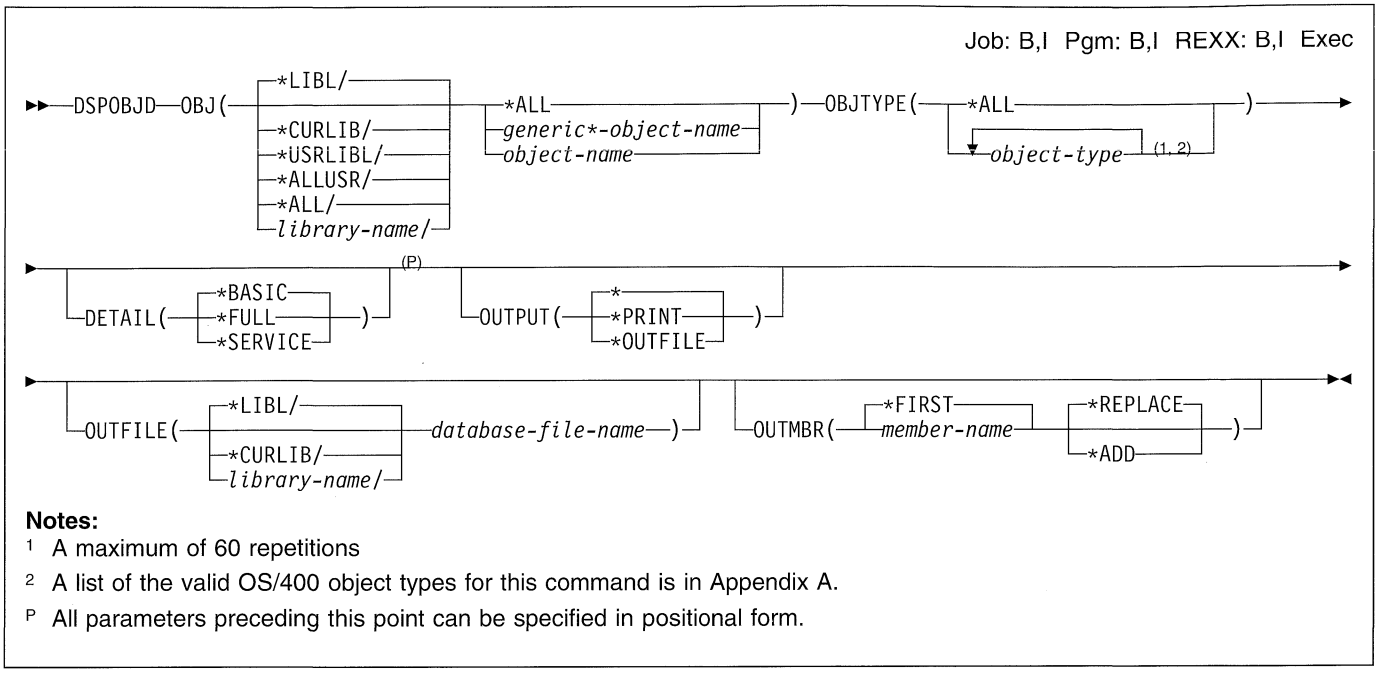
This command shows the authorized users and their authorities for the object named PROG1 to the user who entered the command, if that user has object management authority for the object. If the user does not have object management authority, only personal authorities are shown. PROG1 is a program (*PGM) located in the library named ARLIB. The system assumes * for the device that shows the output list. If the command was entered in the batch subsystem, the output is placed in the default output queue for the job. If the command was entered in the interactive subsystem, the output is shown on the device where the user entered the command.

Example 2: Printing List of Users

```
DSPOBJAUT   OBJ(ARLIB/PROG2)  OBJTYPE(*PGM)
    OUTPUT(*PRINT)
```

This command causes the list of authorized users of the program named PROG2 in the ARLIB library to be printed. If the user who enters the command does not have object management authority for the program, only that user's name and authorities are printed.

DSPOBJD (Display Object Description) Command



Notes:
1 A maximum of 60 repetitions
2 A list of the valid OS/400 object types for this command is in Appendix A.
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Object Description (DSPOBJD) command shows the names and attributes of specified objects in the specified library or in the libraries of the job's library list. The command can also be used to show the names and attributes of libraries themselves.

Only the object attributes of each object are shown; the data attributes of data in the object, and the actual data in the object, are not shown. Also indicated is whether an object being shown has been damaged (possibly by a system failure).

Any OS/400 system object for which the user has some authority can be shown by this command. Objects or libraries for which the user has no object operational authority cannot be shown, even if specified in the command. If only one object is to be shown, the user can specify it by entering the object name, object type, and the name of the library where it is located. Depending on the specified library qualifier, either the first object found in the specified libraries is shown, or all objects in the specified libraries for which the user has object operational authority are shown.

Notes:

1. For objects that are damaged or locked, the information shown, printed, or written to the output file is incomplete. An indication that the object is locked or damaged is written to the output file. If the description of the object is shown or printed, the text for the damaged or locked object indicates either the damaged or locked status.
2. The size of the library object is not determined by the sizes of the objects in the library. The size of a library, including the sizes of the objects in the library, can be obtained by using the DSPLIB command.

Restriction: The user who submits this command must either own the objects or have authority for the specified libraries, and some authority (that is, any one of the object authorities) for each of the objects. If the user does not have read authority for a library, none of its objects are shown.

Required Parameters

OBJ

Specifies which objects in the libraries are to have their object attributes shown. If no library qualifier is specified, *LIBL is assumed, and all libraries in the job's library list are searched for the objects. Objects in the library for which the user does not have some authority are not shown.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB     #DFULIB     #RPGLIB     #SEULIB
#COBLIB     #DSULIB     #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX       QPFRDATA    QUSER38
QGPL        QRCL        QUSRSYS
QGPL38      QS36F       QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All objects in the libraries identified in the library qualifier, which are of the types specified by the OBJTYPE parameter, are shown.

Note: If the library qualifier is *ALL, *ALLUSR, or a library name, all objects of the specified types for which the user has some authority (for example, object operational authority) and that are in the specified libraries are shown. If the qualifier is *USRLIBL or *LIBL, only one object type (not *ALL) can be specified, and only the first object found is shown. The user can show only those objects for which the user has some authority.

generic-object-name:* Specify the generic name of the object. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be searched only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

object-name: Specify the name of the object that is shown.

OBJTYPE

Specifies the types of objects that are shown. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ALL:** All object types that have the specified object name are shown.

object-type: Specify one or more values for the types of OS/400 system objects to be shown. Any of the OS/400 system object types can be specified. Objects in the specified libraries, as well as libraries themselves, that have the object types specified have their object attributes shown. If the library qualifier specified or assumed for the OBJ parameter is *USRLIBL or *LIBL, only one object type (not *ALL) can be specified.

Optional Parameters

DETAIL

Specifies which set of attributes is shown for each object.

***BASIC:** The display or printed output contains the name and a basic set of object attributes for each object.

***FULL:** The display or printed output contains the name and a full set of object attributes for each object, which includes the basic set of attributes.

***SERVICE:** The display or printed output contains the service-related attributes for each object.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

Note: All data is written to the file specified on the OUTFILE parameter regardless of the value specified on the DETAIL parameter.

OUTFILE

Specifies the qualified name of the database file to which the output of the display is directed. If the OUTFILE does not exist, this command creates a database file in the specified library. If the file is created, the text is "output file for DSPOBJD". The public authority is the same as the create authority specified for the library in which the file is created. The database format (QLIDOBJD) of the output file is the same as that used in the IBM-supplied file database QADSPOBJ.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that receives the output of the display.

OUTMBR

Specifies the name of the database file member to which the output is directed. If a member already exists, the system uses the second element of this parameter to determine whether the member is cleared before the new records are added. If the member does not exist and a member name is not specified, the system creates a member with the name of the output file specified on the OUTFILE parameter. If an output file member name is specified, but the member does not exist, the system creates it.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Displaying a Basic Description of Objects

```
DSPOBJD  OBJ(X/PAY)  OBJTYPE(*ALL)
```

A basic description of all the objects (for which the user has some authority) that are named PAY in library X are shown. Objects in the library for which the user has no authority are not shown.

Example 2: Displaying a Full Description of a Program

```
DSPOBJD  OBJ(X/PAY)  OBJTYPE(*PGM)  DETAIL(*FULL)
```

A full description of the program named PAY in library X is shown. The display includes all the attributes of the program.

Example 3: Displaying Program Information

```
DSPOBJD  OBJ(*USRLIBL/PAY)  OBJTYPE(*PGM)
```

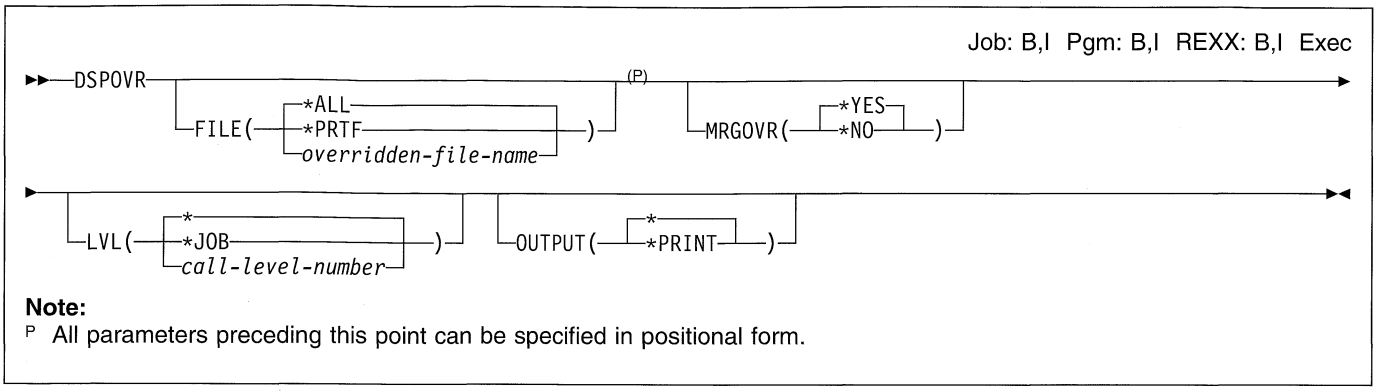
This command shows information about the first program named PAY that is found in the user portion of the job's library list.

Example 4: Displaying a Basic Description of Files

```
DSPOBJD  OBJ(Z/ABC*)  OBJTYPE(*FILE)
```

A basic description of all of the files whose names begin with ABC (generic name) located in library Z, for which the user has some authority, are shown.

DSPOVR (Display Override) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Override (DSPOVR) command displays file overrides at any active call level for a job. All file overrides, or file overrides for a specific file name, can be displayed.

The file overrides can be merged before being displayed. A merged file override is the result of combining all overrides for a file from call level one to the specified call level, producing the override which is applied when the file is opened at the specified call level.

A call level is associated with each call stack entry in the call stack. Calling a program or procedure adds another call stack entry to the call stack. When a program or procedure is called using the TFRCTL (Transfer Control) command, the call stack entry replaces a call stack entry that is already on the call stack; a new call level number is not created.

Note: This function can also be accessed through option 15 of the Work with Job (WRKJOB) command.

Optional Parameters

FILE

Specifies whether all file overrides, or file overrides for a specific file name, are displayed.

***ALL:** All the file overrides from call level one to the specified call level are displayed.

***PRTF:** The *PRTF file override, which exists in the call level where this command is entered, is displayed.

overridden-file-name: Specify the name for which all the file overrides for the specified file name, from call level one to the specified call level, are displayed.

MRGOVR

Specifies whether the file overrides are merged. Only those parameters on the overrides of the same type as the last override used for the merged override are used in determining the effective override for the specified call level.

***YES:** The file overrides displayed are merged.

***NO:** The file overrides displayed are not merged.

LVL

Specifies the call levels of the file overrides displayed. There is a one-to-one correspondence between the call stack entries displayed on the call stack from the WRKJOB command and the call level for that call stack entry.

The first call stack entry name displayed on the call stack (at the top of the list) is the program or procedure at call level one. The second call stack entry name displayed is the program or procedure at call level two. The last call stack entry name displayed is the program or procedure at the highest call level for the job.

- If a merged file override is displayed, file overrides from call level one to the specified call level contributes to the creation of the merged file override.
- If MRGOVR(*NO) and FILE(*ALL) are specified, all file overrides (and the call levels at which they were found) from call level one to the specified call level are displayed.
- If MRGOVR(*NO) is specified, and a file override name is specified on the FILE parameter, all file overrides for the file specified (and the call levels at which they were found) from call level one to the specified call level are displayed.

***:** The call level of the file override displayed is the call level of the program that called the DSPOVR command processing program. If the DSPOVR command is called to QCMDEXC, the call level is the same level as that of QCMDEXC. Overrides at call level numbers greater than 999 are not displayed.

***JOB:** The call levels of the file overrides displayed are all call levels for the job.

call-level-number: Specify the specific call levels of the file overrides displayed. A specific call level is used to display file overrides at call levels lower than the call level at which the user is running. Valid values range from 1 through 999.

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Merged Overrides

```
DSPOVR FILE(REPORTS) MRGOVR(*YES) LVL(3) OUTPUT(*)
```

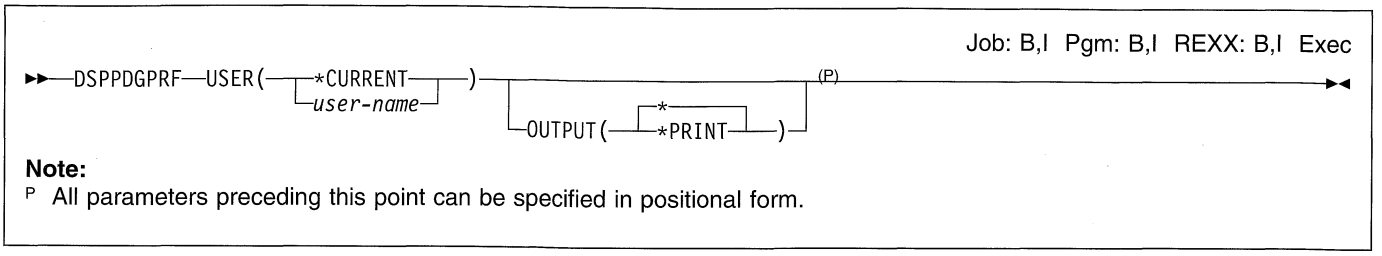
This command produces a display showing the merged override for the file REPORTS at call level 3 with text descriptions of each keyword and parameter. Applicable overrides at call levels 1, 2 and 3 are used to form the merged override.

Example 2: Displaying File Overrides

```
DSPOVR FILE(REPORTS) MRGOVR(*NO) LVL(2) OUTPUT(*)
```

This command displays all file overrides for the file reports up to call level 2. It produces a display showing the file name, the call level for which the override was requested, the type of override, and the override parameters. If no file overrides are found for the file up to and including the specified call level, escape message CPF9842 is sent.

DSPPDGPRF (Display Print Descriptor Group Profile) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Print Descriptor Group Profile (DSPPDGPRF) command displays the print descriptor group (PDG) and print descriptor name for the specified user profile.

Restriction: The user must have *OBJMGT authority to use this command.

Required Parameters

USER

Specifies the name of the user whose PDG profile is to be displayed.

***CURRENT:** The user profile under which the current job is running is used.

user-name: Specify the name of the user whose PDG profile is to be displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

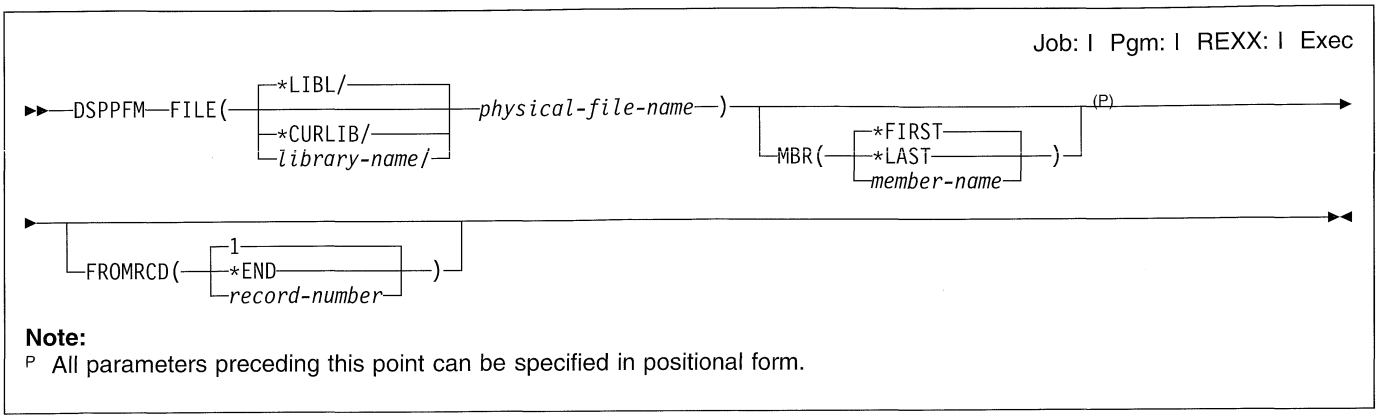
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPPDGPRF  USER(TPDEXTER)
```

This command displays the print descriptor and the print descriptor group for user profile TPDEXTER.

DSPPFM (Display Physical File Member) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Physical File Member (DSPPFM) command displays a physical database file member. Records are shown in arrival sequence, even if the file has a keyed access path. Users can page through the file, locate a particular record by record number, or specify a starting position in the record. They can also select a character or hexadecimal display of the records.

Required Parameters

FILE

Specifies the name of the physical file that contains the member being shown. If no library qualifier is specified, the library list is used to locate the file.

| The name of the physical file member can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

physical-file-name: Specify the name of the physical file.

Optional Parameters

MBR

Specifies the name of the physical file member being shown.

***FIRST:** The first member in the database file is used.

***LAST:** The last member of the specified physical file is shown.

member-name: Specify the name of the physical file member.

FROMRCD

Specifies which record in a physical file is shown on the top line of the initial display. If the specified record number is a deleted record, the display is positioned on the first record that follows the record that has been deleted. The user can display as many records as needed using the page-up and page-down keys.

1: Record number one, or the first non-deleted record, of the physical file is shown.

***END:** The last non-deleted record in the physical file is shown. The *END value shows the last complete page so that the last record in the physical file appears at the bottom of the screen.

record-number: Specify the number of the record shown on the top line of the initial display.

Examples

Example 1: Displaying the First File Member

```
DSPPFM   FILE(TESTA)
```

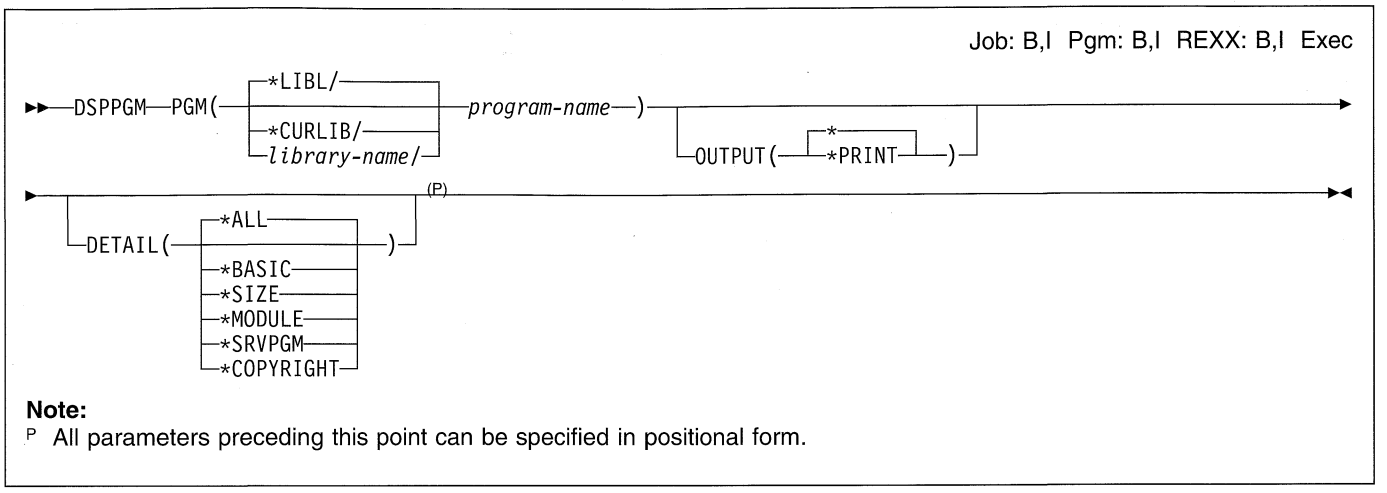
This command shows the first member of a physical file named TESTA. The library list is used to locate the file.

Example 2: Displaying a File Member

```
DSPPFM   FILE(SAMPLE/TESTB) MBR(PROGRAM)
```

This command shows member PROGRAM of physical file TESTB in library SAMPLE.

DSPPGM (Display Program) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Program (DSPPGM) command displays information about a program. The display includes information about the compiler, the type of program, certain processing attributes of the program, the type of program (ILE or OPM), the size of the program, and the number of parameters that can be passed to the program when it is called.

Restrictions You must have *USE authority to the program and *READ authority to the library to use this command.

Required Parameters

PGM

Specifies the qualified name of the program for which information is displayed.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program for which information is displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

DETAIL

Specifies the type of information displayed for the service program. More than one value can be specified, but a list of values must not include *ALL. *ALL must be specified as a single value.

Note: DETAIL(*ALL) or DETAIL(*BASIC) are the only values valid for original program model (OPM) programs. All values other than *ALL or *BASIC are ignored for an OPM program.

***ALL:** All information types are shown on the display.

***BASIC:** General program information is shown.

***SIZE:** The size and size limits for this program are shown.

***MODULE:** A list is shown of the module objects bound by this program.

***SRVPGM:** A list is shown of the service programs bound by this program.

***COPYRIGHT:** A list is shown of the copyrights for this service program.

Note: The DETAIL values *SIZE, *MODULE, *SRVPGM, and *COPYRIGHT are valid only for integrated language environment (ILE) programs. Specifying one of these values for an original program model (OPM) program results in the *BASIC information being shown.

Examples

Example 1: Displaying Program Information

```
DSPPGM   PGM(LIB01/PAYROLL)
```

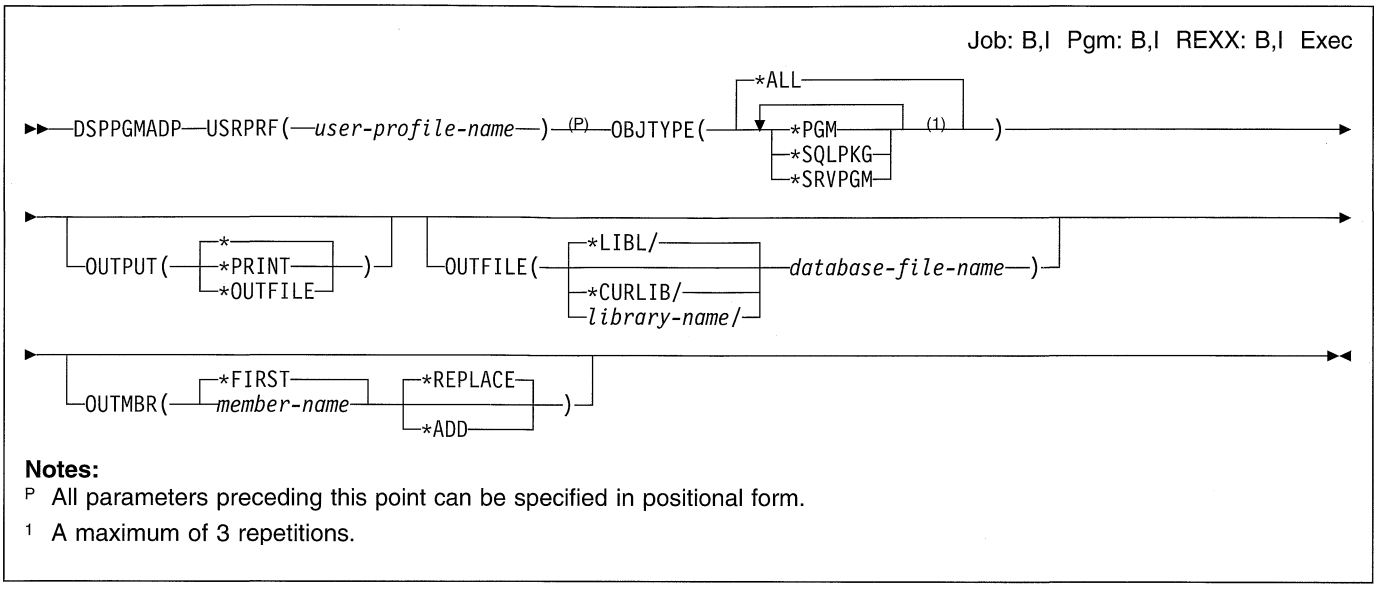
This command displays information about the program named PAYROLL in library LIB01. The display is shown at the display station if requested by an interactive job, or printed if requested by a batch job.

Example 2: Printing Program Information

```
DSPPGM  PGM(CUSINQ)  OUTPUT(*PRINT)
```

This command displays information about a program named CUSINQ. The library list is used to find the program, and the information is printed.

DSPPGMADP (Display Programs that Adopt) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 3 repetitions.

Purpose

The Display Programs that Adopt (DSPPGMADP) command is used to show objects that adopt the special and private authorities of the specified user profile. This is a convenient way to check security exposure due to program adoption.

Restriction: The user of this command must have *OBJMGT authority to the user profile.

Required Parameter

USRPRF

Specifies the name of the user profile whose authorities are adopted. Up to 10 characters can be specified.

Optional Parameters

OBJTYPE

Specifies the type of object shown.

***ALL:** All objects that adopt the user profile specified on the USRPRF parameter are shown.

***PGM:** Only programs that adopt the specified user profile are shown.

***SQLPKG:** Only Structured Query Language (SQL) packages that adopt the specified user profile are shown.

***SRVPGM:** Only service programs that adopt the specified user profile are shown.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file.

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file to which output of the display is directed. If this file does not exist, this command creates a database file in the specified library. If this function creates the file, the text states 'Outfile for DSPPGMADP', and the public authority is *EXCLUDE.

The name of the output database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the user has the option of either adding records to the end of the existing member or clearing the existing member and adding the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

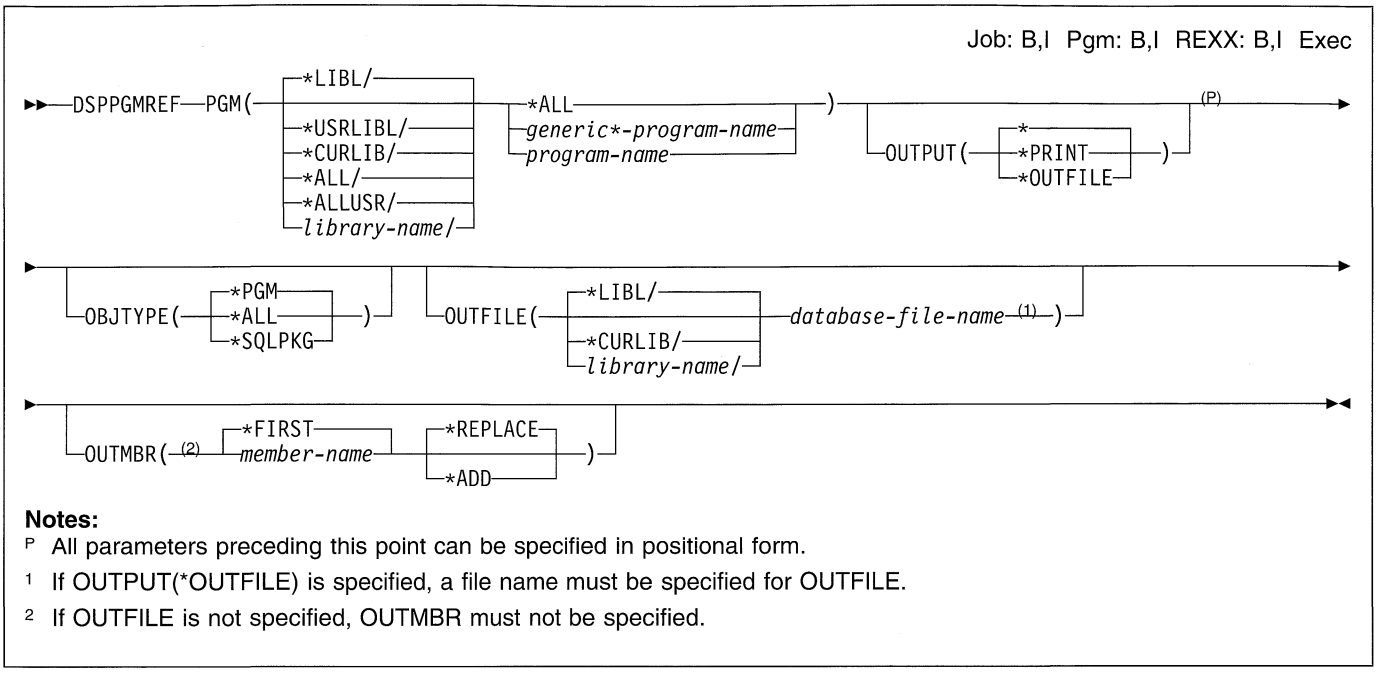
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPPGMADP  USRPRF(xxx)  OUTPUT(*PRINT)
```

This command directs the information about the user profile (xxx) that programs use to print the job's output spooling queue.

DSPPGMREF (Display Program References) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 If OUTPUT(*OUTFILE) is specified, a file name must be specified for OUTFILE.

2 If OUTFILE is not specified, OUTMBR must not be specified.

Purpose

The Display Program References (DSPPGMREF) command provides a list of the system objects to which the specified programs refer.

Program Type	System Objects
CL	*FILE, *PGM, and *DTAARA
RPG	*FILE, *DTAARA, *PGM
COBOL	*FILE and *PGM (literal names on the CALL command)
BASIC	*FILE (externally described) and *PGM
PL/I	*FILE and *PGM
PASCAL	No information is provided
CSP	*FILE, *PGM, *MSGF, *CSPMAP, and *CSPTBL
C	No information is provided

This information can be shown, printed, or written to (stored in) a database output file (an OUTFILE).

If the information is shown or printed, a list (by library) of the specified user-authorized programs, along with the objects referenced by each program, is created. For files, information about how each file is used (input, output, update, unspecified, or any combination of these four) is also shown or printed.

If the information is written to a database file, the database file will have a record format named QWHDRPPR. The fields in record format QWHDRPPR are the same as the fields in the IBM-supplied format QWHDRPPR in file QADSPPGM in the library QSYS. The following information is contained in the database file:

- The name of the program and its text description
- The name of the library containing the program
- The number of objects referenced by the program
- The qualified name of the system object
- The information retrieval dates
- The object type of the referenced object

For files, the record contains the following additional fields:

- The name of the file in the program (possibly different from the system object name if an override was in effect when the program was created)
- The program use of the file (input, output, update, unspecified, or a combination of these four)
- The number of record formats referenced, if any
- The name of the record format used by the file and its record format level identifier
- The number of fields referenced for each format

Note: The referenced object names and libraries listed may be different than the actual names of the objects, since this information is stored when the program is created. If the object has been moved since the program was created, or an override was in effect during creation, the names listed may differ from the actual names.

Restrictions:

1. The user must have object operational authority for the program.
2. Also, of the libraries specified by the library qualifier, only the libraries for which the user has read authority are searched for the programs.

Required Parameters

PGM

Specifies the qualified name of the program (or all programs) that has a list of the files and other related system objects that are referenced by the program. A specific program name or a generic program name is specified; either type can be optionally qualified by a library name. Only the libraries in the specified library qualifier that the user either owns or is authorized to use are searched for the program(s).

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All programs in the specified library (or all the libraries identified in the library qualifier to which the user has access) have their information shown.

generic-program-name:* Specify the name of the program or the generic name of several programs in the specified library (or all libraries identified in the library qualifier) that has information shown. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

program-name: Specify the name of the program to have information shown.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or placed in a database outfile. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OBJTYPE

Specifies the object type for which information is displayed.

***PGM:** Only program information is displayed.

***ALL:** Program information and SQL package information are displayed.

***SQLPKG:** Only SQL package information is displayed.

OUTFILE

Specifies the qualified name of the database output file in which the program-related information is written.

When creating the database output file, the current date, time, and system name must be included. The system name is the name of the source system, not the target system. If the specified file does not exist, the system creates a database output file and member. If the file is created, the description text is *'Outfile for DSPPGMREF'* and the public authority is *EXCLUDE.

Note: The outfile format must be the same as QWHDRPPR of the system file QADSPPGM. More information on the OUTFILE format is in the *Database Guide*.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database output file in which the program-related information is placed. If the specified file is not found, a file by that name is created and stored in the specified library, or in *CURLIB if not qualified. This file can be used again when other DSPPGMREF commands are entered. The IBM-supplied database file QADSPPGM *cannot* be specified.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the user has the option to add records to the end of the existing member or to clear the existing member and add the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Storing a List of Programs

```
DSPPGMREF   PGM(LIBRARY1/*ALL)  OUTPUT(*OUTFILE)
  OUTFILE(LIB2/FILE2)
```

This command creates a list of all authorized programs found in LIBRARY1, and of the files and other system objects that the programs reference. It stores the list in a database file named FILE2 in LIB2.

Example 2: Printing a List of Objects

```
DSPPGMREF   PGM(LIBRARY1/BILLING)  OUTPUT(*PRINT)
```

This command creates a list of system objects that are referenced by the BILLING program in LIBRARY1. The output is spooled for printing.

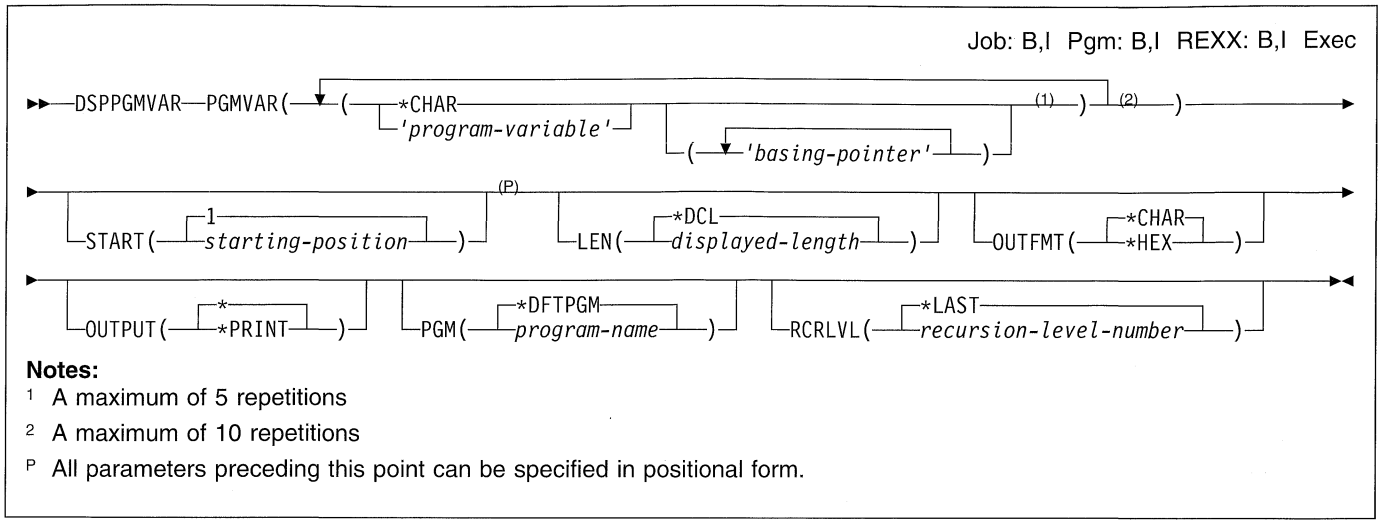
Additional Considerations

When the DSPPGMREF command is entered, the library is searched for the program or programs specified on the PGM parameter; then a group of records that give system object-related information about each program is created. The records are placed in the printer device file named QPDSPPGM. If OUTPUT(*PRINT) is specified on the command, the records are listed on the printer in the following order:

- Header information: Lists the DSPPGMREF command input values.
- Program-related information: Lists, for each program identified by the PGM parameter, all of the system objects (including files) that are referenced by the program. If more than one program is identified, the beginning of the next one follows the end of the previous one.

If the DSPPGMREF command is entered interactively and OUTPUT(*) is specified or assumed, the records in the printer device file are shown instead of being printed.

DSPPGMVAR (Display Program Variable) Command



Notes:

1 A maximum of 5 repetitions

2 A maximum of 10 repetitions

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Program Variable (DSPPGMVAR) command shows the current value of one or more program variables in a program that is being debugged. The variables can be specified either by their variable names or by their machine-interface object-definition-table-vector (MI ODV) numbers. Up to 10 variables can be specified.

Restrictions:

1. This command is valid only in debug mode. To start debug mode, refer to the STRDBG (Start Debug) command.
2. This command cannot be used if the user is servicing another job, and that job is on a job queue, or is being held, suspended, or ended.
3. This command cannot be used to display variables in a bound program.
4. This command cannot be used to display variables within the system domain unless the user has *SERVICE special authority.

Required Parameter

PGMVAR

Specifies the names of from 1 through 10 program variables whose values are shown. The variables can be in a high-level language (HLL) or machine-interface (MI) program.

Element 1: Program Variables

***CHAR:** This special value is specified instead of a variable name if a basing pointer is also specified. This special value displays a character view of a pointer being shown without the use of a based variable.

'program-variable': Specify the names of from 1 through 10 program variables to be shown. The name must be enclosed in apostrophes if it contains special characters.

If the program variable is an array, the subscripts representing an element in the array can be specified. If an array name is specified without any subscripts, all of the array elements are recorded. A single-dimensional cross-section can also be specified. Up to 132 characters may be specified for this program variable entry. This includes any qualifiers, subscripts, embedded blanks, parentheses, and commas. It does not include the enclosing apostrophes when special characters are used. An integer, MI ODV number, asterisk (single-dimensional cross-section), or a numeric variable name can be specified for a subscript. For more information on the program-variable value, refer to Appendix C, "Parameter Values Used for Testing and Debugging."

Some examples are:

```
PGMVAR(A)
PGMVAR('A(2,B)')
PGMVAR('B(I1,*,I3)')
PGMVAR('VAR1 OF A(I,J IN B)')
```

Element 2: Basing Pointers

'basing-pointer': Specify up to five basing pointers for the each program variable being displayed. In some languages, the program variable may be based on a pointer variable. This set of values allows the user to specify the basing pointers for the variable to be displayed. Each basing pointer name must be enclosed in apostrophes if it contains special characters.

If the basing pointer is an array, the subscripts representing an element in the array must be specified. Up to 132 characters can be specified for a basing pointer name. This includes any qualifiers, subscripts, embedded blanks, parentheses, and commas. It does not include the enclosing apostrophes when special characters are used. An integer, MI ODV number, or a numeric variable name can be specified for a subscript. For more information on the basing pointer value, refer

to Appendix C, "Parameter Values Used for Testing and Debugging."

Some examples are:

```
PGMVAR(('VAR1(B,5)' 'PTR2(C,P2)'))
PGMVAR((VAR2 (BASEPTRA BASEPTRB)))
```

Optional Parameters

START

Specifies, for string variables only, the starting position in the string from which its value is being displayed. If more than one string variable is specified in the PGMVAR parameter, the same starting position value is used for each one. For a bit string, the value specifies the starting bit position; for a character string, the value specifies the starting character position.

1: The variable is shown from the first position through the length specified on the LEN parameter.

starting-position: Specify the first position in the string being displayed. The START value specified must not be larger than the maximum string length for any variable specified, except that START(1) is allowed if the maximum length for a string is zero. The LEN value, plus the START position minus one, must not be greater than the maximum string length. These checks are made for each string variable specified in the PGMVAR parameter.

LEN

Specifies, for string variables only, the length of the string shown when the breakpoint is reached, starting at the position specified by the START parameter. If more than one string variable is specified in the PGMVAR parameter, the same value is used for each one. For a bit string, the value specifies the number of bits shown; for a character string, the value specifies the number of characters shown.

***DCL:** The string variable is shown to the end of the string or for a value of 200 bytes, whichever is less. If the string variable has a maximum length of zero, only LEN(*DCL) is allowed.

displayed-length: Specify the length of the data shown. The length (as well as the combination of START and LEN) must be no greater than the length of the shortest string specified in the PGMVAR parameter.

OUTFMT

Specifies the format in which the objects are shown.

***CHAR:** The program variables are shown in character form.

***HEX:** Program variables are shown in both character format and hexadecimal format.

OUTPUT

Specifies whether the output from the command is shown at the requesting display station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

PGM

Specifies the name of the program that contains the program variables shown.

***DFTPGM:** The program currently specified as the default program has its variables shown.

program-name: Specify the name of the program whose program variables are shown.

RCRLVL

Specifies which recursion level of the program contains the variables being shown. Recursion level 1 is the first (or earliest) call of the program, recursion level 2 is the second call of the program, and so on to the last (most recent) recursion level in the stack. For example, if program A calls program B, and then program B calls program A, a new recursion level of program A is formed. Some high-level languages also allow recursive procedures. For these programs, refer to the appropriate high-level language manual for more information.

***LAST:** The last (most recent) call of the program contains the pointer being shown.

recursion-level-number: Specify the number of the recursion level of the program containing the pointer being shown.

Examples

Example 1: Displaying Program Variables

```
DSPPGMVAR   PGMVAR('&QUANT')  PGM(MYPROG)
```

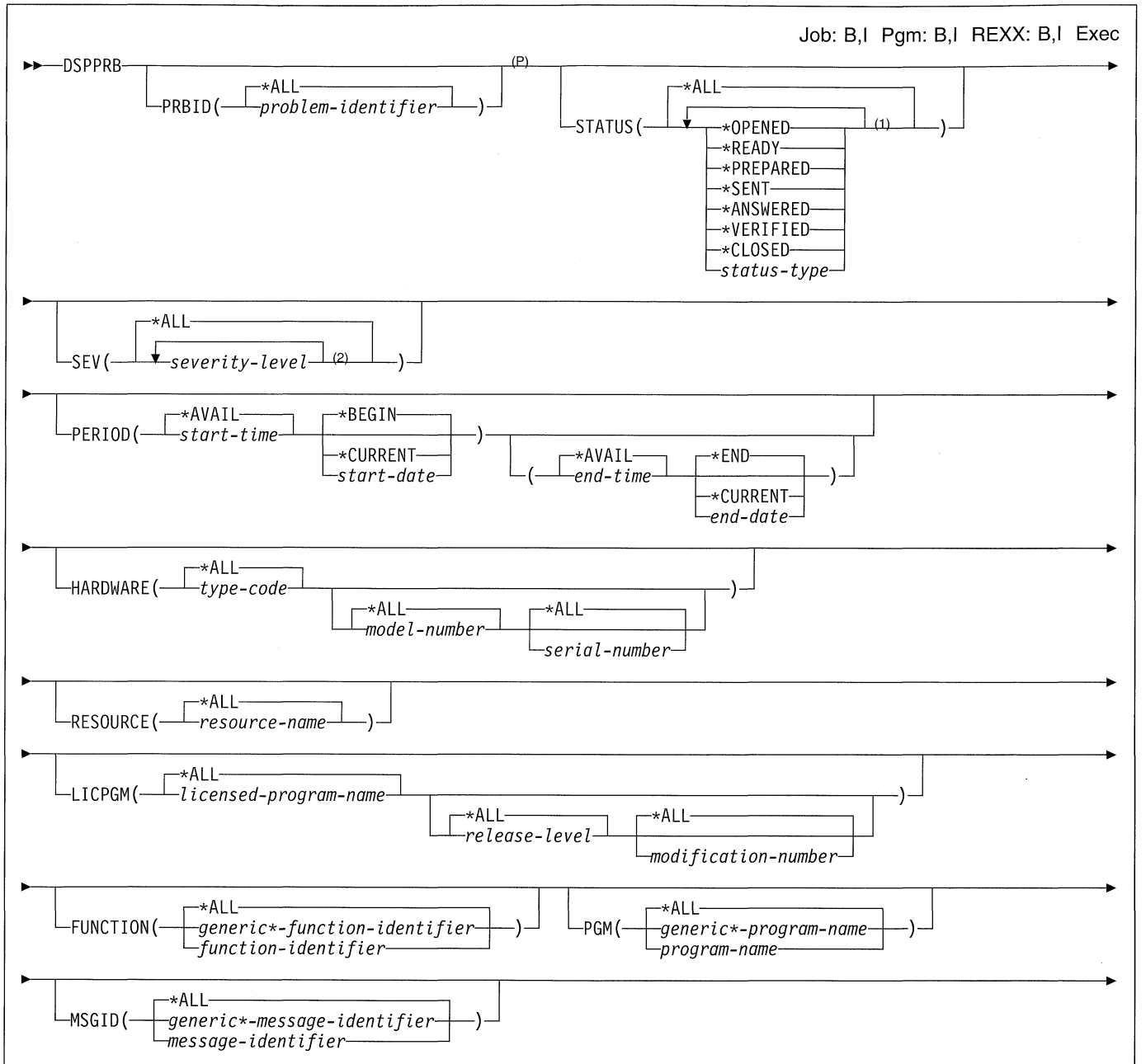
Assuming that the program MYPROG is in debug mode, this command shows the name and current value of the CL variable called &QUANT; its type and length are also shown.

Example 2: Displaying Program Variables

```
DSPPGMVAR   PGMVAR(TOTSALES  MANHRS)
  PGM(REGION)  RCRLVL(1)
```

This command shows the program variables TOTSALES and MANHRS of the first call of the program REGION.

DSPPRB (Display Problem) Command

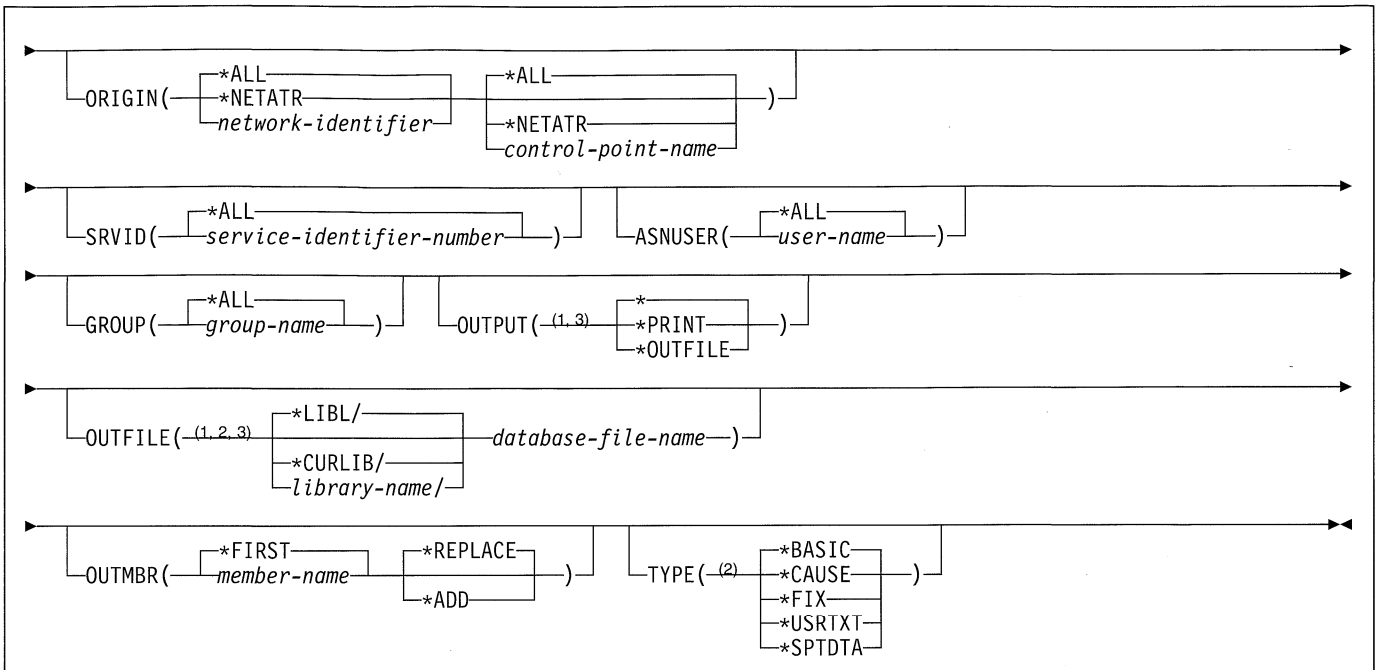


Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 6 repetitions

2 A maximum of 3 repetitions



Notes:

1 The OUTFILE and OUTMBR parameters are valid only when OUTPUT(*OUTFILE) is specified.

2 The TYPE Parameter is valid only when a value is specified for the OUTFILE parameter.

3 If OUTPUT(*OUTFILE) is specified, a database file name is required for the OUTFILE parameter.

Purpose

The Display Problem (DSPPRB) command allows the user to show or print service information related to performing hardware and software maintenance. The service information, contained in the problem log entries, can be shown on the DSPPRB display, printed with the job's output, or stored in a database file.

Optional Parameters

PRBID

Specifies the problem identifier of the problem being selected. Problems with different system origins can have the same identifier. This parameter can be used with the ORIGIN parameter to select a single problem from a particular system origin.

***ALL:** All problem identifiers are selected.

problem-identifier: Specify the 10-character problem identifier of the problem being selected.

STATUS

Specifies the status of the problem log entries. The seven status types are as follows:

*OPENED The problem is in OPENED status. The problem has been identified and a problem record has been created.

*READY The problem is in READY status. Problem analysis information has been added to the problem record.

*PREPARED The problem is in PREPARED status. The problem has been prepared for reporting.

*SENT The problem is in SENT status. The problem has been sent to a service provider, but no answer has been returned.

*ANSWERED The problem is in ANSWERED status. An answer was returned by the service provider or added by an operator on this system.

*VERIFIED The problem is in VERIFIED status. The problem has been resolved and the system operator has verified that the problem has been corrected.

*CLOSED The problem is closed.

***ALL:** Problem log entries with any status are shown.

status-type: Specify up to six status types that identify the problem log entries being displayed.

SEV

Specifies the severity level of the problem log entries being shown on the display. The severity level is assigned by the user when the problem is prepared for reporting. The four severity levels are:

1 High

2 Medium
3 Low
4 None

***ALL:** Problem log entries with any severity level are shown.

severity-level: Specify at least one or as many as three of the four severity levels.

PERIOD

Specifies the period of time for which the data log is printed. The following values can be coded in this parameter, which contains two sets of two values each (four list elements).

Element 1: Starting Time

***AVAIL:** The error data that is available for the specified starting or ending date is printed.

start-time: Specify the starting time of the specified starting date for which the error data is printed. The time is specified in 24-hour format with or without a time separator as follows:

- With a time separator, specify a string of 5 or 8 digits where the time separator separates the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command fails.
- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where **hh** = hours, **mm** = minutes, and **ss** = seconds. Valid values for **hh** range from 00 through 23. Valid values for **mm** and **ss** range from 00 through 59.

Element 2: Starting Date

***BEGIN:** The logged data from the beginning of the log is displayed.

Note: If PERIOD(*BEGIN) is specified, then any time value other than *AVAIL for start-time is ignored.

***CURRENT:** The logged data for the current day, and the time that has elapsed between the specified starting time and ending time for the day, is displayed.

start-date: Specify the date shown. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

Element 3: Ending Time

***AVAIL:** The logged data that is available for the specified ending date is displayed.

end-time: Specify the time that the data logging ends. See the description of *start-time* for details about how time can be specified.

Element 4: Ending Date

***END:** The last day on which data was logged is shown. If PERIOD(*END) is specified, a time value other than *AVAIL for end-time is ignored.

***CURRENT:** The logged data for the current day, and the time that has elapsed between the specified starting time and ending time for the day, is displayed.

end-date: Specify the date that the data logging ends. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

HARDWARE

Specifies the name of the problem log entries identifying the failing device.

Note: If *ALL is specified, then any value other than *ALL for model and serial number is ignored.

Element 1: Machine Type

***ALL:** Entries are shown regardless of which device, if any, is identified as failing.

type-code: Specify the 4-character type code of the device.

Element 2: Model Number

***ALL:** All entries identifying the specified type of failing device are shown.

model-number: Specify the 3-character model number of the device.

Element 3: Serial Number

***ALL:** All entries identifying the specified type and model of the failing device are shown.

serial-number: Specify the serial number of the device in one of the following formats where n is a decimal digit (0-9).

- nnnnn
- nnnnnnn
- nn-nnnnn
- nn-nnnnnnn

RESOURCE

Specifies the name of the failing resource, or all failing resources, for which problem log entries are shown.

***ALL:** Entries are shown regardless of which resource name, if any, is identified by the problem.

resource-name: Specify the name of the resource for which logged entries are being displayed.

LICPGM

Specifies the name of the failing licensed program for which problem log entries are shown.

Element 1: Licensed Program

***ALL:** All licensed programs are shown regardless of whether any are identified as failing.

Note: If *ALL is specified, any value other than *ALL for release and modification is ignored.

licensed-program-name: Specify the name of the failing licensed program.

Element 2: Release Level of the Licensed Program

***ALL:** All entries identifying failing licensed programs are shown.

release-level: Specify the release level of the failing licensed program.

Element 3: Modification Number of the Release

***ALL:** All entries identifying a failing licensed program of the specified program product and release are shown.

modification-number: Specify the modification number of the release of the failing licensed program.

FUNCTION

Specifies the name of the function identifier for which the problem log entries are shown.

***ALL:** Entries are shown regardless of which function identifier, if any, is identified.

generic-function-identifier:* Specify the generic name of the function identifier. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

function-identifier: Specify the complete function identifier of the function. If blank characters are included, the character string must be enclosed in apostrophes.

PGM

Specifies that only problem log entries identifying the failing program are shown. For machine-detected problems, the failing program is identified by the possible cause with the highest probability of failure.

***ALL:** Entries are shown regardless of which program, if any, is identified.

generic-program-name:* Specify the generic name of the program.

program-name: Specify the name of the program.

MSGID

Specifies the message or range of messages for which the problem log entries are shown. This is the problem message identifier shown in the problem details display. For user-detected problems, the message identifier is entered by the user.

***ALL:** Entries are shown regardless of which problem message identifier is associated with a problem.

generic-message-identifier:* Specify the generic name of the message identifier.

message-identifier: Specify the problem message identifier.

ORIGIN

Specifies the nodes for which the problem log entries are shown.

Element 1: Network Identifier

***ALL:** Entries are shown regardless of the network identifier of the origin system of each entry.

***NETATR:** Only entries that originated on systems with the same local network identifier as the one defined in the network attributes for this system are shown.

network-identifier: Specify a network identifier. Only entries originating on systems with this local network identifier are shown.

Element 2: Control Point Name

***ALL:** All entries originating on systems using the network identifier are shown.

***NETATR:** Only entries originating on systems with the same control point name as the one defined in the network attributes are shown.

control-point-name: Specify the name of the control point. Only entries originating on systems with this control point name are shown.

SRVID

Specifies that only problem log entries using the IBM service-assigned number are shown. This number is assigned when the problem is reported to IBM service support.

***ALL:** Entries are shown regardless of which service number, if any, is assigned to the problem.

service-identifier-number: Specify the service identifier number.

ASNUSER

Specifies which problem log entries are displayed.

***ALL:** All problem log entries are shown, regardless of the user name assigned to them.

user-name: Specify the user name assigned to the problem log entries to be displayed.

GROUP

Specifies the group in the filter to which the problem is assigned.

***ALL:** All problem log entries are shown, regardless of the group assigned to them.

group-name: Specify the 10-character problem filter group assigned to the entry.

Note: The values are blank if problem log filtering is not used.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file.

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file that receives the output of the command. If the file does not exist, this command creates a database file in the specified library. If a file is created, the text reads "Outfile for DSPPRB," and the authority for users other than those who have specific authority, group authority, or authority from an authorization list is *EXCLUDE. Refer to Table 48, Table 49, Table 50, and Table 51 for the format used.

Note: The format must be the same as that of the TYPE parameter.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file to which the output of the command is written.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, records may be either added to the end of the member or replace existing records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

TYPE

Specifies the type of problem information that is placed in the output file. Only one type of data can be placed in each file. This parameter is valid only when a value is also specified for the OUTFILE parameter.

***BASIC:** Basic problem data includes problem type and status; device type, model, and serial number; product ID, contact information; and tracking data.

***CAUSE:** Possible problem causes include point-of-failure, partial, isolation, or answer cause records. Answer causes are used if they are available. If they are not, isolation causes are used. If isolation causes are not available, partial causes are used. If partial causes are not available, then the point-of-failure causes are used.

***FIX:** Fixes to the problem are program temporary fix (PTF) records.

***USRTXT:** User-supplied text, consisting of note records, is placed in the output file.

***SPTDTA:** Output files with supporting data identifiers are created.

Basic Problem Data (*BASIC)

The output file produced for type *BASIC has one record for each problem in the problem log. Each record has a unique problem identifier.

The record format is QSXPBOF. See Table 48 for the fields in the record.

Table 48 (Page 1 of 3). Fields for Basic Problem Data

Field Name	Type	Length	Description
PBRCEN	C	1	Century that the data was retrieved and placed in the output file: **0** 20th century **1** 21st century
PBRDAT	C	6	Date that the data was retrieved and placed in the output file.
PBRTIM	C	6	Time that the data was retrieved and placed in the output file.
PBSYSN	C	8	Name of the system in which the DSPPRB command is run to create this output file.
PBFORM	C	8	Record format, always *BASIC.

Table 48 (Page 2 of 3). Fields for Basic Problem Data

Field Name	Type	Length	Description
PBID	C	10	Problem identifier.
PBORG	C	20	System origin.
PBORGI	C	1	Problem origin indication: **L** Local **R** Remote
PBSEV	C	1	Problem severity levels: **1** High **2** Medium **3** Low **4** None
PBTYPE	C	1	Problem type (which identifies how the problem was detected): **1** Machine-detected **2** User-perceived **3** PTF order
PBSTAT	C	1	Problem status: **O** Opened **R** Ready **P** Prepared **S** Sent **A** Answered **V** Verified **C** Closed
PBDESC	C	7	Message identifier of a message in the QCPFMSG message file that describes the problem.
PBPMSG	C	7	Message identifier of a message in the QCPFMSG message file that describes the reference code.
PBRFCD	C	4	Original reference code.
PBSLID	C	8	Identifies the entry recorded in the error log when the system detected a problem.
PBMACH	C	4	Identifies the type of device with which this problem is associated.
PBMODL	C	3	Identifies the model of the device with which this problem is associated.
PBMSER	C	10	Identifies the serial number of the device with which this problem is associated.
PBEC	C	12	Identifies the engineering change (EC) level of the device with which this problem is associated.
PBLPP	C	7	Shows the identification number of the program with the problem. This number could be for a licensed program or for licensed internal code.
PBVER	C	4	Identifies the version of the program with the problem.
PBREL	C	2	Identifies the release level of the program with the problem.
PBMDF	C	2	Identifies the modification level that applies to the release level of the program product with the problem.
PBCNT	C	36	The name of the person whom service personnel should contact to help resolve this problem.
PBTEL1	C	30	Primary telephone number. The number service support personnel should use to contact the user's company.
PBTEL2	C	30	Alternate telephone number. The number service support personnel should use to contact the user's company if the primary telephone number is not available.
PBENTR	C	36	Company name.
PBSTR1	C	36	First line of street address.
PBSTR2	C	36	Second line of street address.
PBSTR3	C	36	Third line of street address.
PBCITY	C	36	City and state.

Table 48 (Page 3 of 3). Fields for Basic Problem Data

Field Name	Type	Length	Description
PBCOUN	C	20	Country.
PBZIP	C	12	ZIP code.
PBSNUM	C	10	Service assigned number.
PBEPDP	C	8	Identifies the program used to run problem analysis.
PBPDPE	C	2	Identifies the code the system returned after problem analysis is run.
PBISOS	C	1	Identifies the result of running problem analysis: **1** Problem analysis was successful, the problem still exists. **2** Problem analysis was successful, the problem has been resolved. **3** Problem analysis was not successful. **4** Problem analysis was not completed. **5** Problem analysis was partially completed.
PBDATO	C	6	Date the problem was opened.
PBTIMO	C	6	Time the problem was opened.
PBUSRO	C	10	User who opened the problem.
PBDATA	C	6	Date the problem was analyzed.
PBTIMA	C	6	Time the problem was analyzed.
PBUSRA	C	10	User who analyzed the problem.
PBDATP	C	6	Date the problem was prepared.
PBTIMP	C	6	Time the problem was prepared.
PBUSRP	C	10	User who prepared the problem.
PBDATS	C	6	Date the service request was sent.
PBTIMS	C	6	Time the service request was sent.
PBUSRS	C	10	User who sent the service request.
PBDATN	C	6	Date the service request was answered.
PBTIMN	C	6	Time the service request was answered.
PBUSRN	C	10	User who answered the problem.
PBDATF	C	6	Date fix was verified.
PBTIMF	C	6	Time fix was verified.
PBUSRF	C	10	User who verified the fix.
PBDATC	C	6	Date the problem was closed.
PBTIMC	C	6	Time the problem was closed.
PBUSRC	C	10	User who closed the problem.
PBSYMP	C	256	Symptom string.
PBUSRD	C	10	User assigned to the problem.
PBFTRN	C	10	Filter assigned to the problem.
PBFTRL	C	10	Filter library.
PBFTRG	C	10	Group in filter assigned to the problem.
PBALIB	C	10	Library containing APAR data saved for the problem.

Notes:

1. All dates are represented in YYMMDD format where:

   YY Year
   MM Month
   DD Day

   For example 890515 represents May 15, 1989.

2. All times are for a 24-hour clock and are represented in HHMMSS format where:

   HH Hour
   MM Minutes
   SS Seconds

For example 144526 represents 26 seconds past 2:45 PM.

3. The contact name, telephone numbers, company name, and address are only present if different than the service contact information.
4. For those activities that can be done more than once, the date and time represent the last time the activity was done.

Possible Causes (*CAUSE)

The output file produced for type *CAUSE has one record for each possible cause identified at the time the failure was detected or as a result of running problem analysis. The cause list type indicates when the possible causes were identified and is the same for all records with a given problem identifier.

The sum of the possibilities will equal 100 percent for each problem identifier.

The record format is QSXCAOF. See Table 49 for the fields in the record.

Table 49 (Page 1 of 2). Fields for Possible Cause Data

Field Name	Type	Length	Description
CARCEN	C	1	Century that the data was retrieved and placed in the output file: **0** 20th century **1** 21st century
CARDAT	C	6	Date that the data was retrieved and placed in the output file.
CARTIM	C	6	Time that the data was retrieved and placed in the output file.
CASYSN	C	8	Name of the system where the DSPPRB command was run to create this output file.
CAFORM	C	8	Record format, which is always *CAUSE.
CAID	C	10	Problem identifier.
CAORG	C	20	System origin.
CALIST	C	1	Cause list type: **P** The possible cause list was identified when the failure occurred. **L** The possible cause list was identified by problem analysis which was not completed. **I** The possible cause list was identified by problem analysis. **A** The possible cause list was returned by RETAIN as result of a service request.
CACODE	C	4	Cause code.
CANAME	C	7	A message identifier of a message in QCPFMSG that describes a possible cause.
CADOC	C	7	Message identifier of a message in QCPFMSG that identifies the document being referenced when solving this problem.
CADEVL	C	40	Location of the rack or system unit when the device is installed.
CAPRT1	C	12	Part 1, a part which describes a possible hardware problem cause. Up to six part numbers are possible for a part because different part numbers are used to identify the part when it has different features.
CAPRT2	C	12	Part 2
CAPRT3	C	12	Part 3
CAPRT4	C	12	Part 4
CAPRT5	C	12	Part 5
CAPRT6	C	12	Part 6
CATYPE	C	1	Cause type: ***Field*** ***Type (Instances)*** **A, K, N** Hardware (communications controller or power supply) **C** Software (OS/400) **D** Device (terminal, printer, or tape unit) **E** S/370 channel attached device **G** Configuration (controller description parameter incorrect) **H** General (modem or cable) **M** Media (tape or diskette) **U** User (door open or diskette upside down)
CACHNC	P	3,0	The possibility that this part or action will correct the problem (1-100%).
CADEVN	C	10	Name of the device where the part is installed.

Table 49 (Page 2 of 2). Fields for Possible Cause Data

Field Name	Type	Length	Description
CAMACH	C	4	Type of device.
CAMODL	C	3	Model number of device.
CAMSER	C	10	Serial number of the device.
CARSER	C	10	Serial number of the rack where the device is installed.
CAADDR	C	9	Address (1234-5678)
CALPP	C	7	Identifies a licensed program product or licensed internal code. Only present for cause type 'C'.
CAVER	C	4	Version of the program product. Only present for cause type 'C'.
CAREL	C	2	Release level of the program product. Only present for cause type 'C'.
CAMDF	C	2	Modification level that applies to the release of the program product. Only present for cause type 'C'.
CAMOD	C	10	Module in the program product. Only present for cause type 'C'.
CAVOL	C	8	Removable media volume identifier. Only present for cause type 'M'.
CARSRC	C	10	Resource name assigned to the device by the system.
CAPORT	C	2	Port where the work station cable connects to the work station attachment.
CAPLOC	C	25	Location of the part in the device.
CAPCOD	C	4	Symptom code, also called the Fault Symptom Code (FSC). It is generated by the sense bits collected in the hardware device at the time of failure. Only present for cause type 'E'.
CASENS	C	64	Sense data bytes are made up of the bits that are collected in the hardware device to show the status of the device. Only present for cause type 'E'.
CAEADDR	C	13	Address of device where part is installed. **1-4** Identifies the device controller card. **5** Separator character (-). **6-13** Identifies the failing device or resource.

Program Fixes (*FIX)

The output file produced for type *FIX has one record for each program temporary fix (PTF) identified as a result of a service call for a machine-detected or user-perceived problem. Multiple program fix records may have the same problem identifier.

The record format is QSXFXOF. See Table 50 for the fields in the record.

Table 50 (Page 1 of 2). Fields for Program Fix Data

Field Name	Type	Length	Description
FXRCEN	C	1	Century that the data was retrieved and placed in the output file: **0** 20th century **1** 21st century
FXRDAT	C	6	Date that the data was retrieved and placed in the output file.
FXRTIM	C	6	Time that the data was retrieved and placed in the output file.
FXSYSN	C	8	Name of the system where the DSPPRB command was run to create this output file.
FXFORM	C	8	Record format, which is always *FIX.
FXID	C	10	Problem identifier.
FXORG	C	20	System origin.
FXPTF	C	7	Program temporary fix (PTF) number.
FXPST	C	1	Program temporary fix (PTF) status: **N** Not requested. **R** Requested.
FXSFS	C	1	PTF save file status: "Y" = yes; "N" = no; "R" = yes (released)

Table 50 (Page 2 of 2). Fields for Program Fix Data			
Field Name	**Type**	**Length**	**Description**
FXPRD	C	7	PTF product ID or special value *ONLY. *ONLY represents PTFID(*ONLYPRD) specified on the SNDPTFORD command.
FXRLS	C	6	PTF release level (VxRxMx) or special value *ONLY. *ONLY represents PTFID(*ONLYRLS) specified on the SNDPTFORD command.

User Text (*USRTXT)

The output file produced for type *USRTXT has one record for each line of user-entered text (notes). Multiple text records may have the same problem identifier.

The record format is QSXTXOF. See Table 51 for the fields in the record.

Table 51. Fields for User Text Data			
Field Name	**Type**	**Length**	**Description**
TXRCEN	C	1	Century that the data was retrieved and placed in the output file: **0** 20th century **1** 21st century
TXRDAT	C	6	Date that the data was retrieved and placed in the output file.
TXRTIM	C	6	Time that the data was retrieved and placed in the output file.
TXSYSN	C	8	Name of the system where the DSPPRB command was run to create this output file.
TXFORM	C	8	Record format, which is always *USRTXT.
TXID	C	10	Problem identifier.
TXORG	C	20	System origin.
TXTYPE	C	1	Type of text: **N** Note **S** Problem description
TXTEXT	C	80	User-entered text.

Supporting Data Identifiers (*SPTDTA)

The output file produced for type *SPTDTA has one record for each item identified. Multiple records may have the same problem identifier. Items identified contain information related to the problem. The data identifier type indicates the type of item being identified. The information in other fields is dependent on the type of item being identified.

The record format is QSXSDOF. See Table 52 for the fields in the record.

Table 52 (Page 1 of 2). Fields for Supporting Data Identifier Data			
Field Name	**Type**	**Length**	**Description**
SDRCEN	C	1	Century that the data was retrieved and placed in the output file: **0** 20th century **1** 21st century
SDRDAT	C	6	Date that the data was retrieved and placed in the output file.
SDRTIM	C	6	Time that the data was retrieved and placed in the output file.
SDSYSN	C	8	Name of the system where the DSPPRB command was run to create this output file.
SDFORM	C	8	Record format, which is always *SPTDTA.
SDID	C	10	Problem identifier.
SDORG	C	20	System origin.
SDTYPE	C	1	Type of identifier: **S** Spooled file identifier
SDNAM1	C	10	Identifier name 1: **Type S** Spooled file name

Table 52 (Page 2 of 2). Fields for Supporting Data Identifier Data

Field Name	Type	Length	Description
SDNAM2	C	10	Identifier name 2: **Type S** Job name
SDNAM3	C	10	Identifier name 3: **Type S** User name
SDJNBR	C	10	Job number
SDFNBR	C	10	Spooled file number

Examples

Example 1: Displaying Today's Problem Log Entries

```
DSPPRB  PERIOD(*AVAIL  *CURRENT)  (*AVAIL  *CURRENT)
```

This command shows all problem log entries that were created today.

Example 2: Creating an Output File

```
DSPPRB  OUTPUT(*OUTFILE)  OUTFILE(*CURLIB/NEWFILE)
```

This command creates a member in the physical file NEWFILE in the current library which contains a record for each problem log entry in the problem log.

Example 3: Displaying a List of Hardware Problems

```
DSPPRB  SEV(1 2)  HARDWARE(9347 001 10-7523489)
```

This command shows a list containing problems with the hardware specified by the user. The user has specified that the command track medium-to-high levels of severity.

Example 4: Displaying a List of Problems That Have Been Opened

```
DSPPRB  STATUS(*OPENED)  PERIOD((*AVAIL  *CURRENT)
  (120000  *CURRENT))  LICPGM(5728SS1  03  00)
  PGM(QNOPGM)
```

This command shows a list containing problems that have been opened during the period starting at midnight and ending at noon on the current day, and have not yet been analyzed. This command also identifies the specified program product identifier and program name as the probable cause of the failure.

Example 5: Displaying a List of Machine-Detected Problems

```
DSPPRB  RESOURCE(TAP01)  MSGID(CPF6788)
```

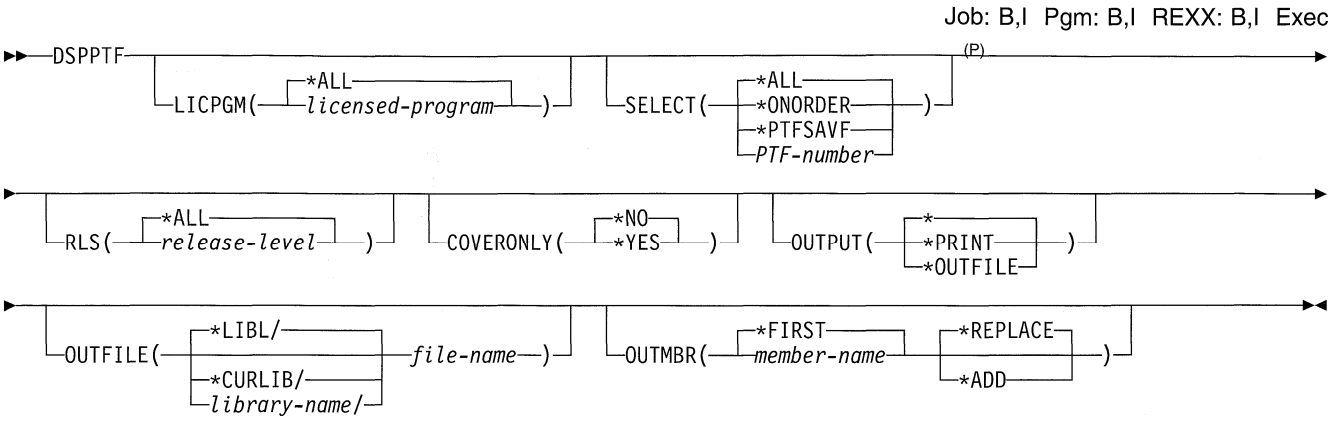
This command shows a list containing machine-detected problems that were opened due to the message, CPF6788, having been sent to the system operator message queue. The list of problems includes user-detected problems. To get the user-detected problems, the user specified the resource name and message identifier by using the Analyze Problem (ANZPRB) command.

Example 6: Displaying a List of Reported Problems

```
DSPPRB  SRVID(12345)
```

This command shows a list containing problems that have been reported to an IBM service support center and have 12345 as the service identifier.

DSPPTF (Display Program Temporary Fix) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Program Temporary Fix (DSPPTF) command shows the program temporary fixes (PTFs) for a specified licensed program.

Restriction: This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.

Optional Parameters

LICPGM

Specifies the licensed program for which PTFs are shown. When LICPGM(*ALL) is specified, PTFs for all licensed programs are shown.

***ALL:** The PTFs of all licensed programs are shown.

licensed-program: Specify the licensed program for which PTFs are shown.

SELECT

Specifies which PTF is shown for the specified licensed program. *ALL cannot be specified for the LICPGM parameter if a PTF number is specified on the SELECT parameter.

When LICPGM(*ALL) and SELECT(*ALL) are specified, all PTFs for all licensed programs are shown. Also, when SELECT(*ALL) is specified, PTFs in QGPL library save files and PTFs that have been ordered are shown.

When LICPGM(*licensed-program*) and SELECT(*ALL) are specified, PTFs for all releases of the specified licensed program are shown.

***ALL:** The status of all PTFs for the specified licensed program is shown.

***ONORDER:** The PTFs that are currently on order are shown. This special value allows both a base system user and a service provider to get a list of PTFs that are currently on order. This special value is valid for PTFs both on order and on the system, that is, in the QGPL library save file or in a loaded status.

***PTFSAVF:** Only PTFs for which save files exist within the QGPL library are shown. This special value is useful to the base system user for determining which PTF save files are no longer needed. It is also useful for the service provider for determining which PTFs can be distributed.

PTF-number: Specify the PTF identification number of the PTF that is shown.

RLS

Specifies the release level of the PTFs being displayed.

***ALL:** All PTFs are displayed.

release-level: Specify the release level in the format VxRxMx, where Vx is the version number, Rx is the release number, and Mx is the modification number.

COVERONLY

Specifies whether a cover letter is displayed.

***NO:** A cover letter is not displayed.

***YES:** A cover letter is displayed.

OUTPUT

Specifies whether the output from this command is displayed, printed, or directed to a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output. The name of the spooled file is QSYSPRT.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the physical database file where the PTFs are directed. If the output file already exists, the system attempts to use it. Existing data in the file member is replaced. If the output file does not exist, the system creates a physical database file (with the name specified in the OUTFILE parameter) in the designated library. A member is created for the file with the name specified in the OUTMBR parameter. More information about OUTFILE is in the table at the end of this command.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file that receives the PTFs.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

Note: The output file produced has one record for each PTF. The record format, QSCPTF, is described in the table at the end of this command.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Printing Status of PTFs

```
DSPPTF  LICPGM(5738SS1)  OUTPUT(*PRINT)
```

This command produces a printout containing the status of PTFs for the licensed program 5738-SS1.

Example 2: Printing Information

```
DSPPTF  LICPGM(5738SS1)  SELECT(SF00034)  OUTPUT(*PRINT)
```

This command produces a printout containing detailed information about PTF SF00034 for licensed program 5738-SS1.

Table 53 (Page 1 of 3). Fields for Display PTF

Field name	Type	Length	Description
SCCENTRY	C	1	Century the data was retrieved and placed in the output file: **0** 20th century **1** 21st century
SCDATE	C	6	Date the data was retrieved and placed in the output file. Date is in the format YYMMDD.
SCTIME	C	6	Time the data was retrieved and placed in the output file. Time is in the format HHMMSS.
SCSRLNBR	C	8	Serial number of the system where the DSPPTF command was run to create this output file.
SCXPFID	C	7	ID of the OS/400 LPP that produced this OUTFILE. **5738SS1** ID for release 3 of OS/400.
SCXPFREL	C	3	Not used for V2R1
SCXPFMOD	C	3	Not used for V2R1
SCXPFDAT	C	6	Release date of the OS/400 LPP that produced this OUTFILE. Date is in the format YYMMDD.
SCLIB	C	10	Name of library containing PTF information for LPP that this PTF is for.
SCPPID	C	7	LPP ID of LPP that this PTF is for.
SCPTFID	C	7	PTF ID.
SCPPREL	C	6	Release and modification of LPP that this PTF is for.

Table 53 (Page 2 of 3). Fields for Display PTF

Field name	Type	Length	Description
SCSTATUS	C	37	Status of this PTF. The following values are the English version of what may appear in this field. **On order only** PTF is on order only - no PTF save file exists and PTF has not been loaded. **Save file only** PTF save file exists but PTF has not been loaded. **Not applied** PTF has been loaded but not applied. **Temporarily applied** PTF is temporarily applied **Permanently applied** PTF is permanently applied **Superseded** PTF has been superseded by another PTF. See field SCSUPPTF for the superseding PTF ID. **Damaged** PTF is damaged and needs to be reloaded.
SCSTDATE	C	8	Date of last PTF activity. Date is in the format specified by the DATFMT and DATSEP attributes of the job invoking the DSPPTF command. This field will be all blanks when this information is not available.
SCSTTIME	C	8	Time of last PTF activity. Time is in the format HH:MM:SS:. This field will be all blanks when this information is not available.
SCTYPE	C	18	Type of this PTF. The following values are the English version of what may appear in this field. **all blanks** Information is not available **Delayed** PTF may only be applied at IPL time. **Immediate** PTF may be applied immediately or at IPL time.
SCIPLACT	C	34	Unattended IPL action. The following values are the English version of what may appear in this field. **None** No action will occur. **Apply temporarily** PTF will be temporarily applied during the next unattended IPL. **Apply permanently** PTF will be permanently applied during the next unattended IPL. **Remove temporarily** PTF will be temporarily removed during the next unattended IPL. **Remove permanently** PTF will be permanently removed during the next unattended IPL.
SCLANG	C	8	Language feature that this PTF is for. The following values are the English version of what may appear in this field. **all blanks** Information is not available **None** PTF is a code PTF. **29nn** Identifier of language that PTF is for. "nn" is a two digit number that identifies the specific language.
SCOBJLIB	C	10	Name of library that contains PTF objects. This field will be all blanks when this information is not available.
SCISRC	C	8	Which copy of licensed internal code is currently in use. **##MACH#A** Copy "A" is in use. **##MACH#B** Copy "B" is in use.
SCSYSNM	C	8	System name.
SCONOR	C	1	On order status. **Y** PTF is currently on order. **N** PTF is not on order.
SCSAVF	C	1	PTF save file status. **Y** A PTF save file exists for this PTF. **N** A PTF save file does not exist for this PTF. **R** A PTF save file exists for this PTF and the PTF is distributable.
SCOPTP	C	4	ID of LPP optional part that this PTF is for. **all blanks** Information is not available **0000** PTF is for the LPP base. **0001-0025** PTF is for an optional part of the LPP. The number is the ID of the optional part.

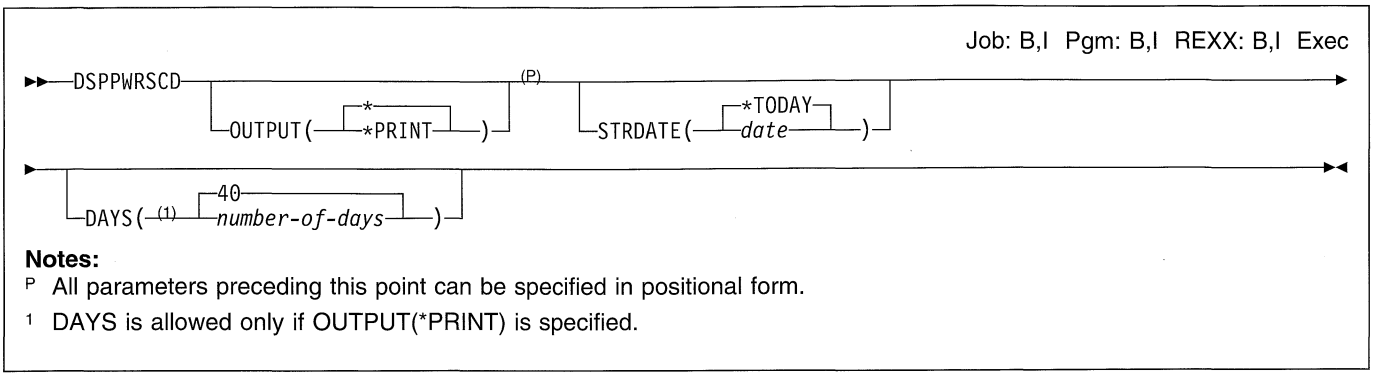
Table 53 (Page 3 of 3). Fields for Display PTF

Field name	Type	Length	Description
SCSPTF	C	7	Superseding PTF ID. This field is all blanks if there is no other PTF that supersedes this PTF.
SCENST	C	1	Encoded PTF status. This field has the same meaning as SCSTATUS, but contains an encoded value. **O** PTF is on order only - no PTF save file exists and PTF has not been loaded. **F** PTF save file exists but PTF has not been loaded. **N** PTF has been loaded but not applied. **T** PTF is temporarily applied **P** PTF is permanently applied **S** PTF has been superseded by another PTF. See field SCSUPPTF for the superseding PTF ID. **D** PTF is damaged and needs to be reloaded.
SCENDT	C	6	Date of last PTF activity. This field has the same meaning as SCSTDATE, but contains the date in a YYMMDD format. This field will be all blanks when this information is not available.
SCENTP	C	1	Encoded PTF type. This field has the same meaning as SCTYPE, but contains an encoded value. **blank** Information is not available **D** PTF may only be applied at IPL time. **I** PTF may be applied immediately or at IPL time.
SCENIA	C	1	Encoded unattended IPL action. This field has the same meaning as SCIPLACT, but contains an encoded value. **blank** No action will occur. **1** PTF will be temporarily applied during the next unattended IPL. **2** PTF will be permanently applied during the next unattended IPL. **3** PTF will be temporarily removed during the next unattended IPL. **4** PTF will be permanently removed during the next unattended IPL.
SCENLG	C	4	Encoded language feature. This field has the same meaning as SCLANG, but contains an encoded value. **all blanks** Information is not available **CODE** PTF is a code PTF. **29nn** Identifier of language that PTF is for. "nn" is a two digit number that identifies the specific language.
SCXPFV	C	6	Version release modification ID of the OS/400 LPP that produced this OUTFILE. **V2R1M1** ID for version 2 release 1 of OS/400.
SCPTFV	C	6	Version Release modification ID of the OS/400 LPP that produced this OUTFILE. **V2R1M0** ID for version 2 release 1 of OS/400.
SCMCGG	C	7	Licensed Internal Code group to which this fix applies.

Note: Information contained in a field may not be available for one or more of the following reasons:

1. The PTF has not been loaded.
2. The PTF is not for a supported product.

DSPPWRSCD (Display Power On/Off Schedule) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 DAYS is allowed only if OUTPUT(*PRINT) is specified.

Purpose

The Display Power On/Off Schedule (DSPPWRSCD) command is used to display or print the power on/off schedule.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

STRDATE

Specifies the first date to be displayed or printed on the power on/off schedule.

***TODAY:** The current date is used.

date: Specify the first date to be displayed. The date must be specified in the format specified on the job attributes. Specify the current date or a date in the future.

DAYS

Specifies the number of days for which the power on/off schedule is to be printed.

40: Forty days of the report are printed.

number-of-days: Specify the number of days. Valid values range from 1 through 366.

Examples

Example 1: Displaying Power On/Off Schedule

```
DSPPWRSCD
```

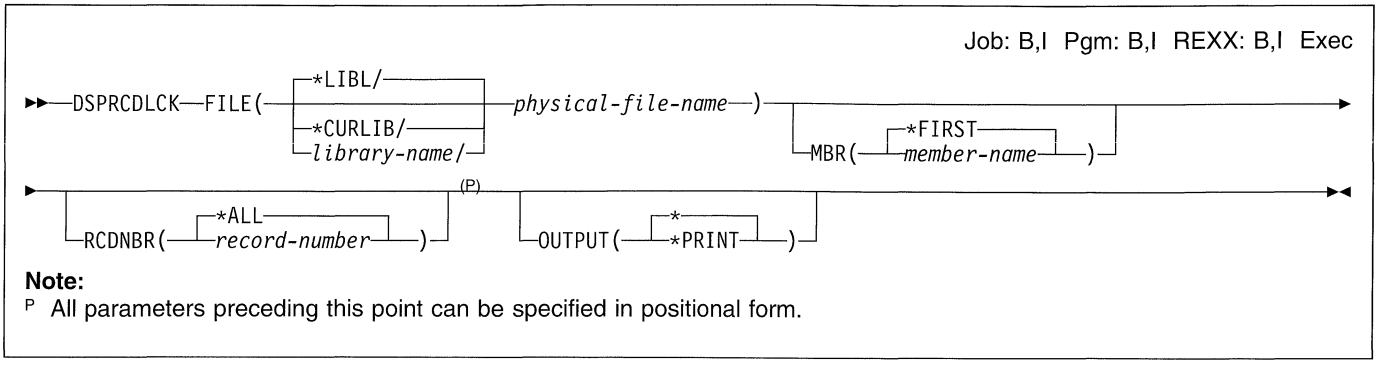
This command displays the power on/off schedule.

Example 2: Printing 30 Days of Power On/Off Schedule

```
DSPPWRSCD   OUTPUT(*PRINT)   DAYS(30)
```

This command prints 30 days of the power on/off schedule, starting with the current date.

DSPRCDLCK (Display Record Locks) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Record Locks (DSPRCDLCK) command allows the user to show the current record lock status of a particular database physical file member. This command displays the lock status for a particular relative record number, or the lock status of all locked records in the member.

Required Parameters

FILE

Specifies the qualified name of the physical file that contains the member whose record locks are shown on the display.

| The name of the physical file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

physical-file-name: Specify the name of the physical file.

Optional Parameters

MBR

Specifies the name of the member in the file whose record locks are shown on the display.

***FIRST:** The first member in the database file is used.

member-name: Specify the name of the physical file member that is shown.

RCDNBR

Specifies a particular relative record number or all records of a member.

***ALL:** The lock status of all records currently locked in a physical file member is shown on the display.

record-number: Specify the number of the record whose lock status is shown on the display.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

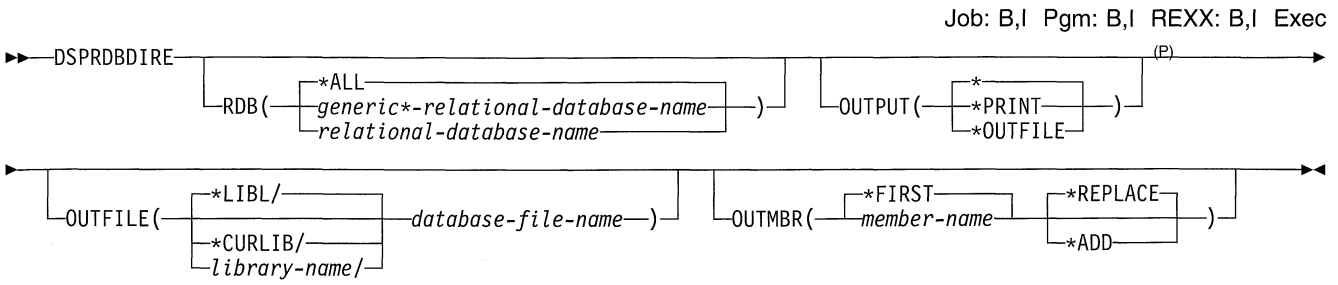
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPRCDLCK   FILE(MASTER/PAYROLL)  MBR(*FIRST)
  RCDNBR(1)  OUTPUT(*)
```

This command shows the lock status of relative record number 1, in the first member of the physical file named PAYROLL in the MASTER library.

DSPRDBDIRE (Display Relational Database Directory Entry) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Relational Database Directory Entry (DSPRDBDIRE) command is used to display one entry, generic entries, or all entries in the relational database directory.

Optional Parameters

RDB

Specifies the relational database directory entries being displayed.

***ALL:** All entries in the relational database directory are displayed.

generic-relational-database-name:* Specify a generic relational database name. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

relational-database-name: Specify the name of the relational database being displayed.

OUTPUT

Specifies whether the relational database directory information is displayed at the requesting work station, printed, or directed to an output file.

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The requested information is printed with the job's spooled output.

***OUTFILE:** The requested information is stored in an output file.

OUTFILE

Specifies the qualified name of the output file to which the requested information is directed. If the file does not exist, it is created. If a new file is created, system file QADSPDE in system library QSYS with a record format name of RWRDDSP is used as a model. If the file already exists, it must have this format. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the file in which the requested information is to be located.

OUTMBR

Specifies the output file member in which the requested relational database information is to be located. If the member does not exist, it is created. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

Element 1: Member to Receive Output

***FIRST:** The requested information is stored in the first file member.

member-name: Specify the name of the output file member in which the requested information is to be located. The member name can be from 1 to 10 characters in length.

Element 2: Operation to Perform on Member

***REPLACE:** If the member exists, the system clears it and adds the new information.

***ADD:** If the member exists, the system adds the new information to the existing records.

Examples

Example 1: Directing Information to an Output File

```
DSPRDBDIRE   OUTPUT(*PRINT)
```

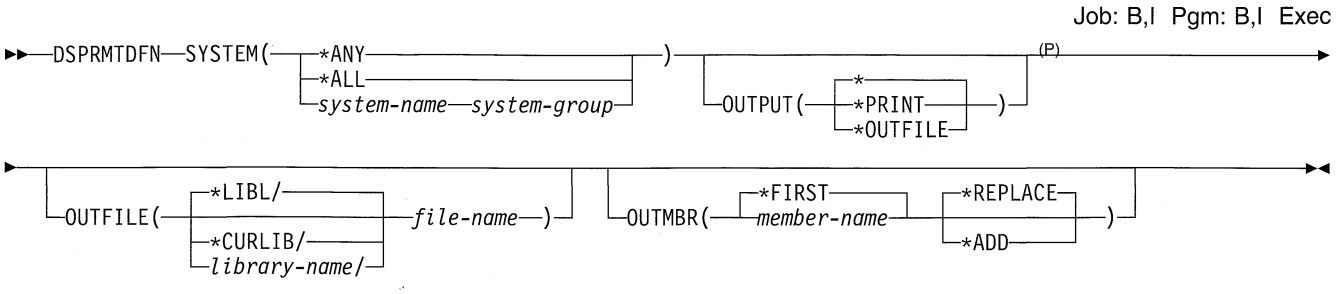
This command directs information from all of the relational database directory entries to a spooled file.

Example 2: Directing Information to an Output File

```
DSPRDBDIRE   OUTPUT(*OUTFILE)  OUTFILE(SAVEDIR)
```

This command directs all of the relational database directory entries to an output file named SAVEDIR. This is the usual method for backing up the contents of the relational database directory. The entries can be restored using a CL program that reads the information from the output file and issues Add Relational Database Directory Entry (ADDRDBDIRE) commands to add the information back into the relational database directory.

DSPRMTDFN (Display Remote Definition) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Remote Definition (DSPRMTDFN) command allows the user to display or print remote definitions for a system. The output can be displayed, printed, or directed to a database file.

Required Parameters

SYSTEM

Specifies the system name and system group of the remote system being displayed.

***ANY:** Displays the default definition for a remote system not covered by the other entries.

***ALL:** Displays all definitions for remote systems.

Element 1: System Name

system-name: Specify the name of the remote system to be displayed.

Element 2: System Group

system-group: Specify the group name of the remote system to be displayed. Do not specify this value if the group name is blank.

Optional Parameters

OUTPUT

Specifies whether the output from this command is displayed, printed, or directed to a database file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file to which the output of this command is directed. If the file does not exist, the system creates a file in the specified library. If a new file is created, system file QAOCRMTDFN in system library QSYS with a record format name of RMTDFN is used as a model. If the file already exists, it must have this format.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the output file.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The output data replaces existing records in the specified member.

***ADD:** The output data is added after existing records in the specified member.

Examples

Example 1: Displaying a Specific Remote Definition

```
DSPRMTDFN  SYSTEM(RCHAS1)
```

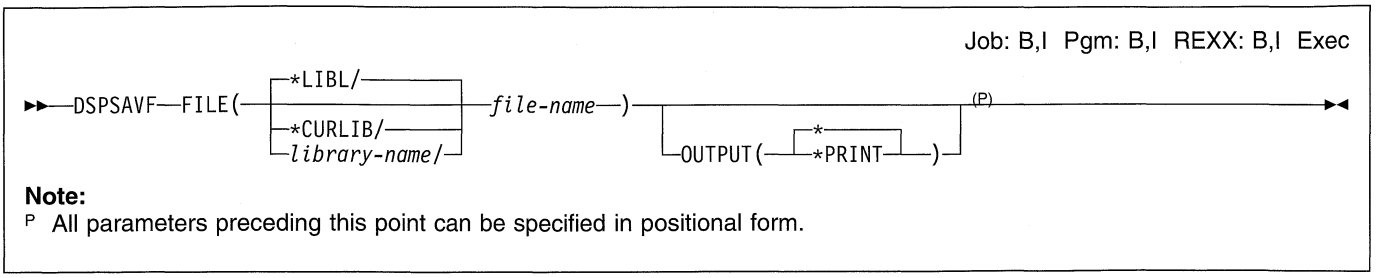
This command displays the current attributes for remote system RCHAS1.

Example 2: Writing a Definition to an Output File

```
DSPRMTDFN  SYSTEM(*ALL)  OUTPUT(*OUTFILE)
  OUTFILE(RMTDFNOUT)
```

This command writes the current attributes for all defined remote systems to the output file RMTDFNOUT.

DSPSAVF (Display Save File) Command



Purpose

The Display Save File (DSPSAVF) command displays the save information in a save file. This information includes a description of each object saved to the save file and summary information.

The information can be written to a printer or shown on a display device.

Restrictions: The user of this command must have use authority for the save file and read authority for the specified library.

Required Parameters

FILE

Specifies the qualified name of the save file to be shown on the display.

The name of the save file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the file.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

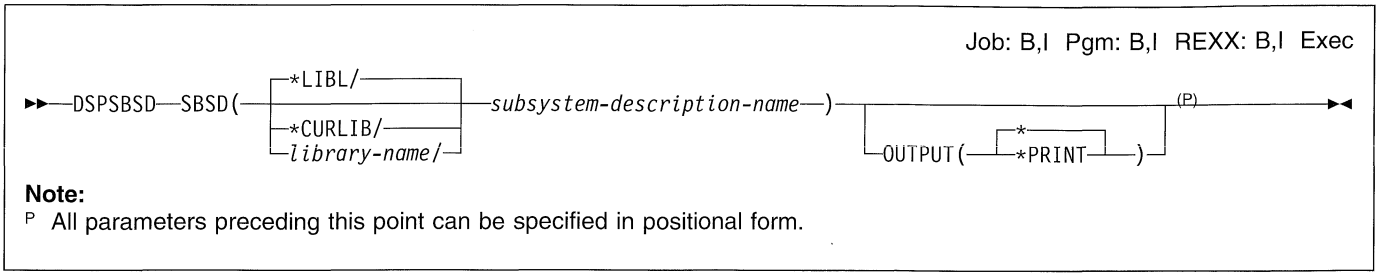
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPSAVF   FILE(ONLINE)  OUTPUT(*PRINT)
```

This command shows the objects saved to save file ONLINE. The output is printed with the job's spooled output.

DSPSBSD (Display Subsystem Description) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The Display Subsystem Description (DSPSBSD) command displays the information contained in a subsystem description. The types of information, which are shown on separate displays, include the following: operational attributes, pool definitions, autostart job entries, work station entries (by name and type), job queue entries, routing entries, communications entries, and remote location name entries. If this command is entered in a batch job, the information for *all* the types is printed with the job's spooled output.

Restriction: The user must have object operational and read authority for the subsystem description before its contents can be shown.

Required Parameters

SBSD

Specifies the qualified name of the subsystem description that is to be displayed.

The name of the subsystem description can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

subsystem-description-name: Specify the name of the subsystem description to be displayed.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

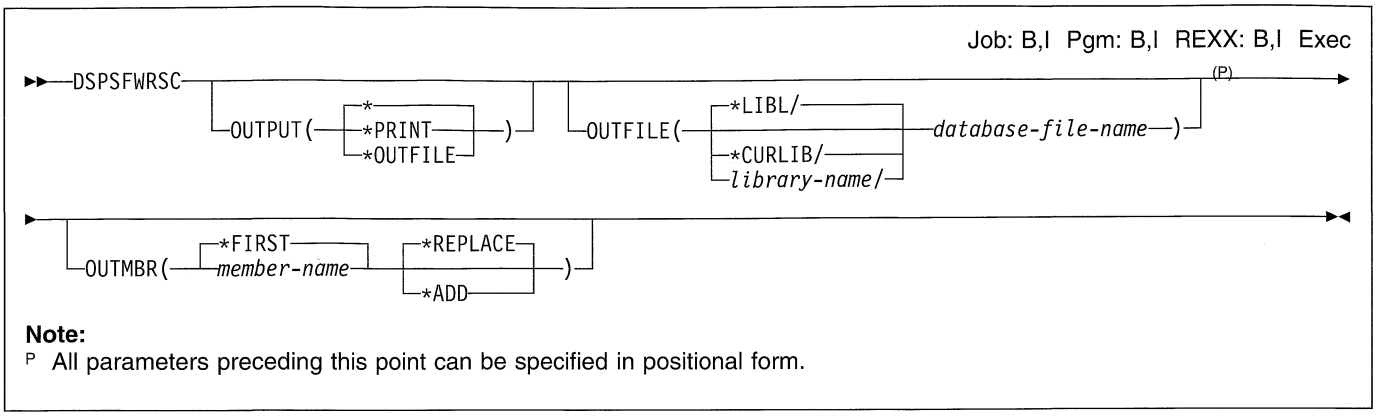
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPSBSD  SBSD(LIB6/ORDER)  OUTPUT(*)
```

This command (if entered from a batch job) sends a complete set of display information about the subsystem description named ORDER (stored in LIB6 library) to the job's spooling queue for printing. The information includes the subsystem's attributes, all of the job entries, and all of the routing entries currently in the subsystem description. If the command is entered in an interactive job, the subsystem description menu is shown on a display from which an option may be chosen.

DSPSFWRSC (Display Software Resources) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The DSPSFWRSC (Display Software Resources) command allows the user to show, print, or write to an output file the list of installed software resources.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station, printed with the job's spooled output, or directed to a database file.

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

Note: If OUTPUT(*OUTFILE) is used, the name of the database file is required.

OUTFILE

Specifies the qualified name of the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library. If a new file is created, the system uses the physical file QARZLCOF in library QSYS as a model file. The file has a record format name of QARZLCGD. Field level information can be obtained using the Display File Field Description (DSPFFD) command and specifying QARZLCOF as the file name and QSYS as the library. If this function creates the file, the text says output file for DSPSFWRSC, and the authority for users is *EXCLUDE (except for users who have specific authority, group authority, or authority from an authorization list). This parameter is valid only if OUTPUT(*OUTFILE) is specified.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file to which the output of the command is directed.

OUTMBR

Specifies the name of the database file member to which the output is directed. This parameter is valid only if OUTPUT(*OUTFILE) is specified.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the records may be added to the end of the existing member or the existing member may be cleared and the records added at the top of the member.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

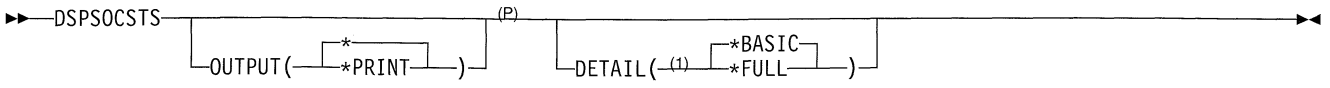
***ADD:** The system adds the new records to the end of the existing records.

Example

```
DSPSFWRSC  OUTPUT(*OUTFILE)  OUTFILE(*CURLIB/NAMES)
```

This command sends the output from the command to the first member of the file NAMES in the current library. The output replaces the information in the member.

DSPSOCSTS (Display Sphere of Control Status) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 This parameter is valid only when OUTPUT(*PRINT) is specified.

Purpose

The Display Sphere of Control Status (DSPSOCSTS) command shows the status of the sphere of control including primary, default, backup, and requested nodes.

More information on sphere of control is in the *Alerts and DSNX Guide*.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requested work station or printed with the job's spooled printer output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

DETAIL

Specifies whether the output from the command is printed as a list of nodes or as a detailed description of each node in the sphere of control.

***BASIC:** A list of nodes in the sphere of control is printed.

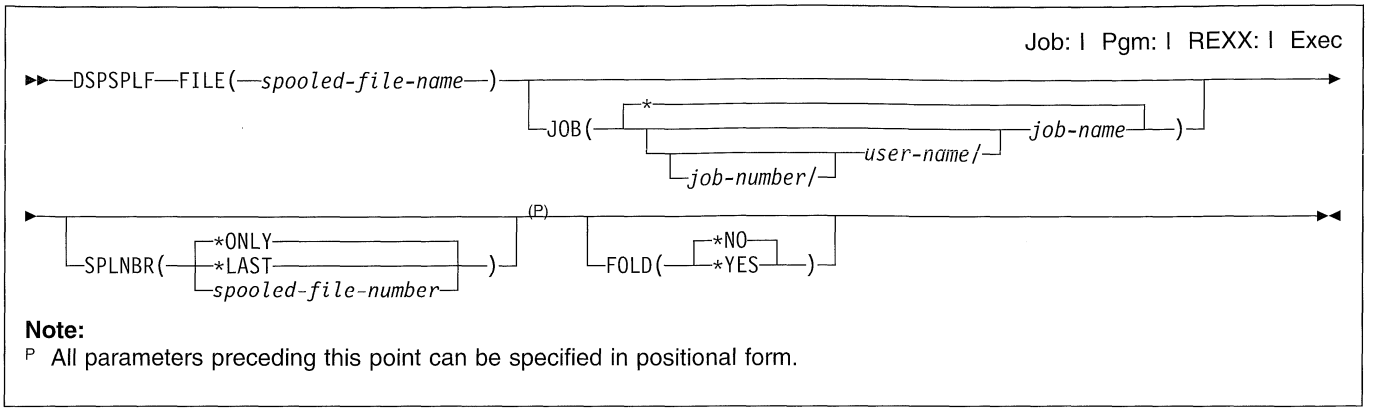
***FULL:** A list of nodes in the sphere of control with detailed information for each node is printed.

Example

```
DSPSOCSTS
```

This command shows the Sphere of Control Status display at the requesting work station.

DSPSPLF (Display Spooled File) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Spooled File (DSPSPLF) command shows the data records in the specified spooled file. The current contents of the file (data records) can be displayed for spooled files which do not contain Advanced Function Printing* Data Stream (AFPDS) data or ASCII data. The display provides various functions to show the file and to scan for a specific character string.

Required Parameters

FILE

Specifies the name of the spooled file whose records are to be displayed.

Optional Parameters

JOB

Specifies the qualified name of the job and consists of as many as three elements. For example:

job-name
user-name/job-name
job-number/user-name/job-name

*N may be used in place of the user-name element to maintain position in the sequence. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** The job that issued this DSPSPLF command is the job that created the spooled file.

job-name: Specify the name of the job that created the spooled file.

user-name: Specify the name of the user of the job that created the spooled file.

job-number: Specify the number of the job that created the spooled file.

SPLNBR

Specifies the number of the job's spooled file that is being displayed. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ONLY:** One spooled file from the job has the specified file name. The number of the spooled file is not necessary. If *ONLY is specified and more than one spooled file has the specified file name, a message is sent.

***LAST:** The highest-numbered spooled file with the specified file name is displayed.

spooled-file-number: Specify the number of the spooled file having the specified file name whose data records are displayed.

FOLD

Specifies whether the records in the first display are folded if they are longer than the length of the display line.

***NO:** The records are not folded. When the length of the record is longer than one line, the remaining positions of the record are not shown.

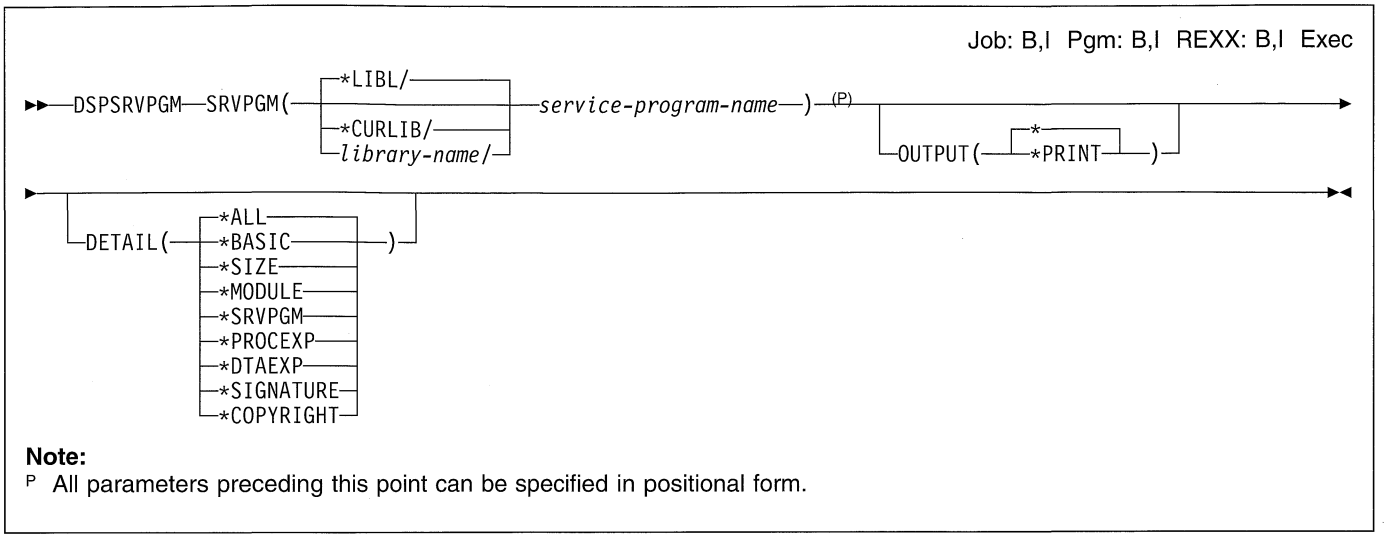
***YES:** The first display shows the first record folded on one or more display lines if it is longer than one line.

Example

```
DSPSPLF  FILE(QPRINT)  JOB(PAYROLL01)  SPLNBR(4)
  FOLD(*NO)
```

In this example, the spooled file QPRINT is displayed. The file is the fourth file produced by the job PAYROLL01. The record positions that are longer than the length of the display line are truncated on the first display.

DSPSRVPGM (Display Service Program) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Service Program (DSPSRVPGM) command displays information about a service program, including the creation and processing attributes of the service program, information about the compiler, and the size of the service program.

Restrictions: You must have *USE authority to the service program being displayed and *READ authority to the library in which the service program exists.

Required Parameters

SRVPGM
Specifies the name of the service program for which information is displayed or printed.

The name of the service program can be qualified by one of the following library values:

__*LIBL:__ All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

service-program-name: Specify the name of the service program for which information is displayed.

Optional Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output.

__*:__ Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

DETAIL
Specifies the type of information displayed for the service program. More than one value can be specified, with the exception of *ALL, which must be specified as a single value.

__*ALL:__ All information types are shown on the display.

***BASIC:** General service program information is shown.

***SIZE:** The size and size limits are shown.

***MODULE:** A list of the module objects bound by this service program is shown.

***SRVPGM:** A list of the service program objects bound by this service program is shown.

***PROCEXP:** A list of the procedures exported from a service program and specified in the binding language is shown. The exports are only for the current signature.

***DTAEXP:** A list of the data items exported from a service program and specified in the binding language is shown. The exports are only for the current signature.

***SIGNATURE:** A list of the signatures is shown. The first signature in the list is the current signature.

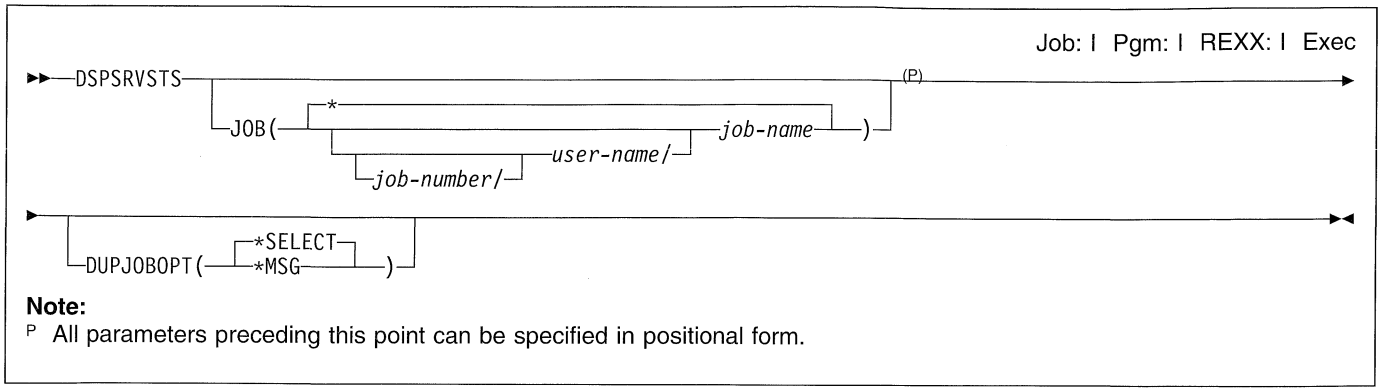
***COPYRIGHT:** A list of the copyrights is shown.

Example

```
DSPSRVPGM   SRVPGM(COACH)
```

This command displays a service program object named COACH.

DSPSRVSTS (Display Service Status) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Service Status (DSPSRVSTS) command shows information about the current service status of the specified job. This includes the name of the job it is servicing and/or the name of the job servicing the specified job.

Restriction: This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.

Optional Parameters

JOB

Specifies which job is having its service status shown on the display. If no user name or job number are given, all of the jobs currently in the system are searched for the simple job name; the name specified must be unique within the system.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
*
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Status information is shown about the job in which the DSPSRVSTS command is entered.

job-name: Specify the name of the job whose status information is shown.

user-name: Specify the name of the user of the job whose status information is shown.

job-number: Specify the number of the job whose status information is shown.

DUPJOBOPT

Specifies the action taken when duplicate jobs are found by this command.

***SELECT:** The selection display is shown when duplicate jobs are found during an interactive session. Otherwise, an escape message is issued.

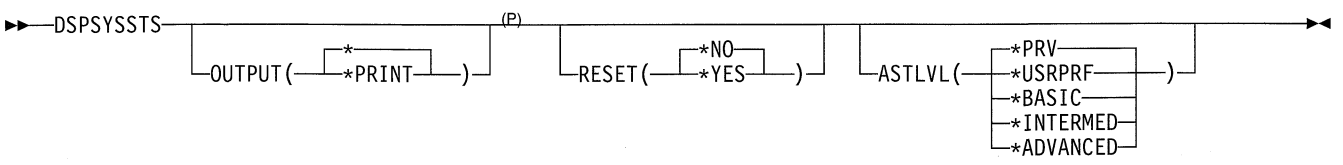
***MSG:** An escape message is issued when duplicate jobs are found.

Example

```
DSPSRVSTS
```

This command shows the service status information for the job from which the command is entered.

DSPSYSSTS (Display System Status) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display System Status (DSPSYSSTS) command allows the user to display or print information about the current status of the system.

Optional Parameters

OUTPUT

Specifies whether the output is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

RESET

Specifies whether system status statistics fields are reset to zero, as if this is the first occurrence of the DSPSYSSTS command in this job. The value specified on this parameter does not affect the information presented for *BASIC assistance level.

***NO:** The system status statistics are not reset.

***YES:** The system status statistics are reset.

ASTLVL

Specifies which user interface to use.

***PRV:** The previously used assistance level is presented.

***USRPRF:** The assistance level defined in the user profile is presented.

***BASIC:** The Operational Assistant user interface is presented.

***INTERMED:** The system user interface is presented.

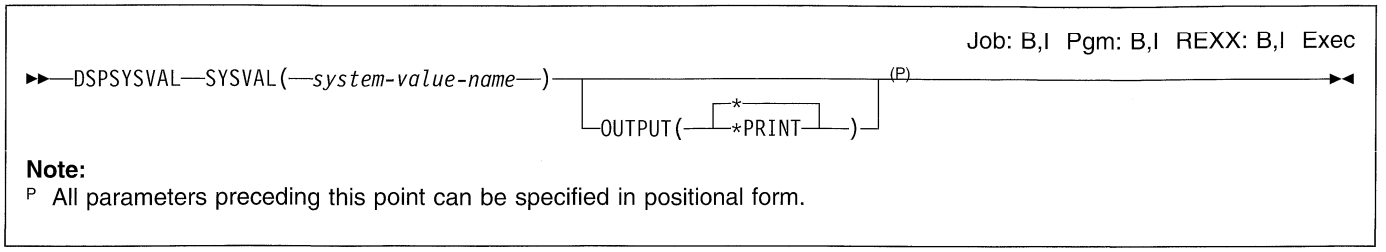
***ADVANCED:** The system user interface expert mode is presented.

Example

```
DSPSYSSTS  OUTPUT(*PRINT)
```

This command prints the current system status information.

DSPSYSVAL (Display System Value) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display System Value (DSPSYSVAL) command displays the name and the value of the specified system value.

Restriction: You must have *ALLOBJ and *SECADM special authorities to change security related system values.

Required Parameter

SYSVAL

Specifies the name of the system value that is displayed. A complete list of system values that are shipped with the system is in the *Work Management Guide* and the *Programming Reference Summary*.

Double-Byte Character Set Considerations:

In addition to the system values specified in the *Work Management Guide*, the user can display the value of QIGC, which indicates whether the double-byte character set (DBCS) version of the operating system is installed on the system. If the system value for QIGC is 1, then the DBCS version of the operating system is installed on the system. If the system value for QIGC is 0, the DBCS version of the operating system is not installed on the system.

Optional Parameter

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

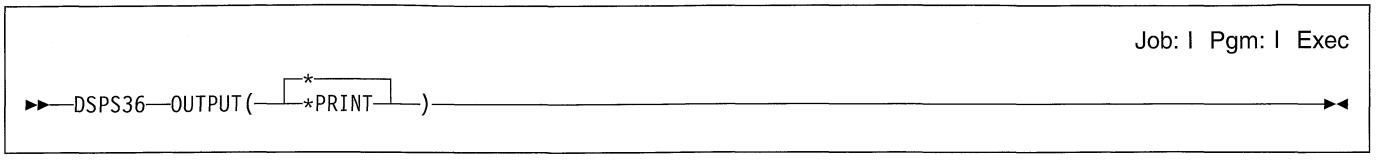
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPSYSVAL   SYSVAL(QHOUR)
```

This command displays the current value of the system value QHOUR.

DSPS36 (Display System/36) Command



Purpose

The Display System/36 (DSPS36) command allows the user to show or print the description of the System/36 environment configuration. The description includes System/36 printers, display stations, general environment values, and (if the user is authorized) MRT security values.

Required Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: The output is displayed at the requesting work station if requested by an interactive job. If this is not an interactive job, the output is printed with the job's spooled output.

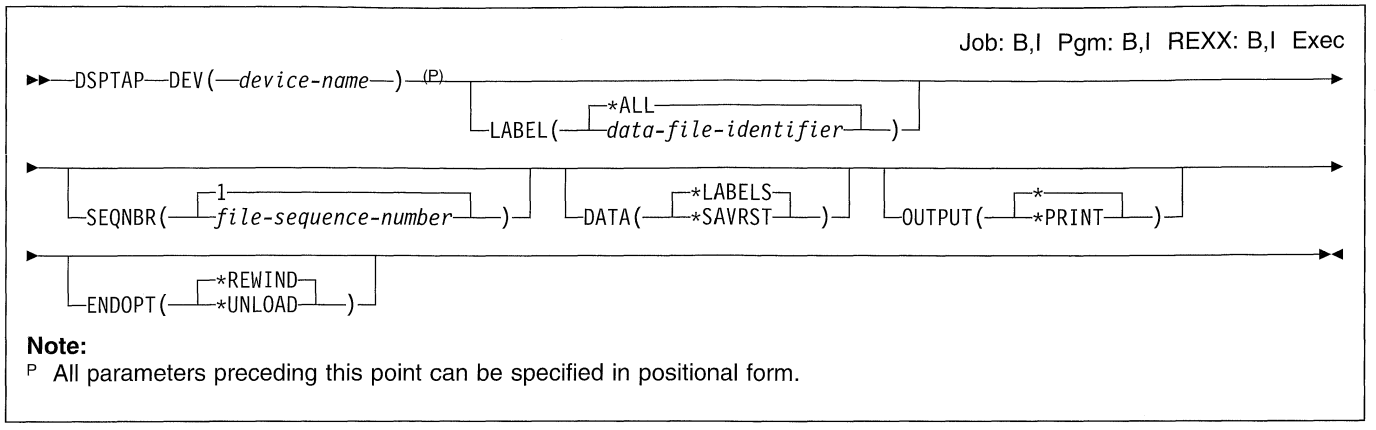
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPS36  OUTPUT(*)
```

This command allows the user in an interactive job to display the System/36 environment description.

DSPTAP (Display Tape) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Tape (DSPTAP) command shows the volume label and data file label information that is contained on a standard labeled magnetic tape or the volume type and density. Information includes an indication of whether the tape is written using improved data recording capability (IDRC) format. If the tape is not labeled, it shows the volume type and density.

If DATA(*SAVRST) is specified, the information includes a description of each object saved to the tape file and summary information about the saved objects.

The information can be printed or shown on a display device. This command allows the user to display a single file label or all file labels on a volume. Additional information can be displayed for save or restore files by specifying DATA(*SAVRST). (The user should know whether the tape being displayed contains data in the basic exchange format or in the save/restore format. But, if not known, the user can specify DATA(*LABELS) and check the data file identifiers listed. If the names of libraries are displayed, the data files are probably in the save/restore format.)

The volume for which information is displayed must be on the specified device. After some information is displayed and DATA(*SAVRST) is specified, the DSPTAP command may read more of the tape before additional information is displayed. If DATA(*SAVRST) is specified, the user may be prompted to load the tape volumes on which the data continues.

Required Parameters

DEV
Specifies the name of the tape device where the volume being displayed is located.

Optional Parameters

LABEL
Specifies the data file identifiers of the data files on the tape whose labels are displayed. The data file identifier is stored in the file label ahead of the data in the file.

***ALL:** All data file identifiers on the tape specified on the DEV parameter are shown.

data-file-identifier: Specify the data file identifier (17 alphanumeric characters maximum) of the data file for which label information is displayed. For save and restore files, the label is usually the library name, but it does not have to be.

SEQNBR
Specifies, for volumes with multiple files, the sequence number of the data file on tape whose label information is displayed. If LABEL(*ALL) is specified or assumed, all files following the file specified by SEQNBR are also displayed after the specified file.

1: The data file being displayed is the first file (or the only file) on the tape. If LABEL(*ALL) is specified, all file labels are displayed, and no verification of label sequence numbers is done.

If a specific LABEL identifier is specified, it is compared with the label identifier of the data file with the specified sequence number (SEQNBR). If the identifiers do not match, an error message is sent. If LABEL(*ALL) is specified, all file labels whose sequence numbers are equal to or greater than the value entered are displayed.

file-sequence-number: Specify the sequence number of the file that is used.

DATA
Specifies the type of information that is displayed.

***LABELS:** The volume label and data file labels are displayed.

***SAVRST:** The tape contains save and restore data. Summary information is shown for the command and each saved object. This option is valid only if the data file was created using one of the CL save commands.

OUTPUT

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

ENDOPT

Specifies the operation that is automatically performed on the tape volume after the operation ends. If more than one volume is included, this parameter applies only to the last tape volume used; all other tape volumes are rewound and unloaded when the end of the tape is reached.

***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

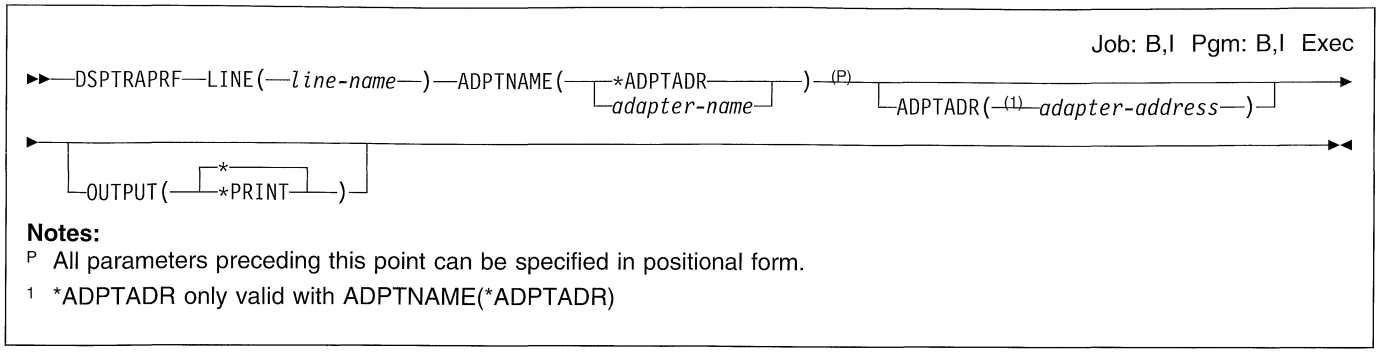
***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

Example

```
DSPTAP    DEV(QTAPE2)   LABEL(*ALL)
```

The volume label and file labels on the tape volume that is on the tape device named QTAPE2 are displayed.

DSPTRAPRF (Display TRLAN Adapter Profile) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 *ADPTADR only valid with ADPTNAME(*ADPTADR)

Purpose

The Display Token-Ring Local Area Network Adapter Profile (DSPTRAPRF) command displays the profile of an active TRLAN (token-ring local area network) adapter.

Required Parameters

LINE

Specifies the name of the line attached to the adapter whose profile is to be displayed.

ADPTNAME

Specifies the name of the adapter whose profile is to be displayed.

***ADPTADR:** The adapter address is used to identify the adapter.

adapter-name: Specify the name of the adapter whose profile is to be displayed.

Optional Parameters

ADPTADR

Specifies the 12-character hexadecimal adapter address.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

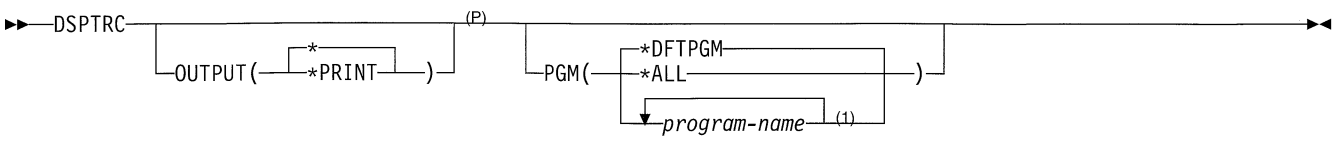
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPTRAPRF  LINE(DETBRANCH)  ADPTNAME(PAYROLL)
```

This command displays the profile of adapter PAYROLL which is attached to line DETBRANCH.

DSPTRC (Display Trace) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 A maximum of 10 repetitions

Purpose

The Display Trace (DSPTRC) command displays all of the data traces that are currently defined in the programs specified in this command. The following data tracing information is displayed:

- The statement ranges or machine instruction ranges in the program
- The name or machine-interface object-definition-table-vector (MI ODV) numbers of all the variables associated with the trace statements
- If the variables are recorded whenever the trace statement is processed or only when their values are changed

Restriction: This command is valid only in the debug mode. To start the debug mode, use the STRDBG (Start Debug) command.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting display station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

PGM

Specifies the programs in the debug mode whose trace data statements and associated program variables are displayed.

***DFTPGM:** The trace data statement of the default program is displayed.

***ALL:** Trace statements of the programs currently in the debug mode are displayed.

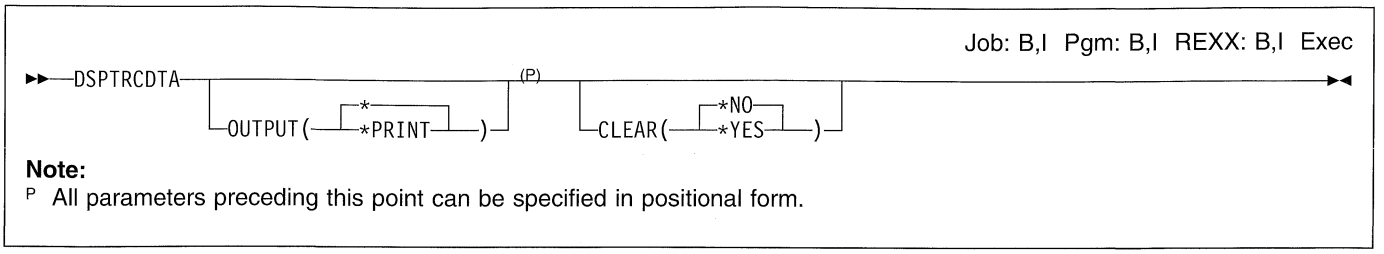
program-name: Specify the names of from 1 through 10 programs already in the debug mode whose trace data statements are displayed.

Example

```
DSPTRC
```

This command shows all of the trace data statement ranges currently specified in the default program of this debugging session. Also displayed are the program variables (but not their values) that are associated with the trace data statements.

DSPTRCDTA (Display Trace Data) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Display Trace Data (DSPTRCDTA) command shows the output of any traces performed since the most recent Clear Trace Data (CLRTRCDTA) command. All of the trace statements and associated program variables within the trace range are displayed. The display shows the sequence in which the traced statements or machine instructions were processed and the name or machine-interface object-definition-table-vector (MI ODV) number and value of any program variables defined for the trace at each point in the sequence. Displaying variable values is controlled by the OUTVAR parameter on the Add Trace (ADDTRC) command that defines which trace is being displayed.

If a job is in debug mode, and that job ends before an ENDDBG is done, this command will be done automatically, printing the output with the job's spooled output.

Restriction: This command is valid only in debug mode. To start debug mode, refer to the STRDBG (Start Debug) command.

Optional Parameters

OUTPUT

Specifies whether the output from the command is displayed at the requesting display station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Note: A program statement appears in the trace data prior to its processing a variable. Therefore, when a program statement (Statement A) changes a variable specified in the trace, the new value appears in the trace data after the statement that follows Statement A.

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

CLEAR

Specifies whether the trace data is cleared after it has been displayed.

***NO:** The trace data is not cleared.

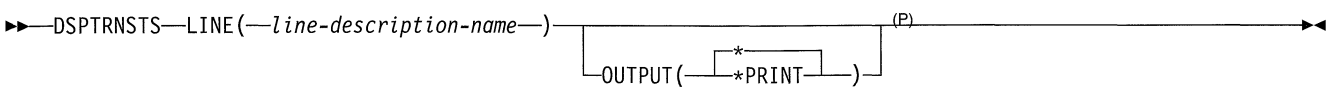
***YES:** The trace data is cleared after it has been displayed.

Example

```
DSPTRCDTA
```

This command shows all of the recorded trace data at the requesting display station. All of the trace statements in the trace range and the values of the associated program variables are displayed. The trace data is not cleared after it has been displayed because CLEAR(*NO) is assumed.

DSPTRNSTS (Display Token-Ring Network Status) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display Token-Ring Network Status (DSPTRNSTS) command displays the status of an active token-ring network line.

Required Parameters

LINE
Specifies the line description name being used by the TRLAN (token-ring local area network) manager.

Optional Parameters

OUTPUT
Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

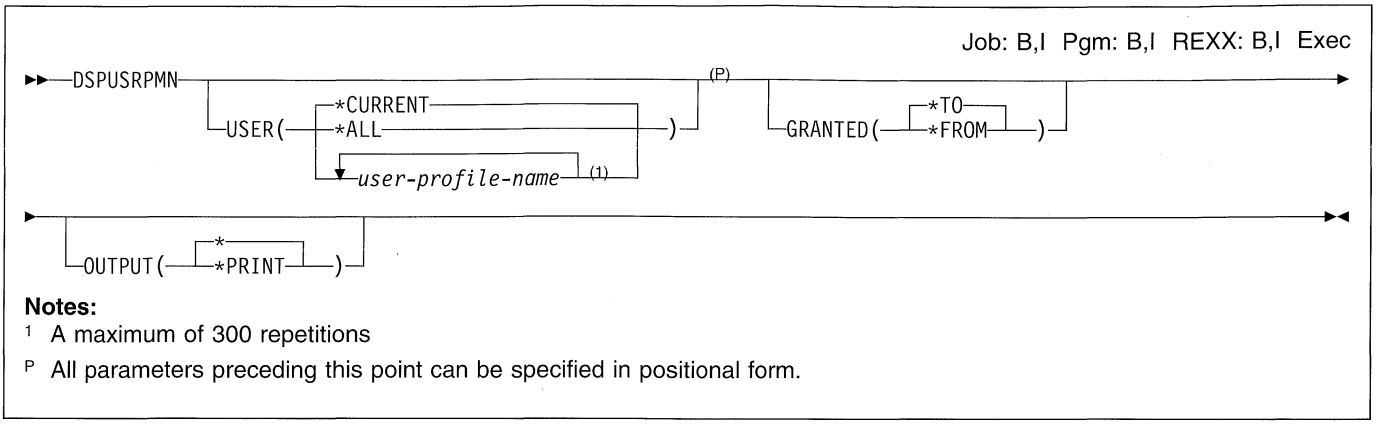
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPTRNSTS   LINE(NYBRANCH)
```

This command shows the status for the line NYBRANCH.

DSPUSRPMN (Display User Permission) Command



Notes:
1 A maximum of 300 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Display User Permission (DSPUSRPMN) command shows the names of the users who are permitted to handle documents or folders and perform tasks on behalf of another user, or shows the names of users who have permitted other users to work on their behalf. More information on user permission is in the Grant User Permission (GRTUSRPMN) command and the *Distribution Services Network Guide*.

Optional Parameters

USER

Specifies the name of the user profile for which the information is displayed. The name in the USER parameter must be enrolled in the system distribution directory before this command is run.

***CURRENT:** The user profile under which the current job is running is used.

***ALL:** Information is displayed for all users in the information directory.

user-profile-name: Specify the name of the user profile for which the information is displayed. Up to 300 user profile names can be specified.

GRANTED

Specifies whether the output produced should display the granted-to relationships or granted-from relationships of a user or users.

***TO:** Users who are permitted to work on behalf of the specified user are displayed.

***FROM:** Users that have permitted the specified user to work on their behalf are displayed.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

Examples

Example 1: Displaying Names of Users

```
DSPUSRPMN  USER(*CURRENT)  GRANTED(*TO)  OUTPUT(*PRINT)
```

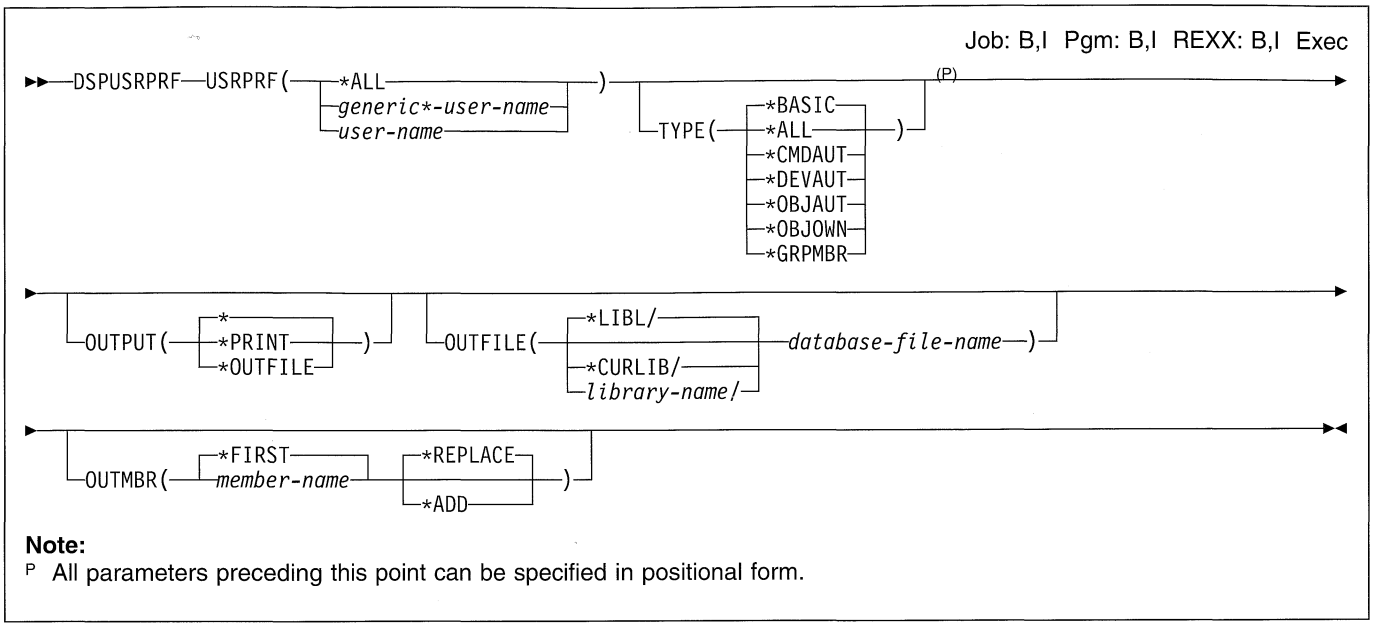
This command prints the names of the users who are permitted to work on behalf of the current user.

Example 2: Printing Names of Users

```
DSPUSRPMN  USER(*ALL)  GRANTED(*TO)  OUTPUT(*PRINT)
```

This command prints the names of all users who have permitted other users to work on their behalf as well as the names of the other users.

DSPUSRPRF (Display User Profile) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Display User Profile (DSPUSRPRF) command displays the contents of a user profile. The user profile contains the user's operational limits for system resources, the names of the objects, commands, and devices that the user has specific authority to use, and the names of the objects that the user owns.

Objects owned by the user profile are not shown on the *CMDAUT, *DEVAUT, or *OBJAUT displays.

This command does not show the password, nor does it show information about objects authorized for public use. The document password is not shown on the *BASIC display or on any CL command output. Any user on the system can be authorized to use the DSPUSRPRF command, but the requesting user must have read authority for the user profile being displayed.

The DSPUSRPRF function may be a long-running function, depending upon the number of objects the user profile owns and is authorized to use.

Restriction: The user name can be specified as USRPRF(*ALL) or USRPRF(generic*-user-name) only when TYPE(*BASIC) and OUTPUT(*OUTFILE) are specified.

Required Parameter

USRPRF

Specifies the name of the user profile.

***ALL:** All user profiles are displayed.

generic-user-name:* Specify the generic name of the user. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be displayed only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

user-name: Specify the name of the user profile to be displayed.

Optional Parameters

TYPE

Specifies the types of user profile information that can be displayed. All, or one, of the following can be displayed:

- The basic portion of the user profile that describes the user
- Commands for which the user profile has specific authority
- Devices for which the user profile has specific authority
- All objects (including commands and devices) for which the user has some specific authority and the authorities assigned with those objects
- Objects that are owned by the user
- Members of the group, if the user profile is a group profile

***BASIC:** The basic portion of the user profile is displayed. This includes:

- The user identifier and text describing the user profile
- The user's special authorities (if any) and the scheduling priority limit
- The user class
- The special environment
- The maximum amount of storage that can be assigned, and the amount currently being used, for the storage of owned objects
- The user's sign-on program
- The user's password expiration interval
- The user's allowed number of device sessions
- The date and time of the user's previous sign-on
- The user's number of sign-on attempts that were not valid
- The user's sign-on menu
- The user's current library
- The level of detailed information in the user options
- The group profile name and, if specified, the owner of the created objects, and the authority that is granted to the group profile when an object is created
- The accounting code assigned to this user
- The job description name assigned to this user
- The message queue assigned to this user
- The printer device for this user
- The output queue assigned to this user
- The ATTN key handling program for this user

***ALL:** All of the information in the user profile is displayed.

***CMDAUT:** The control language (CL) commands, to which the user has specific authority, are displayed.

***DEVAUT:** The system devices to which the user has specific authority are displayed.

***OBJAUT:** The total number and the names of the objects to which the user has specific authority (except those authorized for public use), the user's authority for those objects, and the object types are displayed. Commands and devices are included if *OBJAUT is specified.

***OBJOWN:** For each object owned by the user operating under the user profile, the total number of owned objects and the object names, the types, and the libraries in which the objects reside are displayed.

***GRPMBR:** The members of a group are displayed. This display is available only if the user profile being displayed is a group profile.

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***:** Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

***PRINT:** The output is printed with the job's spooled output.

***OUTFILE:** The output is directed to the database file specified on the OUTFILE parameter.

OUTFILE

Specifies the qualified name of the database file to which the output of the display is directed. If the file does not exist, this command creates a database file in the specified library. If this function creates the file, the text reads 'Outfile for DSPUSRPR', and the public authority is *EXCLUDE.

Note: The outfile format must be the same as QWHDRFFD of system file QADSPFFD. For more information on the outfile format, refer to the *Database Guide*.

The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file.

OUTMBR

Specifies the name of the database file member to which the output is directed.

Element 1: Member to Receive Output

***FIRST:** The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified on the OUTFILE parameter. If the member exists, the user can choose to either add records to the end of the existing member or to clear the existing records in the member and then add the new records.

member-name: Specify the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

Element 2: Operation to Perform on Member

***REPLACE:** The system clears the existing member and adds the new records.

***ADD:** The system adds the new records to the end of the existing records.

Examples

Example 1: Displaying Basic Information

```
DSPUSRPRF  USRPRF(THSMITH)
```

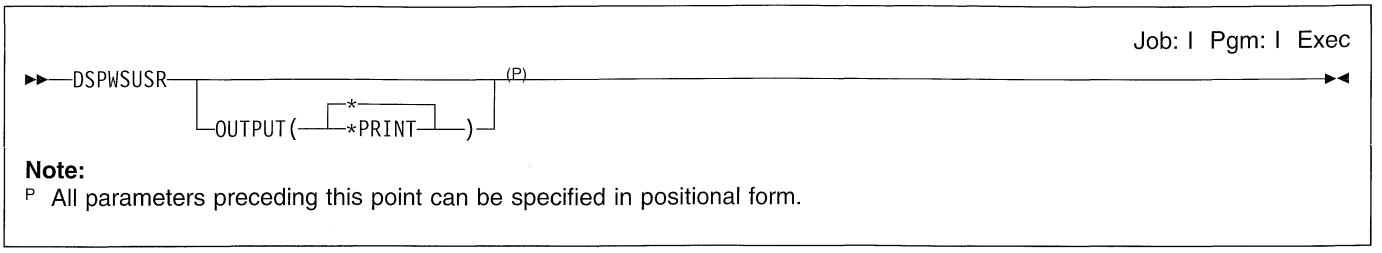
This command shows the basic portion of the user profile named THSMITH because TYPE(*BASIC) is assumed. The commands, devices, and objects that the user is authorized to use are not displayed. Because OUTPUT(*) is also assumed, the operational information is either displayed or printed, depending on where the command is submitted.

Example 2: Printing a List of Objects

```
DSPUSRPRF  USRPRF(RTJOHNSON)  TYPE(*OBJOWN)
  OUTPUT(*PRINT)
```

This command causes the list of objects that are owned by the user named RTJOHNSON to be printed. The list contains the object names, object types, and the names of the libraries where the objects are located.

DSPWSUSR (Display Work Station User) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Display Work Station User (DSPWSUSR) command allows the user to display information describing the current job. Information displayed includes user name or description, system name, work station ID or description, job name, job group, date, and time.

Optional Parameters

OUTPUT

Specifies whether the output from the command is shown at the requesting work station or printed with the job's spooled output. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: Output requested by an interactive job is shown on the display. Output requested by a batch job is printed with the job's spooled output.

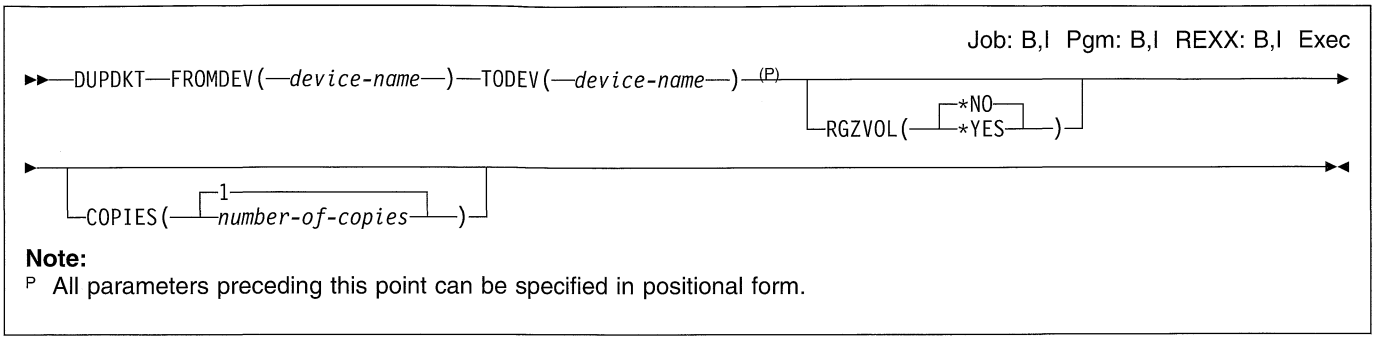
***PRINT:** The output is printed with the job's spooled output.

Example

```
DSPWSUSR   OUTPUT(*)
```

This command displays the information describing the current job.

DUPDKT (Duplicate Diskette) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Duplicate Diskette (DUPDKT) command copies the contents of a single diskette onto one or more diskettes. The volume table of contents (VTOC) and initial program load (IPL) record information and the data records can be copied to a diskette in a specified device. Diskette data in either the basic exchange data format or the save/restore E-format can be copied. The volume identifiers of the diskettes do not have to be unique.

If the diskettes to which data is being copied do not have the same sector size as the diskette from which data is being copied, then a message is sent to the system operator. The copying of diskettes may then be stopped, or the output diskette may be initialized to the same sector size as the input diskette before the copying continues.

Deleted sectors on the input diskette are ignored when copying. They are not copied to the new diskette. The address of the last data record in the file label containing the deleted sector is adjusted on the output diskette, according to the number of sectors found deleted. Therefore, if the input diskette has deleted sectors, the output diskette is not an exact copy.

Diskettes that have an extended label area (up to nine cylinders, in addition to cylinder 0, which are allocated as system area for data set labels), can be copied if the following conditions are followed:

- The RGZVOL option must be *NO.
- No deleted sectors can exist on the input diskette. If deleted sectors are found on the input diskette with an extended label area, a message is sent and the copying function is stopped.

Once a diskette has been copied, the Rename Diskette (RNMDKT) command can be used to rename the copied diskettes so that they have unique volume identifiers.

Restriction: A diskette cannot be copied if it contains control records that indicate records have been relocated, but not in sequence, or that sequential sector addresses are not in consecutive, ascending order. If deleted sectors are found on a diskette with an extended label area, the diskette cannot be copied.

In addition, the following restrictions apply to the type of diskette allowed by the Duplicate Diskette command:

1. A Type 1 diskette may only be copied to another Type 1 diskette.
2. A Type 2 diskette may be copied to a Type 2 diskette or Type 2D diskette. A Type 2D diskette is allowed because it was made for double-density recording. Thus, it is acceptable to use a Type 2D diskette for single-density recording. A message is sent to the system operator if an attempt is made to copy from a Type 2 diskette to a Type 2D diskette. The operator may then use the Initialize Diskette (INZDKT) command to initialize the Type 2D diskette to a Type 2 and continue processing. Note that the Type 2D now logically appears as a Type 2 diskette.
3. A Type 2D diskette may be copied to a Type 2 diskette or a Type 2D diskette. A Type 2 diskette is made for single-density recording and is more prone to media errors if used for double-density recording (Type 2D). If copying from a Type 2D diskette to a Type 2 diskette, a message is sent to the system operator. The operator may then cancel processing or use the Initialize Diskette (INZDKT) command to initialize the Type 2 diskette to a Type 2D diskette and continue processing. Note that the Type 2 diskette now logically appears as a Type 2D diskette and is more prone to media errors since it is meant for single-density recording.

Note: Results when copying to or from a diskette with labels that are not IBM standard labels are unpredictable. The diskette should be initialized by specifying CHECK(*NO) on the Initialize Diskette (INZDKT) command.

Required Parameters

FROMDEV
Specifies the name of the device from which the diskette is being copied.

TODEV
Specifies the name of the device to which the diskette is being copied.

Optional Parameters

RGZVOL

Specifies whether to delete the unused space between files to allow space for additional files to be written on the diskette.

***NO:** The unused space remains as it exists on the input diskette. If no deleted sectors are found, the output diskette is an exact copy of the input diskette.

***YES:** The unused space between files is moved to the end of the last file, making all files on the output diskette connected. If unused space between files currently exists, the files on the output diskette resides at different physical locations.

COPIES

Specifies, for spooled files, the number of copies being printed.

1: One copy of the output is printed.

number-of-copies: Specify the number of diskette copies to make. Valid values range from 2 through 999.

Examples

Example 1: Copying Diskette Contents

```
DUPDKT  FROMDEV(DKT1)  TODEV(DKT2)
```

This command copies the entire contents of the diskette in device DKT1 onto the diskette in device DKT2.

Example 2: Copying Diskette Contents

```
DUPDKT  FROMDEV(DKT2)  TODEV(DKT2)
```

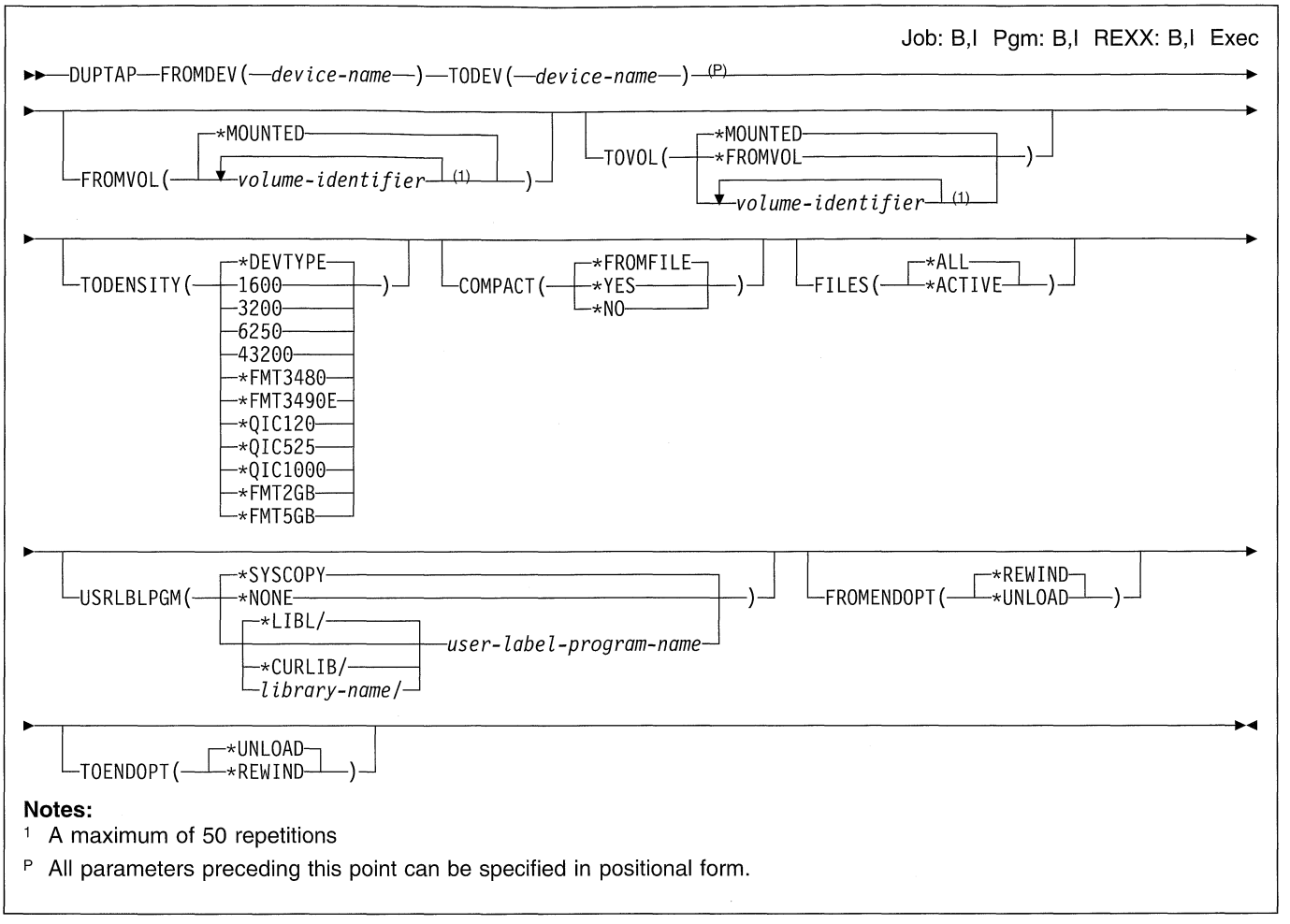
This command copies the entire contents of the diskette in device DKT2, prompts for the next diskette being inserted in device DKT2, and copies the contents on that diskette.

Example 3: Compressing Unused Space

```
DUPDKT  FROMDEV(DKT2)  TODEV(DKT1)  RGZVOL(*YES)
```

This command copies the contents of the diskette in device DKT2 to the diskette in device DKT1. The unused space of the data area is compressed at the end of the diskette. If unused space exists between files on the input diskette, the files on the output diskette reside at different physical locations.

DUPTAP (Duplicate Tape) Command



Notes:
1 A maximum of 50 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Duplicate Tape (DUPTAP) command copies the contents of one tape to another tape.

Notes:

1. The density field in the file header labels is updated to reflect the true density.
2. Byte 80 in the volume label of a tape written on device type 6157 is reset from a 'Q' to a blank.

Restrictions:

- The user must have two tape drives to use this command.
- Non-labeled tapes cannot be duplicated to 1/4 inch or 8mm cartridge devices.

Required Parameters

FROMDEV
Specifies the name of the device from which the tape is copied.

TODEV
Specifies the name of the device to which the tape is copied.

Optional Parameters

FROMVOL
Specifies the volume identifier of the tape being duplicated, or indicates that the tape currently on the magnetic tape device is being duplicated.

***MOUNTED:** The volume currently placed in the device is used.

volume-identifier: Specify the identifier of the labeled volume being duplicated. If the tape placed on the specified device has a different volume identifier than specified, or if it is an unlabeled volume, an error message is sent.

TOVOL
Specifies the volume identifiers of the tapes to be created (destination volumes).

***MOUNTED:** The volume currently placed in the device is used.

***FROMVOL:** The volume label of the tape currently mounted on the source device is used to initialize the tape on the output device.

volume-identifier: Specify the identifier to use on the output volume.

FILES

Specifies which data files are copied.

***ALL:** All data files on the tape volume are copied.

***ACTIVE:** Only data files with an expiration date later than the current system date are copied.

TODENSITY

Specifies (in bits per inch) the density or format in which to write the data to the device specified on the TODEV parameter.

***DEVTYPE:** The data written on the tape volume is based on the device type.

Tape Device	Default Density (bpi)
2440	6250
3422	6250
3430	6250
3480	*FMT3480
3490	*FMT3490E
6341	*QIC120
6342	*QIC525
6343	*QIC1000
6346	*QIC120
6347	*QIC525
6348	*QIC1000
6366	*QIC1000
6368	*QIC1000
7208	43200
7208	*FMT2GB
7208	*FMT5GB
9346	*QIC120
9347	3200
9348	6250

1600: The data density on this tape volume is 1600 bits per inch.

3200: The data density on this tape volume is 3200 bits per inch.

6250: The data density on this tape volume is 6250 bits per inch.

43200: The data density on this tape volume is 43200 bits per inch.

***FMT3480:** The data density on this tape volume is formatted to support a 3480 device.

***FMT3490E:** The data density on this tape volume is formatted to support a 3490 device.

***QIC120:** The format of this tape is *QIC120.

***QIC525:** The format of this tape is *QIC525.

***QIC1000:** The format of this tape is *QIC1000.

***FMT2GB:** The format of this tape is *FMT2GB.

***FMT5GB:** The format of this tape is *FMT5GB.

COMPACT

Specifies whether device data compaction is performed. If the device specified does not support compaction, this parameter is ignored.

***FROMFILE:** Device data compaction is performed only if the file being read from the device specified on the FROMDEV parameter was written using device data compaction.

***YES:** Device data compaction is performed on all files written to the device specified on the TODEV parameter.

***NO:** Device data compaction is not performed.

USRLBLPGM

Specifies whether user tape labels are processed by a user program. The user label program passes user labels that are written to tape to the device specified on the TODEV parameter. The device specified on the FROMDEV parameter passes user labels to the user label program.

***SYSCOPY:** User tape labels are processed to allow proper duplication of System/36 save and restore tapes. If the tape volume specified on the FROMDEV parameter has user header labels, they are copied verbatim to the tape volume specified on the TODEV parameter. The same is done for the user trailer labels at the end of the file or for the trailer labels at the end of the file section.

If an end-of-volume condition occurs on the device specified on the TODEV parameter before the logical end-of-tape is found on the device specified on the FROMDEV parameter, user trailer and header labels are created and written to the current and next tape volumes on the device specified on the TODEV parameter. These labels replicate the data from the user header labels read at the beginning of the file.

***NONE:** No user tape labels are processed. Any user labels read from the tape volume are ignored. No user labels are written to the tape volume.

The name of the user label program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

user-label-program-name: Specify the name of the user program that processes the user tape labels.

FROMENDOPT

Specifies whether the tape volume placed on the device specified on the FROMDEV parameter is rewound or rewound and unloaded after the operation is complete.

***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

TOENDOPT

Specifies whether the tape volume placed in the device specified on the TODEV parameter is rewound, or rewound and unloaded after the operation is complete.

***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

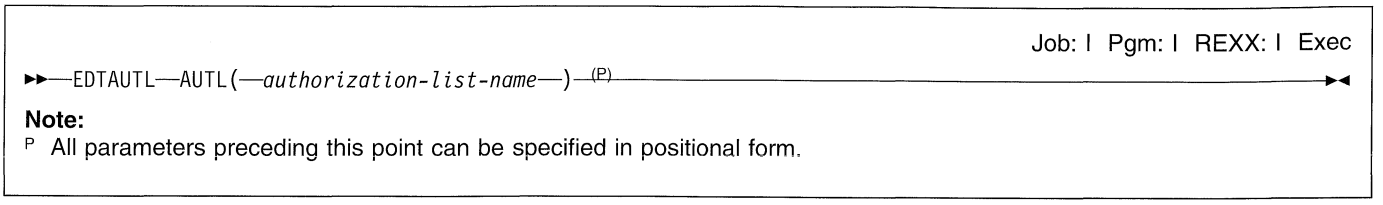
***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

Example

```
DUPTAP    FROMDEV(TAPE01)  TODEV(TAPE02)
```

This command duplicates the tape volume mounted on device TAPE01 onto the tape volume mounted on device TAPE02.

EDTAUTL (Edit Authorization List) Command



Purpose

The Edit Authorization List (EDTAUTL) command shows the list of users and their authorities. From this display, the user can add and remove users and change users' authorities on the authorization list.

Restriction: The user must have authorization list management authority to, or ownership of, the list to use this command.

Required Parameter

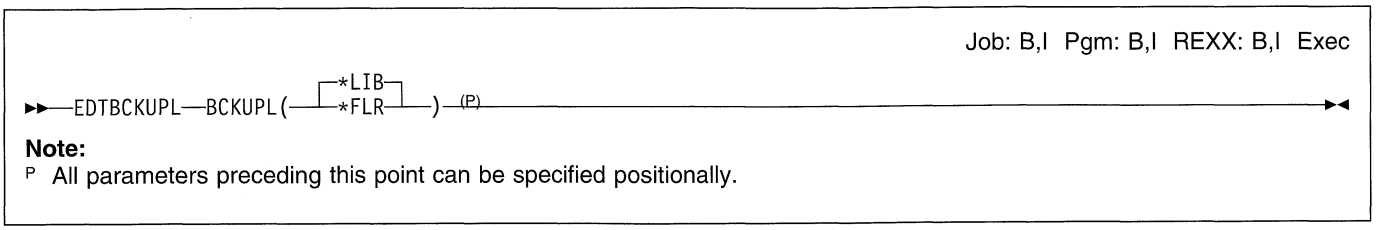
AUTL
Specifies the name of an authorization list with which to work.

Example

```
EDTAUTL   AUTL(MYLIST)
```

This command shows the authorization list MYLIST and allows it to be changed.

EDTBCKUPL (Edit Backup List) Command



Note:
P All parameters preceding this point can be specified positionally.

Purpose

The Edit Backup List (EDTBCKUPL) command allows the user to select libraries and folders for backup. More information on backup is in the *Advanced Backup and Recovery Guide*.

Required Parameters

BCKUPL

Specifies the backup list to be changed.

***LIB:** The library backup list is changed.

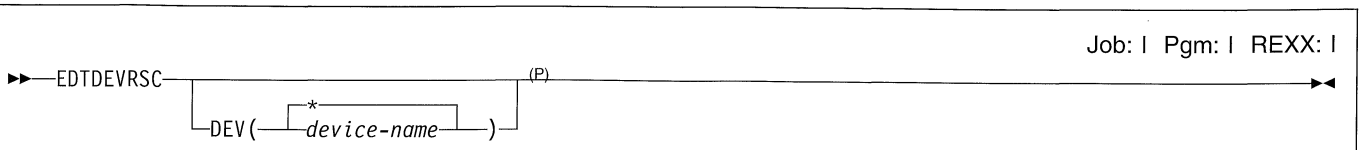
***FLR:** The folder backup list is changed.

Example

```
EDTBCKUPL  BCKUPL(*LIB)
```

This command displays the library backup list stored in user index QUSRSYS/QEZBACKUPL, and allows the user to change it.

EDTDEVRSC (Edit Device Resources) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Edit Device Resources (EDTDEVRSC) command allows the user to display and change device information.

Optional Parameters

DEV

Specifies the name of the device being changed.

*: The current display device is changed.

device-name: Specify the name of the device. The device can be a:

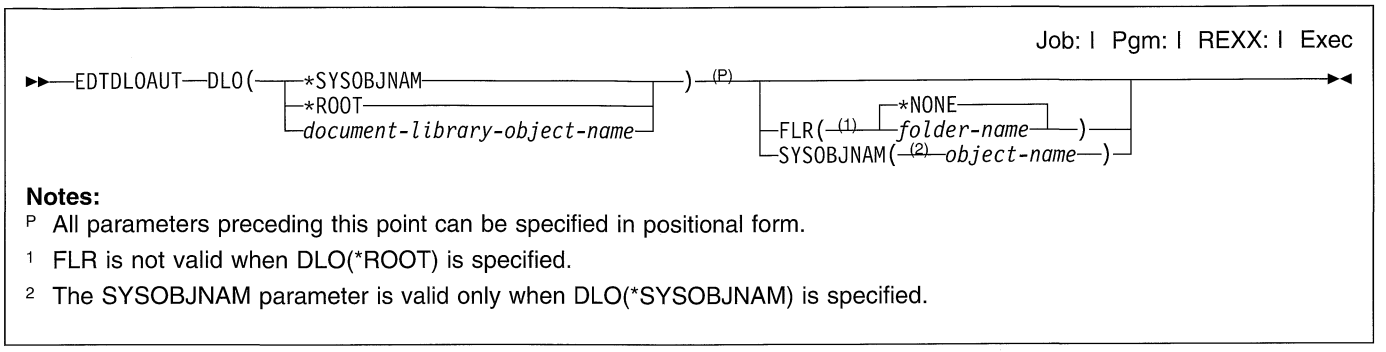
- Work station
  - display
  - printer
- Tape device
- Diskette device

Example

```
EDTDEVRSC   DEV(*)
```

This command shows the Edit Work Station Device Resource display that contains information about the locally attached display device currently being used.

EDTDLOAUT (Edit Document Library Object Authority) Command



Notes:

P All parameters preceding this point can be specified in positional form.

1 FLR is not valid when DLO(*ROOT) is specified.

2 The SYSOBJNAM parameter is valid only when DLO(*SYSOBJNAM) is specified.

Purpose

The Edit Document Library Object Authority (EDTDLOAUT) command is used to change authorization to a document and/or folder object.

The following information is shown for the specified document or folder:

- The name of the document library object
- The name of the authorization list securing the document or folder (if there is one)
- Personal status of the document library object
- Specific user authority for the document or folder
- The authority given to the users with no specific authority for the document or folder
- Access codes can be shown by pressing a function key

Restrictions:

1. A user must have *ALL authority to the document or folder to change the authority or *ALLOBJ special authority.
2. You must have authority to use the ADDDLOAUT, CHGDLOAUT, and the RMVDLOAUT commands to use this command.
3. The user must have *ALLOBJ special authority to change the *ROOT folder public authority.

Required Parameters

DLO

Specifies the name of the document or folder that is changed.

***SYSOBJNAM:** The system object name specified in the SYSOBJNAM parameter is changed.

***ROOT:** The public authority value of the *ROOT folder is changed.

document-library-object-name: Specify the user-assigned name of the document or folder that is changed. Up to 12 characters can be specified.

Optional Parameters

FLR

Specifies the name of the folder that contains the document.

***NONE:** A folder name is not specified. If DLO(name) is specified and the object is located in a folder, FLR(*NONE) cannot be specified.

folder-name: Specify the name of the folder that contains the object. The name can consist of a series of folder names if the folder containing the object is located in another folder. Up to 63 characters can be specified.

SYSOBJNAM

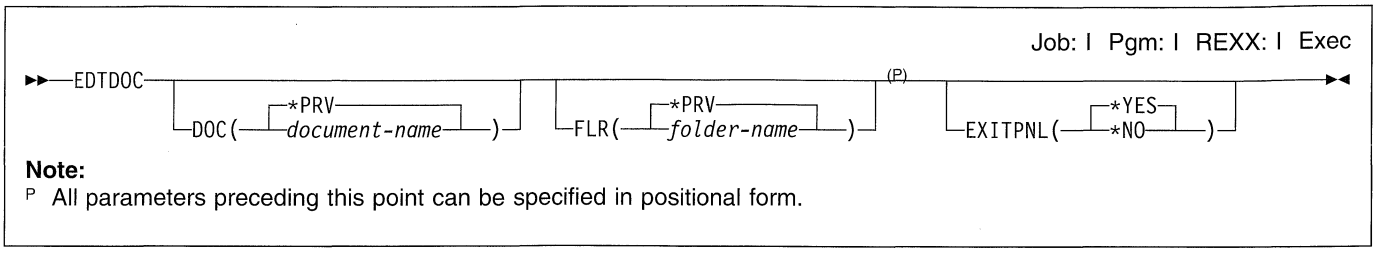
Specifies the system object name of the folder or document. This parameter is valid only when DLO(*SYSOBJNAM) is specified. Ten characters must be specified.

Example

```
EDTDLOAUT  DLO(DOCA)  FLR(MYFLR)
```

This command allows the user of this command to change the list of authorized users and their authorities to the document library object named DOCA in folder MYFLR. The user of this command must have *ALL authority to the object or be the owner of the object.

EDTDOC (Edit Document) Command



Purpose

The Edit Document (EDTDOC) command allows the user to edit a document using the word processing function of OfficeVision/400. More information on editing documents is in the *Using OfficeVision/400* Word Processing*.

Optional Parameters

DOC

Specifies the name of the document to edit.

***PRV:** The name used in the previous session is used.

document-name: Specify the name of the document that is edited.

FLR

Specifies the name of the folder that contains the document.

***PRV:** The name used in the previous session is used.

folder-name: Specify the name of the folder that contains the document to be edited.

EXITPNL

Specifies whether the Exit Document display is shown when F3(Exit) or F12(Cancel) is pressed to end the editing.

***YES:** The Exit Document display is shown when F3(Exit) or F12(Cancel) is pressed to end the editing.

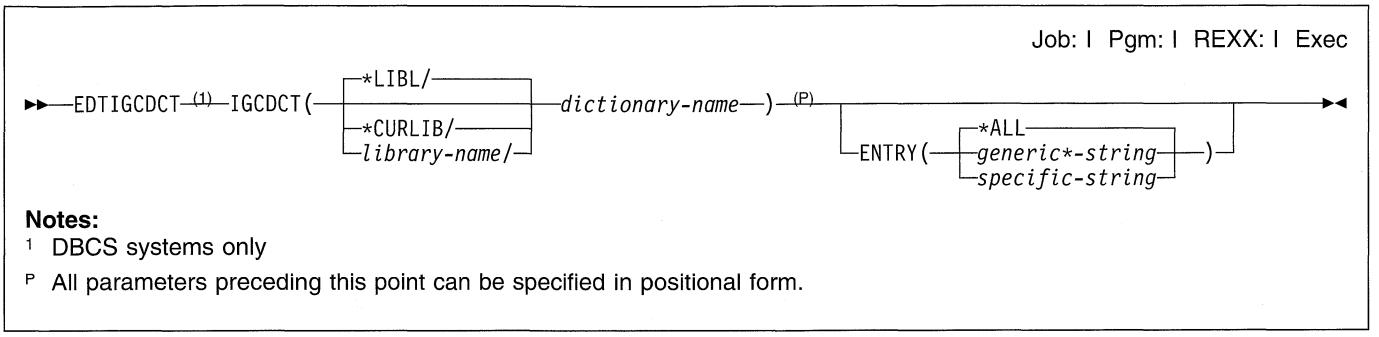
***NO:** The Exit Document display is not shown when F3(Exit) or F12(Cancel) is pressed to end the editing.

Example

```
EDTDOC   DOC(TASK4) FLR(INSTTXT)
```

This command displays the document TASK4 of the folder INSTTXT, and allows the user to edit the document TASK4.

EDTIGCDCT (Edit DBCS Conversion Dictionary) Command



Purpose

The Edit DBCS Conversion Dictionary (EDTIGCDCT) command lets the user add, change, and delete alphameric entries and their related words from the specified double-byte character set (DBCS) conversion dictionary. The system refers to the DBCS conversion dictionary when doing DBCS conversion. The system displays the entries being edited when this command is specified.

Note: Use of the conversion function is not recommended for Chinese and Korean DBCSs.

Required Parameters

IGCDCT

Specifies the qualified name of the DBCS conversion dictionary being edited. If a library name is not specified, the first dictionary found when searching the library list is edited.

The name of the dictionary can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

dictionary-name: Specify the name of the dictionary that will be edited.

Optional Parameters

ENTRY

Specifies the alphanumeric entries being edited with their related words.

***ALL:** Any entry in the dictionary can be edited. The system first shows the Work with DBCS Conversion Dictionary display showing all alphanumeric entries in the dictionary. From this display, specific entries are chosen to be edited.

generic-string:* Specify the generic name of the string to be edited. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be edited only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

specific-string: Specify the entry to be edited. The system displays the Edit Related Words display, showing a single alphanumeric entry and its related words. You can edit the related words on this display. The string cannot be longer than 12 characters.

Examples

Example 1: Showing the Work with DBCS Conversion Dictionary Display

```
EDTIGCDCT   IGCDCT(DBCSLIB/QUSRIGCDCT)  ENTRY(123*)
```

This command shows the Work with DBCS Conversion Dictionary display showing all the alphanumeric entries that start with 123 in the dictionary QUSRIGCDCT, which is stored in the library DBCSLIB.

Example 2: Showing the Edit Related Words Display

```
EDTIGCDCT   IGCDCT(DBCSLIB/QUSRIGCDCT)  ENTRY(WORDS)
```

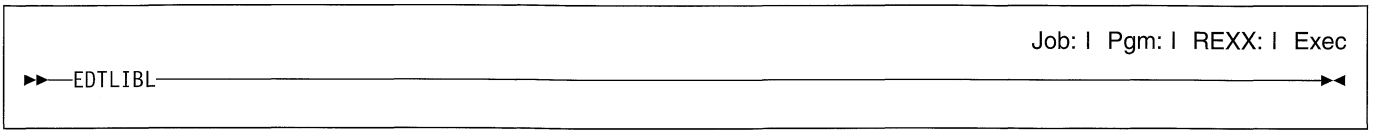
This command shows the Edit Related Words display showing the alphanumeric entry WORDS and its related words from the dictionary QUSRIGCDCT, which is stored in library DBCSLIB.

Additional Considerations

Consider the following when using this command:

- You can only rearrange the words related to alphanumeric entries when editing the IBM-supplied DBCS conversion dictionary (QSYS/QSYSIGCDCT). You cannot add or remove related words or alphanumeric entries.
- Editing can only be done at a display station. The following display stations are needed:
  - If a specific string is specified with the ENTRY parameter or to display DBCS characters, use a DBCS-capable display station.
  - If a specific string is not specified with the ENTRY parameter, or if the user does not want to display DBCS characters, use either a DBCS-capable display station or a 24-row by 80-column alphanumeric display station.
- How to specify specific and generic strings for the ENTRY parameter:
  - You must enclose a string of characters in apostrophes (') if the string to be edited contains Katakana, lowercase alphabetic, or special symbols. The system translates lowercase alphabetic characters to uppercase characters unless they are specified as parameter values in apostrophes. For example, specify the string abc as 'abc'.
  - When specifying a string that contains only numeric or uppercase alphabetic characters, the string does not have to be enclosed in apostrophes ('). That means that ABC is as acceptable to the system as 'ABC'.
  - Do not include embedded blanks ( ) in a string of characters. In other words, the system would not accept the string ABC D.
- Specify the correct library name.
- The first display produced by the EDTIGCDCT command, the Work with Japanese Dictionary display, shows all of the dictionary entries. The second display, the Edit Related Words display, shows individual entries and their related words.

EDTLIBL (Edit Library List) Command



Purpose

The Edit Library List (EDTLIBL) command shows an entry display that allows users to make changes to the library list. This command cannot be used to change the system portion of the library list or the library list of any other job.

Restriction: This command is valid only in an interactive environment.

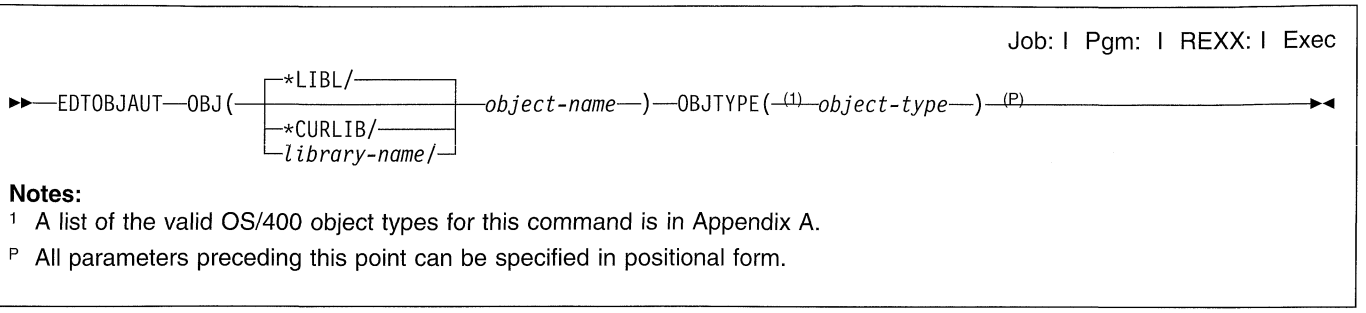
There are no parameters for this command.

Example

```
EDTLIBL
```

This command shows the Edit Library List display from which the user can add libraries, remove libraries, and change the order of the libraries in the library list.

EDTOBJAUT (Edit Object Authority) Command



Notes:

1 A list of the valid OS/400 object types for this command is in Appendix A.

P All parameters preceding this point can be specified in positional form.

Purpose

The Edit Object Authority (EDTOBJAUT) command displays the list of authorized users of an object and their associated authorities. If you own the object or are the security officer, you can add, change, or remove authority for the object. If you have object management authority for the object, you can remove your specific authorities or grant or remove them for other users.

The following are displayed for the specified object:

- the object name
- the name of the library containing the object
- the name of the object's owner, the object's type, and a list of all the users who are authorized to use the object
- the authorities that each user has for the object

If an object does not have an owner name associated with it, no authorities for the object are shown.

Restrictions:

1. The user must have object management authority to the object to use this command.
2. If the object is a file, the user must have object operational and object management authorities.

Required Parameters

OBJ

Specifies the qualified name of the object for which the authorized users and their authorities are to be shown.

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

object-name: Specify the name of the object for which its authorized users and their authorities are to be shown.

OBJTYPE

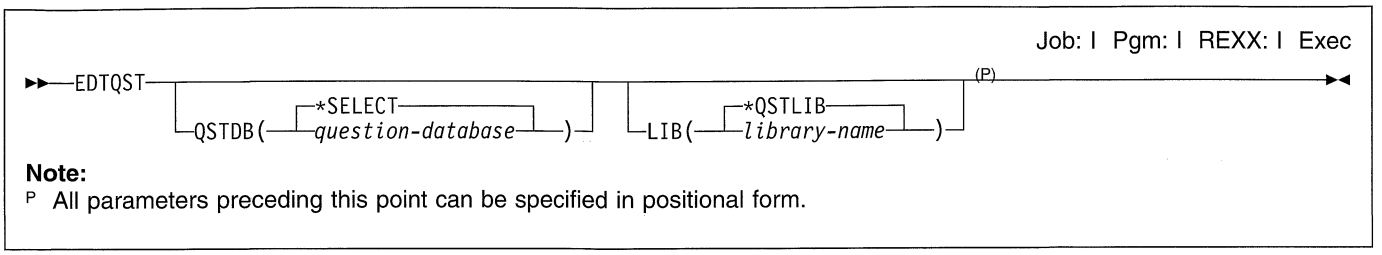
Specifies the object type of the object that is to have its authorized users and their authorities shown. Any one of the operating system object types can be specified. For example, to display a list of users authorized to use a file, specify the value *FILE. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

Example

```
EDTOBJAUT  OBJ(ARLIB/PROG1)  OBJTYPE(*PGM)
```

This command causes the list of authorized users and their authorities for the object named PROG1 to be shown, but only if the user has object management authority for the object. PROG1 is a program (*PGM) located in the library named ARLIB.

EDTQST (Edit Questions and Answers) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Edit Questions and Answers (EDTQST) command allows authorized users to edit questions and answers for publication in a specified database. More information is in the *Q & A Database Coordinator's Guide*.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. A user must have authority to the command and be a Q & A coordinator for any Q & A database referred to by the command.
3. This command is interactive only.

Optional Parameters

QSTDB

Specifies the Questions-and-Answers (Q & A) database in which to edit questions and answers.

***SELECT:** The user is asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database: Specify the name of the Q & A database in which to edit questions and answers.

LIB

Specifies the name of the library that contains the Q & A database.

***QSTLIB:** The library containing the specified Q & A database is searched. If *SELECT is specified on the QSTDB parameter, any Q & A database in any library for which the user is authorized can be selected.

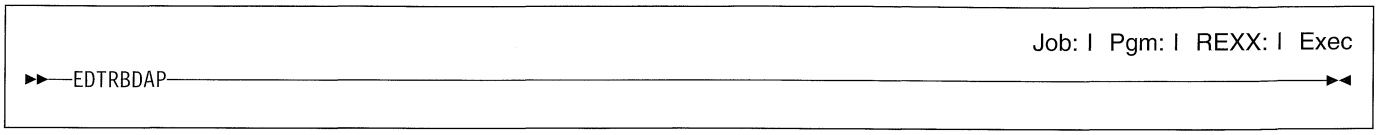
library-name: Specify the name of the library to be searched. If *SELECT is specified on the QSTDB parameter, any database in the library for which the user is authorized can be selected.

Example

```
EDTQST
```

This command shows the Work with Candidate Questions display.

EDTRBDAP (Edit Rebuild of Access Paths) Command



Purpose

The Edit Rebuild of Access Paths (EDTRBDAP) command shows the editing options that are available to the user when rebuilding an access paths. From this display, users can selectively control the rebuilding of access paths.

Restrictions:

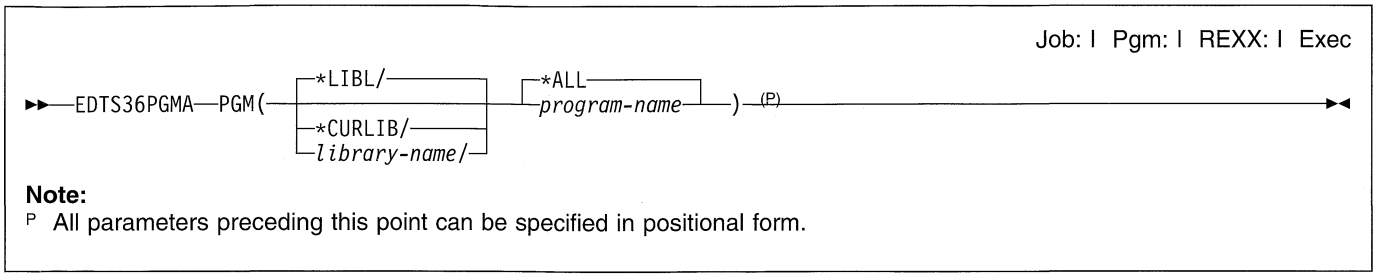
1. This command is shipped with public *EXCLUDE authority and the QSYSOPR user profile has private authority to use the command.
2. If an access path is marked with HELD, the user cannot open the access path or run a query that uses the access path.

Example

```
EDTRBDAP
```

This command shows the controls that are available when editing rebuild access paths.

EDTS36PGMA (Edit System/36 Program Attributes) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Edit System/36 Program Attributes (EDTS36PGMA) command presents the attributes of the specified program on your display to allow you to change them. The attributes of a specified program or of all programs in the specified library can be changed.

Required Parameters

PGM

Specifies the qualified name of the program having its attributes updated.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

***ALL:** The attributes of all programs in the library are shown for update. *ALL is not allowed if the library specified is *LIBL.

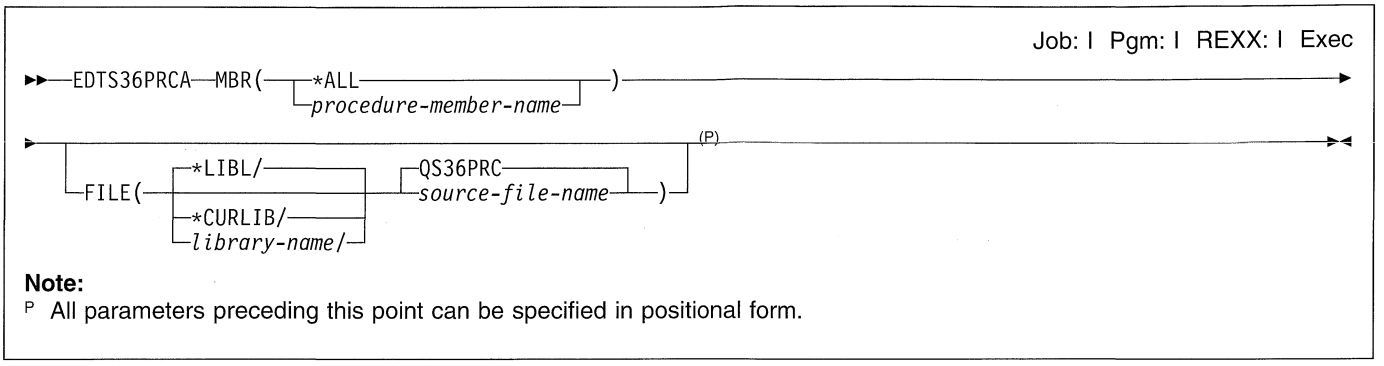
program-name: Specify the name of the program whose attributes you want to update.

Example

```
EDTS36PGMA   PGM(RPGLIB/*ALL)
```

This command shows the program attributes of all the programs in RPGLIB and allows them to be changed.

EDTS36PRCA (Edit System/36 Procedure Attributes) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Edit System/36 Procedure Attributes (EDTS36PRCA) command presents the attributes of the specified procedure on your display for you to change. The attributes of a specified procedure or of all procedures in the specified file can be changed.

Required Parameters

MBR

Specifies the name of the procedure member for which attributes are shown or updated.

***ALL:** The attributes of all procedure members in the file are shown for update.

procedure-member-name: Specify the name of the procedure member whose attributes you want to update.

Optional Parameters

FILE

Specifies the qualified name of the source physical file that contains the procedure members to be updated.

| The name of the file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QS36PRC: The name of the default source physical file that contains the procedure members is used.

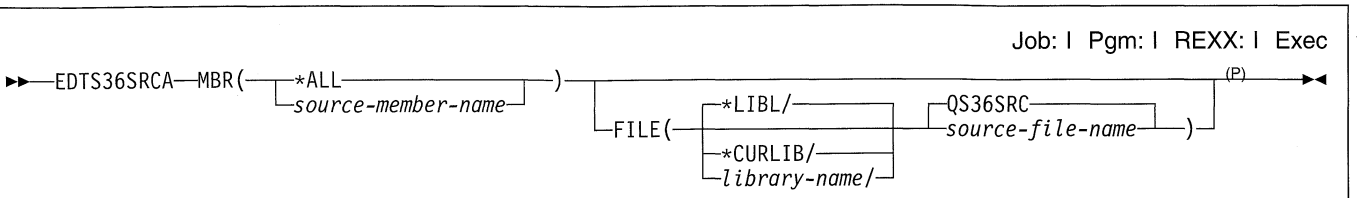
source-file-name: Specify the name of the source physical file that contains the procedure members.

Example

```
EDTS36PRCA   MBR(RPGPROC) FILE(RPGLIB)
```

This command shows the attributes of procedure RPGPROC in file QS36PRC in library RPGLIB and allows them to be changed.

EDTS36SRCA (Edit System/36 Source Attributes) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Edit System/36 Source Attributes (EDTS36SRCA) command presents the attributes of the specified source member on your display for you to change. The attributes of a specified source member or of all source members in the specified file can be changed.

Required Parameters

MBR

Specifies the name of the source member that is having its attributes shown or updated.

***ALL:** The attributes of all source members in the file are shown for update.

source-member-name: Specify the name of the source member whose attributes you want to update.

Optional Parameters

FILE

Specifies the qualified name of the source physical file containing the source members.

| The name of the source physical file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

QS36SRC: The name of the default source physical file containing the source members.

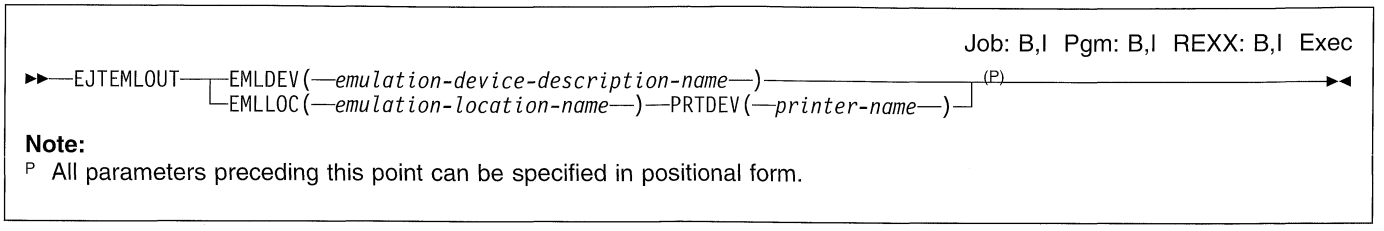
source-file-name: Specify the name of the source physical file containing the source members.

Example

```
EDTS36SRCA   MBR(*ALL) FILE(SDALIB/QS36SRC)
```

This command shows the source attributes of all the source members in file QS36SRC in library SDALIB and allows them to be changed.

EJTEMLOUT (Eject Emulation Output) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Eject Emulation Output (EJTEMLOUT) command finishes the current emulation output, and then ejects to a new page. It forces the last data received from the host system to the spooled file or printer by closing the printer file and then reopening another, so that more data can be spooled or printed. If SCHEDULE(*FILEEND) is specified on the Create Printer File (CPTPRTF) command, printing starts. The request may not take effect immediately, because the printer utility must complete printing a transmission block from the host system before doing this function. See "Additional Considerations" at the end of this command description for more information about running this command.

More information about device emulation is available in the *3270 Device Emulation Guide*.

Required Parameters

EMLDEV
Specifies the name of the printer emulation device requested to receive data from the host system. The printer emulation job using this device is informed of the request and closes the printer file. This forces all of the data received from the host system to the spooled file or printer. The printer file is then reopened and printer emulation continues. To use this function, the user must be authorized to the device.

EMLLOC
Specifies the remote location name associated with this session. This name is defined during configuration and refers to the remote location where communication takes place. This value must be the same as that specified for the Start Printer Emulation (STRPRTEML) command.

PRTDEV
Specifies the name of the printer device that is used to print the spooled output. This value must match the value specified on the Start Printer Emulation (STRPRTEML) command. This parameter must be specified when the EMLLOC parameter is specified.

Example

```
EJTEMLOUT   EMLDEV(HOSTPRT1)
```

This command closes the printer file in the printer emulation job using the emulation device HOSTPRT1, forcing the latest data from the host system out to the spooled file or printer.

Additional Considerations

You must use care when running this command. Before entering the command, you should look at the printed output (if SPOOL(*NO) was specified) or use the Display Spooled File (DSPSPLF) command to look at the spooled file (if SPOOL(*YES) was specified), to determine whether the printer data is at a logical breaking point. If this function is requested when printer emulation is in the middle of a group of print data from the host system, the group is split into separate printer files on the system.

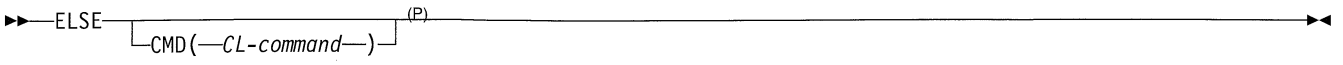
The effect of this command on the printer emulation output varies, depending on the values specified for the SPOOL and SCHEDULE parameters on the printer file.

The possible values and their conditions are:

- **SPOOL(*NO):** All the data received from the host system is printed, and the printer moves to the top of the next page.
- **SPOOL(*YES) and SCHEDULE(*IMMED):** If a writer is active to the output queue and is printing this file, all the data received from the host system is printed, and the printer moves to the top of the next page. If a writer is not active (printing this file), the effect is the same as if SCHEDULE(*FILEEND) was specified. Another printer file is opened on the output queue.
- **SPOOL(*YES) and SCHEDULE(*FILEEND):** The status of the printer file on the output queue changes from *open* to *ready to print*. If a writer is active, the data can be printed. Another printer file is opened on the output queue.
- **SPOOL(*YES) and SCHEDULE(*JOBEND):** The status of the printer file on the output queue changes from *open* to *closed*. The file is not ready to print until the end of the job is reached. Another printer file is opened on the output queue.

ELSE (Else) Command

Pgm: B,I



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Else (ELSE) command is used with an IF command to specify another command that is to be conditionally processed. The ELSE command is processed only if the result of evaluating the logical expression on the preceding IF command is false, the ELSE command can specify a CL command to be processed for the false condition. If the DO command is specified on the ELSE command, a Do group can be processed. If the result of processing the IF command is true, the ELSE command and commands associated with it are not processed.

An ELSE command does not have to follow each IF command, but each ELSE command that is coded must have an associated IF command preceding it. If nested levels of IF commands are used, a given ELSE is always matched with the innermost IF command that has not already been matched with another ELSE command. Although the ELSE command is optional, coding all of the matching ELSE commands makes it easier to see where all of the nesting levels start and end.

Restriction: The ELSE command is valid only in a CL program. It must have an associated IF command preceding it.

Optional Parameters

CMD

Specifies the command or commands (in a Do group) to be processed if the result of evaluating the expression on the corresponding IF command is false. If the command specified in this parameter is a DO command, all of the commands specified within the Do group are considered to be part of the command specified by the parameter. If no command is specified, no action is taken for a false condition.

If the command specified by the CMD keyword is not coded on the same line as the keyword, the left parenthesis following CMD must be coded on the same line, followed by a + or - to show continuation. The command and the right parenthesis can then be coded on the next line. For example:

```
ELSE  CMD(  +
    GOTO C)
```

If any part of the command continues on the next line, a continuation character (+ or -) must be specified.

If a DO command is specified, only the DO command (not the commands specified as part of the Do group) is placed in parentheses. For example:

```
ELSE  CMD(DO)
    CMD1
    CMD2
      •
      •
      •
    ENDDO
```

The following commands, although valid in CL programs, cannot be specified on the ELSE command:

ENDDO (End Do)
MONMSG (Monitor Message)
PGM (Program)
ENDPGM (End Program)
DCL (Declare CL Variable)
DCLF (Declare File)
another ELSE command

In addition, the MONMSG command cannot be specified as the next command after the ELSE command.

Examples

Example 1: Using Else and If Commands

```
IF   (&A *GT &B)  THEN(CHGVAR  VAR(&A)  VALUE(&B))
  ELSE   (CHGVAR &B &A)
```

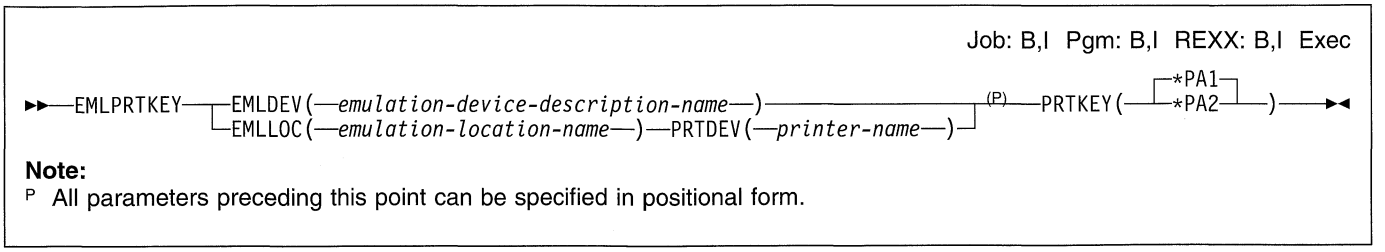
If the value of &A is greater than the value of &B, &A is set equal to &B. If &A is less than or equal to &B, the test result is false. The CHGVAR command on the ELSE command is processed, and the value of &B is set to the same value as &A. (Refer to the CHGVAR (Change Variable) command for the description of the command and its parameters.)

Example 2: Nested Levels of Commands

```
IF   COND(&A *EQ &B)
  THEN(IF   (&C *EQ &D)
    THEN(IF   (&C *EQ &F)   THEN(DO)))
                          CMD1
                          CMD2
                           •
                           •
                           •
                          ENDDO
          ELSE   CMDX
     ELSE   CMDY
ELSE   DO
```

This example shows the use of nested levels of IF commands where an ELSE command is associated with each IF. The use of the ELSE commands makes the nested levels of IF commands easier to identify.

EMLPRTKEY (Emulate Printer Key) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The Emulate Printer Key (EMLPRTKEY) command causes the printer emulation job or session that is using the specified printer emulation device to send either a PA1 or PA2 key signal to the host system. PA keys are program attention keys that are used to signal the host system. The command may not take effect immediately because the printer emulation routine will complete printing the last data received from the host system before doing this function. In addition, the PA key signal, although sent to the host system, may not immediately be received.

Required Parameters

EMLDEV

Specifies the name of a printer emulation device that receives data from the host system. This device must be a 3287 Printer (EMLDEV(3287)) or a 3289 Printer (EMLDEV(3289)), and must currently be operating as an LU1 unit. The printer emulation job or session that is using this device will be informed of the request. If the LU1 session is between brackets, printer emulation starts a bracket and sends the PA key signal to the host system with Change Direction (CD) request. If the LU session is in receive condition, a signal (request for CD) is sent to the host system, and printer emulation waits for the CD. When the CD is received, the PA key signal is sent to the host system with the CD. If the LU session is in send condition, the PA key signal is sent to the host system with the CD.

EMLLOC

Specifies the remote location name associated with this session. This name is defined during configuration and refers to the remote location where communication takes place. This value is the same as the value specified for the EMLLOC parameter for the Start Printer Emulation (STRPRTEML) command.

PRTDEV

Specifies the name of the printer device that is used to print the spooled output. This value must match the value specified on the Start Printer Emulation (STRPRTEML) command. This parameter must be specified when the EMLLOC parameter is specified.

PRTKEY

Specifies the PA key signal that is sent to the host system. The host system program determines how the PA keys work.

***PA1:** The PA1 key signal is sent to the host system.

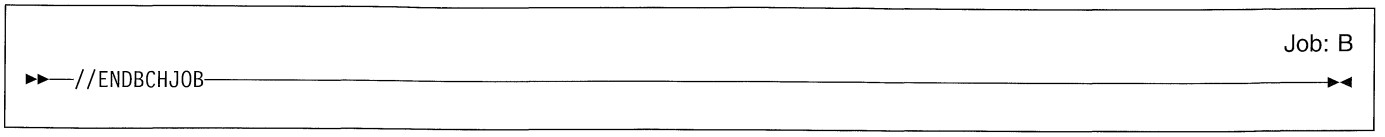
***PA2:** The PA2 key signal is sent to the host system.

Example

```
EMLPRTKEY  EMLDEV(HOSTPRT2)  PRTKEY(*PA2)
```

This command causes the printer emulation session using emulation device HOSTPRT2 to send the PA2 key signal to the host system.

ENDBCHJOB (End Batch Job) Command



Purpose

The End Batch Job (//ENDBCHJOB) command is a delimiter that indicates the end of a job in a batch job stream. The //ENDBCHJOB command also can indicate the end of an inline data file if the command is detected while the inline file is being processed. Other conditions that can indicate the end of a job are:

- Batch Job (BCHJOB) command
- An end-of-file condition while an input stream is being processed

Restriction: The ENDBCHJOB command cannot be used from a work station. Two slashes must precede this command name when entering it in the data record, for example, `//ENDBCHJOB`. The user can separate the slashes from this command name with blank spaces, for example, `//  ENDBCHJOB`.

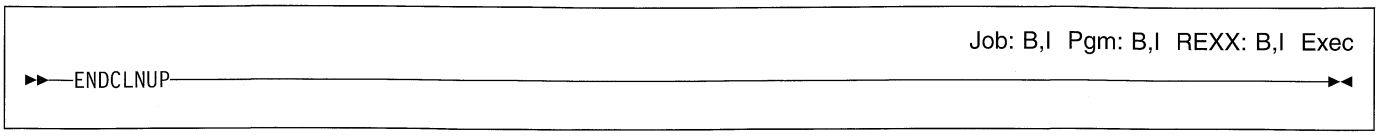
There are no parameters for this command.

Example

```
//ENDBCHJOB
```

This command indicates the end of a job that began with the BCHJOB command.

ENDCLNUP (End Cleanup) Command



Purpose

The End Cleanup (ENDCLNUP) command allows the user to end the cleanup operation. Any active batch cleanup jobs, either processing or on the job queue, are ended immediately.

This command does not alter the parameters specified on the Change Cleanup (CHGCLNUP) command. The cleanup operation can be restarted by specifying the Start Cleanup (STRCLNUP) command.

More information on cleanup commands is in the *Operator's Guide*.

Restriction: The user must have *JOBCTL authority to use this command.

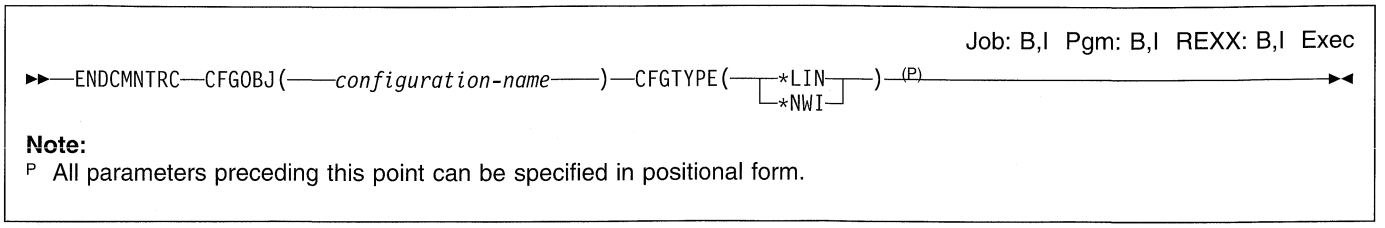
There are no parameters for this command.

Example

```
ENDCLNUP
```

This command ends the cleanup operation.

ENDCMNTRC (End Communications Trace) Command



Note:

(P) All parameters preceding this point can be specified in positional form.

Purpose

The End Communications Trace (ENDCMNTRC) command ends the trace running on the specified line or network interface description.

Restrictions:

1. The user must have *USE authority for the line or network interface to be traced.
2. *SERVICE authority is required to use this command.
3. This command is shipped with public *EXCLUDE authority.
4. The following user profiles have authority to this command:
   - QSECOFR
   - QSRV

Required Parameters

CFGOBJ

Specifies the name of the configuration object being traced. The object is either a line description or a network interface description.

CFGTYPE

Specifies the type of configuration description being traced.

***LIN:** The type of configuration object is a line description.

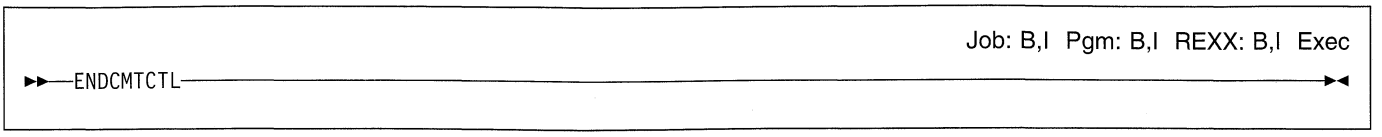
***NWI:** The type of configuration object is a network interface description.

Example

```
ENDCMNTRC  CFGOBJ(*QESLINE)  CFGTYPE(*LIN)
```

This command ends the communications trace of line description QESLINE.

ENDCMTCTL (End Commitment Control) Command



Purpose

The End Commitment Control (ENDCMTCTL) command notifies the system that the commitment definition associated with the activation group for the program that issued the command is to be ended and changes to commitment resources associated with the commitment definition are no longer to be made.

This command either ends the activation group level or the job level commitment definition associated with the activation group for the program that issued the command. A commitment definition is first established by the Start Commitment Control (STRCMTCTL) command.

If there are uncommitted changes for an interactive job, a message is sent asking the user whether the changes should be committed or rolled back before a commitment definition is ended. For a batch job, the changes are rolled back.

More information on the use of commitment control is in the *Advanced Backup and Recovery Guide*.

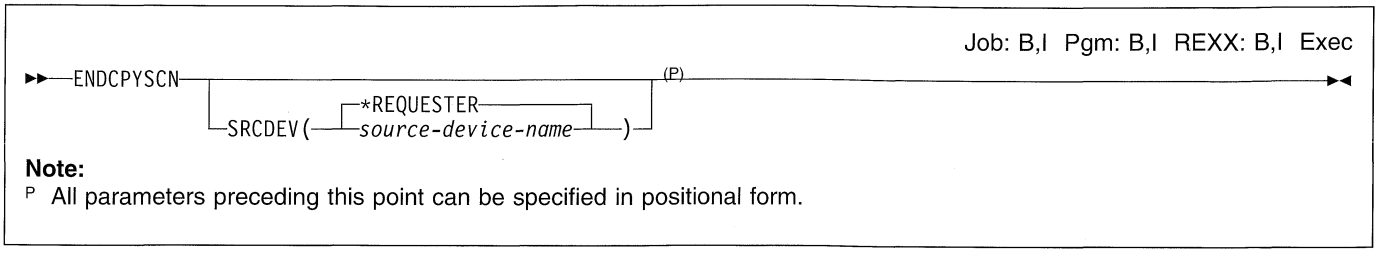
There are no parameters for this command.

Example

```
ENDCMTCTL
```

This command specifies that the commitment definition established with the STRCMTCTL command is to end. The system determines if any changes have been made to the commitment resources after the last commitment boundary (at the last completed Commit (COMMIT) command or Rollback (ROLLBACK) command). If changes have been made for an interactive job, a message is sent asking the user whether the changes should be made permanent (committed) or removed (rolled back). For batch jobs, any changes are rolled back.

ENDCPYSCN (End Copy Screen) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Copy Screen (ENDCPYSCN) command ends the copy screen image operation. This command can be entered from any display station capable of command entry.

Note: The target display station also stops the copy screen image operation when the user presses the System Request key and types ENDCPYSCN (End Copy Screen command) on the command line. No parameters can be specified.

Optional Parameters

SRCDEV

Specifies the name of the display station that is currently having its screen images copied. If no display station name is specified, processing for the requesting display station is canceled.

***REQUESTER:** The source display station that requested the command is used.

source-device-name: Specify the display station name whose screen images are being copied.

Example

```
ENDCPYSCN  SRCDEV(CHARLIE)
```

The command sends a message to 'CHARLIE' (the source display station). The message indicates the copy screen image operation is about to end. The target work station display is restored to the same display image that was shown before the operation started. The sign-on display is normally shown.

ENDCTLRCY (End Controller Recovery) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The End Controller Recovery (ENDCTLRCY) command ends error recovery procedures for a specific controller. If any type of failure occurs after this command is run, an inquiry message is sent.

Use the Resume Controller Recovery (RSMCTLRCY) command to reestablish error recovery procedures for the controller.

Required Parameters

CTL

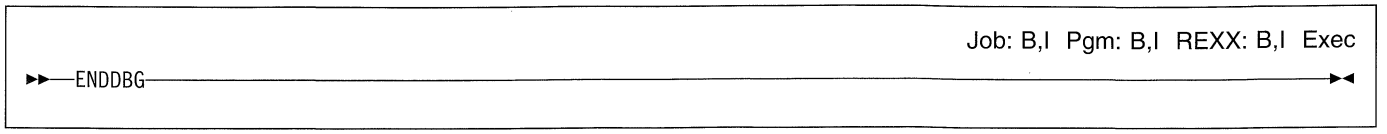
Specifies the controller whose recovery is ended. Specify the name of the controller in the controller description.

Example

```
ENDCTLRCY  CTL(TROLL3)
```

This command ends error recovery procedures for the controller TROLL3.

ENDDBG (End Debug) Command



Purpose

The End Debug (ENDDBG) command ends debug mode for a job, removes all breakpoints and traces, clears any trace data, and removes all programs from debug mode. This command cannot be entered when one or more of the programs in the call stack are stopped at a breakpoint. All breakpoints must be canceled by Resume Breakpoint (RSMBKP) or End Request (ENDRQS) commands. After this command has been entered, all database files in production libraries can be updated normally.

If ENDDBG is not done before the job has ended, all trace data is printed.

Restriction: This command is valid only in debug mode. To start debug mode, refer to the STRDBG (Start Debug) command.

If the user is servicing another job while in debug mode, then this command must be specified before the ENDSRVJOB (End Service Job) command is allowed.

There are no parameters for this command.

Example

```
ENDDBG
```

Assuming that this command is entered interactively and no program in the call stack is stopped at a breakpoint, debug mode for the job is ended.

ENDDEVRCY (End Device Recovery) Command

Job: B,I Pgm: B,I REXX: B,I Exec

Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The End Device Recovery (ENDDEVRCY) command ends error recovery procedures for a specific device. If any type of failure occurs after this command is run, an inquiry message is sent. The user must have object operational authority for the device.

Use the Resume Device Recovery (RSMDEVRCY) command to reestablish error recovery procedures for the device.

Required Parameters

DEV

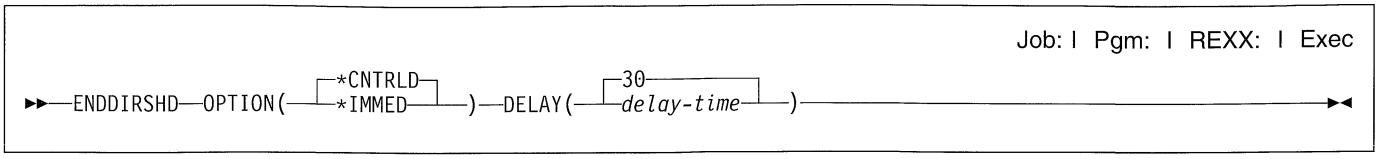
Specifies the device whose recovery is ended. Specify the name specified for the device in the device description.

Example

```
ENDDEVRCY  DEV(WSPR03)
```

This command ends error recovery procedures for the device WSPR03.

ENDDIRSHD (End Directory Shadowing) Command



Purpose

The End Directory Shadowing (ENDDIRSHD) command ends the directory shadow controlling job in the system work subsystem (QSYSWRK).

Any active collector jobs running are allowed to complete. No new collector jobs are started. Supplier jobs are prevented from starting if a collector system requests data through directory shadowing. The Start Directory Shadowing (STRDIRSHD) command can be used to re-start directory shadowing.

Restriction: You must have job control (*JOBCTL) authority to use this command.

Required Parameters

OPTION

Specifies whether the directory shadow controlling job is ended in a controlled manner or immediately.

***CNTRLD:** The directory shadow controlling job is ended in a controlled manner. This allows the directory shadow controlling job to perform cleanup (end-of-job processing).

***IMMED:** The directory shadow controlling job is ended immediately. The directory shadow controlling job is not allowed to perform any cleanup.

Note: Using the *IMMED option can cause unexpected results if data has been only partially updated.

DELAY

Specifies the amount of time (in seconds) allowed for the directory shadow controlling job to complete its cleanup processing during a controlled end. This parameter is not valid if OPTION(*IMMED) is specified. If the cleanup is not complete before the end of the delay time, the directory shadow controlling job is immediately ended.

30: A maximum delay time of 30 seconds is allowed for cleanup before the directory shadow controlling job is ended.

delay-time: Specify the maximum amount of delay time in seconds before the controlling job is ended. Valid values range from 1 through 999999.

Examples

Example 1: Ending Directory Shadowing in a Controlled Manner

```
ENDDIRSHD  OPTION(*CNTRLD)  DELAY(60)
```

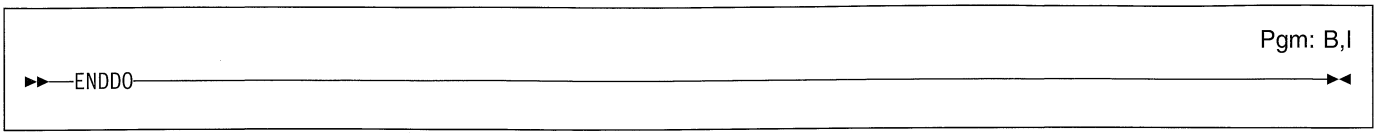
The directory shadow controlling job is ended in the system work subsystem in a controlled manner and will have 60 seconds to complete its end-of-job processing.

Example 2: Ending Directory Shadowing Immediately

```
ENDDIRSHD  OPTION(*IMMED)
```

The directory shadow controlling job is ended in the system work subsystem immediately. The directory shadow controlling job does not perform end-of-job processing.

ENDDO (End Do) Command



Purpose

The End Do (ENDDO) command is used with the DO command to identify a group of commands that are run together as a group. The ENDDO command specifies the end of the Do group that is started with an associated DO command. The ENDDO command must be specified after the last command in the Do group.

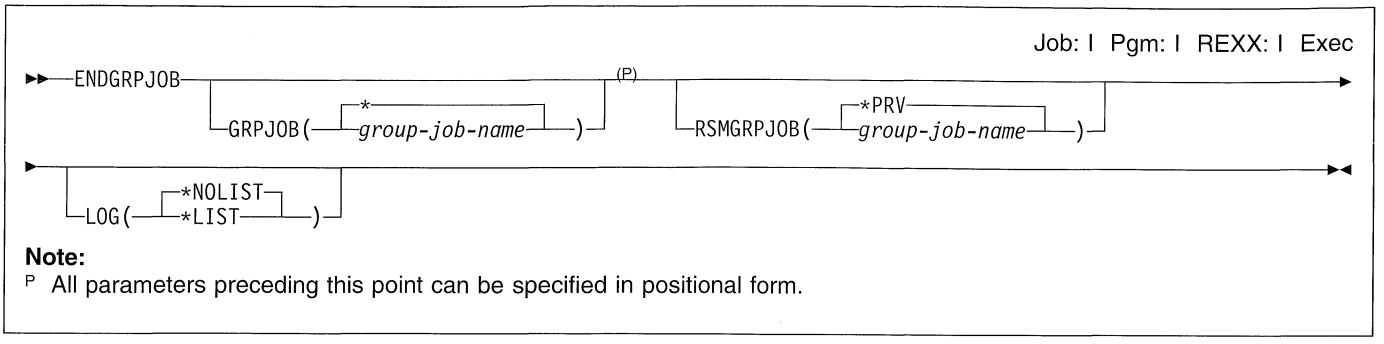
When Do groups are nested, each group must have its own ENDDO command at its end. Every ENDDO command must be associated with a DO command; if too many ENDDO commands occur in the CL program source, a message is issued and the program is not created.

Restriction: This command is valid only within a CL program.

There are no parameters for this command.

See the examples for the DO and IF commands.

ENDGRPJOB (End Group Job) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The End Group Job (ENDGRPJOB) command ends a single job in a group and resumes another job in the group. (Issue the Sign Off (SIGNOFF) command to end all jobs in a group.) The user can specify the following:

- Which job in the group is ended
- Which job in the group gains control (this is valid only when a job is ending itself)
- Whether the job ended has a job log spooled to an output queue

Note: The job that issues the ENDGRPJOB command can only end itself or some other job in the group to which this job belongs. To end a job outside the group, the END Job (ENDJOB) command must be used.

Optional Parameters

GRPJOB

Specifies the group job name of the job that is ended.

***:** This special value ends the group job that issued the ENDGRPJOB command.

group-job-name: Specify the group job name of the job that is ended.

RSMGRPJOB

Specifies the group job name of the job that is to be resumed after the active job in the group has ended. This parameter is valid only when the job that issues this command is ending itself.

***PRV:** The group job most recently active is resumed.

group-job-name: Specify the group job name of the job that gains control after the active job in the group ends.

LOG

Specifies whether the job log of the ending group is spooled to an output queue.

When the only job in a group is ending, one of two situations can occur:

- The ending group job is part of a secondary job pair. In this case, the other job in the secondary job pair is resumed, and control is passed back to that job.
- The ending group job is not part of a secondary job pair. In this case, the sign-on display is shown at the work station (if the work station entry for the work station device in the subsystem specifies *SIGNON).

***NOLIST:** An intermediate representation of the program listing is not created.

***LIST:** The information in the job log is spooled to an output queue.

Examples

Example 1: Ending Group Job that Issued Command

```
ENDGRPJOB  GRPJOB(*)  LOG(*PRINT)  RSMGRPJOB(GROUPJOB1)
```

This command ends the job that is currently running. Its job log is spooled to an output file for printing. When the job completes running, group job GROUPJOB1 becomes the active job in the group.

Example 2: Printing Output of Ended Job

```
ENDGRPJOB  GRPJOB(GROUPJOB2)  LOG(*PRINT)
```

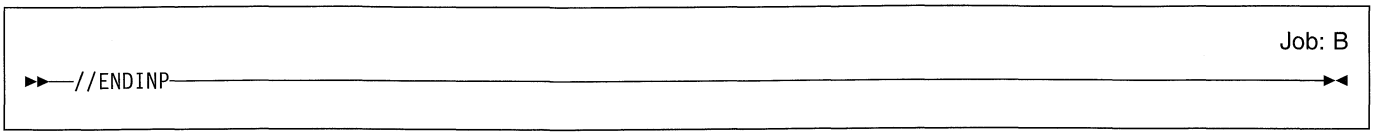
Assume that the job issuing the ENDGRPJOB command is group job GROUPJOB1, which wants to end GROUPJOB2. Group job GROUPJOB2 ends. Its job log is spooled to an output file for printing.

Example 3: Ending a Job That's Part of a Secondary Job Pair

```
ENDGRPJOB  GRPJOB(*)  LOG(*NOLIST)
```

Assume that the job issuing the ENDGRPJOB command is the only job in the group and is part of a secondary job pair. The job issuing the command ends. The job's job log is not spooled to an output file. When the job ends, the other job in the secondary job pair is resumed.

ENDINP (End Input) Command



Purpose

The End Input (//ENDINP) command is a delimiter that indicates the end of the input in a batch input stream. The //ENDINP command also can indicate the end of an inline data file if the command is detected while the inline file is being processed. If the inline file is using ending characters which are not defaults, the //ENDINP command is embedded without being recognized.

Restrictions: The ENDINP command cannot be used from a work station. Two slashes must precede this command name when entering it in the data record, that is, //ENDINP. The user can separate the slashes from this command name with blank spaces, for example, // ENDINP.

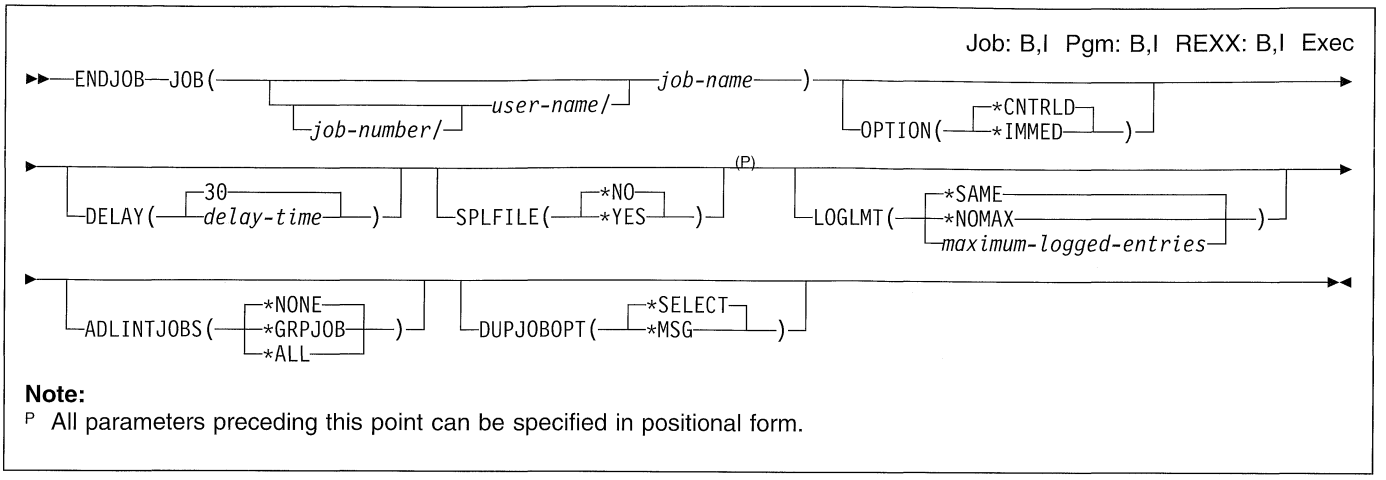
There are no parameters for this command.

Example

```
//BCHJOB
•
•
•
//DATA
•
•
•
//ENDINP
```

The ENDINP command indicates the end of a input stream that began with the Batch Job (BCHJOB) command.

ENDJOB (End Job) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Job (ENDJOB) command ends the specified job and any associated inline data files. The job may be on a job queue, it may be active within a system, or it may have already completed running. All spooled files associated with the job being ended can also be deleted or allowed to remain on the output queue.

Restriction: To use this command, the user must have the special job control authority.

Required Parameter

JOB

Specifies the name of the job that is ended. If no job qualifier is given, all of the jobs currently in the system are searched for the simple job name. If more than one of the specified name is found, a qualified job name must be specified.

A job identifier is a qualified name with up to three elements. For example:

```
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

job-name: Specify the name of the job that is ended.

user-name: Specify the name of the user of the job that is ended.

job-number: Specify the number of the job that is ended.

Optional Parameters

OPTION

Specifies that the job is either ended in a controlled manner, which lets the application program perform end-of-job processing, or is ended immediately. In either case, the system performs certain job cleanup processing.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

Note: If a controlled end is issued against a job that is suspended because of a Hold Job (HLDJOB) command, system request option 1, Transfer Secondary Job (TFRSECJOB) command, or Transfer Group Job (TFRGRPJOB) command, the delay time does not start until the job is active.

***IMMED:** The job is ended immediately. This option may cause undesirable results (for example, if data has been partially updated) and should be used only after a controlled end has been attempted unsuccessfully.

Note: Ending a job with the *IMMED option does not necessarily remove the job from the system immediately. The system starts to perform end-of-job cleanup even though the immediate option is specified. This cleanup may be very brief or it may last for several minutes. Functions performed during end-of-job cleanup can include:

- Closing of database files
- Spooling of the job log to an output queue
- Cleanup of OS/400 system internal objects
- Showing the end-of-job display (for interactive jobs)

DELAY

Specifies the amount of time (in seconds) allowed for the job to complete its cleanup processing during a controlled end. This parameter is not used if OPTION(*IMMED) is specified. If the cleanup is not completed before the end of the delay time, the job is immediately ended. (Only system cleanup is performed.)

30: A maximum delay time of 30 seconds is allowed for cleanup before the job is ended.

delay-time: Specify the maximum amount of delay time (in seconds) before the job is ended. Valid values range from 1 through 999999 seconds. For additional information on the use of the DELAY parameter for ending an end-of-file delay job, refer to the EOFDLY (End of File Delay) parameter of the Override Database File (OVRDBF) command.

SPLFILE

Specifies whether spooled files created by this job are kept for normal processing or are deleted.

***NO:** The spooled files created by the job being ended are kept for normal processing by a writer.

***YES:** The spooled files created by the job being ended are deleted. The job log is not deleted.

LOGLMT

Specifies the maximum number of entries in the message queue of the job being ended that are written to the job log. This parameter can be used to limit the number of messages written to the job log printer file (QPJOBLOG) for a job that is ended. This option is particularly useful when a job is ended and its message queue contains an excessive number of entries.

If the ENDJOB command is used to change the message logging limit while the messages for the job that is ended are written to the spooled file, and the new limit is greater than the number written at the time the command is entered, messages continue to be written until the new limit is reached. If the new limit is less than the number of messages already written to the spooled file, a message indicating that the limit has been reached is immediately put in the spooled file as the last entry, and the remaining messages on the queue are ignored. If the limit is set to zero before any messages are written to the spooled file, no job log is produced for the job that is ended.

***SAME:** The value does not change.

***NOMAX:** There is no limit to the number of messages logged; all messages on the job message queue are written to the job log.

maximum-logged-entries: Specify the maximum number of messages written to the job log. This value is the maximum only if it is entered before the job log contains that many messages; otherwise, the limit just stops the process of writing any more messages to the job log. If 0 is specified before any messages are written to the log, no job log is produced.

ADLINTJOBS

Specifies whether additional interactive jobs associated with the job specified in the JOB parameter are being ended. The additional interactive jobs can be group jobs or all jobs associated with the work station (group and secondary jobs) where the job specified in the JOB parameter is running. The job specified in the JOB parameter does not have to be the active job. A job name entered on the JOB parameter must resolve to a single interactive job for this parameter to be used. If the job is not an interactive job, an error message is sent.

***NONE:** Only the job specified in the JOB parameter is ended.

***GRPJOB:** If the job specified in the JOB parameter is a group job, all jobs associated with the group are ended. If the job is not a group job, the job specified in the JOB parameter is ended.

***ALL:** All interactive jobs running on the work station associated with the job specified in the JOB parameter are ended, including group jobs and secondary jobs.

DUPJOBOPT

Specifies the action taken when duplicate jobs are found by this command.

***SELECT:** The selection display is shown when duplicate jobs are found during an interactive session. Otherwise, a message is issued.

***MSG:** A message is issued when duplicate jobs are found.

Examples

Example 1: Ending a Job Immediately

```
ENDJOB  JOB(JOB1)  OPTION(*IMMED)  SPLFILE(*YES)
```

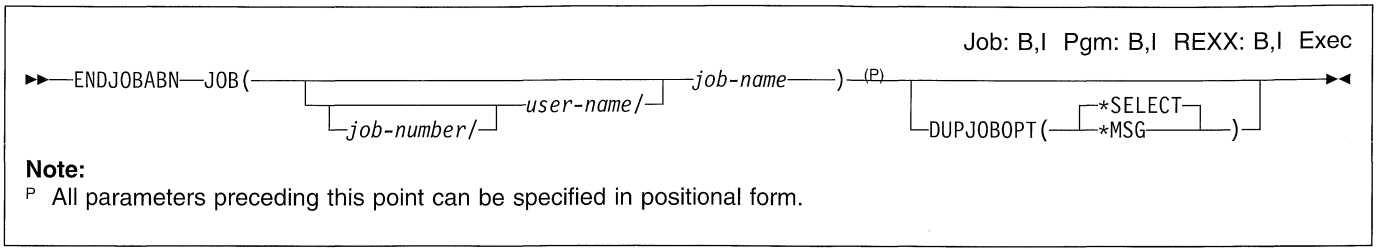
This command ends a job named JOB1 immediately. Spooled output produced by the job is deleted; the job log is saved.

Example 2: Saving Spooled Output

```
ENDJOB  JOB(001234/XYZ/JOB2)  OPTION(*CNTRLD)
  DELAY(50)  SPLFILE(*NO)
```

This command ends a job named 001234/XYZ/JOB2. Spooled output is saved for normal processing by the spooling writer. The job has 50 seconds to perform any cleanup routines, after which it is ended immediately.

ENDJOBABN (End Job Abnormal) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Job Abnormal (ENDJOBABN) command ends a job that cannot be ended successfully by running the End Job (ENDJOB) command with OPTION(*IMMED) specified. The ENDJOBABN command cannot be issued against a job until 10 minutes have passed following the request for immediate ending. This allows sufficient time for normal job ending functions to occur.

When the ENDJOBABN command is issued, most of the end-of-job processing is bypassed (including spooling of the job log, the end of job display for interactive jobs, and the end-of-job processing for the specific functions that are being performed). The part of the end-of-job processing that is attempted is allowed only five minutes to complete. If it does not do so in five minutes, the job is forced to end at that point. Because some of the job cleanup is not performed, the ENDJOBABN command should only be used when a job that is in the process of immediate ending does not finish ending and resources in use by the job are needed by another job or by the system. When the ENDJOBABN command is used, some resources in use by the ended job may be left unavailable until the next IPL.

Use of the ENDJOBABN command causes the next system end to be marked as ABNORMAL. Certain system functions are then called during the subsequent IPL to clear up conditions that may have resulted from running the ENDJOBABN command. This does not, however, cause any machine recovery functions to be called, nor do any access paths need to be rebuilt. Some storage in use by the job may become unavailable after the ENDJOBABN command is run and that available storage can be reclaimed by using the Reclaim Storage (RCLSTG) command.

Bypassing the job log writing process causes the job to have the status of JOBLOG PENDING (as shown on the DSPJOB status attributes display) after it has been ended with the ENDJOBABN command. The job log writing is not performed until the next IPL. However, the contents of the job log can be printed or shown by using the Display Job Log (DSPJOBLOG) command.

When the ENDJOBABN command is run, the following functions are performed successfully:

- Journaling entries
- Commitment control
- Making database files available for use by others
- Releasing file locks

This command fails to end a job or takes more than five minutes to do so in the following situations:

- When the job runs under a subsystem monitor that is hung, is abnormally slow, or has ended abnormally (the subsystem monitor performs part of the ending function).
- When the machine interface (MI) instruction running in the job is hung or is abnormally slow. The job cannot end until the MI instruction that is currently running completes or reaches a point of interruption.

Restrictions:

1. This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, and QSRV user profiles have private authorities to use the command.
2. After the ENDJOBABN command is run, subsequent ENDJOBABN commands cannot be issued against the job.
3. Users cannot end a reader, writer, subsystem monitor, or system job.
4. Users cannot run the ENDJOBABN command until ten minutes *after* immediate ending of the job is started. Immediate ending of the job is started in the following ways:
   - When the End Job (ENDJOB) command with OPTION(*CNTRLD) is specified and the delay time ends
   - When the End Job (ENDJOB) command with OPTION(*IMMED) is issued
   - When the End Subsystem (ENDSBS) command with OPTION(*CNTRLD) is issued against the subsystem in which the job is running and the delay time ends
   - When the End Subsystem (ENDSBS) command with OPTION(*IMMED) is issued against the subsystem in which the job is running
   - When the End System (ENDSYS) command with OPTION(*IMMED) is issued, or OPTION(*CNTRLD) is issued and the delay time ends

Required Parameters

JOB

Specifies the name of the job to be ended. If no job qualifier is given, all of the jobs currently in the system

are searched for the simple job name. If more than one of the specified names are found, a qualified job name must be specified to distinguish them.

A job identifier is a qualified name with up to three elements. For example:

```
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

job-name: Specify the name of the job to be ended.

user-name: Specify the name of the user of the job to be ended.

job-number: Specify the number of the job to be ended.

DUPJOBOPT

Specifies the action taken when duplicate jobs are found by this command.

***SELECT:** The selection display is shown when duplicate jobs are found during an interactive session. Otherwise, a message is issued.

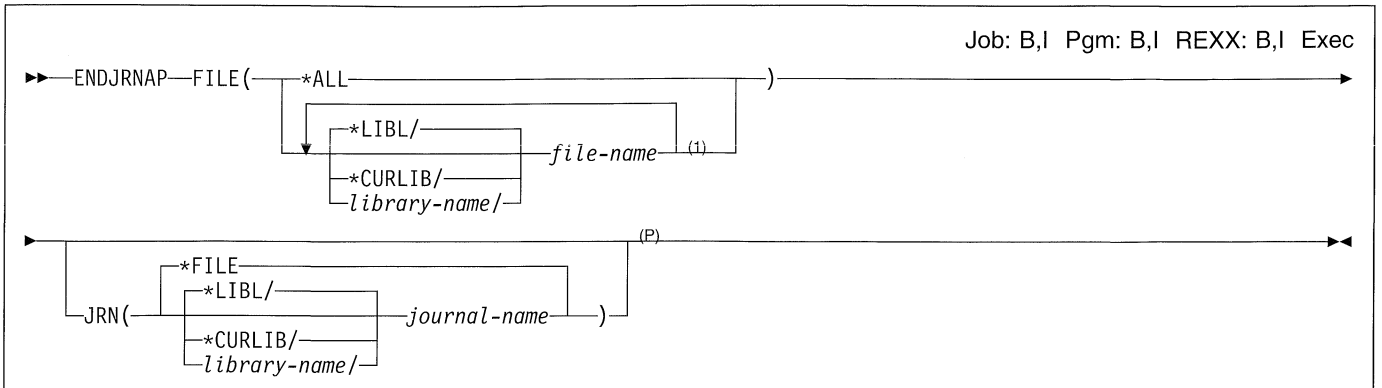
***MSG:** A message is issued when duplicate jobs are found.

Example

```
ENDJOBABN  JOB(000310/SMITH/PAYROLL)
```

This command ends the batch job 000310/SMITH/PAYROLL after the failure of an earlier attempt to end it with the ENDJOB command. The ENDJOBABN command can be issued only after waiting at least ten minutes for the job to end after issuing the ENDJOB command.

ENDJRNAP (End Journal Access Path) Command



Notes:
1 A maximum of 50 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The End Journal Access Path (ENDJRNAP) command is used to end journaling of the access paths of a file.

All access paths currently being journaled to a specific journal may also have journaling stopped.

Restrictions:

1. The access paths for the files specified on the command cannot be in use for any reason at the time the command is running.
2. Overrides are not applied to the files listed in the FILE parameter.
3. If FILE(*ALL) is specified, a journal name must be specified.
4. If a journal name and a list of file names are specified, then all the access paths for the listed files must be currently journaled to the indicated journal.
5. Journaling entries for any physical file does not end by the running of this command.

Required Parameter

FILE

Specifies a maximum of 50 qualified names of the database files for which journaling for their access paths is ended.

***ALL:** Journaling ends for all access paths that are currently being journaled to the specified journal.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specifies the name of the database file that has journaling ended for its access paths.

Optional Parameter

JRN

Specifies the qualified name of the journal to which the access paths for the indicated files are currently being journaled.

***FILE:** The journal is determined by the system from the specified file names.

The name of the journal can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

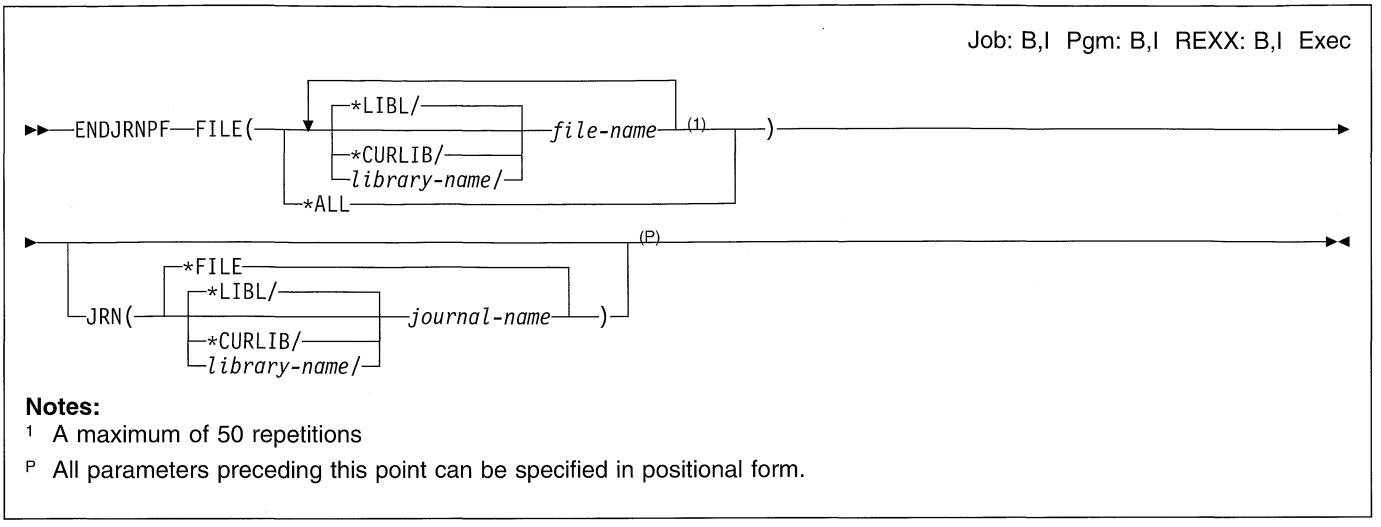
journal-name: Specifies the name of the journal to which the access paths for the indicated files are currently being journaled.

Example

```
ENDJRNAP   FILE(MYLIB/MYFILE)
```

This command ends the journaling for all access paths of the file MYFILE in the library MYLIB.

ENDJRNPF (End Journal Physical File) Command



Notes:
1 A maximum of 50 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The End Journal Physical File (ENDJRNPF) command is used to end journaling of changes for a specific physical file and all of its members.

All physical files currently being journaled to a specific journal may also have journaling stopped.

Restrictions:

1. Members in the files specified on the command cannot be in use for any reason at the time the command is running.
2. Overrides are not applied to the files listed in the FILE parameter.
3. If FILE(*ALL) is specified, a journal name must be specified.
4. If a journal name and a list of file names are specified, all files must be currently journaled to the indicated journal.

Required Parameter

FILE

Specifies a maximum of 50 qualified names of the physical files for which changes are no longer being journaled .

***ALL:** All files currently being journaled to the indicated journal are to stop having their changes journaled.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

file-name: Specify the name of the database file for which journaling has ended.

Optional Parameter

JRN

Specifies the qualified name of the journal to which changes in the files are currently being journaled.

***FILE:** The journal is determined by the system from the specified file names.

The name of the journal can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

journal-name: Specify the name of the journal to which the indicated files are currently being journaled.

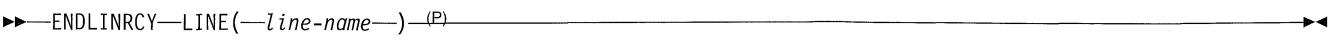
Example

```
ENDJRNPF    FILE(MYLIB/MYFILE)
```

This command stops the journaling of all changes to all members of file MYFILE in library MYLIB. Changes made after this command is run are not journaled.

ENDLINRCY (End Line Recovery) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The End Line Recovery (ENDLINRCY) command ends error recovery procedures for a specific line. If any type of failure occurs after this command is run, an inquiry message is sent.

Use the Resume Line Recovery (RSMLINRCY) command to reestablish error recovery procedures for the line.

Required Parameters

LINE

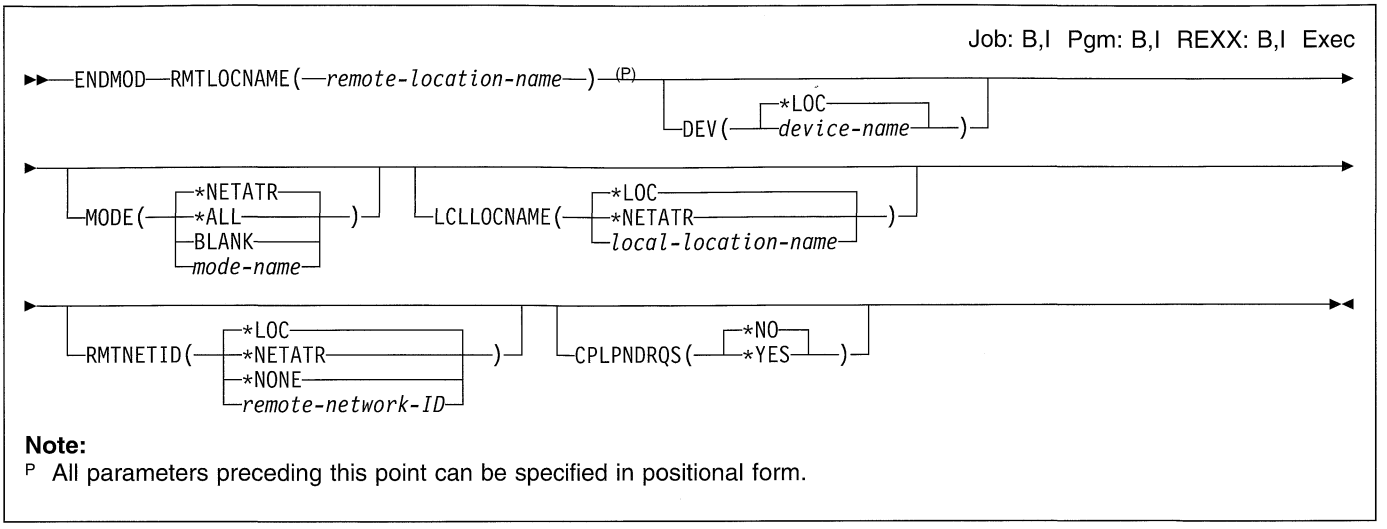
Specifies the name of the line whose recovery is stopped.

Example

```
ENDLINRCY  LINE(NYC2)
```

This command ends error recovery procedures for the line named NYC2.

ENDMOD (End Mode) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The End Mode (ENDMOD) command ends (deactivates) a single mode or all active modes for a specific advanced program-to-program communications (APPC) remote location. The mode remains inactive until a Start Mode (STRMOD) command is run to start the mode. This command can be used to end all the sessions for a particular remote location and to cause an active switched connection to disconnect. The user can also specify how activities that have been requested on the remote system but have not yet been performed are to be handled.

The *APPC Programmer's Guide* has more information on the ENDMOD command.

Restriction: This command cannot be used to end (deactivate) PC Support mode (QPCSUPP) at a remote location.

Required Parameters

RMTLOCNAME

Specifies the remote location name for one or more modes which are being ended.

Optional Parameters

DEV

Specifies the device description name used with the remote location.

***LOC:** The device associated with the remote location is used. If several devices are associated with the remote location, the system determines which device is used.

device-name: Specify the name of the device for which one or more modes are being ended.

MODE

Specifies the mode that is ended.

***NETATR:** The mode name specified in the network attributes is used.

***ALL:** All modes currently active by the remote location are ended.

BLANK: The mode name consisting of 8 blank characters is used.

mode-name: Specify a mode name for the device.

LCLLOCNAME

Specifies the local location name.

***LOC:** The device associated with the remote location is used. If several devices are associated with the remote location, the system determines which device is used.

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

local-location-name: Specify the local location name.

RMTNETID

Specifies the remote network ID used with the remote location.

***LOC:** The remote network identifier (ID) associated with the remote location is used. If several remote network IDs are associated with the remote location, the system determines which remote network ID is used.

***NETATR:** The LCLLOCNAME value specified in the system network attributes is used.

***NONE:** No remote network identifier (ID) is used.

remote-network-ID: Specify the name of the remote network ID used.

CPLPNDRQS

Specifies how the remote location processes all pending user work before ending the mode.

***NO:** Requested activities currently in progress at the remote location can complete; activities that have been requested, but not started at the remote location will not be performed.

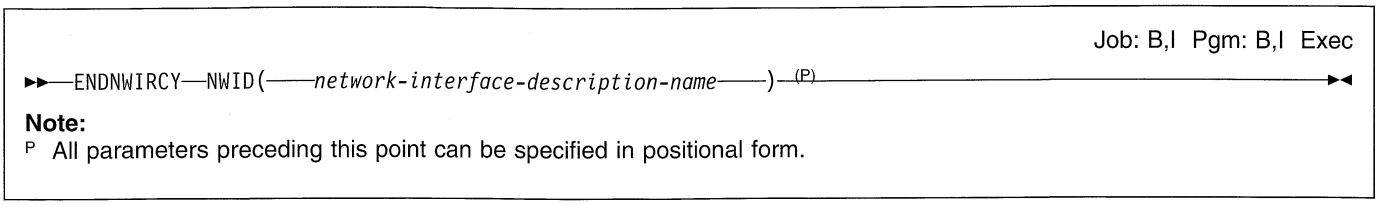
***YES:** All requested activities are allowed to complete before the mode is ended.

Example

```
ENDMOD  RMTLOCNAME(APPCRLOC)  MODE(APPCMOD)
```

This command ends a mode named APPCMOD for remote location APPCRLOC.

ENDNWIRCY (End Network Interface Recovery) Command



Purpose

The End Network Interface Recovery (ENDNWIRCY) command ends automatic error recovery procedures for a network interface description.

The Resume Network Interface Recovery (RSMNWIRCY) command is used to reestablish error recovery procedures for the network interface description.

Required Parameters

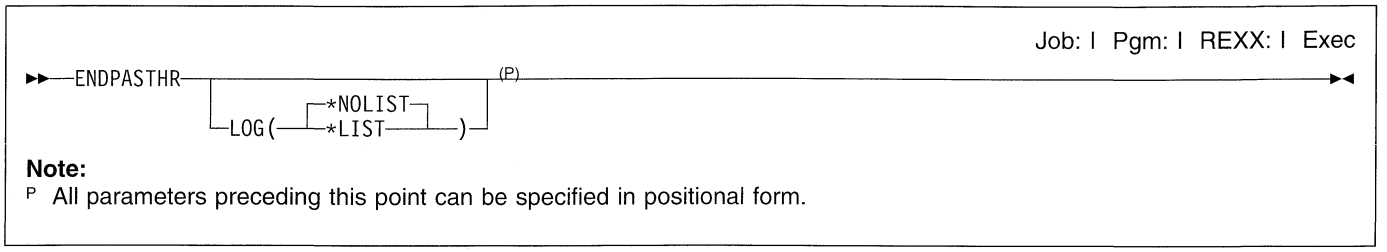
NWID
Specifies the name of the network interface description for which recovery is to be ended.

Example

```
ENDNWIRCY   NWID(ISDNNET)
```

This command ends automatic error recovery procedures for the network interface named ISDNNET.

ENDPASTHR (End Pass-Through) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The End Pass-Through (ENDPASTHR) command ends a pass-through session. The ENDPASTHR command signs you off the target system, and ends the advanced program-to-program communications (APPC) session. This releases the virtual display device from the subsystem and returns it to the vary-on pending condition. The job at each intermediate node for the pass-through session also ends. Control returns to the source system for the next command following the Start Pass-Through (STRPASTHR) command.

Note: The ENDPASTHR command uses the SIGNOFF command as part of its processing. If the system has a SIGNOFF command that appears in the library list before QSYS/SIGNOFF, the SIGNOFF command is used by ENDPASTHR. The SIGNOFF command should not use the ENDPASTHR command. It sends the system into a loop when you end your pass-through session.

The ENDPASTHR command does not end the passthrough session when there is a secondary interactive job at the target system. One of the jobs must be ended (by using SIGNOFF or ENDJOB) before the ENDPASTHR command can be entered.

If the ENDPASTHR command is entered and there is not a pass-through session, an error message is sent.

More information about pass-through is in the *Remote Work Station Guide*.

Optional Parameters

LOG

Specifies whether the JOBLOG is saved at the remote system.

***NOLIST:** The information in the job log is deleted when the job ends.

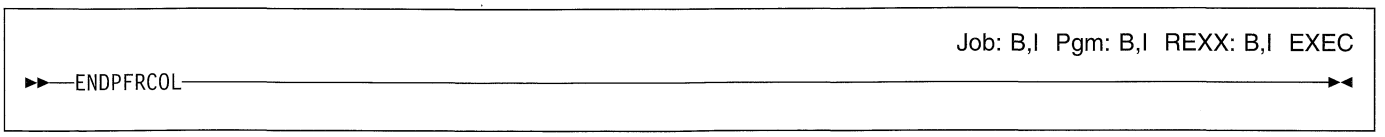
***LIST:** The JOBLOG is saved at the remote system.

Example

```
ENDPASTHR  LOG(*LIST)
```

This command ends a pass-through session and prints a job log.

ENDPFRCOL (End Performance Collection) Command



Purpose

The End Performance Collection (ENDPFRCOL) command stops the automatic collection of performance data that is defined on the Work with Performance Collection (WRKPFRCOL) command.

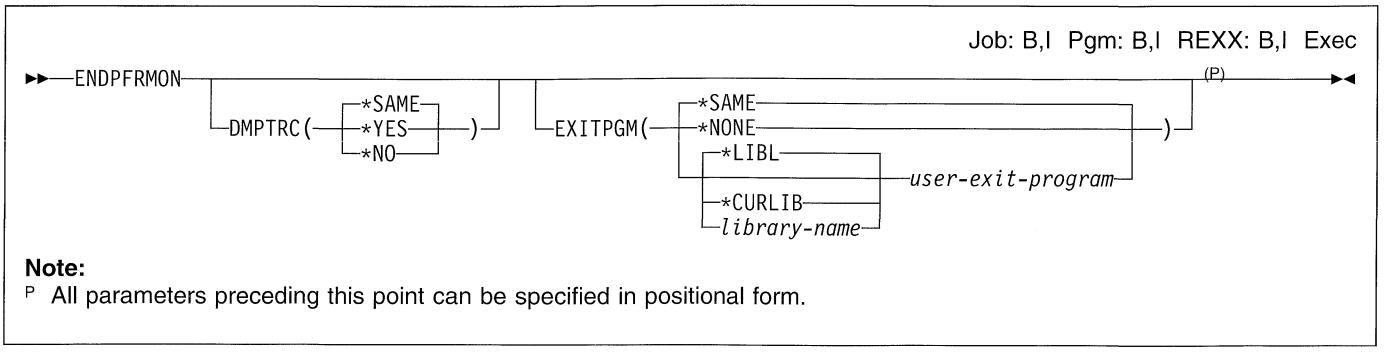
There are no parameters for this command.

Example

```
ENDPFRCOL
```

This command stops the collection of performance data.

ENDPFRMON (End Performance Monitor) Command



Purpose

The End Performance Monitor (ENDPFRMON) command stops the collection of performance data. After the collection of performance data stops, all of the database files are closed, and the performance monitor job ends.

Optional Parameters

DMPTRC

Specifies whether the trace is dumped to a database file (QAPMDMPT). If the trace is started, but is not dumped when the performance monitor ends, the trace can be dumped at a later time by using the Dump Trace (DMPTRC) command.

***SAME:** The value does not change.

***YES:** The trace data, if any, is dumped.

***NO:** The trace data is not dumped.

EXITPGM

Specifies the user-written exit program that is called to process the performance data collected by this command.

***SAME:** The value does not change.

***NONE:** No exit program is specified.

The name of the exit program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

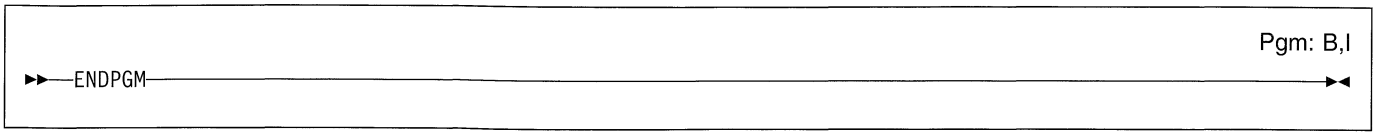
user-exit-program: Specify the name of the user exit program to be used.

Example

```
ENDPFRMON
```

This command stops the collection of performance data. Depending on what was specified by the DMPTRC parameter of the STRPFRMON command, the trace data may be dumped to the QAPMDMPT database file.

ENDPGM (End Program) Command



Purpose

The End Program (ENDPGM) command specifies the end of a CL program. When the command is processed, it performs the same function as a RETURN command. That is, control is returned to the command immediately following the CALL command in the calling program.

The ENDPGM command is not required at the end of a CL program. If the last statement in a CL program source file is reached and no ENDPGM command is found, an ENDPGM command is assumed by the compiler.

Restriction: This command is valid only within a CL program.

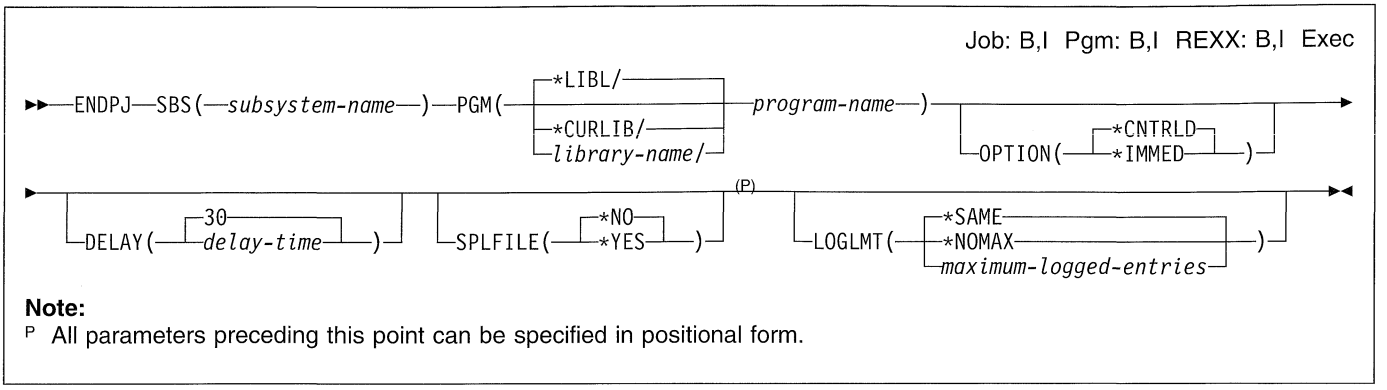
There are no parameters for this command.

Example

```
PGM
  •
  •
  •
ENDPGM
```

This program is identified by a PGM command that contains no parameters and is ended by the ENDPGM command.

ENDPJ (End Prestart Jobs) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Prestart Jobs (ENDPJ) command ends all jobs and any associated inline data files for a prestart job entry in an active subsystem. Jobs may be waiting for a program start request or may already be associated with a program start request. Spooled files associated with the jobs being ended can also be deleted or allowed to remain on the output queue. The limit on the number of messages being written to each of the job logs can also be changed.

Restriction: This command is restricted to a user with job control special authority.

Required Parameters

SBS

Specifies the name of the active subsystem that contains the prestart job entry. Specify the name of the subsystem.

PGM

Specifies the qualified name of the program for the prestart job entry.

The name of the program can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

program-name: Specify the name of the program for the prestart job entry.

Optional Parameters

OPTION

Specifies that the job is either ended in a controlled manner, which lets the application program perform end-of-job processing, or is ended immediately. In either case, the system performs certain job cleanup processing.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

Note: If a controlled end is issued against a job that is suspended because of a Hold Job (HLDJOB) command, a system request option 1, Transfer Secondary Job (TFRSECJOB) command, or Transfer Group Job (TFRGRPJOB) command, the delay time does not start until the job is active.

***IMMED:** The job is ended immediately. This option may cause undesirable results (for example, if data has been partially updated) and should be used only after a controlled end has been attempted unsuccessfully.

Note: Ending a prestart job with the *IMMED option does not immediately remove the job from the system. The system starts to perform end-of-job cleanup even though the immediate option is specified. This cleanup can be very brief or last several minutes. Functions performed during end-of-job cleanup can include:

- Closing of database files
- Spooling of the job log to an output queue
- End-of-job processing of OS/400 system internal objects
- Displaying the end-of-job display (for interactive jobs)

DELAY

Specifies the time (in seconds) allowed for the program to complete end-of-job processing during a controlled end. This parameter is not used if OPTION(*IMMED) is specified. If the cleanup is not complete before the end

of the delay time, the job is immediately ended. (Only end-of-job processing is performed.)

30: Up to 30 seconds of delay time is allowed for cleanup before the job is ended.

delay-time: Specify the maximum delay time (in seconds) before the job is ended. Valid values range from 1 through 999999 seconds. For additional information on ending an end-of-file delay job, refer to the EOFDLY parameter in the Override Database File (OVRDBF) command description.

SPLFILE

Specifies whether spooled files created by the job are retained for normal processing by a writer or are deleted.

***NO:** The spooled files created by the job being ended are retained for normal processing by a writer.

***YES:** The spooled files created by the job being ended are deleted. The job log is not deleted.

LOGLMT

Specifies the maximum number of entries in the message queue of the job being ended that are written to the job log. This parameter can be used to limit the number of messages written to the job log printer file, QPJOBLOG, for a job that is ended. This option is particularly useful when a job is ended and its message queue contains an excessive number of entries.

If this command is used to change the message logging limit while the messages for the ended job are being written to the spooled file, and if the new limit is greater than the number written at the time the command is entered, messages continue to be written until the new limit is reached. If the new limit is less than the number of messages already written to the spooled file, a message indicating that the limit is reached is immediately put in the spooled file as the last entry, and the rest of the messages on the queue are ignored. If the limit is set to zero before any messages are written to the spooled file, no job log is produced for the ended job.

***SAME:** The value does not change.

***NOMAX:** There is no limit on the number of messages being logged. Messages on each job message queue are written to the log of each job.

maximum-logged-entries: Specify the maximum number of messages being written to the job log for each job. This value is the maximum if it is entered before the job log contains that many messages. Otherwise, the limit merely stops the process of writing any more messages to the job log. If zero is specified before any messages are written to the log, no job log is produced.

Examples

Example 1: Ending a Job Immediately

```
ENDPJ  SBS(SBS1)  PGM(PJLIB/PJPGM)  OPTION(*IMMED)
  SPLFILE(*YES)
```

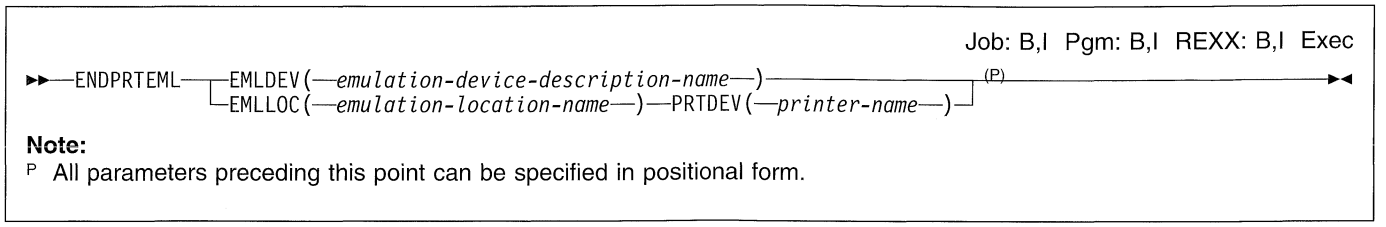
This command ends all jobs associated with prestart job entry PJPGM in subsystem SBS1 immediately. Spooled output produced by these prestart jobs is deleted and the job log is saved.

Example 2: Delaying a Job End

```
ENDPJ  SBS(SBS2)  PGM(PJPGM2)  OPTION(*CNTRLD)
  DELAY(50)  SPLFILE(NO)
```

This command ends all the jobs associated with prestart job entry PJPGM2 in subsystem SBS2. Spooled output for these prestart jobs is saved for normal processing by the spooling writer. The jobs have 50 seconds to perform any cleanup routines, after which they are immediately ended.

ENDPRTEML (End Printer Emulation) Command



Purpose

The End Printer Emulation (ENDPRTEML) command ends printer emulation without ending the job. If there is another request in the job, that request is then processed.

This command closes the file to the host system, and then writes the last data received from the host system to the spooled file or printer by closing the printer file.

In some cases, the request does not take effect immediately. The request is delayed while any of the following conditions exist in the printer emulation request:

- Printing a block sent from the host system.
- Waiting for a printer error to be cleared (for example, a paper jam).
- Waiting for a reply to a PA1 or PA2 inquiry message.
- Waiting for error recovery to be done to the host system or printer device.
- The job has been held by using the HLDJOB command. When the condition is cleared, the End Printer Emulation request takes effect, and the printer emulation request ends.

Required Parameters

EMLDEV
Specifies the name of the printer emulation device requested to receive data from the host system. The printer emulation job using this device is informed of the request and closes the printer file. This forces all of the data received from the host system to the spooled file or printer. The printer file is then reopened and printer emulation continues. To use this function, the user must be authorized to the device.

EMLLOC
Specifies the remote location name associated with this session. This name is defined during configuration and refers to the remote location where communication takes place. This value was specified on the Start Printer Emulation (STRPRTEML) command.

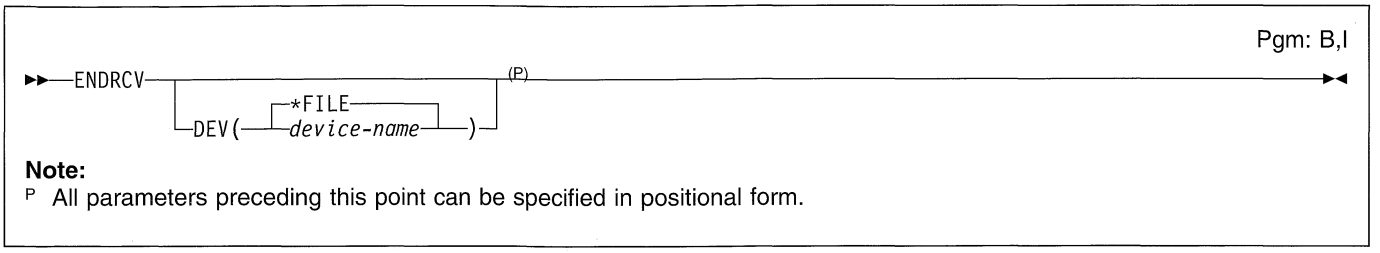
PRTDEV
Specifies the name of the printer device that is used to print the spooled output. This value must match the value specified on the Start Printer Emulation (STRPRTEML) command. This parameter must be specified when the EMLLOC parameter is specified.

Example

```
ENDPRTEML   EMLDEV(HOSTPRT3)
```

This command ends the printer emulation request that is using the device HOSTPRT3.

ENDRCV (End Receive) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Receive (ENDRCV) command is used to end (cancel) a request for input made by a previously issued Receive File (RCVF) or Send/Receive File (SNDRCVF) command that had WAIT(*NO) specified. The ENDRCV command ends an input request even if the user enters the requested data at the display station at the same time that the command is processed. If the requested data is entered and is being sent to the program when the end receive operation is performed, the entered data is lost. If there is no outstanding input request, the command is ignored.

Restriction: This command is valid only for display files within CL programs. It cannot be used for database files.

Optional Parameters

DEV

Specifies the name of the display device for which the request for input is being ended.

***FILE:** The name of the device having the response from it ended is contained in the device file that was declared in the FILE parameter of the Declare File (DCLF) command. If the device file has more than one device name specified in it, *FILE cannot be specified.

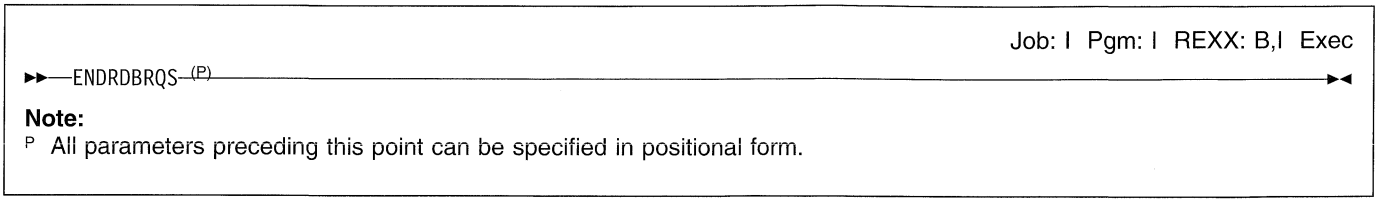
device-name: Specify the name of the display device from which a response is being ended.

Example

```
ENDRCV    DEV(MYDISPLAY)
```

Assume that a RCVF command with WAIT(*NO) was issued earlier in the CL program to request input from the device file declared earlier in the DCLF command and from the display device MYDISPLAY. When this ENDRCV command is processed, that request for input from MYDISPLAY is ended.

ENDRDBRQS (End Relational Database Request) Command



Purpose

The End Relational Database Request (ENDRDBRQS) command ends a previously requested SQL operation that is running on a remote system.

This command can achieve the same result as Option 50 on the System Request menu by using it from an attention program. Under commitment control, once the cancelation is finished, only the SQL request that is running on the remote system is rolled back. The request originator can then continue to commit or roll back the entire unit of work.

Restriction: This command can only be used by the originator of the remote SQL request within the same job.

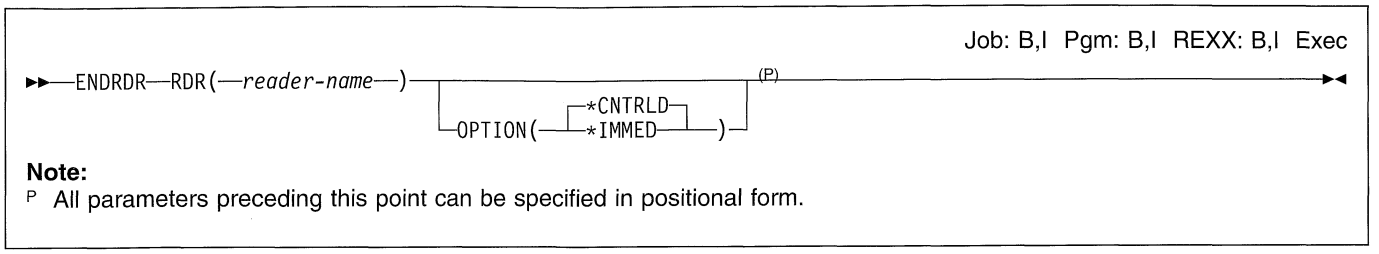
There are no parameters for this command.

Example

```
ENDRDBRQS
```

This command allocates a second conversation to send this request to the remote system to cancel the SQL request that is running. After the cancelation is finished, the request originator continues to the next request.

ENDRDR (End Reader) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Reader (ENDRDR) command ends the specified diskette or database reader and makes its associated input device available to the system. The reader can be stopped either immediately, without completing the current job being read, or at the end of the current job. If the reader is in a hold state when this command is issued, the reader is stopped immediately.

Required Parameters

RDR

Specifies the name of the diskette or database reader being ended. The reader's associated input device is made available to the system.

Optional Parameters

OPTION

Specifies when the ended reader stops processing.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

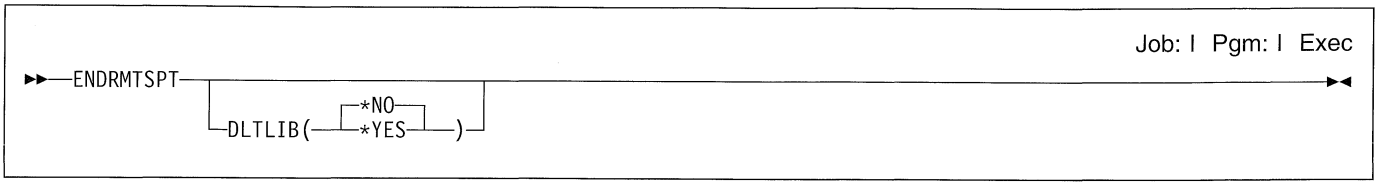
***IMMED:** The reader stops processing in a controlled manner immediately. The job being read in is not placed on the job queue.

Example

```
ENDRDR  RDR(DISKETTE)
```

This command stops the reader DISKETTE as soon as the current job is completely read in and releases that device to the system.

ENDRMTSPT (End Remote Support) Command



Purpose

The End Remote Support (ENDRMTSPT) command varies off and deletes the configuration objects created by the Start Remote Support (STRRMTSPT) command.

Restriction: This command is not valid when the user is signed on the remote support work station.

Optional Parameters

DLTLIB

Specifies whether the remote support library QTILIB is deleted when remote support is ended.

***NO:** The remote support library is not deleted.

***YES:** The remote support library is deleted.

Example

```
ENDRMTSPT
```

This command varies off and deletes the configuration objects that have been created.

ENDRQS (End Request) Command

Job: I Pgm: I REXX: I Exec



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End Request (ENDRQS) command ends (cancels) a previously requested operation (command). One common use of the ENDRQS command is to cancel a request that is currently stopped at a breakpoint. This command function is also available as an option on the System Request menu.

If the ENDRQS command cannot be processed immediately because a system function that cannot be interrupted is currently running, the command is delayed until interruption is allowed.

When a request is ended, an escape message (see the *Programming Reference Summary*) is sent to the request processing program that is currently called at the request level being canceled. Request processing programs can monitor for the escape message so that cleanup processing can be done when the request is canceled. The static storage and open files are reclaimed for any program that was called by the request processing program. None of the programs called by the request processing program are notified of the cancelation, so they have no opportunity to stop processing. To become a request processing program, the program must receive a request message.

If the ENDRQS command is in a program, that program must become a request processor before it issues this command.

More information on how to set up a program to become a request processor is in the *CL Programmer's Guide*.

Note: External objects that are locked by the Allocate Object (ALCOBJ) command are not unlocked (deallocated) by the canceled request.

Optional Parameter

RQSLVL

Specifies the (command) request level at which the command being canceled was entered.

***PRV:** The command entered at the immediately previous level is being canceled.

request-level: Specify the request level at which the command being canceled was entered. All request levels from the level specified to the current level are canceled.

Examples

Example 1: Ending a Command

```
CALL PROGA      (This is level 1)
     •
     •
     •
Breakpoint occurs

CALL PROGB      (This is level 2)
     •
     •
     •
Breakpoint occurs

ENDRQS          (This is level 3)
```

In this example, because RQSLVL(*PRV) is the default, the request made at level 2 is canceled. The user can then enter another command at level 2 or press F3 to show the PROGA breakpoint display again.

Example 2: Ending a Command

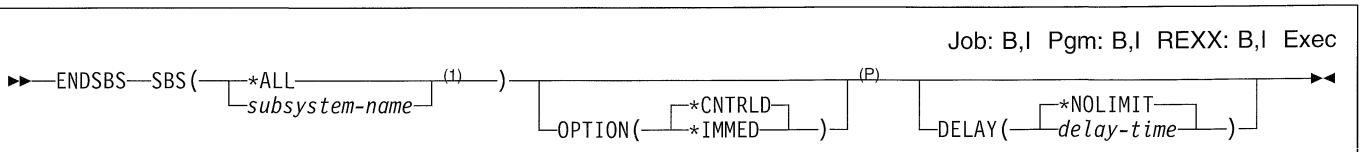
```
CALL PROGA          (This is level 1)
     •
     •
     •
Breakpoint occurs

CALL PROGB          (This is level 2)
     •
     •
     •
Breakpoint occurs

ENDRQS  RQSLVL(1)   (This is level 3)
```

In this example, the request made at the highest level (CALL PROGA) is canceled. Consequently, any requests made between level 1 and level 3 are also canceled.

ENDSBS (End Subsystem) Command



Notes:
1 SBS(*ALL) is not valid in a pass-through job or in a work station function job.
P All parameters preceding this point can be specified in positional form.

Purpose

The End Subsystem (ENDSBS) command ends the specified subsystem (or all active subsystems) and specifies what happens to active work being processed by that subsystem or all subsystems. No new jobs are started in the subsystem or subsystems after this command is run.

Interactive jobs that have been transferred to a job queue by the Transfer Job (TFRJOB) command are ended as part of ending the subsystem. If an initial program load (IPL) occurs while either a batch or interactive job is on a job queue (because of the TFRJOB command), that job is removed from the job queue during IPL and its job log is produced.

Restrictions: 1) If the controlling subsystem is being ended, because either its name or *ALL is specified for the SBS parameter, this command can be entered only in an interactive job. The interactive job must be in the controlling subsystem, and the command can be entered only from a work station (associated with the interactive job) whose work station entry in the controlling subsystem description specifies AT(*SIGNON). 2) To use this command, you must have job control (*JOBCTL) authority. 3) To use this command, you must have object operational authority to the subsystem.

Required Parameters

SBS

Specifies the name of the subsystem to be ended, or it specifies that all active subsystems are to be ended.

***ALL:** All the subsystems that are currently active are ended. All jobs are ended except the job in which this command is entered. When this value is specified, the QSYSOPR message queue should be in break delivery mode in the job issuing the ENDSBS command.

subsystem-name: Specify the simple name of the subsystem to be ended. If the subsystem specified is the controlling subsystem, the interactive job from which the command was issued remains active. Also, if the subsystem specified is the controlling subsystem and the job that issues this command is one of two jobs that are active at the work station (through the use of the system request key or the Transfer Secondary Job (TFRSECJOB) command), neither of the jobs is forced to end. The controlling subsystem does not end until you end one of the jobs (either by signing off in one job or by ending one job from the other).

Optional Parameters

OPTION

Specifies whether jobs in the subsystem are ended in a controlled manner (ending jobs in a controlled manner lets the application programs perform end of job processing) or immediately.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

***IMMED:** The job is ended immediately. This option may cause undesirable results (for example, if data has been partially updated) and should be used only after a controlled end has been attempted unsuccessfully.

DELAY

Specifies the amount of time (in seconds) allowed in which to complete a controlled subsystem end operation. If this amount of time is exceeded, any jobs still running in the subsystem are ended immediately.

***NOLIMIT:** The amount of time in which to complete a controlled end operation is not limited.

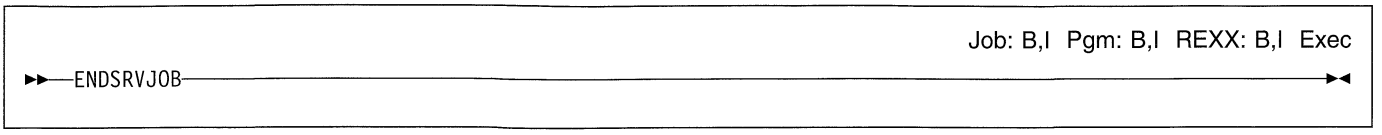
delay-time: Specify the number of seconds in which the end operation is completed. Valid values range from 1 through 99999 seconds.

Example

```
ENDSBS   SBS(QBATCH)  OPTION(*CNTRLD)  DELAY(60)
```

This command ends all active jobs in the QBATCH subsystem and ends the subsystem. The active jobs are allowed 60 seconds to perform application-provided end-of-job processing.

ENDSRVJOB (End Service Job) Command



Purpose

The End Service Job (ENDSRVJOB) command ends the remote job service operation. This command stops the service operation that began when the Start Service Job (STRSRVJOB) command was entered.

Restrictions:

1. If tracing or debugging is active in the serviced job when this command is entered, the remote service operation is *not* ended.
2. This command is shipped with public *EXCLUDE authority.
3. The following user profiles have private authorities to use the command:
   - QPGMR
   - QSYSOPR
   - QSRV
   - QSRVBAS

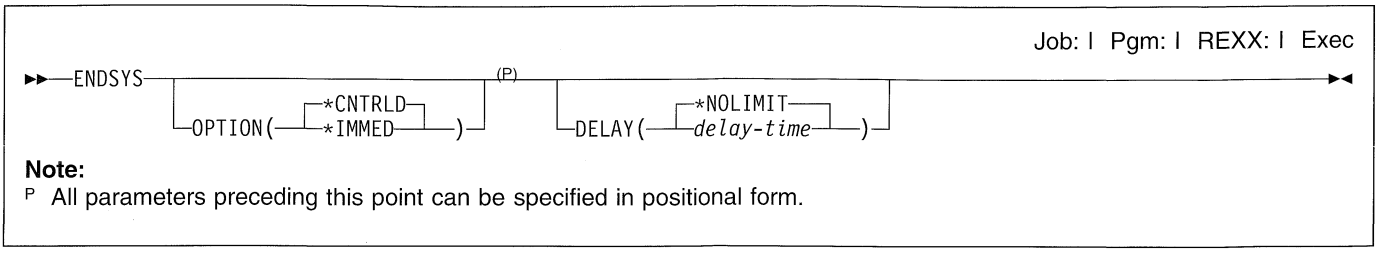
There are no parameters for this command.

Example

```
ENDSRVJOB
```

This command stops the service operation of the job currently being serviced.

ENDSYS (End System) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The End System (ENDSYS) command ends most activity on the system and leaves the system in a condition in which only the console is active in the controlling subsystem. This is normally done so that the programs to diagnose equipment problems can be run. This condition is called the restricted state and is required for operations like SAVSYS or RCLSTG.

If two jobs are active in the controlling subsystem at the console (through the use of the system request key), neither of the jobs is forced to end. The ENDSYS command cannot complete running until the user ends one of the jobs (either by signing off in one job or by canceling one job from the other).

All active subsystems are notified that an end system operation is in process. No new jobs or routing steps can be accepted by the subsystems. The ENDSYS command also specifies what happens to all active work.

Interactive jobs that are transferred to a job queue by the Transfer Job (TFRJOB) command are ended as part of subsystem ending. If an initial program load (IPL) occurs while either a batch or interactive job is on a job queue (because of the TFRJOB command), that job is removed from the job queue during IPL and its job log is produced.

Restriction: This command can be entered only in an interactive job in the controlling subsystem. To use this command, the user must have job control (*JOBCTL) authority. This command is not allowed in a pass-through job or in a work station function job.

Optional Parameters

OPTION

Specifies whether all active jobs are ended in a controlled manner (which lets the application programs perform end of processing) or immediately. In either case, the system performs certain job cleanup functions.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

***IMMED:** The job is ended immediately. This option may cause undesirable results (for example, if data has been partially updated) and should be used only after a controlled end has been attempted unsuccessfully.

DELAY

Specifies the amount of time (in seconds) that the controlled end operation is allowed. If this amount of time is exceeded and the end operation is not complete, any jobs still being processed are ended immediately, except for those running long-running system instructions.

***NOLIMIT:** The amount of time to complete a controlled end operation is not limited.

delay-time: Specify the number of seconds in which the end operation is completed. Valid values range from 1 through 99999 seconds.

Examples

Example 1: Ending System Activity

```
ENDSYS
```

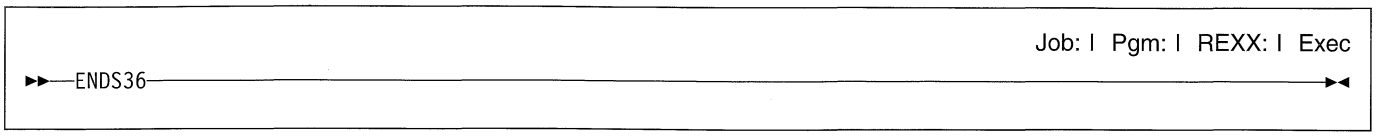
This command ends the system activity after all active jobs in the system are allowed to perform their own end of processes. The amount of time the end can take is not limited.

Example 2: Ending System Activity After Jobs are Ended

```
ENDSYS   OPTION(*IMMED)
```

This command ends system activity after all active jobs are immediately ended.

ENDS36 (End System/36) Command



Purpose

The End System/36 (ENDS36) command allows the user to end the System/36 Environment session that was started with a Start System/36 (STRS36) command.

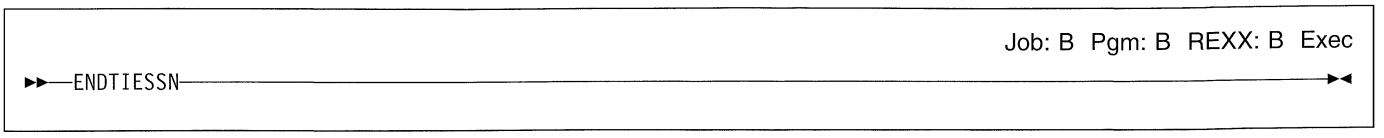
There are no parameters for this command.

Example

```
ENDS36
```

This command immediately ends the System/36 Environment session and any programs or procedures that are running in the System/36 Environment. If the ENDS36 command is in a procedure or in a program, the statements following the command are ignored.

ENDTIESSN (End Technical Information Exchange Session) Command



Purpose

The End Technical Information Exchange Session (ENDTIESSN) command allows the user to disconnect the communications line used for TIE batch commands.

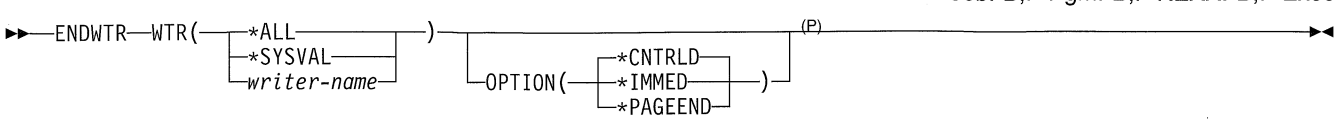
There are no parameters for this command.

Example

```
ENDTIESSN
```

This command ends the TIE function by disconnecting the communications line used for TIE batch commands.

ENDWTR (End Writer) Command



Note:
(P) All parameters preceding this point can be specified in positional form.

Purpose

The End Writer (ENDWTR) command ends the specified spooling writer and makes its associated output device available to the system. The writer can be ended immediately or in a controlled manner. If it is ended immediately, the writer stops writing the file and the file is again available on the output queue. If it is ended in a controlled manner, the writer finishes writing the current file (or a copy of a file), or it finishes printing a page of the file, if it is a printer writer, before it is ended.

Required Parameters

WTR

Specifies the name of the spooling writer being ended. The writer's output device is available to the system.

***ALL:** All writers that are started are ended.

***SYSVAL:** The writer started to the system default printer is ended.

writer-name: Specify the name of the writer being ended.

Optional Parameters

OPTION

Specifies when the writer stops processing. Output stops at the end of the spooled file (or copy of a file) currently being written to an output device.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

Note: The spooled file that is currently being processed remains on the output queue. If the printer is a printer writer, printing stops and a form feed is done.

***IMMED:** The operation ends immediately.

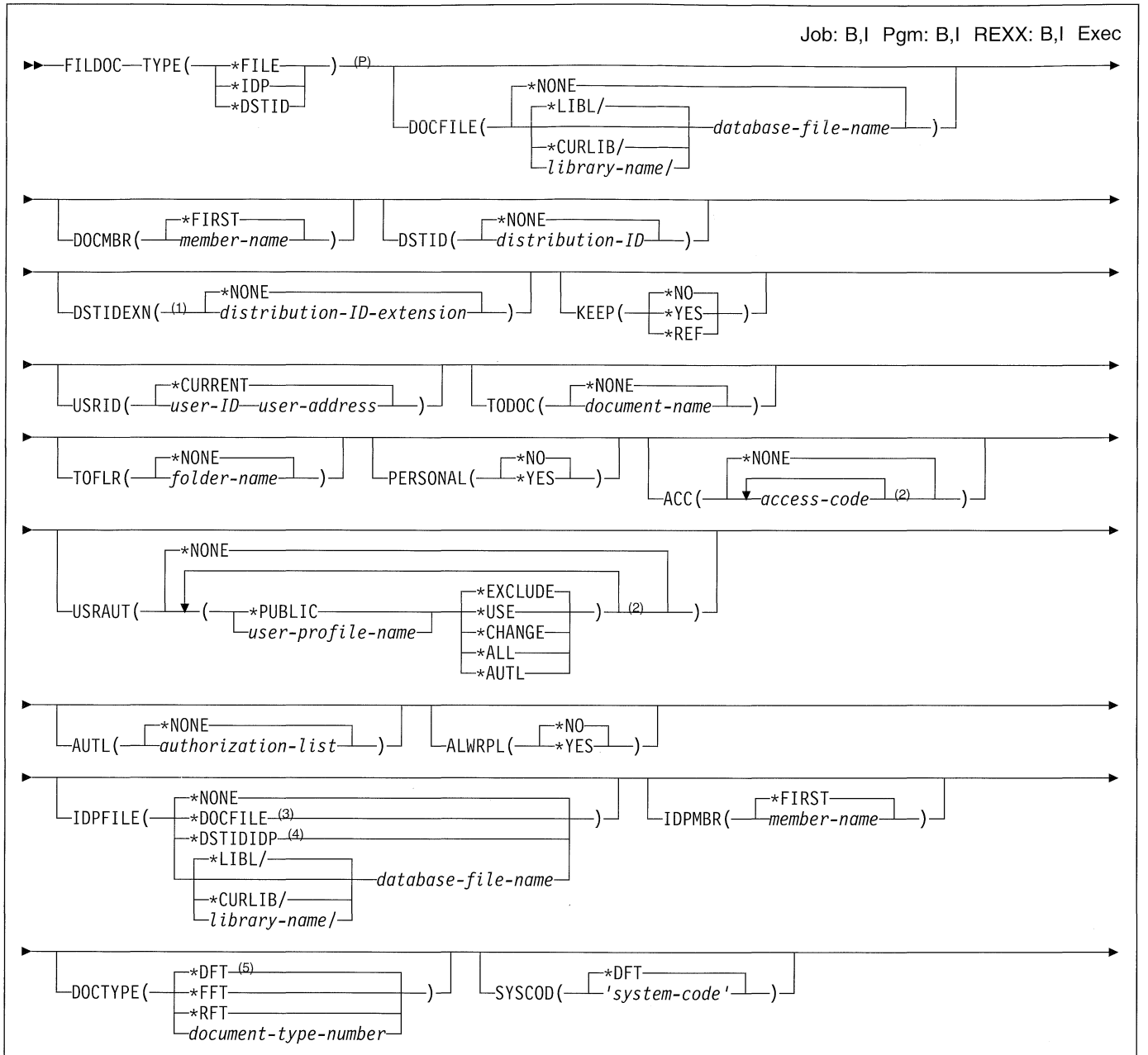
***PAGEEND:** The writer is stopped at the end of a page. This value is valid only if the spooling writer is a printer writer.

Example

```
ENDWTR  WTR(PRINTER)
```

This command stops the writer named PRINTER at the end of the spooled file whose output is being printed, and then releases the device to the system.

FILDOC (File Document) Command



Notes:

1 This parameter is valid only when TYPE(*IDP) is specified.

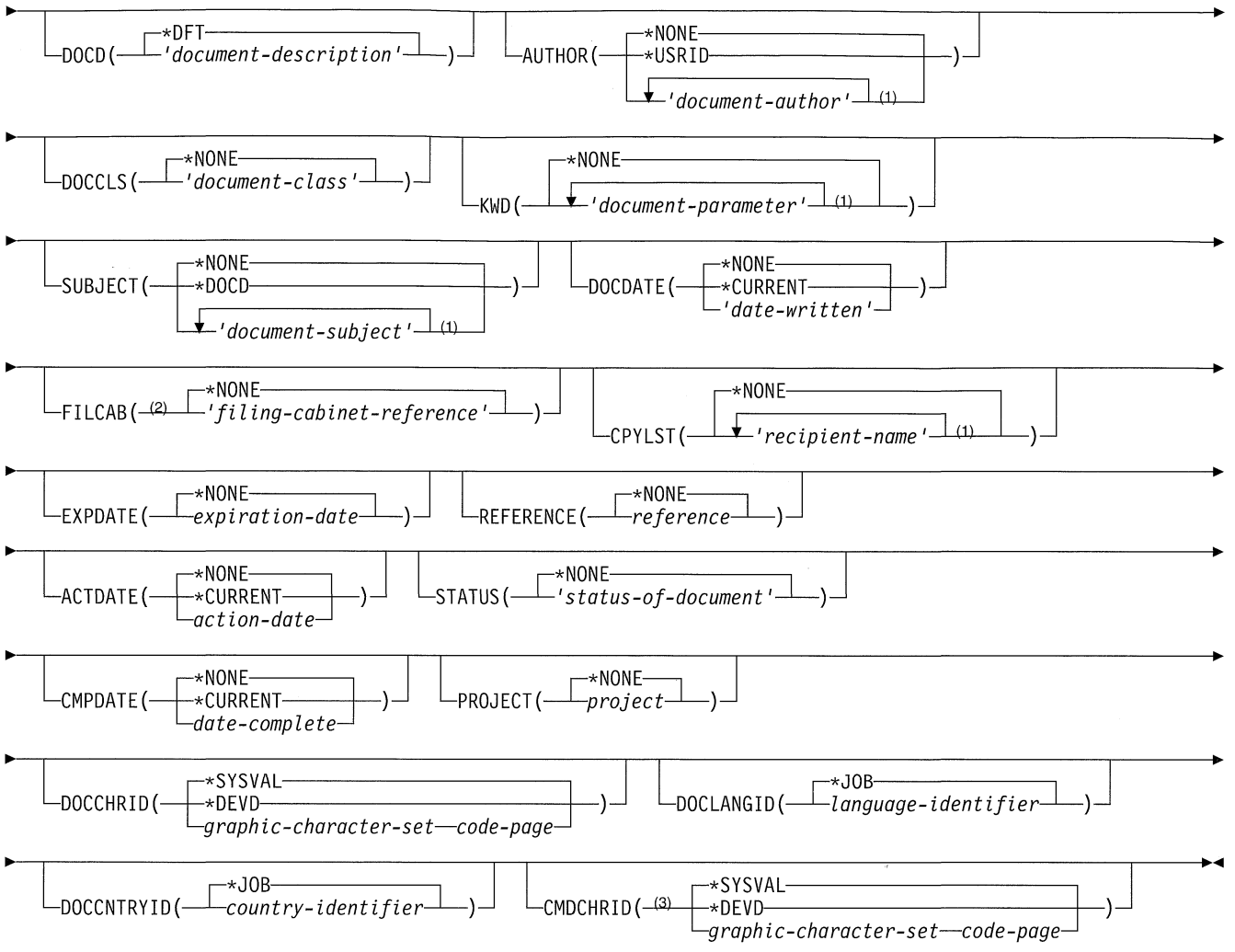
2 A maximum of 50 repetitions

3 This value is valid only when TYPE(*FILE) is specified.

4 This value is valid only when TYPE(*DSTID) is specified.

5 If DOCFILE(*database-file-name*) is specified, this value cannot be specified.

P All parameters preceding this point can be specified in positional form.



Notes:

1 A maximum of 50 repetitions

2 This parameter is required when TYPE(*IDP) and IDPFILE(*NONE) are specified.

3 This parameter applies to the following parameters: DSTID, USRID, SYSCOD, DOCD, AUTHOR, DOCCLS, KWD, SUBJECT, FILCAB, CPYLST, REFERENCE, STATUS, and PROJECT.

Purpose

The File Document (FILDOC) command is used to file information from a database file, a mail item, or command parameters to a document or as an Interchange Document Profile in the document library.

The database file from which the information is taken can be created using the Retrieve Document (RTVDOC) or Receive Distribution (RCVDST) commands. The user of these commands typically does not have OfficeVision/400 installed but wants to use portions of it to interface with other OfficeVision/400 functions.

The user also uses the FILDOC command to create a hard-copy reference (details, no content) to search the library for the document.

Restrictions:

1. You can file a document on behalf of another user if you are authorized to work on behalf of the other user. You must be granted authority to work on behalf of another with the Grant User Permission (GRTUSRPMN) command.
2. The user ID and address must be enrolled in the system distribution directory.
3. Security for the new document is taken from the parameters in the FILDOC command and not inherited from the folder.

Required Parameters

TYPE

Specifies the type of information that is filed and the parameters that are valid on this command. Dependent parameter codes for this parameter (see syntax diagram) determine which parameters are valid. If the dependent parameter code is listed in the syntax diagram, the parameter is valid when that type is specified.

***FILE:** The database file specified in the DOCFILE and the DOCMBR parameter is filed.

Note: If this value is specified, you must specify the default values on the FILCAB, DSTID, DSTIDEXT, and KEEP parameters and you cannot specify DOCFILE(*NONE).

***IDP:** The interchange document profile (IDP) specified in the IDPFILE and the IDPMBR parameter or the document profile built from the parameters DOCD, AUTHOR, DOCCLS, KWD, SUBJECT, REFERENCE, STATUS, PROJECT, ACTDATE, CMPDATE, DOCDATE, FILCAB, CPYLST, and EXPDATE is filed.

Note: If this value is specified, you must specify the default values on the DOCFILE, DOCMBR, DOCTYPE, SYSCODE, DOCCHRID, DSTID, DSTIDEXT, and KEEP parameters. If this value is specified, IDPFILE and FILCAB cannot both specify *NONE.

***DSTID:** The distribution document identified by the distribution ID specified in the DSTID parameter is filed from the mail log into the document library.

Note: If this value is specified, you cannot specify DSTID(*NONE).

Optional Parameters

DOCFILE

Specifies the qualified name of the database file that contains the document data being filed. The database file can be a user-defined file, or the file specified in the OUTFILE parameter of the Receive Distribution (RCVDST) command or the Retrieve Document (RTVDOC) command. Database files are filed as they exist without any changes. If an output file is specified, only the document data record format is read from the output file, and the prefix is removed from the document data. More information on defining the format of database files (output files) is in the *Database Guide*.

***NONE:** A database file containing document data is not specified.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that contains the document data being filed.

DOCMBR

Specifies the document database file member that is being filed.

***FIRST:** The first member (in order of creation) in the database file is filed.

member-name: Specify the name of the database file being filed.

DSTID

Specifies the unique distribution ID of the distribution. The ID is assigned to the distribution by the system that originates it. Distribution IDs can be found by using the Query Distribution (QRYDST) command. IDs are also returned from the Send Distribution (SNDDST) command.

***NONE:** No distribution ID is used.

distribution-ID: Specify a distribution ID composed of the second part of the sender's user ID (padded on the right with blanks to 8 characters), the first part of the sender's user ID (padded on the right with blanks to 8 characters), and a 4-digit zoned sequence number with the leading zeros. For example, *NEWYORK SMITH 0204*. The distribution ID is entered this way because a blank character is valid with a user ID address.

The status of the distribution is changed to FILED if KEEP(*REF) is specified and the mail log entry is changed to OPENED if KEEP(*YES) is specified.

DSTIDEXN

Specifies the extension of the distribution ID specified on the DSTID parameter. This extension uniquely identifies duplicate distributions. This extension is a 2-digit extension that ranges from 01 through 99. For example, if the distribution ID is *NEWYORK SMITH 0204* and two copies of this distribution are sent to a user, then the user has two distributions with the same distribution ID. To uniquely distinguish the two distributions, an extension is added to each distribution. One distribution is identified by distribution ID and extension as *NEWYORK SMITH 020401* and the other by *NEWYORK SMITH 020402*. If there are no duplicates, the extension defaults to 01. This extension will map one to one with the distribution identifier specified on the DSTID parameter.

***NONE:** There is no duplicate distribution. This value is equivalent to an extension of 01.

distribution-ID-extension: Specify the extension that is associated with the distribution. This is used to uniquely identify duplicate distributions.

KEEP

Specifies whether to keep a copy of the distribution document by filing it from the mail log, to delete it from the

mail log, or to keep a reference in the mail log of the filed document.

***NO:** The distribution document is deleted from the mail log after the file is complete.

***YES:** The distribution document is filed and a copy is kept in the mail log.

***REF:** The distribution is deleted and a reference to the filed distribution document is kept in the mail log.

USRID

Specifies the user ID and address of the user for whom the request is made. The current user of this command must have the authority to work on behalf of the specified user profile. Authority to work on behalf of another user is granted with the Grant User Permission (GRTUSRPMN) command. If a user files a document on behalf of another user's user ID, the other user is the owner of the document.

***CURRENT:** The user profile under which the current job is running is used.

Element 1: User ID

user-ID: Specify the user ID of the user for whom the document is filed.

Element 2: User Address

user-address: Specify the user address of the user for whom the document is filed.

TODOC

Specifies the name of the newly filed document.

***NONE:** The newly filed document does not have a user-defined document name and is not filed in a folder. Up to 12 characters can be specified. This document name must not exist in the folder into which the document is being filed.

document-name: Specify the name of the document that is filed.

TOFLR

Specifies the name of the folder that contains the document being filed. TOFLR can be specified only when TODOC is also specified.

***NONE:** The newly filed document does not have a user-assigned name and is not filed in another folder.

folder-name: Specify the name of the folder to contain the document being filed. A folder name can consist of a series of folder names (FLR1/FLR2/and so forth) if the document is being filed in a folder that is contained in another folder. Up to 63 characters can be specified.

PERSONAL

Specifies whether the document is being filed as a personal document. Personal documents cannot be accessed by users working on behalf of another user if they are not authorized to the document.

***NO:** The document is not filed as personal.

***YES:** The document is filed as personal.

ACC

Specifies the access codes that are used with this document. Access codes are specified to allow groups of users access to the document. Access codes must be added on the system with the Add Access Code (ADDACC) command.

***NONE:** No access codes are assigned to this document. Authority for this document is controlled by the values specified in the AUTUSR and AUTL parameters.

access-code: Specify the access codes, ranging from 1 through 2047, that control who can use the document. Up to 50 access codes can be specified. Access code 0 gives *USE authority to all users. Other access codes specified must exist on the system.

USRAUT

Specifies the users that can access the document and the authority each user has. This parameter is used to change the effect of the authorization list for this document by giving more users authority to the document, removing a user's authority for the document, or changing the user's authority.

***NONE:** No users have specific authority to access the document.

Element 1: Users Affected by Authority Level Change

***PUBLIC:** The users who do not have specific authority to the document, who are not on the authorization list, and whose users profile group does not have any specific authority to the document are given authority.

user-profile-name: Specify the user profile names of one or more users that are being granted authority.

Element 2: User Authority Levels

***EXCLUDE:** The user cannot access the file document.

***USE:** The user can perform basic operations on the file document, such as running a program or reading a file. The user cannot change the file document. *USE authority provides object operational authority and read authority.

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** All authority allows the user to perform all operations on the document except those limited to the owner. In addition to the functions allowed with use (*USE) and change (*CHANGE) authorities, the user can control the document's existence and specify the security for the document. The user cannot transfer ownership of the document.

***AUTL:** The authority of the authorization list specified in the AUTL parameter is used for the document. AUTL is valid only if *PUBLIC is also specified.

AUTL

Specifies that the authority for the document named in the TODOC parameter comes from the authorization list.

***NONE:** An authorization list is not specified.

authorization-list: Specify the name of the authorization list whose authority is used for the document.

ALWRPL

Specifies the setting to allow replacement of the document content of the document being filed. If this parameter is specified when filing a document that cannot be replaced, it is ignored. A document that cannot be replaced cannot be changed back to a document that can be replaced. The current user of this command must have *ALL authority to the document to request this change.

***NO:** The document content of the document being filed cannot be replaced.

***YES:** The document content can be replaced.

IDPFILE

Specifies where the document profile information is located.

***NONE:** The document profile is supplied by other parameters on this command. There is no database file containing the document profile information. The IDPMBR parameter is ignored if *NONE is specified.

***DOCFILE:** The database file specified for the document also contains the profile information. The DOCMBR parameter is used for the IDPMBR if *DOCFILE is specified and the IDPMBR parameter is ignored.

***DSTIDIDP:** The IDP information associated with the distribution document is used. If the IDPMBR parameter is specified, this value is ignored.

The name of the database file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the database file that contains the IDP information. The document profile database file can be a user-defined file or the output file specified on the Receive Distribution (RCVDST) or Retrieve Document (RTVDOC) commands. If an output file is specified, only the document profile record format is read from the output file and the prefix is removed from the data. If the user specifies this parameter, the remaining parameters after IDPMBR parameter are ignored except for the DOCCHRID and CMDCHRID parameters. More information on defining the format of database files (output files) is in the *Database Guide*.

IDPMBR

Specifies the interchange document file member name that is filed. This parameter is used only when IDPFILE (database-file-name) is also specified.

***FIRST:** The first member (in order of creation) in the database file is filed.

member-name: Specify the name of the database file member being filed.

DOCTYPE

Specifies the type of document being filed. This identifier is used by the system to determine whether it can handle the data stream correctly.

***DFT:** The system creates the appropriate document type identifier depending on where the data comes from.

Document Type	Description
2	Final form text document (FFTDCA)
3	5520 revisable form text document (RFT5520)
4	Word processing EBCDIC (EBCDIC)
5	Word processing information file (WRDPRCINFF)
6	Image-data subset document (IMAGE)
7	3730 text data stream (TEXT3730)
8	DIA document library descriptor document (DOCDESCDOC)
9	3732 display document data stream (DATAST3732)
10	DIA display document data stream (DOCUNITCTL)
11	Revisable form text document (RFTDCA)
12	1403 compatible data stream with variable length, unblocked records (PRT1403)
13	Digitized ADS audio (AUDIO)
14	IBM PC data file (PCFILE)
15	Hard copy of document (HARDCOPY)
223	Data file
32753	List of documents created from search (DOCLIST)
32756	Version 2 DIA document, library document, descriptor document
32768	AS/400 revisable-form document (RFTAS400)
32769	AS/400 final-form document (FFTAS400)
32770	IBM DW4 revisable-form document (RFTDW4)
32912	ASCII data (ASCIIDATA)
65262	IBM S/38 revisable document (RFTS38)

If TYPE(*IDP) is specified, the document type is 15 (printed document).

If TYPE(*FILE) is specified and the file is an output file created by the RTVDOC command, the document is taken from this file. If this file does not contain a document type or if the value in this file for the document type is 0, then an error message is sent. If the file is a user-defined file, the document type is 223 (data file).

If TYPE(*DSTID) is specified, the document type is the same document type associated with the distribution.

***FFT:** The document is final form text. This type of document is intended to be shown and printed and not edited by the receiver.

***RFT:** The document is revisable form text. This type of document can be viewed, printed, and edited by the receiver.

document-type-number: Specify a number from 2 through 65,535. The numbers from 2 through 32,767 are controlled by registering them with the IBM Document Interchange Architecture and are used for IBM-defined document types. The numbers 32,768 through 65,535 are not registered with IBM and can be used for non-IBM-defined document types. The meaning of these document types must be determined with the value of the SYSCOD parameter.

SYSCOD

Specifies the text that can be used with the DOCTYPE parameter to help uniquely identify that type of document being filed. The receiver of the data stream can determine the document data stream and processing requirements to edit, view, print, or change the document.

***DFT:** The system supplies a default system code. If the DOCTYPE value is between 1 and 32,767, the default `IBM AS/400 CL` is retrieved from message CPX9026. The message of this text can be changed to an installation-defined default but cannot exceed 13 characters. If the DOCTYPE is between 32,768 and 65,535, a system code must be specified.

'system-code': Specify the text that helps uniquely identify the type of document being filed. Up to 13 characters can be specified.

DOCD

Specifies a description for the document being filed. This is the Document Interchange Architecture document name field.

***DFT:** The system creates a document description for the database files. The default is (library-name/file-name/member-name) for database files. If TYPE(*IDP) is specified to file only a reference to a printed document, the default document name is Hardcopy Document Reference and is retrieved from the message CPX9025. An installation may change this message but only the first 44 characters are used in the document name. If TYPE(*DSTID) is specified, the document description specified when the distribution is used.

'document-description': Specify the description of the document. Up to 44 characters can be specified.

AUTHOR

Specifies the authors of the document.

***NONE:** No author is identified for the document.

***USRID:** The user ID and address of the user whose document is being done is used as the author's name. This can be the current user or the user ID and address specified in the USRID parameter.

'document-author': Specify the names of the authors. Up to 50 authors can be specified and each author name can have up to 20 characters.

DOCCLS

Specifies the class associated with this document, such as memo, form, or sheet.

***NONE:** No class is assigned to the document.

'document-class': Specify the document class. Up to 16 characters can be specified.

KWD

Specifies the parameters that can be used to describe the document.

***NONE:** No parameters are defined for this document.

'document-parameter': Specify the parameters used to describe the document. Up to 50 parameters can be specified and each parameter can have up to 60 characters.

SUBJECT

Specifies the subject of the document.

***NONE:** No subject is defined for the document.

***DOCD:** The document description is used as the subject for the document.

'document-subject': Specifies the subject of the document. Up to 50 subjects can be specified and each subject can have up to 60 characters of text.

DOCDATE

Specifies any date the user wants to assign to the document.

***NONE:** No date is assigned to the document.

***CURRENT:** The current date is used.

date-written: Specify the value that is used as the document date. The date must be in the format specified by the system value QDATFMT. A description of the possible date formats is in the *CL Programmer's Guide*.

FILCAB

Specifies the location where the document is stored. This parameter is intended for printed documents. All that is filed is the interchange document profile that refers to the printed document. This parameter is required if TYPE(*IDP) and IDPFILE(*NONE) is specified.

***NONE:** No filing cabinet reference is defined for this document.

'filing-cabinet-reference': Specify text that describes where the printed document is located. Up to 60 characters can be specified.

CPYLST

Specifies the names of the users who receive this document.

***NONE:** No copy list is included for this document.

'recipient-name': Specify the names of the users who receive the document. Up to 50 names can be specified and each name can have up to 60 characters.

EXPDATE
Specifies the date on which the document is no longer needed.

***NONE:** No expiration date is specified.

expiration-date: Specify an expiration date. The date must be entered in the format specified by the system value QDATFMT.

REFERENCE
Specifies a reference associated with the document.

***NONE:** No reference field is included in the interchange document profile.

reference: Specify the text that describes the reference associated with the document. Up to 60 characters can be used.

ACTDATE
Specifies the date when the requested action is due.

***NONE:** No due date is specified.

***CURRENT:** The current date is used.

action-date: Specify the value that is used as the action due date. The date must be entered in the format specified by the system value QDATFMT.

STATUS
Specifies the user-defined status (In Process, Pending Approval, or Retired).

***NONE:** No status is included in the interchange document profile.

'status-of-document': Specify text that describes the status of the document. Up to 20 characters can be specified.

CMPDATE
Specifies the date when the requested action is completed.

***NONE:** No completion date is included.

***CURRENT:** The current date is used.

date-complete: Specify the date when the action is completed. The date must be entered in the format specified by the system value QDATFMT.

PROJECT
Specifies the project with which the document is associated.

***NONE:** No project field is included in the interchange document profile.

project: Specify the text that describes the project of the document. Up to 10 characters can be specified.

DOCCHRID
Specifies the character identifier (graphic character set and code page) for the data being filed. The character identifier is related to the display device that was used to create the data being filed. More information about character identifier processing is in the *Database Guide*.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the data being filed.

Element 2: Code Page

code-page: Specify the code page value used to create the command parameters. Valid values range from 1 through 999.

DOCLANGID
Specifies the language identifier being placed in this document's interchange document profile (IDP). This parameter is ignored if the IDPFILE parameter is specified.

***JOB:** The language identifier specified for the job in which this command is entered is used.

language-identifier: Specify the language identifier. More information on valid language identifiers is in the *National Language Support Planning Guide*.

DOCCNTRYID
Specifies the country identifier being placed in this document's interchange document profile (IDP). This parameter is ignored if the IDPFILE parameter is specified.

***JOB:** The country identifier specified for the job in which this command is entered is used.

country-identifier: Specify an ISO 3166 Alpha-2 code from the country code table. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

CMDCHRID
Specifies the character identifier (graphic character set and code page) for data being specified as parameter values on this command. This character identifier (CHRID) is related to the display device used to specify the command. More information about CHRID processing is in the *Guide to Programming Displays*.

Note: The CMDCHRID parameter applies to the following parameters and means that the character set and code page are stored with the fields to allow the display station that accesses the document to correctly print or display the fields. The fields are translated to a common character set

and code page when the fields are written to the search database. This value translates the following parameters to character set and code page of '697 500'. USRID and DSTID parameter values are in character set 930 and code page 500.

The following fields are translated:

USRID
DSTID
SYSCOD
DOCD
AUTHOR
DOCCLS
KWD
SUBJECT
FILCAB
CPYLST
REFERENCE
STATUS
PROJECT

This value translates the USRID parameter to character set and code page of '930 500'. The *Communications: Distribution Services Network Guide* contains the character set and code page table for '930 500'.

***SYSVAL:** The system determines the graphic character set and code page values for the command parameters from the QCHRID system values.

***DEVD:** The system determines the graphic character set and code page values for the command parameter from the display device description where the command is entered. This option is valid only when specified from an interactive job. If this value is specified in an interactive CL program or a batch job, an error message is sent.

Element 1: Character Set

graphic-character-set: Specify the graphic character set values used to create the command parameter.

Element 2: Code Page

code-page: Specify the code page. Valid values range from 1 through 9999.

Examples

Example 1: Filing a Personal Document

```
FILDOC  TYPE(*FILE)  DOCFILE(MARYLIB/MARYFILE)
  PERSONAL(*YES)  IDPFILE(*DOCFILE)
```

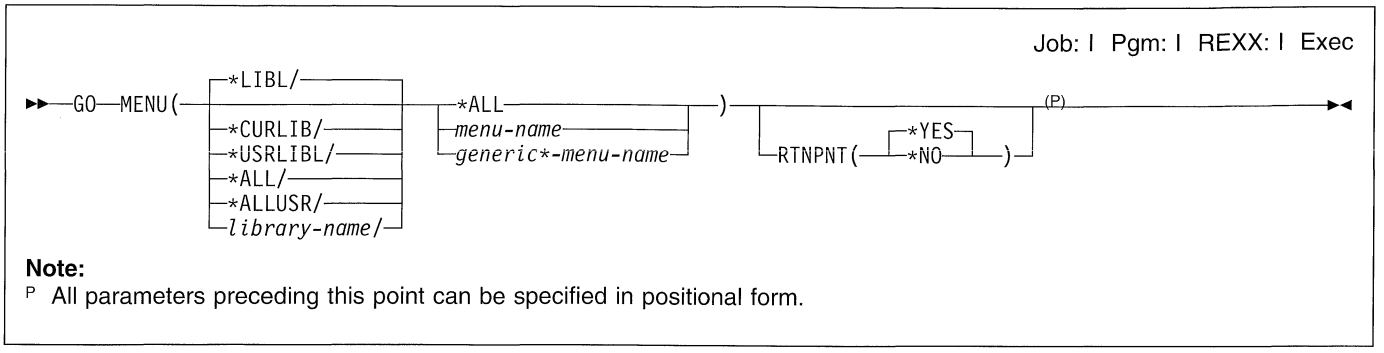
This command files a personal document using a database file that has the document content and document profile information. The default for the distribution ID extension is 01 (DSTID(01)).

Example 2: Filing a Distribution Document

```
FILDOC  TYPE(*DSTID)  DSTID('NEWYORK SMITH 0201')
  DSTID(02)  DOCCLS('NEW CLASS') TODOC(DST0201)
  TOFLR(FLRDST)
```

This command files a distribution document in the document library QDOC in document DST0201 and folder FLRDST. The document class is changed in the distribution document, and the second distribution that was sent to the user is filed.

GO (Go to Menu) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Go to Menu (GO) command allows the user to specify either a particular menu or a generic menu name. The user may optionally specify whether or not to return to the menu or display from which the command is entered after running the menu specified by the GO command.

Using the Previous and Exit Keys

A menu is placed on an internal menu stack before it is run. If a stack is not available for the menu, one is created. When the Cancel key is pressed for a menu, the previous menu in the stack is shown. Each menu stack is ten elements (menus) deep. When the eleventh menu is placed on the menu stack, the first, or oldest, menu is removed from the stack. This menu cannot be returned to by using the Cancel key.

Pressing the Exit key returns the user to the last display or menu from which a GO command was entered with RTNPNT(*YES). (Note that *YES is the default value for the optional RTNPNT parameter.) The display that the user is returned to is found by removing menus from the menu stack until a return point is found. This process may also cause a program in the call stack to return to its calling program unless the program is a return point.

Pressing either the Exit or Cancel key while viewing help for a menu returns the user to the menu.

Restrictions:

1. The user must have *USE authority for the menu and its display and message files or program (whichever applies) to use this command.
2. The user must also have *USE authority for the library where the menu is located.

Required Parameters

MENU

Specifies the qualified name of the menu being run.

The name of the menu can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries in the system portion of the job's library list, including QSYS, are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** A list of all menus in the given library are presented to allow the selection of the menu to be run.

menu-name: Specify the name of the menu being run.

generic-menu-name:* Specify the generic name of the menu. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can

be specified only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

Optional Parameters

RTNPNT

Specifies whether the display where the command is entered is returned to when the Exit key is pressed.

***YES:** The display where the command is entered is shown when the Exit key is pressed.

***NO:** The display where the command is entered is not shown when the Exit key is pressed.

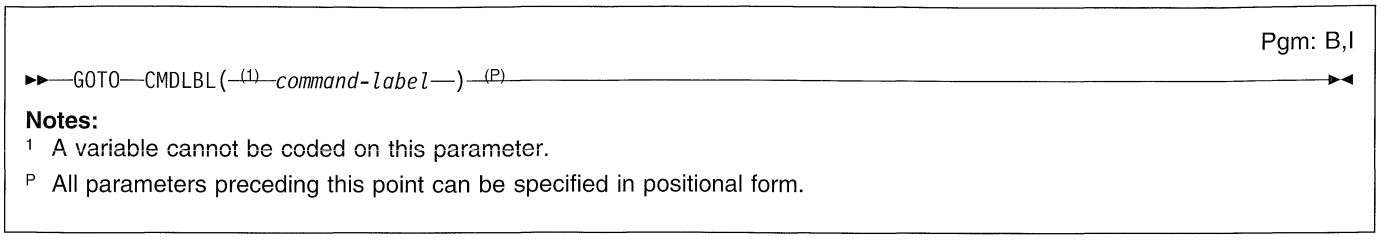
Example

```
GO    MENU(PERSMENU)
```

This command runs a menu called PERSMENU, located in a library found by searching the library list (*LIBL default value).

If the Exit key is pressed while PERSMENU is being shown, the display where the GO command was entered (*YES default value on the RTNPNT parameter) is shown.

GOTO (Go To) Command



Purpose

The Go To (GOTO) command is used in CL programs for branching from one part of the program to another. The branching is to the label on another command that is specified on the GOTO command. Branching can be either forward or backward, but the specified label must be inside the program.

Restriction: This command is valid only within a CL program.

Required Parameters

CMDLBL

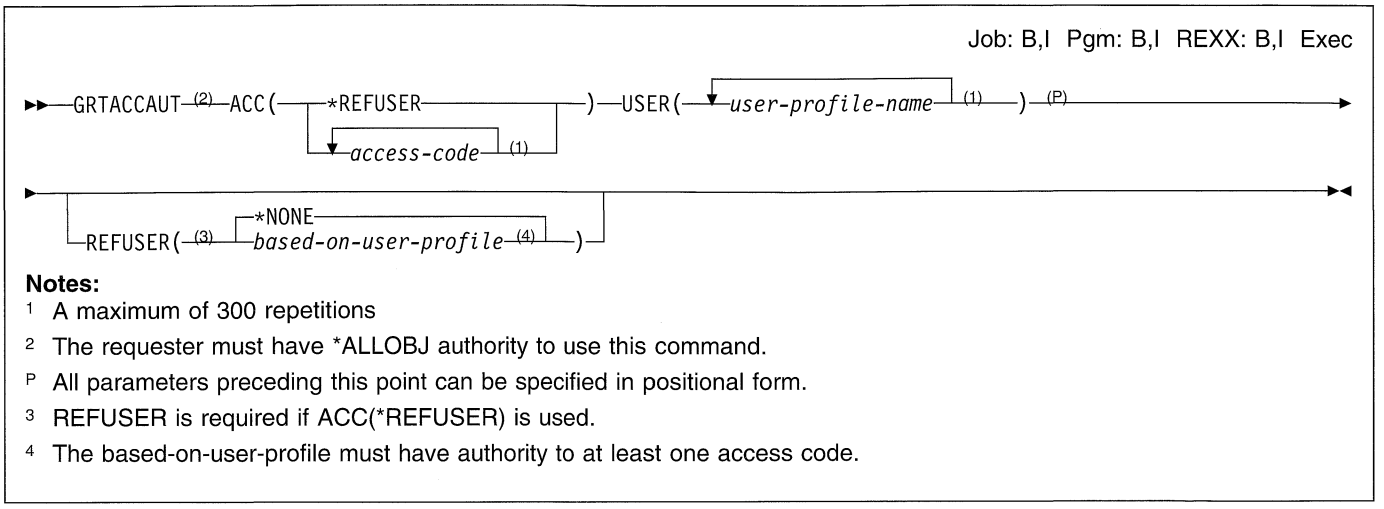
Specifies the label name of the command to which control is transferred when the GOTO command is processed. The command with the label is then processed. If the specified command cannot be run (for example, if it is a DCL command), control is transferred to the next command following the command with the specified label. The label must be within the same program as the GOTO command. A CL variable name cannot be used to specify the label name.

Example

```
LOOP:  CHGVAR &A (&A + 1)
  IF  (&A *LT 30)  THEN(GOTO  LOOP)
```

The Change Variable (CHGVAR) command increases the value of &A by 1 until &A is equal to or greater than 30. The GOTO command is processed each time that the IF command tests the expression and the result is true; the GOTO command following the THEN parameter causes the program to branch back to the label LOOP on the CHGVAR command. Refer to the descriptions of the CHGVAR command and the IF command for additional explanations of their functions.

GRTACCAUT (Grant Access Code Authority) Command



Notes:
1. A maximum of 300 repetitions
2. The requester must have *ALLOBJ authority to use this command.
P. All parameters preceding this point can be specified in positional form.
3. REFUSER is required if ACC(*REFUSER) is used.
4. The based-on-user-profile must have authority to at least one access code.

Purpose

The Grant Access Code Authority (GRTACCAUT) command gives the specified users authority to access documents and folders associated with the access codes. Access is restricted to read only (*USE authority).

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. The access code must be defined to the system (by using the Add Access Code (ADDACC) command) before the user can give access code authority.
3. The user being given access code authority must be enrolled in the system distribution directory.

Note: When a new user profile is created based on an existing user profile, the access codes (if any) associated with the user do not carry over to the new user. The GRTACCAUT command must be used for the new user profile; however, the REFUSER parameter can be specified on this command to get all the codes of the user who is being based on.

Authority for this command is originally restricted to the security officer. Authority to use this command can be granted to other users with the Grant Object Authority (GRTOBJAUT) command.

Required Parameters

ACC

Specifies the access code that is being authorized for use by the user identified on the USER parameter.

***REFUSER:** The access code authority granted is based on a second user profile name; that name must be specified on the REFUSER parameter.

access-code: Specify a number, ranging from 1 through 2047, that specifies the access code of the documents and folders for which the user wants authority to be granted. The access code must be defined to the system using the ADDACC command before being specified in this parameter. Up to 300 access codes can be specified.

USER

Specifies the user profile name of the users to whom access code authority is granted. The user identified has the access code added to the list of access codes to which the user is authorized; this access code is used to verify additional document and folder accesses from the document library. The user must be enrolled in the system distribution directory before being granted authority to use an access code. Specify the name of the user profile. Up to 300 can be specified.

Optional Parameters

REFUSER

Specifies the user referred to, which is the user profile on which to base the access code authority. If this parameter is used, *REFUSER must be specified on the ACC parameter. The *REFUSER's access code authority is added to the existing access code authority specified on the USER parameter. All access code authority of the *REFUSER is copied except for the access code authority the *REFUSER's group profile may have.

***NONE:** No user is referred to.

based-on-user-profile: Specify the user-profile name on which the access code authority is based. This user must also be enrolled in the system distribution directory.

Examples

Example 1: Granting Authority to Multiple Users

```
GRTACCAUT  ACC(3 30 60)  USER(SAM LARRY)
```

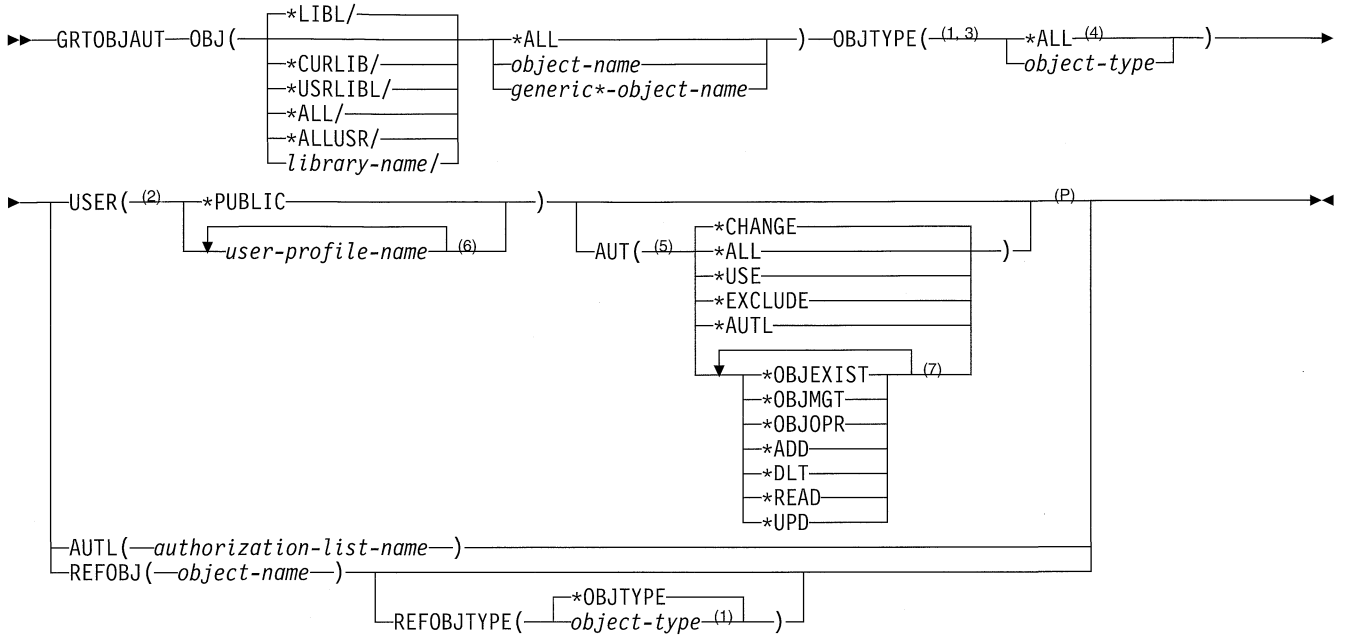
This command gives authority to access codes 3, 30, and 60 to SAM and LARRY.

Example 2: Granting Authority Based on Another User

```
GRTACCAUT  ACC(*REFUSER)  USER(JOE)  REFUSER(TOM)
```

This command grants access code authority to JOE based on TOM's authority. For example, if JOE currently has authority to access codes 1, 12, and 50, and TOM currently has authority to access codes 8 and 9, the GRTACCAUT command authorizes JOE to access codes 1, 8, 9, 12, and 50.

GRTOBJAUT (Grant Object Authority) Command



Notes:

1 A list of the valid OS/400 object types for this command is in Appendix A.

2 Positional order is USER, AUT, AUTL, REFOBJ, REFOBJTYPE. Only one of USER, AUTL, or REFOBJ can be specified.

3 OBJTYPE(*ALL) is not valid if REFOBJTYPE(*OBJTYPE) is specified.

4 OBJTYPE(*ALL) is not valid if one of the following library names is specified: *LIBL, *USRLIBL, *ALL, *ALLUSR.

5 AUT(*AUTL) is valid only if USER(*PUBLIC) is specified. It is not valid if the object type is *USRPRF.

6 A maximum of 50 repetitions

7 Each can be used only once, a maximum of 7

P All parameters preceding this point can be specified in positional form.

Purpose

The Grant Object Authority (GRTOBJAUT) command is used by one user to grant specific authority for the object named in this command to another user or group of users.

Authority can be given to:

- Named users
- Users (*PUBLIC) who do not have authority specifically given to them either for the object or for the authorization list
- Groups of users who do not have any authority to the object or are not on the authorization list that secures the object
- Users of the referenced object (specified in the REFOBJ parameter)
- Users on an established authorization list

The AUTL parameter specifies the name of the authorization list whose members are given authority for the object specified in the OBJ parameter.

The AUT(*AUTL) parameter allows the user to specify the authority for all users who do not have authority specifically given to them for an object, who are not on the authorization list that secures the object, whose users' group does not have authority specifically given to it, or whose users' group is not on the authorization list that secures the object. It is only allowed with USER(*PUBLIC).

The AUTL parameter is used to secure an object with an authorization list. User profiles cannot be secured by an authorization list (*AUTL). Therefore, the AUTL parameter is not valid when the object type is *USRPRF.

This command can be used by an object's owner, by the security officer, or by a user with object management authority for the specified object. A user with object management authority can grant to other users any authority that

user has, except object management authority. Only the owner of the object, or someone with all object authority (*ALLOBJ), can grant object management authority to a user.

A user with *ALL authority can assign a new authorization list.

Restrictions:

1. This command must get an exclusive lock on a database file before read or object operational authority can be given to a user.
2. If a user requests authority for another specified user to a device currently in use by another authorized user, authority to the device is not given.
3. Object type *AUTL cannot be specified.
4. AUT(*AUTL) is valid only with USER(*PUBLIC).
5. A user must either be the owner of the object or have *ALL authority to use the AUTL parameter.
6. The user must have object management authority to the object.
7. If the object is a file, the user must have object operational and object management authorities.
8. For display stations or for work station message queues associated with the display station, if this command is not entered at the device for which authorities are being granted, it should be preceded by the Allocate Object (ALCOBJ) command and followed by the Deallocate Object (DLCOBJ) command.

Required Parameters

OBJ

Specifies the qualified name of the objects for which specific authorities are given to one or more users or to an authorization list. A specific object name or a generic object name can be qualified by a library name. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

The name of the object can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***USRLIBL:** Only the libraries in the user portion of the job's library list are searched.

***ALL:** All libraries on the system are searched.

***ALLUSR:** All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

```
#CGULIB    #DFULIB    #RPGLIB    #SEULIB
#COBLIB    #DSULIB    #SDALIB
```

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered *user libraries*, and are also searched:

```
QDSNX      QPFRDATA   QUSER38
QGPL       QRCL       QUSRSYS
QGPL38     QS36F      QUSRVxRxMx
```

Note: A different library name, of the form QUSRVxRxMx, is added with each release. VxRxMx is the version, release, and modification level of the library.

library-name: Specify the name of the library to be searched.

***ALL:** All objects of the specified type (OBJTYPE) found in the search have specific authorities granted. A specific library name must be specified.

object-name: Specify the name of the object for which specific authorities are given to one or more users.

generic-object-name:* Specify the generic name of the object. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be granted only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names."

OBJTYPE

Specifies the object type of the object for which specific authorities are given to the specified users or to an authorization list. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ALL:** Specific authorities for all object types (except *AUTL) are given to the specified users or to the authorization list.

object-type: Specify the specific object type of the object for which specific authorities are given to the specified users.

USER

Specifies the user names of one or more users to whom authorities for the named object are being given. If user names are specified, the authorities are given specifically to those users. Authority given by this command can be revoked specifically by the Revoke Object Authority (RVKOBJAUT) command.

***PUBLIC:** All users of the system, who do not have authority specifically given to them for the object, who are not on the authorization list, whose user group does not have any authority, or whose user group is not on the authorization list, are authorized to use the object as specified in the AUT parameter.

user-profile-name: Specify the user names of one or more users who have specific authority for the object. Up to 50 user profile names can be specified.

AUT

Specifies the authority given to users specified on the USER parameter. Users must have *AUTLMGT authority to manage the authorization list. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***CHANGE:** The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

***ALL:** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence and specify the security for the object, change the object, and perform basic functions on the object. The user can change ownership of the object.

***USE:** The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. *USE authority provides object operational authority and read authority.

***EXCLUDE:** The user cannot access the object.

***AUTL:** The public authority of the authorization list specified in the AUTL parameter is used for the public authority for the object.

A maximum of seven of the following values can be specified:

***OBJEXIST:** Object existence authority provides the authority to control the object's existence and ownership. This authority is necessary for users who want to delete the object, free storage of the object, perform save and restore operations for the object or transfer ownership of an object. (If a user has special save system authority (*SAVSYS), object existence authority is not required.) Object existence authority is required to create an object that has been named by an authority holder.

***OBJMGT:** Object management authority provides the authority to specify the security for the object, move or rename the object, and add members to database files.

***OBJOPR:** Object operational authority provides authority to look at the description of an object and use the object as determined by the data authorities that the user has to the object.

***ADD:** Gives the authority to add entries to an object (for example, job entries to a queue or records to a file).

***DLT:** Delete authority allows the user to remove entries from an object, for example, remove messages from a message queue or records from a file.

***READ:** Read authority provides the authority needed to get the contents of an entry in an object or to run a program.

***UPD:** Update authority provides the authority needed to change the entries in an object.

AUTL

Specifies the name of the authorization list whose users are given authority for the object specified in the OBJ parameter.

REFOBJ

Specifies the name of the object being queried to obtain authorization information. Those authorizations are given to the object specified by the OBJ parameter. Users authorized to the referenced object are authorized in the same manner to the object for which authority is being given. If the referenced object is secured by an authorization list, that authorization list secures the object specified in the OBJ parameter. Specify the name of the object.

REFOBJTYPE

Specifies the object type of the referenced object (REFOBJ parameter).

***OBJTYPE:** The object type of the referenced object is the same type as the object being given authority (OBJTYPE parameter).

object-type: Specify the type of the object. Any one of the operating system object types can be specified.

Examples

Example 1: Granting Authority to All Users

```
GRTOBJAUT   OBJ(USERLIB/PROGRAM1)  OBJTYPE(*PGM)
  USER(*PUBLIC)
```

This command gives authority to use the object named PROGRAM1 to all users of the system who do not have authorities specifically given to them, who are not on an authorization list, whose user group does not have authority to the object, or whose user group is not on the authorization list. The object is a program (*PGM) located in the library named USERLIB. Because the AUT parameter is not specified, the authority given to all users is change authority. This allows all users to run the program and to debug it.

Example 2: Granting Object Management Authority

```
GRTOBJAUT   OBJ(ARLIB/PROGRAM2)  OBJTYPE(*PGM)
  USER(TMSMITH)  AUT(*OBJMGT)
```

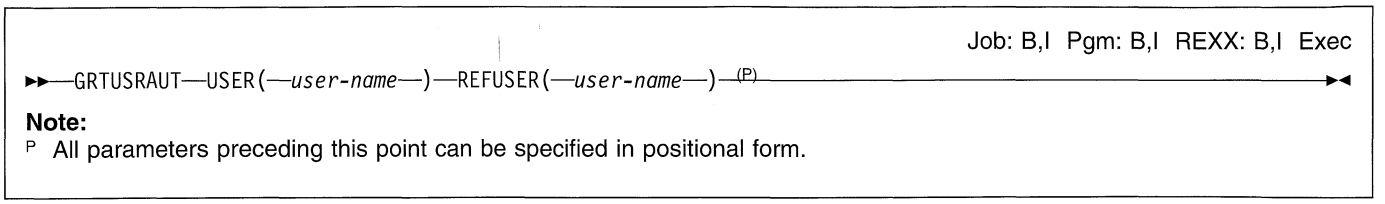
This command gives object management authority to user named TMSMITH. This authority allows TMSMITH to grant to others personally possessed authorities for the object named PROGRAM2, which is a program located in the library named ARLIB.

Example 3: Granting Authority to Users on Authorization List

```
GRTOBJAUT OBJ(MYLIB/PRGM3) OBJTYPE(*PGM)
   AUTL(KLIST)
```

This command gives to users the authority specified for them on authorization list KLIST for the object named PRGM3. The object is a program located in library MYLIB.

GRTUSRAUT (Grant User Authority) Command



Purpose

The Grant User Authority (GRTUSRAUT) command grants authority to a user by referring to another user profile. Authorization lists or group profiles should be used instead of GRTUSRAUT command support if possible. The GRTUSRAUT command can take a long time to run, and the time to run a subsequent SAVSYS or SAVSECDTA command also increases. The authority granted to the user depends on the user profile being referred to and the user using the command.

If the security officer enters this command, the user specified in the USER parameter is granted the same authority for each object as granted in the user profile being referred to, including object management authority.

If the user whose user profile is specified for the REFUSER parameter enters this command, the user specified for the USER parameter is given all authorities for each object owned by the referenced profile, including object management authority.

For objects that the user profile being referred to does not own but is authorized to use, the user of this command must have object management authority and the authorities being granted for the object, or must own the object. Otherwise, no authority is granted for that object. Object management authority is not granted unless the user of this command owns the object being authorized.

Ownership of an object or authorities held by the referred to user cannot be changed by this command. Authorities to objects granted to a user profile are added to any authorities that the user profile already has.

Restrictions: User profiles QSYS, QDBSHR, QSPL, QSNADS, QDOC, QGATE, QDFTOWN, QTSTRQS, QDSNX, or QRJE cannot be specified for either of the parameters on this command.

Required Parameters

USER
Specifies the name of the user profile to which authority is being granted.

REFUSER
Specifies the name of the user profile being referred to for authority.

Examples

Example 1

User SECOFR is entering this command:

```
GRTUSRAUT   USER(USRB) REFUSER(USRA)
```

This command grants the user profile USRB the same authorities that USRA has for all objects that USRA owns (including object management authority) or has authority to.

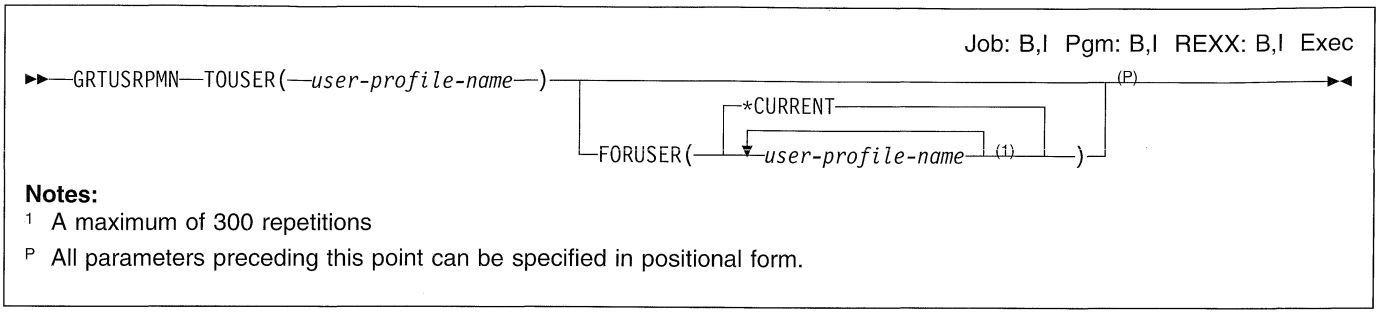
Example 2

User USRA is entering this command:

```
GRTUSRAUT   USER(USRB) REFUSER(USRC)
```

This command grants the user profile USRB the same authorities that USRC has for all objects that USRC has authorities to only if USRA, entering this command, has object management authority to the objects or is the owner of the objects being referred to.

GRTUSRPMN (Grant User Permission) Command



Notes:
1 A maximum of 300 repetitions
P All parameters preceding this point can be specified in positional form.

Purpose

The Grant User Permission (GRTUSRPMN) command gives permission to a user to handle documents and folders or to perform tasks on behalf of another user. Access is restricted to documents, folders, and mail that is not personal. For example, user A gives user B the ability to work on behalf of user A. User B is now allowed to obtain, file, delete, retrieve, list, distribute, cancel, and search a document library on behalf of user A; however, user B is not allowed to obtain any of user A's personal mail or get any of user A's private documents or folders from the library.

The users specified must be enrolled in the system distribution directory before using this command. To enroll a user, use the Work with Directory (WRKDIR) command.

Restrictions: The user must have *ALLOBJ authority to grant permission for a user to work on behalf of another user.

Additional information on giving document authority is in the *Communications: Distribution Services Network Guide*.

Required Parameters

TOUSER
Specifies the name of the user profile that is permitted to work on behalf of the user specified in the FORUSER parameter. Access is restricted to objects that are not personal. The user profile must exist at the time the command is run, and the user must be enrolled in the system distribution directory before running the command.

Optional Parameters

FORUSER
Specifies the name of the user profile that the user specified in the TOUSER parameter works on behalf of. The user must be enrolled in the system distribution directory before running the command.

***CURRENT:** The user profile under which the current job is running is used.

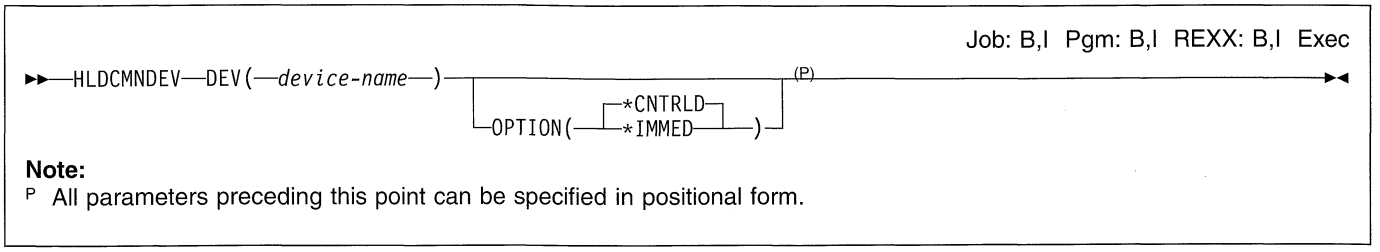
user-profile-name: Specify the name of the user profile.

Example

```
GRTUSRPMN   TOUSER(JUDY)   FORUSER(PEGGY)
```

JUDY is the administrative assistant for an executive. This command allows JUDY to work with documents or folders for PEGGY that are not personal.

HLDCMNDEV (Hold Communications Device) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Communications Device (HLDCMNDEV) command allows the operator to hold communication through the specified device description in two ways: controlled or immediate. The controlled option allows any program using the communications device to continue to do input/output operations, but no new uses of the device are started. The immediate option stops a communications device immediately, and a permanent input/output error is sent to the program. Communications are restarted with the Release Communications Device (RLSCMNDEV) command or by varying the device off and then on (Vary Configuration (VRYCFG) command).

Restriction: This command is shipped with public *EXCLUDE authority and the QPGMR, QSYSOPR, QSRV, and QSRVBAS user profiles have private authorities to use the command.

Required Parameters

DEV

Specifies the name of the device whose communications are held. Devices whose communications are held are:

DEV Value	Device
3180	Display station
3277	Display station
3278	Display station
3279	Display station
3287	Printer (work station)
5219	Printer (work station)
5224	Printer (work station)
5225	Printer (work station)
5251	Display station
5252	Display station
5256	Printer (work station)
5291	Display station
5292	Display station
PLU1	Primary logical unit, type 1 (for SNA)
BSC	Binary synchronous device (Base and RJE)
BSCT	This AS/400 system as a BSC multipoint tributary station
APPC	Logical unit in advanced program-to-program communications (APPC) network

Specify the name of the device (as specified in the device description) whose communications capability is suspended.

Optional Parameters

OPTION

Specifies the manner in which communications with this device is held.

***CNTRLD:** The specified device is not capable of communications at the next OPEN or ACQUIRE operation.

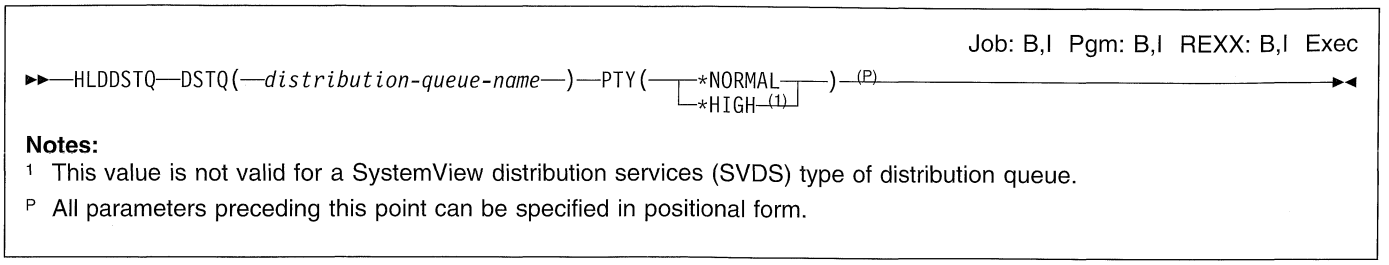
***IMMED:** The specified device is not capable of communications at the next READ, WRITE, OPEN, or ACQUIRE operation.

Example

```
HLDCMNDEV  DEV(WSPR05)
```

This command holds the communications capability of the device WSPR05 at the time of the next OPEN or ACQUIRE operation.

HLDDSTQ (Hold Distribution Queue) Command



Notes:

1 This value is not valid for a SystemView distribution services (SVDS) type of distribution queue.

P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Distribution Queue (HLDDSTQ) command prevents a distribution queue from being sent.

Distribution queue names are translated to the graphic character set and code page 930 500, using the job's coded character set identifier (CCSID).

Restrictions:

1. This command is shipped with public *EXCLUDE authority, and the QPGMR and QSYSOPR user profiles have private authorities to use the command.
2. Messages that report errors about distribution queues may display or print different characters than the user entered for the distribution queue name because of internal system transformations. Similarly (depending on the language used for the work station), the internal value for a distribution queue name may differ from the characters shown on the Work with Distribution Queue (WRKDSTQ) command. An error may be reported if the character-string value specified for the DSTQ parameter does not match the rules for an internal distribution queue value or if it does not match the internal value for any defined distribution queue (ignoring case differences).

Required Parameters

DSTQ

Specifies the name of the distribution queue being held. Both normal and high priority portions of the specified distribution queue are shown or printed. The queue specified must have been previously configured. See the Configure Distribution Services (CFGDSTSRV) command or the Add Distribution Queue (ADDDSTQ) command.

PTY

Specifies whether the normal priority or high priority portion of the specified queue is held.

***NORMAL:** Holds the normal priority queue, which is for distributions with a service level of data low.

***HIGH:** Holds the high priority queue, which is for distributions with a service level of fast, status, or data high.

Examples

Example 1: Holding the Normal Priority Portion of a Queue

```
HLDDSTQ  DSTQ(CHICAGO)  PTY(*NORMAL)
```

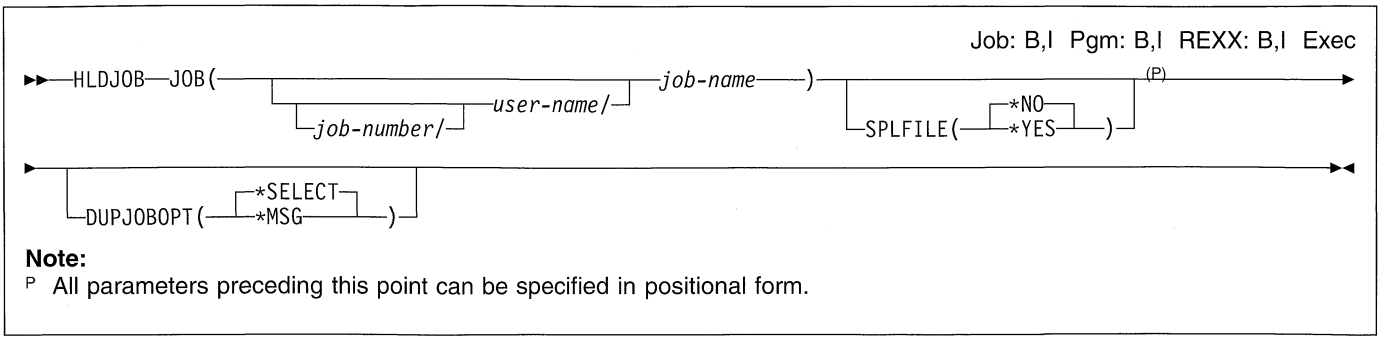
This command holds the normal priority portion of the CHICAGO distribution queue.

Example 2: Holding the High Priority Portion of a Queue

```
HLDDSTQ  DSTQ(ATLANTA)  PTY(*HIGH)
```

This command holds the high priority portion of the ATLANTA distribution queue.

HLDJOB (Hold Job) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Job (HLDJOB) command makes a job ineligible for processing by the system. The job is held until it is:

- Released by the Release Job (RLSJOB) command
- Cleared by the Clear Job Queue (CLRJOBQ) command
- Ended by the End Job (ENDJOB) command
- Ended (while the job is active) by the End Subsystem (ENDSBS) command, the End System (ENDSYS) command, or the Power Down System (PWRDWNSYS) command

If the job is not run before the OS/400 system is ended, the queue can be cleared (and the job ended) when the OS/400 system is started again. The specified job to be held can be on the job queue or on the output queues, or it can be active in a subsystem. This command also specifies whether the job's spooled files are held.

Note: If you use this command to hold a job that has exclusive access to any resources on the system, these resources are not available to other jobs. Other jobs which require access to those resources will either fail or wait indefinitely.

Restriction: The job must belong to the user issuing the command, or the user must have job control authority (*JOBCTL).

Required Parameter

JOB

Specifies the qualified name of the job being held. If no job qualifier is given, all of the jobs currently in the system are searched for the job name. If more than one of the specified names are found, a qualified job name must be specified.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
job-name
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

job-name: Specify the name of the job being held.

user-name: Specify the name of the user of the job being held.

job-number: Specify the number of the job being held.

Optional Parameters

SPLFILE

Specifies whether spooled files created by the job being held are also held.

***NO:** The spooled files produced by the job are not held.

***YES:** The spooled files produced by the job are also held.

DUPJOBOPT

Specifies the action taken when duplicate jobs are found by this command.

***SELECT:** The selection display is shown when duplicate jobs are found during an interactive session. Otherwise, a message is issued.

***MSG:** A message is issued when duplicate jobs are found.

Examples

Example 1: Making a Job Ineligible for Processing

```
HLDJOB  JOB(PAYROLL)  SPLFILE(*YES)
```

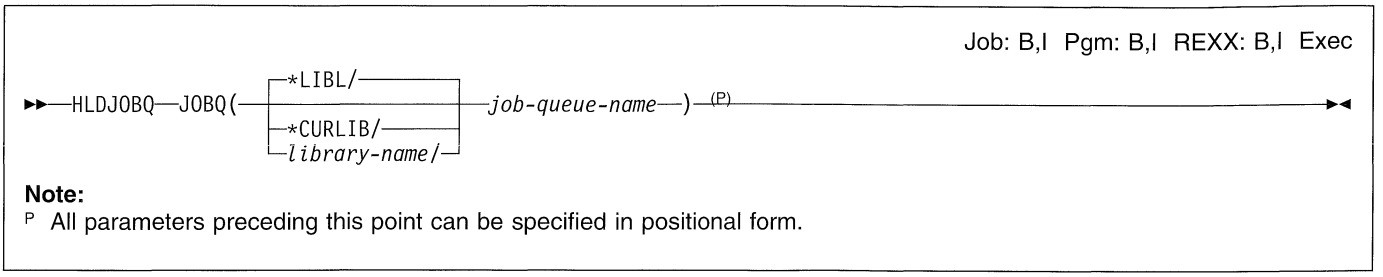
This command makes the job named PAYROLL ineligible for processing. All spooled files for this job are also held.

Example 2: Holding a Job that has a Duplicate Name

```
HLDJOB  JOB(DEPTXYZ/PAYROLL)
```

This command holds the job named PAYROLL submitted by a user operating under the user profile DEPTXYZ. The qualified form of the job name is used when jobs with duplicate names exist in the system. Spooled files are not held.

HLDJOBQ (Hold Job Queue) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Job Queue (HLDJOBQ) command prevents the processing of all jobs currently waiting on the job queue and all entries that are added to the queue after the command is issued. The HLDJOBQ command has no effect on jobs that are running. Additional jobs can be placed on the job queue while it is being held, but they are not processed. The jobs are held until a Release Job Queue (RLSJOBQ) command is issued. When a job queue is being held, the jobs can be cleared with the Clear Job Queue (CLRJOBQ) command or a specific job can be canceled by the ENDJOB command.

Restriction: The QLPINSTALL job queue cannot be held.

Required Parameters

JOBQ

Specifies the qualified name of the job queue that has its current and future jobs withheld from further processing.

The name of the job queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

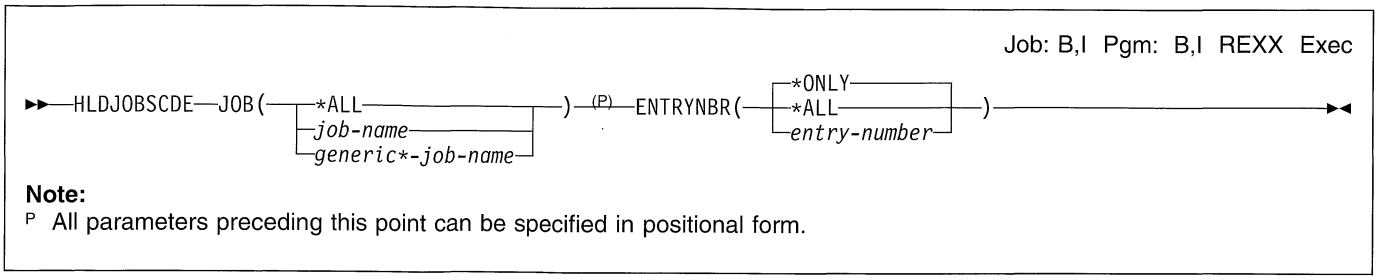
job-queue-name: Specify the name of the job queue being held

Example

```
HLDJOBQ  JOBQ(QBATCH)
```

This command prevents the processing of the jobs currently on the QBATCH job queue and any jobs added to the queue. They are held until the queue is released or cleared. Individual jobs can also be ended with the ENDJOB command, which removes the job from the job queue.

HLDJOBSCDE (Hold Job Schedule Entry) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Job Schedule Entry (HLDJOBSCDE) command allows you to hold an entry, entries, or generic entries in the job schedule. Each job schedule entry contains the information needed to automatically submit a batch job one time, or at regularly scheduled intervals.

If you hold a job schedule entry:

- The entry is held until it is released using the Release Job Schedule Entry (RLSJOBSCDE) or Work with Job Schedule Entries (WRKJOBSCDE) command.
- A job is not submitted when the entry is released, even if the date and time at which it was scheduled to be submitted passed while the entry was held.

Restriction: To hold entries, you must have *JOBCTL special authority; otherwise you can hold only those entries that you added.

Required Parameters

JOB

Specifies the name of the job schedule entry.

***ALL:** All of the job schedule entries for which you have authority are held. If JOB(*ALL) is specified, ENTRYNBR(*ALL) must also be specified.

job-name: Specify the name of the job schedule entry.

generic-job-name:* Specify a generic name. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified, and multiple libraries are searched, multiple objects can be held only if *ALL or *ALLUSR library values can be specified for the name. For more information on the use of generic functions, refer to "Rules for Specifying Names." If a generic name is specified, ENTRYNBR(*ALL) must also be specified.

Optional Parameters

ENTRYNBR

Specifies the number of the job schedule entry you want to hold. The message sent when an entry is successfully added contains the entry number. You can also determine the entry number by using the Work with Job Schedule Entries (WRKJOBSCDE) command. Press F11 from the Work with Job Schedule Entries display to show the entry numbers of the selected entries.

***ONLY:** One entry in the job schedule has the job name specified on the JOB parameter. If *ONLY is specified and more than one entry has the specified job name, no entries are held and a message is sent.

***ALL:** All entries with the specified job name are held.

entry-number: Specify the number of the job schedule entry you want to hold.

Examples

Example 1: Holding a Job Schedule Entry

```
HLDJOBSCDE   JOB(CLEANUP)
```

This command holds the job schedule entry with the job name CLEANUP.

Example 2: Holding All Job Schedule Entries

```
HLDJOBSCDE   JOB(*ALL)  ENTRYNBR(*ALL)
```

This command holds all entries in the job schedule.

Example 3: Holding an Individual Job Schedule Entry

```
HLDJOBSCDE   JOB(PAYROLL)  ENTRYNBR(*ONLY)
```

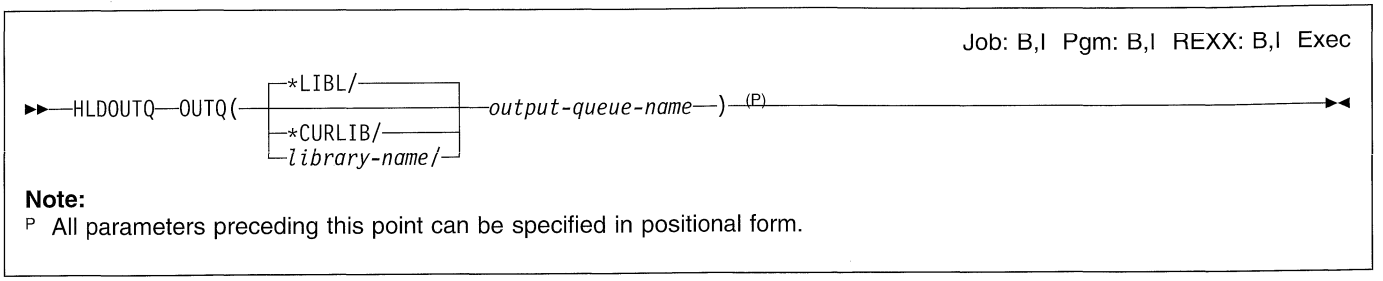
This command holds the entry PAYROLL in the job schedule.

Example 4: Holding a Generic Job Schedule Entry

```
HLDJOBSCDE   JOB(PAY*)  ENTRYNBR(*ALL)
```

This command holds all entries in the job schedule that have the prefix PAY in their names.

HLDOUTQ (Hold Output Queue) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Output Queue (HLDOUTQ) command prevents all currently waiting spooled files, and all spooled files that are added to the output queue after the command is issued, from being processed by a spooling writer. The HLDOUTQ command has no effect on jobs that are running and adding spooled files to the queue or on the spooled file being produced by a spooling writer at the time the command is issued. When the spooling writer completes all copies of the current output file, it cannot begin the output for any other files until the queue is released.

The spooled files are held until a Release Output Queue (RLSOUTQ) command is issued. Otherwise, the output queue can be cleared with the Clear Output Queue (CLROUTQ) command or a specific file can be deleted by the DLTSPLF command.

Required Parameters

OUTQ

Specifies the qualified name of the output queue.

| The name of the output queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

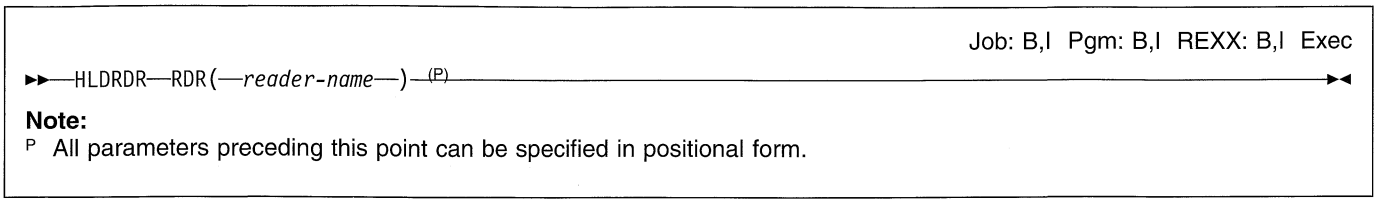
output-queue-name: Specify the name of the output queue.

Example

```
HLDOUTQ  OUTQ(QPRINT)
```

This command prevents the processing of the spooled files on the QPRINT queue and any spooled files added to the queue. They are held until the queue is released or cleared. A specific job (with its spooled files) can also be ended with the ENDJOB command, which removes the spooled files from the output queue.

HLDRDR (Hold Reader) Command



Purpose

The Hold Reader (HLDRDR) command immediately stops the activity of the specified spooling reader. The reader is not ended, and its associated input device is not made available to the system. The reader remains inactive until a Release Reader (RLSRDR) or End Reader (ENDRDR) command is issued. Data is not lost when the reader is held.

Required Parameters

RDR

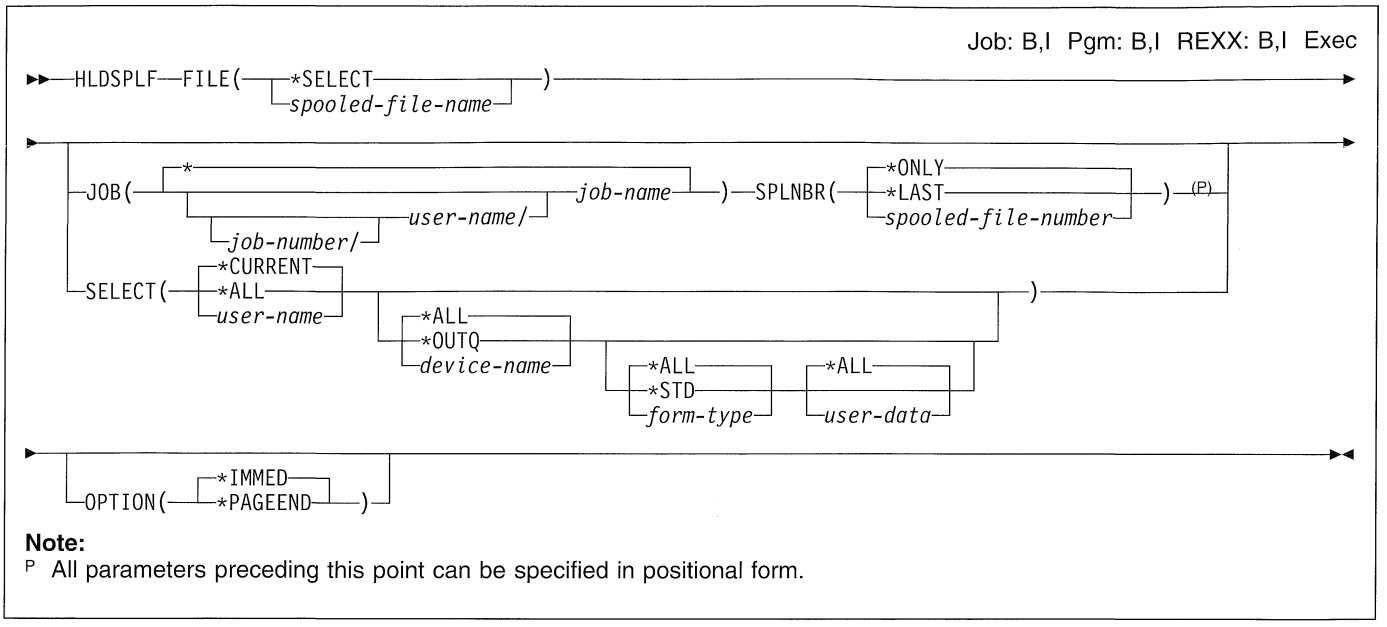
Specifies the name of the spooling reader being held.

Example

```
HLDRDR    RDR(QDKT)
```

This command causes the diskette reader QDKT to immediately stop reading data. To release the reader, so that it can continue to read data, a Release Reader (RLSRDR) command must be entered. If the End Reader (ENDRDR) command is used, the reader is stopped and the job that was being read in is lost because no job entry was added to the job queue.

HLDSPLF (Hold Spooled File) Command



Note:

P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Spooled File (HLDSPLF) command stops the specified spooled file from additional processing by a spooling writer. If the file is being produced on an output device, the writer stops processing that file and gets the next file to be processed. When the file is released and again selected for output, it is again processed starting at the beginning. If several copies are being produced for the file when it is held, the incomplete copy is again produced from the beginning along with the remaining copies that follow it.

If the specified file is still receiving records from a program that is running when the file is held, the program that is running is unaware that the file is on hold. Also, if the held file is part of a job that produces other spooled files, they can be processed before the held file is released. The held file remains on the output queue and it appears to be the only file produced by the job.

The file is held until one of the following commands is entered: the RLSSPLF (Release Spooled File), the DLTSPLF (Delete Spooled File), the ENDJOB (End Job) with keyword SPLFILE(*YES) specified, or the CLROUTQ (Clear Output Queue) command. If the file is released before the writer begins producing it, the file is produced in its normal order within the group of files for the job. Production of a job's output is not delayed because some of its files are being held.

Required Parameters

FILE

Specifies the name of the spooled file to be held.

***SELECT:** All spooled files that meet the selection requirements specified in the SELECT keyword are held. This value is mutually exclusive with the JOB and SPLNBR parameters. Specifying *SELECT causes the JOB and SPLNBR parameters to be ignored.

spooled-file-name: Specify the name of the spooled file to be held.

Optional Parameters

JOB

Specifies the name of the job that created the file being held. If no job qualifier is given, all of the jobs currently in the system are searched for the simple job name.

A job identifier is a special value or a qualified name with up to three elements. For example:

```
job-name
*
user-name/job-name
job-number/user-name/job-name
```

More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

*: The job that issued this HLDSPLF command is the job that produced the file being held.

job-name: Specify the name of the job that created the file being held.

user-name: Specify the name of the user of the job that created the file being held.

job-number: Specify the number of the job that created the file being held.

SPLNBR

Specifies the number of the spooled file to be held that was created by the specified job. More information on

this parameter is in Appendix A, "Expanded Parameter Descriptions."

***ONLY:** One spooled file from the job has the specified file name. The number of the spooled file is not necessary. If *ONLY is specified and more than one spooled file has the specified file name, a message is sent.

***LAST:** The spooled file with the highest number and the specified file name is used.

spooled-file-number: Specify the number of the spooled file that has the specified file name to be held.

SELECT

Specifies which group of files are selected to be held. Files can be selected based on user, device, form type, and user data. Only files that meet each of the requirements are selected.

Element 1: User Values

***CURRENT:** Only files created by the user running this command are held.

***ALL:** Files created by all users are held.

user-name: Specify the names of files to be held that were created by the specified user.

Element 2: Device Values

***ALL:** Files queued for any device or on any object queue are held.

***OUTQ:** All files that are not queued for a device are held. These files are on output queues that are not associated with printers.

device-name: Specify the names of files to be held that are queued for the specified device.

Element 3: Form Type Values

***ALL:** Files for all form types are held.

***STD:** Only files that specify the standard form type are selected.

form-type: Specify the form type of the files to be held.

Element 4: User Data Values

***ALL:** Files with any user data tag specified are held.

user-data: Specify the user data tags of files to be held.

OPTION

Specifies which option to use when holding a spooled file. This parameter allows the user to choose when to hold a spooled file. If the spooled file being held is not currently being processed by a spool writer, this parameter is not valid. If the spooled file is being processed by a printer writer, then printing stops and a form feed is done to prepare the printer for the next spooled file to be printed.

***IMMED:** The spooled file is held immediately.

***PAGEEND:** The spooled file is held at a page boundary.

Examples

Example 1: Holding a File Created by Another Job

```
HLDSPLF  FILE(SHIPITEMS)  JOB(00009/JONES/ORDER)
```

This command withholds the spooled file SHIPITEMS, created by the job ORDER, from additional processing.

Example 2: Holding a File at a Page Boundary

```
HLDSPLF  FILE(QPJOBLOG)  OPTION(*PAGEEND)
```

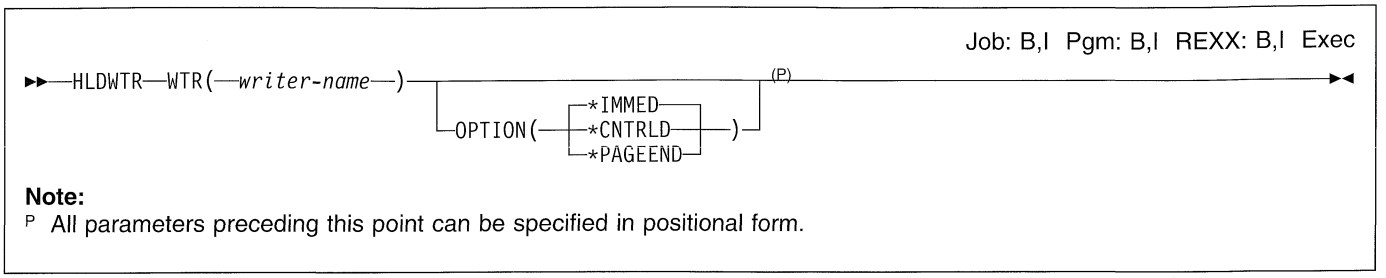
This command holds the spooled file QPJOBLOG at a page boundary.

Example 3: Holding a File Immediately

```
HLDSPLF  FILE(QPJOBLOG)  OPTION(*IMMED)
```

This command holds the spooled file QPJOBLOG immediately. Holding a spooled file by specifying this option causes the CHGSPLFA command RESTART(*NEXT) to be inaccurate if the spooled file is currently being processed by a spool writer.

HLDWTR (Hold Writer) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Hold Writer (HLDWTR) command stops the specified writer at the end of a record, at the end of a spooled file, or at the end of a printed page. If multiple copies of a file are produced, the writer can be held at the end of the copy currently being produced. The writer is not ended and the device is not made available to the system. The writer remains inactive until a RLSWTR (Release Writer) or ENDWTR (End Writer) command is issued. Data is not lost when the writer is held.

Required Parameters

WTR

Specifies the name of the spooling writer being held. Specify the name of the spooling writer.

Optional Parameters

OPTION

Specifies when the spooling writer should stop producing output.

***IMMED:** The writer stops immediately after it has written the last record, in the current block of records, to the output device. Each time the writer finishes producing a block of records on a device, it makes another I/O request to get the next block from the file being spooled to the device. If *IMMED is specified, the writer stops only after it has written the last record in the block being processed, which (for diskette output) is a complete diskette record being written on diskette.

When *IMMED is specified for printed output, the writer stops anywhere within or at the end of a print line or at the end of a complete block, which may not be at the end of a line. This is because some data records (which are blocked to improve performance) may be split in two, with the first part of a record at the end of one block and the last part of the record at the beginning of the next block. If only one copy of the file is being produced or if the last copy is being produced, the entry for the file is removed from the output queue when the output is completed.

***CNTRLD:** The job is ended in a controlled manner. This allows the program to perform cleanup (end-of-job processing).

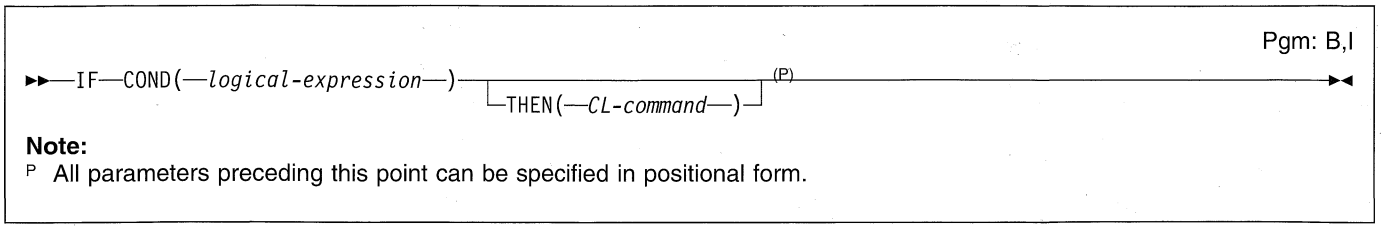
***PAGEEND:** The writer is held at the end of a page. This value is valid only when the spooling writer is a printer writer.

Example

```
HLDWTR  WTR(PRINTER)  OPTION(*CNTRLD)
```

This command stops the writer named PRINTER at the end of the current file. The writer is held until an RLSWTR (Release Writer) or ENDWTR (End Writer) command is issued.

IF (If) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The If (IF) command evaluates a logical expression and conditionally processes CL program commands according to the evaluation of the expression. If the logical expression is true (a logical 1), the command (or the group of commands in a Do group) specified in the THEN parameter is processed, and the ELSE command with its associated command or Do group is not processed. If the logical expression is false (a logical 0), the command specified in the THEN parameter is not processed and control passes to the next command. If that command is an ELSE command, the command or Do group specified in that command is processed. If the ELSE command is not specified, control passes to the next command.

When a DO command is specified, either in the THEN parameter of the IF command or in the CMD parameter of the ELSE command, the Do group is bypassed if the result of the expression is not the one needed for the group being processed. That is, control passes to the command that follows the ENDDO command associated with that DO.

When the command or Do group specified by the THEN parameter or the ELSE command is completed (and no GOTO command has been processed), control passes to the next command following the command or Do group specified by the ELSE command. If a GOTO command is processed, control is passed to the command with the label specified by the GOTO command, and processing continues from that command.

The following command sequence shows this flow. In this example, &TESTSW is a logical variable.

```
IF &TESTSW DO
   Group A  (group of CL commands)
        •
        •
   ENDDO

ELSE DO
   Group B  (group of CL commands)
        •
        •
   ENDDO

Group C  (continued CL commands)
        •
        •
```

The IF command tests the logical variable &TESTSW. If a true condition exists (&TESTSW contains a value of '1'), the commands in Group A are processed, then control passes to the commands in Group C. If a false condition exists (&TESTW contains a value of 0), the commands in group A are bypassed, the commands in Group B are processed, then control passes to the commands in Group C.

Restrictions:

1. The IF command is valid only in CL programs.
2. Up to ten levels of nested IF and ELSE commands are allowed.

Required Parameters

COND

Specifies the logical expression that is evaluated to determine a condition in the program and what is done next. Refer to "Logical Expressions" in Chapter 3, "Expressions in CL Commands" for a description of logical expressions. Note that variables, constants, and the %SUBSTRING, %SWITCH, and %BINARY built-in functions can be used within the expression.

Optional Parameters

THEN

Specifies the commands (in a Do group) that are processed if the result of evaluating the expression is true. After the command or Do group is processed, control is passed to the next command *after* the ELSE command associated with this IF command. If the result is true, the ELSE command associated with the IF command is not processed. If the command specified in this parameter is a DO command, all commands within the Do group are considered to be the command specified by the parameter.

If the command specified by the THEN keyword is not coded on the same line when the keyword is coded, the THEN keyword must be immediately followed (on the same line) either by the left parenthesis or by a plus sign (+) or a minus sign (-) to show continuation. (A blank cannot immediately follow any keyword.) The command and the right parenthesis can then be coded on the next line. For example:

```
IF COND(&A *EQ &B)  THEN(      +
  GOTO C)
```

If any part of the command specified by the THEN parameter continues on the next line, a continuation character (+ or -) must be specified.

If a DO command is specified, only the DO command (not the commands specified within the Do group) is within the parentheses. For example:

```
IF COND(&A *EQ &B)  THEN(DO)
  CMD1
  CMD2
    •
    •
  ENDDO
```

If no command is specified on the THEN parameter (a null THEN) and the ELSE command immediately follows it, the ELSE is processed if the IF expression is false and it is skipped if the expression is true.

Any CL command can be specified on the THEN parameter, except the following commands:

ELSE
PGM, ENDPGM
ENDDO
MONMSG
DCL, DCLF

The command can be another IF, unless there are already ten levels of nested IF and ELSE commands.

Examples

```
IF  COND(&A *EQ &B)  THEN(GOTO X)
IF  (&A *EQ &B)  THEN(GOTO X)
IF  (&A *EQ &B)  (GOTO X)

IF  COND(&A *EQ &B)  THEN(GOTO X)
```

The examples above show a number of different ways the IF command can be specified to test a condition and branch to a label. In each of these examples, if &A equals &B, control passes to a CL command that has a label named X.

```
IF  COND(&TESTSW)  THEN(CHGVAR  VAR(&A)  VALUE(23))
```

If &TESTSW has a logical value of 1 (true), the CHGVAR command is processed to set &A to decimal 23; if &TESTW has a logical value of 0 (not true), the Change Variable (CHGVAR) command is not processed.

```
IF  ((&ALPHA *EQ &BETA) *AND *NOT &GAMMA)  THEN(RETURN)
```

If the value of &ALPHA equals the value of &BETA and if &GAMMA is a logical 0, then return to the program that called this CL program.

```
IF  &LOG1  THEN(IF  (&A *GT 10) THEN(GOTO X))
  ELSE(GOTO Y)
ELSE DO
   •
   •  (group of CL commands)
ENDDO
```

This is an example of nested IF commands. If &LOG1 has a logical value of 1 (true) and if &A is greater than 10, a branch is made to label X. If &LOG1 has a logical value of 1 and &A is *not* greater than 10, a branch is made to label Y. If &LOG1 has a logical value of 0 (false), &A is not compared to 10. Instead, the DO group of the second ELSE command is processed.

```
| IF   &TEST  THEN(     +
        DO)
           CHGVAR  &A (&A + 1)
           GOTO  X
        ENDDO
ELSE
        DO
           CHGVAR &B (&B + 1)
           CALL EXTPGM  (&B)
    ENDDO
```

This example shows how the THEN parameter can be continued on the next line. If &TEST has a logical value of 1, the Do group specified in the THEN parameter is processed. Otherwise, the Do group specified by the ELSE command is processed.

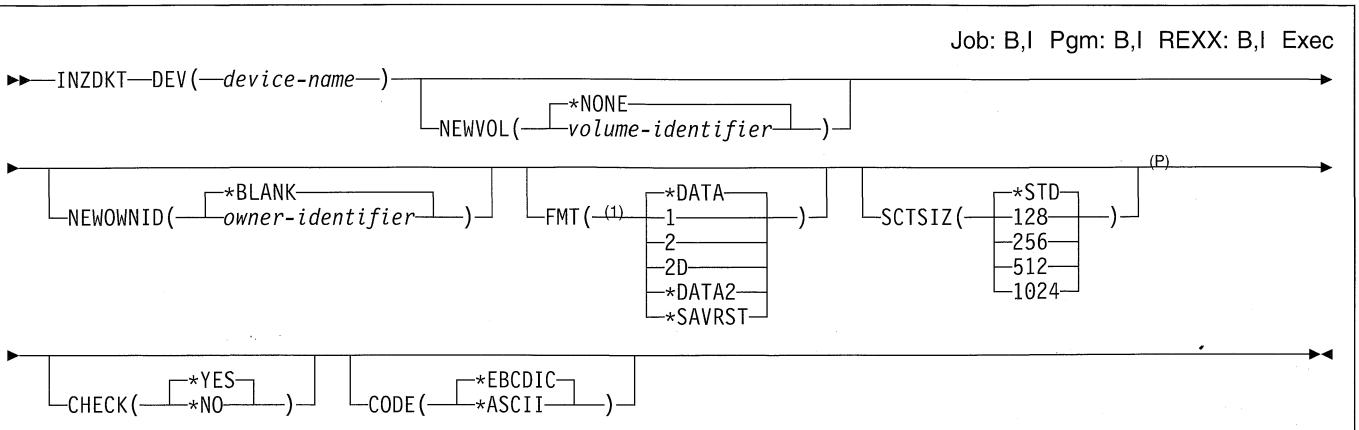
```
IF   (&A *EQ YES)
DO
  CHGVAR &B 1
  CHGVAR &C 'Z'
ENDDO
```

This example shows a Do group as the THEN parameter. The two Change Variable (CHGVAR) commands are processed if, in the relational expression, &A is equal to YES.

```
IF   %SWITCH(10XXXX10)   THEN(GOTO X)
```

This example shows how the built-in function %SWITCH is used to test the eight job switches in a job. Refer to the end of Chapter 3, "Expressions in CL Commands" for a complete description of %SWITCH. In this example, job switches 1, 2, 7, and 8 are tested for the values indicated in the 8-character mask. If switches 1 and 7 contain 1s and switches 2 and 8 contain 0s, then the program branches to the command having the label X. If any of the four switches do not contain the value indicated, the branch does not occur.

INZDKT (Initialize Diskette) Command



Purpose

The Initialize Diskette (INZDKT) command initializes a diskette for use. This command writes identification information on a diskette and sets the format. Initializing a diskette includes the following:

- Checking for active files that should not be cleared.
- Testing each track for physical defects on the recording surface. A diskette is unusable if more than two defective cylinders are found, if cylinder 0 is defective, or if the track identifier of a defective track cannot be read.
- Formatting each track to a specified sector size (128, 256, 512 or 1024 bytes) and recording density (single density or double density). A diskette's format determines what the diskette may be used for in later processing. This is explained more fully in the FMT and SCTSIZ parameter.
- Defining a single (expired) file covering the entire diskette. The file is identified as DATA.

IBM-supplied diskettes are initialized before they are shipped to a customer. A diskette should be reinitialized when:

- Its format is changed.
- The sequence of the records on the diskette is changed (they can only be sequential).
- A defect has occurred in one or two tracks.
- The diskette has been exposed to a strong magnetic field.

One INZDKT command can initialize only one diskette at a time.

Note: When initializing diskettes with labels that are not IBM standard labels, specify CHECK(*NO).

Required Parameters

DEV

Specifies the name of the device in which the diskette being initialized is placed.

Optional Parameters

NEWVOL

Specifies the volume identifier for the diskette being initialized.

***NONE:** No volume identifier is specified; only the system date is written in the volume identifier field of the volume label.

volume-identifier: Specify up to 6 characters for the volume identifier to identify the diskette being initialized; any combination of alphabetic characters (except $, #, or @) or numeric characters can be used.

NEWOWNID

Specifies the identification of the new diskette owner to write in the volume label. Any combination of alphabetic characters (except $, #, or @) or numeric characters can be used.

***BLANK:** Text is not specified.

owner-identifier: Specify up to 14 uppercase alphabetic or numeric characters that identify the new owner of the diskette. Even if the value is enclosed in apostrophes, no lowercase letters, embedded blanks, or special characters can be used. If fewer than 14 characters are specified, the field is left-justified and padded on the right with blanks.

FMT

Specifies the format to use to initialize the diskette. Either *DATA, *DATA2, 1, 2, or 2D can be specified if the diskette will contain data files that are in the basic, H, I, or System/36 environment exchange formats. The basic exchange format has a set of requirements that

ensures that a diskette can be exchanged between systems that are capable of using both the IBM diskette types 1 or 2. *SAVRST (type E general data exchange) must be specified if the diskette is used in the AS/400 system or System/38 save and restore operations; that is, if their data files will contain saved objects. *SAVRST is also a valid format for the System/36 save and restore operations, and it is the preferred format to use.

***DATA:** A one- or two-sided diskette is formatted with single-density recording.

1: A one-sided diskette is formatted with single-density recording.

2: A two-sided diskette is formatted with single-density recording.

2D: A two-sided diskette is formatted with double-density recording.

***DATA2:** A two-sided diskette is formatted with double-density recording (as if 2D is specified).

***SAVRST:** A two-sided diskette is formatted with double-density recording. This format is required if the diskette is used in a save and restore operation.

Notes:

1. Because *DATA2, *SAVRST, and 2D specify that the diskette is used for double-density recording, it is recommended that a type 2D diskette be used, rather than a type 2. A type 2D diskette is made for double-density recording, whereas a type 2 is made for single-density recording and is more prone to media errors if used for double-density recording.
2. If FMT(*SAVRST) is specified, CODE(*ASCII) cannot be specified.

SCTSIZ

Specifies the number of bytes per sector with which each track is initialized.

***STD:** A standard sector size, based on the value of the FMT parameter, is used. Such as the following:

FMT	SCTSIZ (*STD)
*DATA	128
1	128
2	128
2D	256
*DATA2	256
*SAVRST	1024

128: Each track is initialized with 128 bytes per sector.

256: Each track is initialized with 256 bytes per sector.

512: Each track is initialized with 512 bytes per sector.

1024: Each track is initialized with 1024 bytes per sector.

Table 54, at the end of this command description, shows the valid sector sizes.

Format for 8-inch diskette.

Table 55, at the end of this command description, shows the format for 5.25-inch diskette

The following chart shows which exchange types can be used for each sector size.

Exchange Type	128	256	512	1024
Basic	X			
H		X[1]		
I	X	X	X	X
SAVRST				X
S/36-S/R	X	X	X	X

1 H is used only as the exchange type if the diskette is initialized with double-density recording (FMT(*DATA2) or FMT(2D)).

CHECK

Specifies whether a check is made for active files (files having an expiration date greater than the system date).

***YES:** A check is performed on files whose labels are in cylinder 0 only. File labels in an extended file label area are not checked. If any active files are found, an operator message is sent. The operator can continue initialization (active files are destroyed), or end the initialization of that diskette.

***NO:** Diskette initialization proceeds without a check for active files.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The volume label is written in EBCDIC and is an IBM standard label; subsequent data must also be written in EBCDIC.

***ASCII:** The volume label is written in ASCII and is an IBM standard label in the same format as the EBCDIC label; subsequent data must also be written in ASCII. If FMT(*SAVRST) is specified, *ASCII cannot be specified.

Table 54. Valid Sector Size

SCTSIZ	*DATA	1	2	2D	*DATA2	*SAVRST
*STD	X	X	X	X	X	X
128	X	X	X			
256	X	X	X	X	X	
512	X	X	X	X	X	
1024				X	X	X

Table 55. Format for 5.25 Inch Diskette

SCTSIZ	*DATA	1	2	2D	*DATA2	*SAVRST
*STD				X	X	X
256				X	X	
512				X	X	
1024				X	X	X

Examples

Example 1: Initializing a Diskette

```
INZDKT  DEV(DKT1)  NEWVOL(ORIGIN)  NEWOWNID(DEPT504)
```

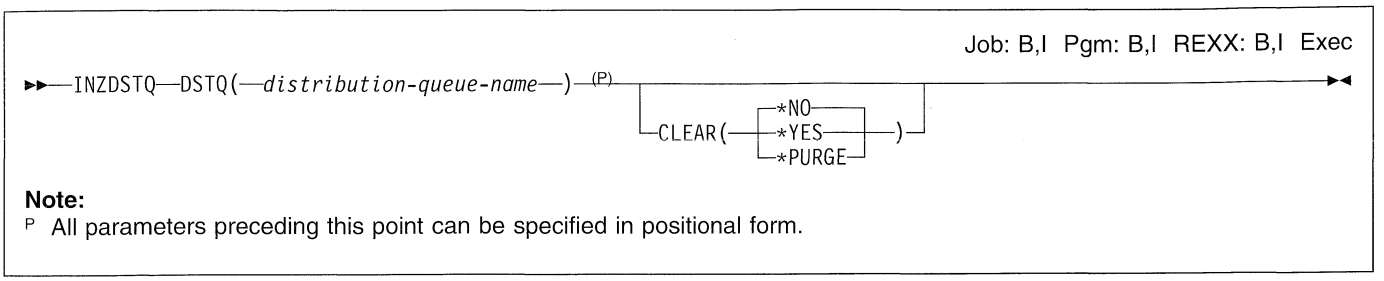
This command initializes the diskette device DKT1. The diskette is checked for active files (CHECK(*YES) is assumed). The diskette is formatted for basic data exchange files (FMT(*DATA) is assumed). The volume label has ORIGIN written in the volume identifier field and DEPT504 written in the owner identifier field.

Example 2: Initializing a Diskette to the Save and Restore Format

```
INZDKT  DEV(DKT2)  NEWVOL(SAVE)
  NEWOWNID(DON)  FMT(*SAVRST)  CHECK(*NO)
```

This command initializes the diskette in device DKT2 to the save and restore format. The diskette is initialized with the NEWVOL parameter of SAVE. The owner identifier field has DON written into it.

INZDSTQ (Initialize Distribution Queue) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Initialize Distribution Queue (INZDSTQ) command resets the status of a distribution queue and the entries on the queue. It also optionally clears all distributions on the queue. This command applies to both the normal and high priority sections of the specified queue.

Warning: Initializing a distribution queue can result in the loss or duplication of distributions in the network, depending on the status of the distributions in transit at the time this command is run.

Initializing a distribution queue includes the following:

- If a SNADS (Systems Network Architecture (SNA) distribution services) sender job is active for the queue, the job is ended. This job cancelation takes effect immediately. Distribution queues being sent are interrupted.
- If the queue type is a SystemView distribution services (SVDS) queue type and a receiver job is active for this connection, the job is ended. This job cancelation takes effect immediately. All partially received distributions are discarded.
- If the distribution queue is to be cleared, all distributions on the queue are deleted as specified on the CLEAR parameter.
- If the queue is not cleared, the distributions on the queue that do not have "Held" status are set to "Ready". Distributions with a status of "Held" remain held.
- The queue status is set to "Ready" unless the queue is in the "Held" status.
- If the QSNADS system is active, a SNADS sender job is submitted for the queue following the same rules used to start the QSNADS subsystem.

Distribution queue names are translated to the graphic character set and code page 930 500, using the job's coded character set identifier (CCSID).

Restrictions:

1. This command is shipped with public *EXCLUDE authority, and the QPGMR and QSYSOPR user profiles have private authorities to use the command.
2. Messages that report errors about distribution queues may display or print different characters than the user entered for the distribution queue name because of internal system transformations. Similarly (depending on the language used for the work station), the internal value for a distribution queue name may differ from the characters shown on the Work with Distribution Queue (WRKDSTQ) command. An error may be reported if the character-string value specified for the DSTQ parameter does not match the rules for an internal distribution queue value or if it does not match the internal value for any defined distribution queue (ignoring case differences).

Required Parameter

DSTQ
Specifies the name of the distribution queue to initialize. The queue must be previously configured using the Configure Distribution Services (CFGDSTSRV) or the Add Distribution Queue (ADDDSTQ) command.

Optional Parameters

CLEAR
Specifies whether distributions on the queue are deleted.

Warning: Using the *PURGE value results in the loss of distributions with no trace.

***NO:** Distributions on the queue are not deleted.

***YES:** Distributions on the queue are deleted. Each deleted distribution is logged and, if the distribution originator requested notification, a notification is sent to the originator or to the report destination specified in the distribution.

Note: System Network Architecture distribution services (SNADS) status distributions and distribution reports are used to report information about a distribution back to the originator. Status report distributions never result in another status report distribution. If a status report distribution is deleted, no notification is sent.

***PURGE:** Distributions on the queue are deleted. Deleted distributions are not logged and no notification is sent to the originator or to the report destination specified in the distribution.

Examples

Example 1: Initializing a Distribution Queue

```
INZDSTQ  DSTQ('SYSTEMA APPN')
```

Connection information is about to be changed for system 'SYSTEMA APPN' by a central site administrator. This command initializes the queue to avoid error conditions that can be encountered by the Change Distribution Queue (CHGDSTQ) command. Distributions on the queue are not deleted.

Example 2: Initializing and Clearing a Distribution Queue

```
INZDSTQ  DSTQ('ERRORQ')  CLEAR(*YES)
```

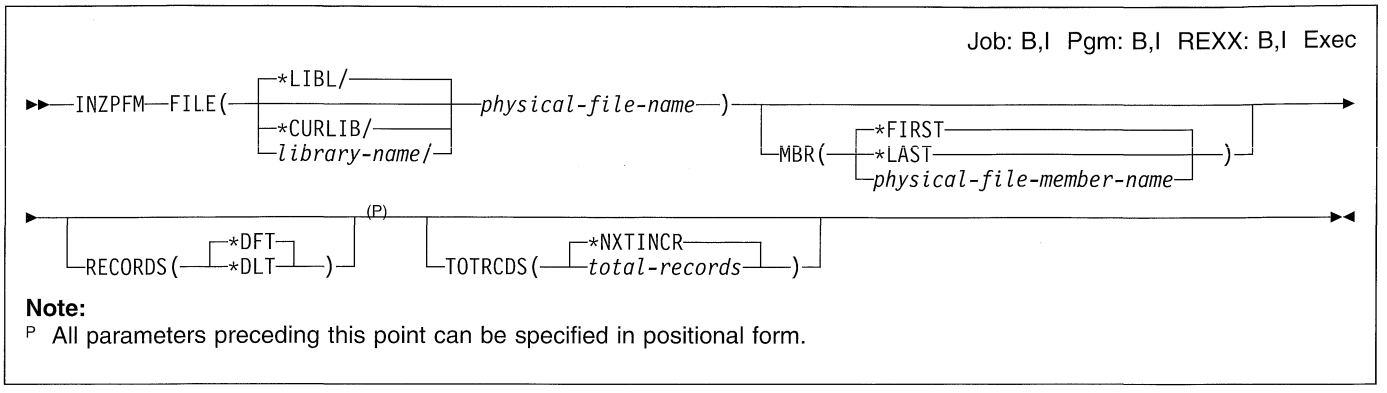
This command clears the distribution queue ERRORQ that is being used as a repository for distributions that would have resulted in routing errors. Distributions that are deleted are logged, and the originators of the distributions are notified.

Example 3: Initializing and Purging a Distribution Queue

```
INZDSTQ  DSTQ('TESTQ')  CLEAR(*PURGE)
```

This command clears the distribution queue TESTQ that is being used for testing a new batch application. Distributions are deleted but not logged, and the originators are not notified.

INZPFM (Initialize Physical File Member) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Initialize Physical File Member (INZPFM) command initializes records in a member of a physical file to the specified type of record (either default or deleted records). This command is usually used for files that are processed in arrival sequence or by relative record numbers. If the initialized member is empty, records are added and initialized to the specified type; if the member is not empty, records of the specified type are added to the member. As many records are added as is necessary to make the total record count specified.

Note: The INZPFM command ignores all file overrides that are currently in effect for the job.

Restrictions: (1) The member may be open for input (read only) while the initialize operation takes place but cannot be open for update, delete, or insert. (2) To initialize the member with default records, the user needs object operational authority, object management authority, and add authority for the file in which the member exists. (3) To initialize the member with deleted records, the user also needs delete authority for the file. (4) An *EXCLRD lock is required on the member to initialize it.

Required Parameters

FILE

Specifies the qualified name of the physical file that contains the member being initialized.

The name of the physical file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

physical-file-name: Specify the name of the physical file.

Optional Parameters

MBR

Specifies the name of the member that is initialized.

***FIRST:** The first member in the database file is used.

***LAST:** The last member of the specified physical file is initialized.

physical-file-member-name: Specify the name of the physical file member that is initialized.

RECORDS

Specifies the type of records that are initialized (added) to the specified member. The specified members are initialized with default records or deleted records.

***DFT:** The specified member is initialized with default records. If a default value was specified in the DDS (DFT keyword) for a field, that field is initialized to the specified default; otherwise, all numeric fields are initialized to zeros and all character fields are initialized to blanks.

***DLT:** The specified member is initialized with deleted records. The records are not eligible for access, but simply hold a place in the file. Deleted records can be updated to reuse the deleted space.

TOTRCDS

Specifies the total number of records in the member after it is initialized. If the value specified in this parameter causes the size of the file to be larger than the size specified when the file was created, a message is sent to the system operator's message queue (QSYSOPR). The operator can either continue or end the operation.

***NXTINCR:** The number of records in the member is increased to extend the file to the next allocation increment. If the member is empty, records are added to meet the first allocation specified for the member. If SIZE(*NOMAX) is specified when the file is created, *NXTINCR is not valid.

total-records: Specify the total number of records, ranging from 1 to 2147483646, by which the member is increased. If the number of existing records in the

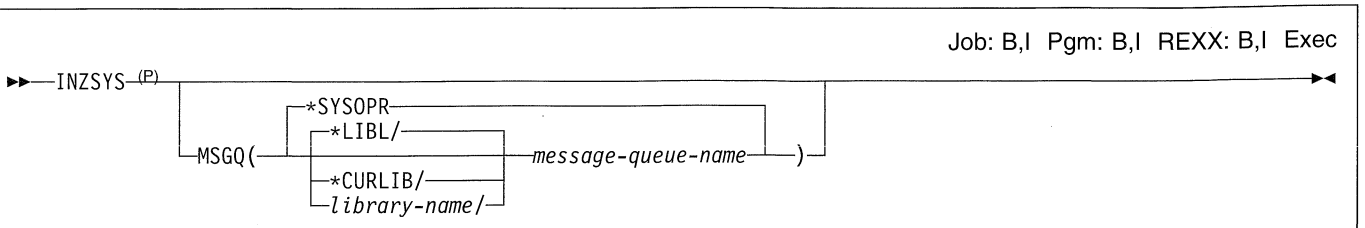
member already meets or is larger than this number, no records are initialized; if the number is less than that specified, enough records are initialized to equal the total specified.

Example

```
INZPFM   FILE(*CURLIB/INV)  TOTRCDS(12000)
```

This command initializes as many as 12,000 records in the first member of the physical file named INV in the job's current library *CURLIB. Only the number of records are added that brings the total to 12,000 records in the member. Any records that are added are initialized to the default format. If a default value is specified in the DDS (DFT keyword) for a field, that field is initialized to the specified default; otherwise, all numeric fields are initialized to zeros and all character fields are initialized to blanks.

INZSYS (Initialize System) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Initialize System (INZSYS) command initializes conversions done during installation procedures. This process is initiated during the first IPL after the software package is installed.

More information is available in the *Licensed Programs and New Release Installation Guide*.

Optional Parameters

MSGQ

Specifies the qualified name of the message queue to which messages are sent.

***SYSOPR:** Messages from the system operator message queue (QSYSOPR) are sent to the system operator.

The name of the message queue can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

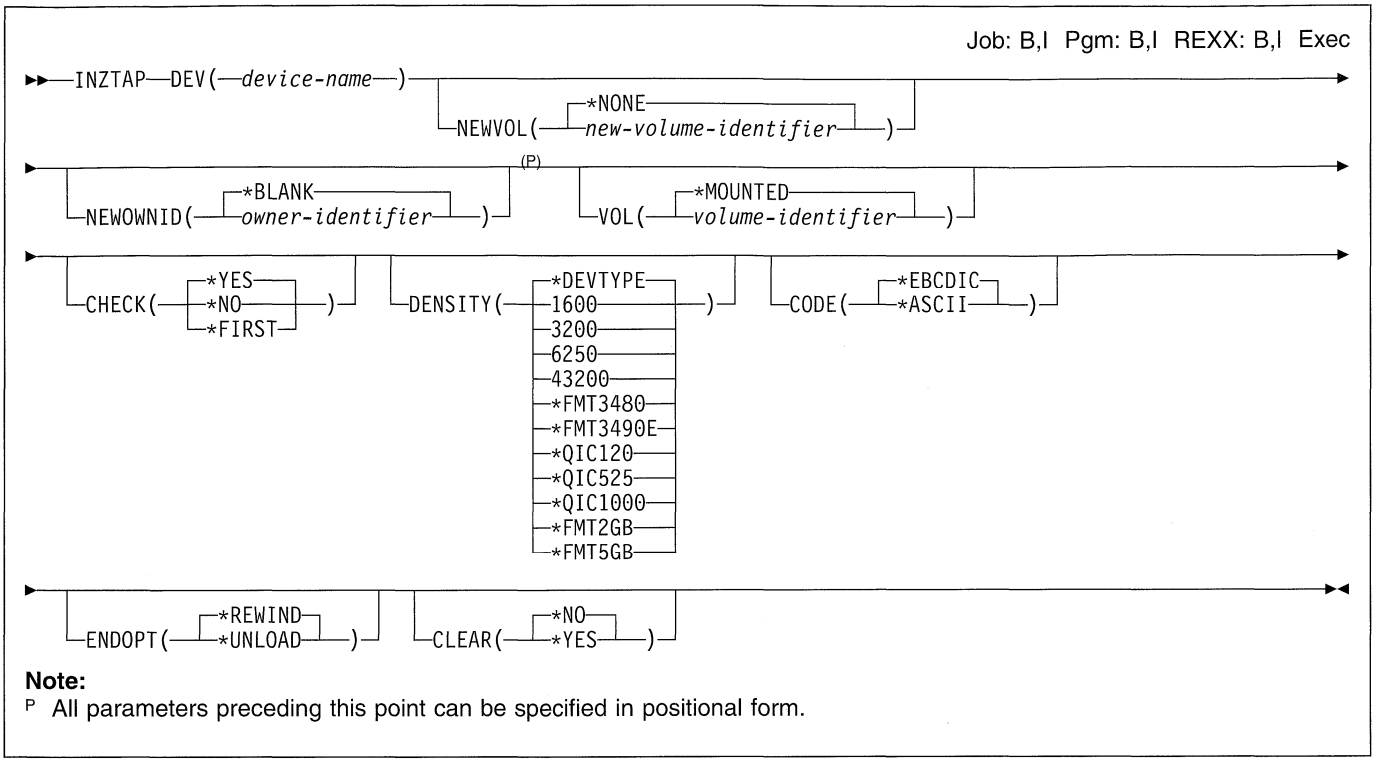
message-queue-name: Specify the name of the message queue name to which messages are sent.

Example

```
INZSYS
```

This command initializes the conversions done during installation procedures.

INZTAP (Initialize Tape) Command



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Initialize Tape (INZTAP) command prepares magnetic tapes for use on the system. This command is used to write volume labels on standard-labeled magnetic tapes so the tape device support can do standard-label processing. Unlabeled tapes must also be initialized by this command or by a similar process on another system before these tapes can be used on the AS/400 system. Note that this command, unlike the Initialize Diskette (INZDKT) command, does not indicate whether the tape is used for data written in the basic exchange format or in the save and restore format. That is done when the data files are actually written on the tape.

Three operations can be done by this command, depending on the values specified for the CHECK and CLEAR parameters.

1. The tape can be checked for active data files (files that have not reached their expiration dates).
2. The tape can be initialized either as a standard-labeled tape (if NEWVOL specifies a volume identifier) or as an unlabeled tape. Initialization writes only on the beginning of the tape, but it makes all data on the reel inaccessible.
3. If the tape is being initialized, any previous data on the tape can also be deleted.

If the tape is being initialized as a standard-labeled volume, a volume label followed by two tape markers is written at the beginning of the tape. If it is initialized as an unlabeled tape, only the two tape markers are written at the beginning of the tape.

All tapes must be initialized before use. Tapes that have been initialized do not need to be reinitialized unless the user wants to write a new volume label, change the tape type from a standard-labeled tape to an unlabeled tape or vice versa, or change the density of a standard labeled tape.

Restriction: Non-labeled tapes cannot be duplicated to 1/4 inch or 8mm cartridge devices.

Required Parameters

DEV
Specifies the name of the device in which the volume being initialized is placed. Specify the name of the device.

Optional Parameters

NEWVOL
Specifies the volume identifier for a tape being initialized as a standard-labeled tape. If no volume identifier is specified, the tape is initialized as an unlabeled tape; that is, it has no volume label and no header labels for data files that are written on it later.

***NONE:** The tape is initialized as an unlabeled tape. Only tape markers are used to indicate the beginning and end of each data file on it, and the beginning and end of the volume itself.

new-volume-identifier: Specify up to 6 characters to identify the new volume. The identifier must contain only alphanumeric characters (A through Z, $, #, @, and 0 through 9), and cannot have a prefix or contain blanks. Each tape reel should have a unique volume identifier to ensure optimum protection and control of the tape volumes.

NEWOWNID

Specifies which tape owner's identifier to write in the volume label of the volume being written. The owner identification contains up to 14 characters (letters and/or numbers in any combination), is left-justified, and padded on the right with blanks if fewer than 14 characters are supplied.

***BLANK:** Text is not specified.

owner-identifier: Specify up to 14 characters that identify the owner of the tape. If fewer than 14 characters are specified, the field is left-justified and padded on the right with blanks.

VOL

Specifies one or more volume identifiers used by the file. More information on this parameter is in Appendix A, "Expanded Parameter Descriptions."

***MOUNTED:** Any labeled or unlabeled volume on the specified tape device is initialized. VOL(*MOUNTED) and CHECK(*NO) must be used to initialize a new or completely erased tape volume. Otherwise, the system attempts to read labels from the tape volume that is on the specified device and completely unwinds from the reel.

volume-identifier: Specify the identifier of the labeled volume being initialized. This parameter value can be used only to reinitialize a tape that is already a labeled volume. If the tape on the specified device has a different volume identifier than the one specified by this value, or if it is an unlabeled volume, an error message is sent.

CHECK

Specifies whether a labeled tape volume is checked for active data files (files with an end date later than the current system date) before it is initialized. If an unlabeled volume is on the specified device for initialization, this parameter is ignored. If the volume must be checked for active files, as much of the tape is read as necessary before initialization is done.

***YES:** All data file labels on the tape are checked. If active files are found, the operation is ended and an error message is sent.

***NO:** Tape initialization continues with no checking for active files. To initialize a new or empty volume, VOL(*MOUNTED) and CHECK(*NO) must be specified; otherwise, the system attempts to read labels from the volume on the specified device until the tape completely unwinds from the reel.

***FIRST:** Only the first data file label on the tape is checked. If there are no data files in the volume, or if the first data file has ended, the volume is initialized without checking for any other files on the reel. If the first data file has not ended (it is active), the operation is stopped and an error message is sent.

If it is known that there is only one data file on the tape, that there are no active files on the volume, or that all files have the same end date, use CHECK(*FIRST) to do the fastest initialization of the tape. Note, however, that any data files past the first one are destroyed, with no end check made.

DENSITY

Specifies (in bits per inch (bpi)) the density or format in which to write the data on the tape after it has been initialized. The density used for all data files written to a standard-labeled tape is specified when the volume is initialized, and cannot be changed unless the tape is reinitialized. For a tape that is not labeled, the DENSITY parameter specified in the Create Tape File (CRTTAPF) or the Change Tape File (CHGTAPF) command can change the density of the volume when the first data file on the tape is created.

***DEVTYPE:** The data written on the tape volume is based on the device type (the type of tape device).

Tape Device	Default Density (bpi)
2440	6250
3422	6250
3430	6250
3480	*FMT3480
3490	*FMT3490E
6341	*QIC120
6342	*QIC525
6343	*QIC1000
6346	*QIC120
6347	*QIC525
6348	*QIC1000
6366	*QIC1000
6368	*QIC1000
7208	43200
7208	*FMT2GB
7208	*FMT5GB
9346	*QIC120
9347	3200
9348	6250

1600: The data density on this tape volume is 1,600 bits per inch. This density is used for one half inch reel tapes.

3200: The data density on this tape volume is 3,200 bits per inch. This density is used for one half inch reel tapes on devices that support this density.

6250: The data density on this tape volume is 6,250 bits per inch. This density is used for one half inch reel tapes on devices that support this density.

43200: The data density on this tape volume is 43,200 bits per inch. This density is used for 8 millimeter cartridge tapes.

***FMT3480:** The data density on this tape volume is formatted to support a 3480 device. This density is used for one half inch cartridge tapes.

***FMT3490E:** The data density on this tape volume is formatted to support a 3490 device. This density is used for one half inch cartridge tapes.

***QIC120:** The format of this tape is *QIC120. This density is used for one quarter inch cartridge tapes.

***QIC525:** The format of this tape is *QIC525. This density is used for one quarter inch cartridge tapes.

***QIC1000:** The format of this tape is *QIC1000. This density is used for one quarter inch cartridge tapes.

| ***FMT2GB:** The format of this tape is *FMT2GB. This
| density is used for one quarter inch cartridge tapes.

| ***FMT5GB:** The format of this tape is *FMT5GB. This
| density is used for one quarter inch cartridge tapes.

CODE

Specifies the character code used. The code can be either extended binary-coded decimal interchange code (*EBCDIC) or the American National Standard Code for Information Interchange (*ASCII).

***EBCDIC:** The extended binary-coded decimal interchange code (EBCDIC) character set code is used.

***ASCII:** The ASCII character set code is used.

ENDOPT

Specifies whether the tape is rewound only or rewound and unloaded after the operation ends.

***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

CLEAR

Specifies whether all previous labels and data are deleted from the tape when it is initialized. If the volume must be cleared of all data, it is spaced from the location of the initializing volume label or tape markers to the end of the tape marker.

***NO:** The existing data is not deleted. Even though the existing data is not deleted, the data on the volume is not accessible after the volume has been initialized.

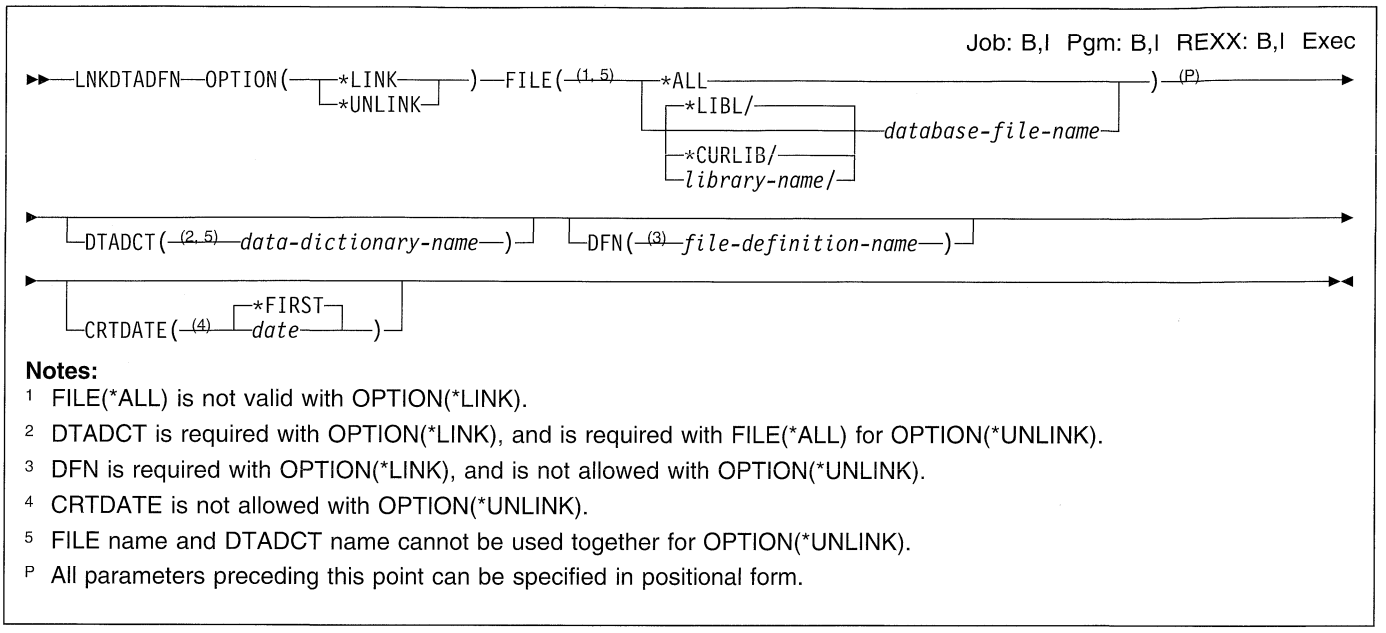
***YES:** After the beginning of the tape has been initialized, the remaining data on the tape is deleted. The *YES value is needed only if there are security concerns with old data. If *YES is specified, the initialize operation can take a long time.

Example

```
INZTAP  DEV(TAPE1)  NEWVOL(T00100)  CHECK(*NO)
  CODE(*ASCII)  ENDOPT(*UNLOAD)
```

This command initializes the volume on the tape device named TAPE1 using the ASCII character code. Its new volume identifier is T00100, regardless of whether it contains a valid volume identifier or files that have not ended (active field). Once the volume has been initialized, the tape is rewound and unloaded. Any previous data beyond the new volume label is not deleted, but is no longer accessible.

LNKDTADFN (Link Data Definition) Command



Notes:

1 FILE(*ALL) is not valid with OPTION(*LINK).

2 DTADCT is required with OPTION(*LINK), and is required with FILE(*ALL) for OPTION(*UNLINK).

3 DFN is required with OPTION(*LINK), and is not allowed with OPTION(*UNLINK).

4 CRTDATE is not allowed with OPTION(*UNLINK).

5 FILE name and DTADCT name cannot be used together for OPTION(*UNLINK).

P All parameters preceding this point can be specified in positional form.

Purpose

The Link Data Definition (LNKDTADFN) command links or unlinks file definitions in a dictionary, program-described files, or externally-described files.

Restriction: A file cannot be linked if it is already linked. However, a definition can be linked to several files at the same time.

Note: If file text and the file definition are not the same, a new version of the definition is created.

Required Parameters

OPTION

Specifies the action performed on the program-described file, externally-described file, or file definition.

***LINK:** The program-described file or the file definition is linked.

***UNLINK:** The program-described file, the externally-described file, or the file definition is unlinked.

FILE

Specifies the qualified name of the program-described file to link or unlink, or specifies the externally-described file to unlink.

***ALL:** All program-described database files linked to definitions in the specified dictionary are unlinked. This value is valid only if OPTION(*UNLINK) is also specified. This value is not valid for externally-described files.

The name of the program-described file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

database-file-name: Specify the name of the file being linked or unlinked.

Optional Parameters

DTADCT

Specifies the name of the dictionary that contains the file definition to link to the program-described file, or the dictionary from which all program-described files are unlinked.

This parameter is required for the *LINK option. It is also required if FILE(*ALL) is specified and the option is *UNLINK.

DFN

Specifies the name of the file definition that is linked to the program-described file. This parameter is only required with the *LINK option.

CRTDATE

Specifies the creation date of the object.

Note: This parameter is valid only with the *LINK option.

***FIRST:** The file definition with the specified definition name and the earliest creation date is used.

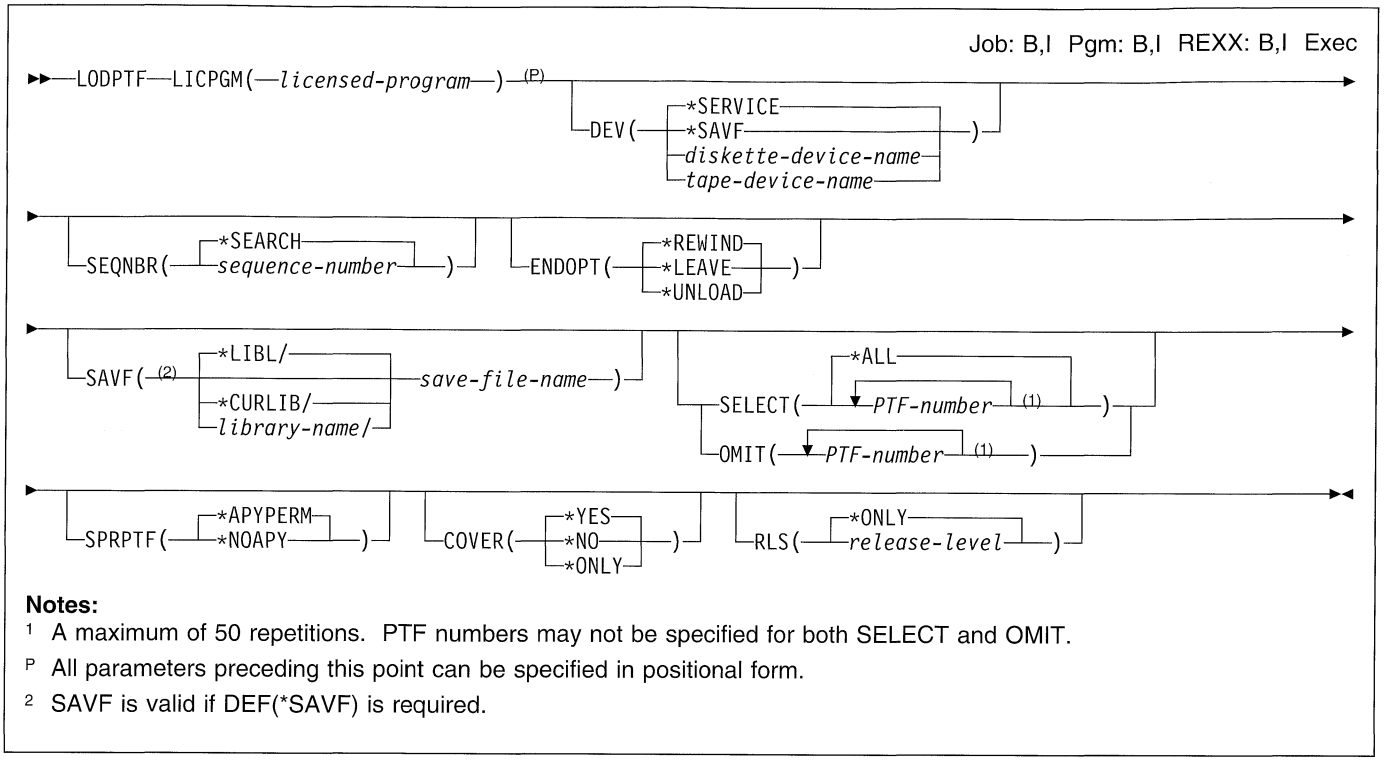
date: Specify the creation date of the file definition to link to the program-described file. If a date is not specified, *FIRST is assumed.

Example

```
LNKDTADFN  OPTION(*LINK)  FILE(MYLIB/MYFILE)
  DTADCT(MINE)  DFN(MYDEF)
```

This command links definition MYDEF in dictionary MINE, to the program-described database file MYFILE located in library MYLIB.

LODPTF (Load Program Temporary Fix) Command



Notes:
1 A maximum of 50 repetitions. PTF numbers may not be specified for both SELECT and OMIT.
P All parameters preceding this point can be specified in positional form.
2 SAVF is valid if DEF(*SAVF) is required.

Purpose

The Load Program Temporary Fix (LODPTF) command loads program temporary fixes (PTFs) for a specified licensed program from diskette, tape, or save files into the licensed program base library. Each PTF contains one or more objects, including programs, that can be applied to a licensed program by the Apply Program Temporary Fix (APYPTF) command.

Only the PTFs for a single licensed program, such as Operating System/400, can be loaded at a time. Not all of the PTFs for the specified program must be loaded, because specific PTFs can be selected and loaded.

Restriction: This command is shipped with public *EXCLUDE authority and the QSRV user profile has private authority to use the command.

Required Parameters

LICPGM
Specifies the licensed program to load that is undergoing a program temporary fix (PTF).

Optional Parameters

DEV
Specifies the device from which the PTFs are loaded. The device name must already be known on the system by a device description.

***SERVICE:** The PTFs sent from the service support system are loaded.

***SAVF:** The PTFs are loaded in save file format. If *SAVF is specified, a save file name must be specified on the SAVF parameter.

diskette-device-name: Specify the name of the diskette device from which the PTFs are loaded.

tape-device-name: Specify the name of the tape device from which the PTFs are loaded.

SEQNBR
Specifies the sequence number on the tape volume where the load operation begins to load the PTF data. This parameter is only valid if a tape-device-name is specified on the DEV parameter.

***SEARCH:** The tape volume is searched for the first program temporary fix (PTF) file for the specified licensed program.

sequence-number: Specify the sequence number of the PTF file being loaded. This sequence number must exist on the tape.

ENDOPT
Specifies the operation that is automatically performed on the tape volume after the operation ends. If more than one volume is included, this parameter applies only to the last tape volume used; all other tape volumes are rewound and unloaded when the end of the tape is reached.

Note: This parameter is ignored if a diskette-device-name is specified on the DEV parameter.

***REWIND:** The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE:** The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

***UNLOAD:** The tape is automatically rewound and unloaded after the operation ends.

SAVF

Specifies the qualified name of the save file from which the PTFs are loaded.

The name of the save file can be qualified by one of the following library values:

***LIBL:** All libraries in the user and system portions of the job's library list are searched.

***CURLIB:** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name: Specify the name of the library to be searched.

save-file-name: Specify the save file name from which the PTFs are loaded.

SELECT

Specifies which of the PTFs for the specified licensed program are loaded. The OMIT parameter cannot be specified if SELECT is specified with PTF numbers.

Note: Permanently removed PTFs will be ignored when *ALL is specified for the SELECT parameter. To load removed PTFs, specify the PTF number on the SELECT parameter.

***ALL:** All the PTFs for the specified licensed program are loaded.

PTF-number: Specify the PTF identification numbers (up to 50) of the single PTFs being loaded.

OMIT

Specifies that all PTFs, except those specified in this parameter, are loaded. Specify the PTF numbers of the PTFs that are omitted when the rest are loaded. Up to 50 PTF numbers can be specified. The OMIT parameter cannot be specified if single PTF numbers are specified on the SELECT parameter.

SPRPTF

Specifies which operation is performed for temporarily applied PTFs that are superseded by PTFs encountered by this load operation.

***APYPERM:** For the specified licensed program, any PTFs that are temporarily applied and are superseded by PTFs contained on the PTF media, are automatically permanently applied before loading the superseding PTFs. If the superseded PTFs have any prerequisite PTFs, they are also permanently applied by this operation.

***NOAPY:** The load operation stops when temporarily applied PTFs are being superseded by PTFs contained on the PTF medium. The temporarily applied PTFs that are superseded must be permanently applied (APYPTF command) or removed (RMVPTF command) before the LODPTF command can be processed again.

COVER

Specifies whether to copy the cover letter for the PTF into a physical file. This parameter is valid only when a tape device is specified on the DEV parameter.

***YES:** After the PTF is loaded, the cover letter is copied into a physical file.

***NO:** The cover letter is not copied into a physical file.

***ONLY:** The cover letter is copied into a physical file but does not load the PTF. If the SEQNBR parameter is specified, it must contain the sequence number of the file that contains the PTF.

RLS

Specifies the release level of the PTFs being loaded.

***ONLY:** This value is valid only when *one* release is installed on the system.

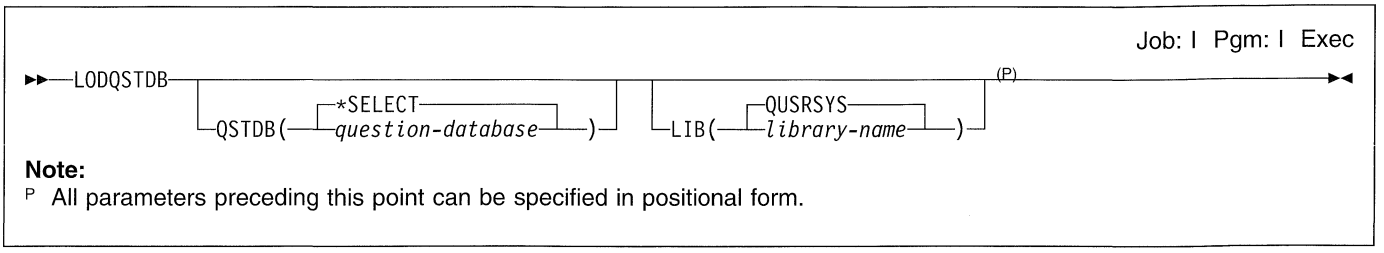
release-level: Specify the release level in the format VxRxMx, where Vx is the version number, Rx is the release number, and Mx is the modification number.

Example

```
LODPTF  LICPGM(5738SS1)  OMIT(SF00003 SF00008 SF000014)
```

This command loads all of the PTFs for the licensed program 5738-SS1, except SF00003, SF00008, and SF000014. The Apply Program Temporary Fix (APYPTF) command can then be used to apply these PTFs to the OS/400 system licensed program.

LODQSTDB (Load Question-and-Answer Database) Command



Note:

| P All parameters preceding this point can be specified in positional form.

Purpose

The Load Question-and-Answer Database (LODQSTDB) command allows the user to load a Question-and-Answer (Q & A) database from an alternative medium (such as tape) to the system. The *Q & A Database Coordinator's Guide* has more information.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. A user must have authority to the command and be a Q & A coordinator for any Q & A database referred to by the command.
3. This command is interactive only.

Optional Parameters

QSTDB

Specify the Q & A database to be loaded to.

***SELECT:** The user is asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database: Specify the name of the Q & A database to be used. If the Q & A database already exists on the system, the supplied subset of the Q & A database will be replaced. If the Q & A database does not exist on the system, it will be created in the specified library.

LIB

Specifies the name of the library that contains the Q & A database.

QUSRSYS: The library default for this command is QUSRSYS.

library-name: Specify the library where the Q & A database is to be loaded. The library must exist on the system.

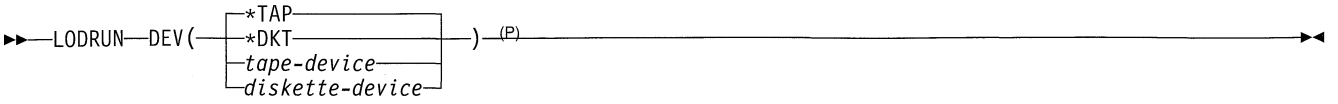
Example

```
LODQSTDB
```

This command shows the Load Database to System display.

LODRUN (Load and Run Media Program) Command

Job: B,I Pgm: B,I REXX: B,I Exec



Note:
P All parameters preceding this point can be specified in positional form.

Purpose

The Load and Run Media Program (LODRUN) command restores a user-written program object from tape or diskette into the library QTEMP. The system passes the device name to the restored program and transfers control to the restored program.

When the LODRUN command is run:

1. The media is searched for the user-written program, which must be named QINSTAPP. If tape is used, the tape is rewound first.
2. If a QINSTAPP program already exists in the QTEMP library on the user's system, it is deleted.
3. The QINSTAPP program is restored to the QTEMP library using the RSTOBJ command. The following values are specified on the RSTOBJ command: OBJTYPE(*PGM), SEQNBR(*SEARCH), ENDOPT(*LEAVE), and ALWOBJDIF(*NONE).

   The device used for the restore operation is determined by the LODRUN command. If DEV(*TAP), the default value, is specified, the LODRUN command examines the QDEVNAMING parameter retrieved by the Retrieve System Value (RTVSYSVAL) command. If the returned value is *S36, DEV is set to T1; otherwise, DEV is set to TAP01. If DEV(*DKT) is specified, and the returned value is *S36, DEV is set to I1; otherwise, DEV is set to DKT01. Any other value specified for the DEV parameter is used as is.
4. Control of the system is passed to the QINSTAPP program. The QINSTAPP program can be used, for example, to restore other applications to the user's system and run those applications.
5. When the user signs off, the QINSTAPP program is removed from the system.

Note: If the system security level on which the LODRUN command is run is 40, and the QINSTAPP program validation value is missing or incorrect, the system attempts to translate the program again. If this fails, the program is restored, but all public and private authorities are revoked and ownership of the program is transferred to the QDFTOWN user profile. LODRUN does not transfer control to the QINSTAPP program in this case.

The user supplying the QINSTAPP program is responsible for writing and supporting it. The QINSTAPP program is not supplied by IBM. The program can be designed to accomplish many different tasks. For example, the program could:

- Restore and run other programs or applications
- Restore a library
- Delete another program or application
- Create specific environments
- Apply fixes to existing applications

The QINSTAPP program is run only once each time the LODRUN command is entered. LODRUN passes only one parameter (DEV), which specifies the device from which the QINSTAPP program is restored. The QINSTAPP program transferred to the LODRUN command should not attempt to use LODRUN again. This will have unpredictable results.

In addition to writing the QINSTAPP program, the user supplying the program is responsible for providing the user with the media containing the program. To save the program to a tape or diskette:

1. Prepare the tape or diskette. For tape, use the Initialize Tape (INZTAP) command. For diskette, use the Initialize Diskette (INZDKT) command with FMT(*SAVRST) specified.
2. Use the Clear Library (CLRLIB) command to clear the QTEMP library.
3. Use the Create Duplicate Object (CRTDUPOBJ) command to create the QINSTAPP program into the QTEMP library. The program must be the only object in the file that contains it.
4. Use the Save Object (SAVOBJ) command to save the QINSTAPP program from QTEMP to the desired tape device or diskette unit. Specify LIB(QTEMP), LABEL(*LIB), and CLEAR(*ALL). Specifying LABEL(*LIB) ensures that the label will be QTEMP. If the QINSTAPP program is being saved to a tape device, ENDOPT(*LEAVE) must also be specified. The correct value for the TGTRLS parameter must also be entered if the target release is not the default, *CURRENT.
5. Use the SAVOBJ or SAVLIB command to save any other necessary applications, programs, or libraries to the tape or diskette. This step is optional and is used to

save applications that the QINSTAPP program restores to the user's system when the LODRUN command is run.

If the QINSTAPP program is saved to tape, the tape is not rewound after the QINSTAPP program is restored; the application or series of applications that the QINSTAPP program restores to the user's system should be next on the tape.

Required Parameters

DEV

Specifies the I/O device from which the program is loaded.

TAP: The program is loaded from the tape device connected to the system.

DKT: The program is loaded from the diskette device connected to the system.

tape-device: Specify the name of the tape device from which the program is loaded onto the system.

diskette-device: Specify the name of the diskette unit from which the program is loaded onto the system.

Example

```
LODRUN  DEV(TAP01)
```

This command restores the program object from the tape on device TAP01 to the library QTEMP. Control is then transferred to the restored program.



Program Number: 5738-SS1

Printed in Denmark by Bonde's

SC41-0030-02

