LOOKOUT™ 4.5.1 RELEASE NOTES

Welcome to Lookout 4.5.1. This edition of the Lookout HMI/SCADA package is primarily a bug fix and maintenance release. However, Lookout 4.5.1 adds two new objects: the Logger object and the IDEC protocol driver object.

If you are upgrading from Lookout 4.0, read the *Changes in Lookout 4.5* section of this release note.

Upgrading to Lookout 4.5.1 from Lookout 4.0 and Lookout 3.*x* is the same as upgrading to Lookout 4.5. Refer to the *Installation* section for more information about upgrading to Lookout 4.5.1. Refer to the *Compatibility and Conversion with Lookout 4.5* section for more information about compatibility issues.

The *Lookout 4.5.1 Release Notes* contain the following information:

- Requirements for running Lookout 4.5.1
- New features and corrections that are not in the print documentation
- Known issues with Lookout 4.5.1 and Lookout 4.5
- A roadmap to where you can find answers to your questions about Lookout 4.5.1 and Lookout 4.5

For the most recent Lookout news and updates, point your browser to `http://www.ni.com/lookout/`

Contents

Requirements

Lookout requires the following:

- Pentium class PC or equivalent, running at 90 MHz or faster
- Monitor display setting of at least 800 × 600 pixels
- Windows 98/95/Me, Windows NT version 4 or later, or Windows 2000. National Instruments strongly recommends using

Windows NT version 4 or later or Windows 2000 for optimum stability for your Lookout applications.

- At least 32 MB RAM
- Between 17 and 112 MB free disk space for Lookout (depending on your installation options), plus as much 100 MB or more for data logging (depending on how much data you intend to log to the Citadel database). You should also have approximately 50 MB of disk space for file swapping on Windows NT/2000 computers.
- Network card and TCP/IP networking installed on the computers you want to connect, if you intend to take advantage of Lookout networking
- Adobe Acrobat Reader version 4 or later, which you can install from the `Lookout\Documentation` folder (launch `rs40eng.exe`) or from the Adobe web site

While minimum requirements for running Lookout will run Lookout processes, you may find that large processes which draw heavily on computer system resources will benefit from faster computers that have more RAM and disk memory than the minimum requirements. For example, if your Lookout processes 25,000 traces or more, you should use a Pentium class 450 MHz computer with at least 256 MB of RAM.

Because Lookout can run 24 hours a day, your computer should have some form of AC power surge protection. An uninterruptible power supply (UPS) provides the ultimate protection. A UPS provides complete isolation between the AC power source and the computer and has backup battery power if there are blackouts and brownouts. A quality surge protector will protect your computer from most electrical surges and spikes if you do not need battery backup.

Caution If you are running Windows 95 with no service pack upgrades, you may experience difficulties in networking with Lookout 4 or greater. If your computer begins to behave erratically after networking with Lookout 4 or greater, you may need to install an updated `kernel32.dll` from Microsoft. You can download this upgrade from Microsoft using the following URL:

`http://www.our-town.com/win95maint/win95maint.htm`

Installation

Installing Lookout is more than just copying files to your computer. You must register Lookout for long term use of your I/O points, and for instances of Lookout that are intended to run server processes, you must also register to activate your client connections. There are also a number of Lookout system configuration options you should set when you install. If you plan on networking with Lookout, you should also test to make sure TCP/IP is present and running on all the computers you intend to use Lookout with over a network.

Avoid installing Lookout 4.5.1 over Lookout 3.8 or earlier. Either uninstall Lookout 3.8, or put Lookout 4.5.1 in a new directory. You can transfer or copy Lookout 3.8 process files for converting. However, you can run Lookout 4.0 and Lookout 4.5.1 on the same computer. If you were using a pre-release version of Lookout 4.5.1 you should uninstall before installing the release version of Lookout 4.5.1.

Note You can run Lookout 3.8 on the same computer with Lookout 4 and higher, and a process in one can be networked to a process in the other using NetDDE. Start the earlier version of Lookout first, though, if you intend to run both versions on one computer.

Refer to Chapter 1, *Installing Lookout*, in the *Getting Started with Lookout* manual for more information about installation, testing TCP/IP networking, and configuring the Lookout system options.

Registration

If you are replacing your current copy of Lookout with Lookout 4.5.1, and your previous version was purchased from National Instruments, your earlier registration keycode will work to maintain your I/O licensing. You must register your client connection levels separately, however. Get your client connection keycode (and an upgraded I/O point keycode as well, if necessary) by registering Lookout with National Instruments, using the *Lookout Registration Form* included in your Lookout package.

You must register your copy of Lookout to have full functionality of the software and access to all your I/O points and client connections. Your I/O points and client connections are activated by a Lookout keycode you get when you register the software. Refer to Chapter 1, *Installing Lookout*, of the *Getting Started With Lookout* manual for more information about registering Lookout and getting your keycode.

You can register by phone or fax using the registration form provided with your software, or you can register and get your keycode over the Internet by pointing your browser to:

`http://www.ni.com/keycode`

If you have not already created your personal profile on NI web, you will have to do so before you can register Lookout and get a keycode across the web.

Corrections and Additions to Lookout 4.5.1 Documentation

This section contains information and corrections which are not in the Lookout manual set.

In Lookout 4.5 the parameter documentation for Lookout object classes was located in the *Lookout Object Reference Manual* PDF. That manual is now obsolete.

Parameter documentation has been incorporated in the *Lookout Help*. A link to the parameter documentation for each object is located in the main topic for that object class. You can also find the parameter documentation table for each object in the *Lookout Help* index by searching for the object name, such as Modbus and Modbus MOSCAD Parameters.

Logger Object

The new Logger object is a unique object class in Lookout used for logging under special circumstances.

Each Logger object records data from one I/O point or data source. Unlike logging with most Lookout objects, you do not have to configure the database to log data with the Logger object. The **Edit Database** dialog box is disabled for this object.

You can set the Logger object to log on command, or to start and stop continuous logging. The two modes make different assumptions about your connections.

When you use the intermittent, **`LogNow`** data member, the Logger object can accept periodic data transmissions from remote hardware that is not connected continuously to the computer running Lookout. For example, an RTU may accumulate data for some time before connecting to your Lookout process, at which time the RTU would then report the accumulated data acquired since the last reporting time. This data is logged as one trace.

You can also use the Logger object to log continuously. Use an expression in the **`LogContinously`** data member field to customize deadbanding instead of using Lookout's built-in capabilities.

When you create a Logger object, a dialog box with several expression fields appears. The Logger object has three parameters: **Resolution**, **Lifespan**, and **Alarm Priority**. The rest of the settings in the **Create New Logger** dialog box expose some of the Logger object data members. When you fill out a field in the dialog box, you are making a connection to the corresponding data member.

When you log data using a Logger object, you can associate each piece of data with a timestamp you specify by using the `TimeN` data member. Refer to the *Discrete Logging with the Logger Object* section of these release notes for more information about associating a timestamp with each piece of data.

The Logger object logs discretely when a signal connected to the `LogNow` data member goes TRUE. The Logger object also logs continuously, starting when a signal connected to the `LogContinuously` data member goes TRUE and stopping when that signal goes FALSE. The Logger object time stamps data using either the `TimeN` data member or the system clock, depending on how you configured and connected the object. Refer to the *Continuous Logging with the Logger Object* and *Discrete Logging with the Logger Object* sections of these release notes for more information about using the Logger object for discrete and continuous logging.

You can log data continuously or at set times. Insert a logical expression into the **`LogNow`** field to enable point logging. You can use a Pulse object, a condition test, or an update register to trigger point logging.

When the value of the expression in the **`Log Continuously`** field is TRUE, Lookout logs data continuously.

Insert an expression into the **Log a break if this transitions to true (break)** field to log a break in your data. The break appears in data records as a period when there is no data to report. While `Break` is TRUE no logging is done, no matter what the input to `LogNow` or `LogContinuously`.

Note When logging breaks between individual points of data, Lookout does not display point-only data in the HyperTrend using symbols. Lookout displays point data by drawing trace lines between the logged points. If you insert breaks after every point logged, the HyperTrend will not display data graphically except by single pixels.

If you are logging from a computer or system in a different time zone, you can make a time zone adjustment by entering a positive or negative integer in the **Data Source TimeZone offset (+/- hours from local time)** field. This only operates when you have made a connection to the `Time` data member. If there is no connection to the `Time` data member, a value in this field does not alter the default use of system time to timestamp your data.

Enter the desired **Resolution** for your data. This sets the level of precision the logged data. For example, entering .001 will log the data precisely to two decimal places. If the value logged is 4.76952, and the **Resolution** setting is .001, when you retrieve the data it will be reported as $4.769 \pm n$, such that $0.000 \leq n \leq 0.001$.

The Citadel database purges data after a given period of time. Enter the number of days to keep logged data in the **Lifespan** field. Select **Perpetual** to save logged data as long as your computer has memory to save the data. Use a CitadelControl object to archive data if you want to maintain the data indefinitely in another location while keeping disk space used by the database on the local computer to a minimum.

Set the **Alarm Priority** (1–10) for the Logger object. The Logger object generates an alarm any time you log data out of historical order, or when logging fails for some reason.

The Logger object allocates memory based on the number of the data member you assign to each data point. For this reason it is best not to skip numbers when connecting different I/O points to the Logger object. For example, connecting a data point to the `Data10000` data member reserves a large amount of memory for logging, whether the previous 9,999 data members are connected or not. It is best to begin at `Data1` and work up. This is also true of the `Time` data members, which must have the same number as the `Data` data members to which they are connected.

Logging Data Backwards

Logging data backwards historically can cause serious difficulties in retrieving data from Citadel. If you are attaching timestamps or using time zone offsets with the Logger object, make sure that the data logs in the correct time order. Logging data out of time sequence can create difficulties for querying and displaying data from Citadel. If this happens often it can be impossible to retrieve the data.

If you log data out of sequence, the display will show split traces for that data, as shown in the Figure 1.

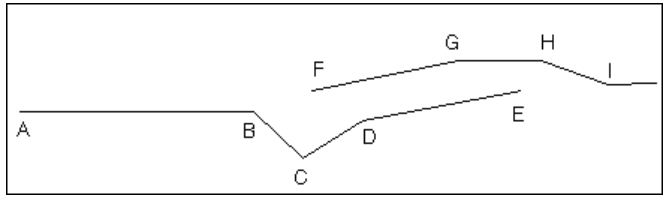

Figure 1. Split Traces from Logging Data Backwards

Points A through E were logged with correct times. Points F and G were logged with timestamps set earlier than points D and E, which were both previously logged. Notice that the HyperTrend will not integrate the data into one trace or display a line connecting E to F.

Continuous Logging with the Logger Object

To log continuously using the Logger object, make only one connection to the `Data1` data member. You can also make a timestamp connection to the `Time1` data member. Lookout logs the data continuously.

If necessary, you can use an Alarm object for any custom alarming you want to do with your data, and (expression) objects for custom deadbanding.

Discrete Logging with the Logger Object

Use the `LogNow` data member to log data at discrete intervals. How you make connections depends on what you are tying to accomplish.

In the simplest case, connect the data you want logged to the `Data1` data member, associate a timestamp if necessary using the `Time1` data member, and activate the `LogNow` data member every time you want the data logged.

You can also perform more intricate logging with data that has been remotely collected over a period of time and stored for delivery in one communications session. For example, you have an RTU that takes a measurement every 15 minutes and stores the results. Once every hour this RTU connects to the computer running Lookout and reports four observations, each with a time stamp for when the data was taken.

To log this data with the Logger object, connect each data register to one of the `Data` data members, preferably `Data1` through `Data4`. Connect the corresponding timestamp information to the `Time1` through `Time4` data members. Once the connection has been established and the data has been transmitted, activate the `LogNow` data member. The Logger object then logs the data as indicated by the associated timestamp data.

The Figure 2 demonstrates how Logger object data is displayed. The top trace shows how data received in batches through `LogNow` appears. The bottom trace shows how data logged through `LogContinuously` or `LogNow` using one data member connection appears.

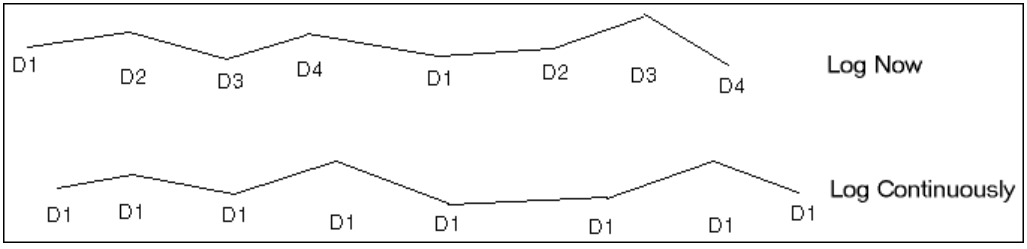

Figure 2. Data Logged with Logger Object

Logger Object Data Members

Data Member	Type	Read	Write	Description
Break	Logical	Yes	Yes	When this value changes to TRUE, the Logger object logs a break in the data trace. The Logger object will not log while this data member remains TRUE.
Data1 – Data*N*	N/A	Yes	Yes	Connect the data you want logged to a `Data` data member. You can log as many I/O points as you want if you have the hard disk space and system memory to support the logging operation. To associate a timestamp with each piece of data, make a timestamp to the `Time` data member with the corresponding number.
LogContinuously	Logical	Yes	Yes	When this value changes to TRUE, the Logger object begins logging data continuously, stopping when the value transitions to FALSE or when interrupted by the `Break` data member.
LogNow	Logical	Yes	Yes	When this value transitions to TRUE, the Logger object logs the data at that time, stopping after logging each connected data point. This works even if you are logging continuously by forcing the Logger object to log data, even if it has not changed.

Data Member	Type	Read	Write	Description
Time1 – Time*N*	Numeric	Yes	Yes	This is the optional timestamp for each data point you choose to log. The default for an unconnected time data member is the time and date of the system clock of the local computer.
TimeZone	Numeric	Yes	Yes	Enter a positive or negative integer to compensate for time zone differences if the data logging is coming from a remote source in another time zone. The Logger object adds or subtracts the indicated number of hours. The default is the time zone of the system clock of the local computer.

Logger Object Parameters

Number	Name	Type	Description
1	Resolution	Numeric Constant	This sets the level of precision for the logged data. For example, entering .001, logs data precisely to 2 decimal places. If the value logged is 4.76952, and the **Resolution** setting is .001, when you retrieve the data it will be reported as $4.769 \pm n$, such that $0.000 \leq n \leq 0.001$.
2	Lifespan	Text Constant	The length of time Lookout keeps data in the Citadel database before purging the data. While there is no real limit to how long to set a lifespan, use a CitadelControl object to archive data for long term storage.
3	Alarm priority	Numeric Constant	Alarm level, 1–10.

IDEC Micro3, IDEC Micro3C, IDEC OpenNet

Use the IDEC driver objects to communicate with IDEC Micro3, Micro3C, and OpenNet PLCs.

Set the **PLC Address** to the address of the device the IDEC object communicates with.

Serial port specifies which COM port on the host computer the Lookout object uses for communicating to the external device. This does not specify

the communication type. Communication type is determined by the **Options»Serial Ports** setting. Refer to Chapter 5, *Serial Port Communication Service*, of the *Lookout Developer's Manual* for more information about configuring serial ports for use with Lookout.

Data rate, **Parity**, **Data bits**, and **Stop bits** reference the settings on the hardware device.

Phone specifies the telephone number dialed if the selected serial port is configured for dial-up. This number only applies to the individual protocol object.

PollRate is a numeric expression that determines how often to poll the device. Lookout polls the device at the specified time interval. Normally, this is a simple time constant such as 0:01 (one second). Refer to the *Numeric Data Members* section of Chapter 2, *How Lookout Works*, in the *Getting Started with Lookout* manual for information about entering time constants.

Poll is a logical expression. When this expression changes from FALSE to TRUE, Lookout polls the device. You can use a simple expression like the signal from a Pushbutton object, or a complex algorithm.

Communication alarm priority determines the priority level of the alarms generated by the AB IDEC object.

Retry attempts specifies the consecutive number of times Lookout attempts to establish communication with a device if it is not getting a valid response. After Lookout tries the number of **Retry attempts** specified, the object generates an alarm and begins to **Skip every n poll requests after comm failure**. Once Lookout reestablishes communications, it polls the device on its regular cycle, as defined by **PollRate**.

Receive timeout is the time delay that Lookout waits for a response from a device before retrying the poll request.

The **Skip every** setting instructs Lookout not to poll a device with which it has lost communication on every scheduled poll. Instead, Lookout polls the device once in the specified number of poll cycles. Once communication is reestablished, the device is polled on the regular cycle.

IDEC Data Members

Data Member	Type	Read	Write	Description
CommFail	Logical	Yes	No	Object-generated signal that is ON if Lookout can not communicate with the device.
Poll	Logical	No	Yes	When this value changes from FALSE to TRUE, Lookout polls the device.
PollRate	Numeric	No	Yes	Lookout expression that determines the device-polling frequency.
Update	Logical	Yes	No	Object-generated signal that pulses low each time Lookout polls the device.
D0 – D7999	Numeric	Yes	Yes	16-bit data registers. All digits are decimal. The range extends to D99 for Micro3 and to D499 for Micro3C.
I0 – I597	Logical	Yes	No	1-bit input operand number. The least-significant digit of the range number is an octal number (0–7). The upper digits are decimal. Consult the IDEC communication protocols documentation for information about using operand numbers. Range only extends to I35 for Micro3 and Micro3C.
M0 – M2557	Logical	Yes	Yes	1-bit internal relay operand number. The least-significant digit of the range number is an octal number (0–7). The upper digits are decimal. Consult the IDEC communication protocols documentation for information about using operand numbers. Range only extends to M287 for Micro3 and Micro3C.
Q0 – Q597	Logical	Yes	Yes	1-bit output operand number. The least-significant digit of the range number is an octal number (0–7). The upper digits are decimal. Consult the Communication Protocols documentation for information about using operand numbers. Range only extends to Q31 for Micro3 and Micro3C.

IDEC Parameters

Number	Name	Type	Description
1	PLC Address	Numeric Constant	PLC address on the network (1)
2	Serial Port	Text Constant	Serial port
3	Baud Rate	Numeric Constant	Baud rate (1200, 2400, 4800, 9600, 19200)
4	Parity	Text Constant	Parity (None, Even, Odd)
5	Data bits	Numeric Constant	Data Bits 0 = 7 1 = 8
6	Stop bits	Numeric Constant	Stop bits 0 = 1 1 = 1.5 2 = 2
7	Phone Number	Numeric Constant, NULL	Phone number
8	Alarm priority	Numeric Constant	Alarm level, 1–10
9	Retry attempts	Numeric Constant	Retries before communications alarm
10	Receive timeout	Numeric Constant	Timeout in milliseconds
11	Skips	Numeric Constant	Number of poll requests skipped after a communication failure

IDEC Status Messages

No response from device
Lookout received no response from the device within the **Receive timeout** period. The object was able to establish a connection, but when it sent a message to the device, the device did not respond. Try increasing **Receive timeout** (and **Poll Rate**) to ensure Lookout allows enough time to receive the expected response. Also, verify the cable connections, serial port wiring, power supply, configuration settings, and IP settings.

Garbled Message
A response was received from the PLC, but the response could not be interpreted.

Wrong PLC Address
Check the **PLC address** setting in the object properties of the IDEC object.

The following alarm messages are generated by the PLC. Consult the IDEC hardware documentation for more information about resolving these alarm conditions.

Improper write/read program size.
Protected against write/read in MICRO97.
Writing user program is attempted while MICRO97 is running.
User program CRC code does not match.
Protect code in the request message does not math that set in FUN27.
Invalid data range designated.
Preset value change is attempted to timer or counter with preset value designated by data register.
Invalid value written to calendar/clock.
Designated data cannot be cleared.
Invalid data other than 0 (30h) - 9(39h) or A(41h) - F(46h).
Incorrect setting for user communication.
Too many write commands in one scan. Only three write commands are accepted in one scan.
Appended BCC code does not match BCC calculated value of received data.
Quantity of received bits differs from the preset value (stop bit is 0 for example).
Parity error or overrun error occurred.
Unsupported request message is received.
Received request message does not match the expected data (include quantity of data).

New FieldPoint Pulse Generator Module Data Members

Please refer to the FP-PG-522 operating instructions for a detailed description of the features and functions available for this module.

Note Lookout supports channel commands, but you must still use the FieldPoint Explorer software to configure FieldPoint modules.

Data Member	Type	Read	Write	Description
CommFail	Logical	Yes	No	Object-generated signal that is ON if Lookout can not communicate with the device.
OffHook	Logical	No	Yes	When TRUE, this flag instructs the FieldPoint object to retain exclusive use of the assigned communication port.

Data Member	Type	Read	Write	Description
Poll	Logical	No	Yes	When this value changes from FALSE to TRUE, Lookout polls the device.
PollRate	Numeric	No	Yes	Lookout expression that determines the device-polling frequency.
Update	Logical	Yes	No	Object-generated signal that pulses low each time Lookout polls the device.
PGPG000.00 – PGPG255.15	Numeric	Yes	No	Number of pulses yet to be generated after the current pulse.
PGPS000.00 – PGPS255.15	Logical	Yes	No	Boolean value that indicates the state of the physical outputs. When TRUE the pulse is ON. When FALSE the pulse is OFF.
PGSI000.00 – PGSI255.15	Logical	No	Yes	Stop pulse generation immediately.
PGSA000.00 – PGSA255.15	Logical	No	Yes	Stop pulse generation after the current pulse.
PGGP000.00 – PGGP255.15	Numeric	No	Yes	Number of pulses to generate. Works only in finite mode. This input also functions as a trigger: when you change this input, the module begins generating the number of pulses set and continues to the last pulse. Use the FieldPoint Explorer to set the pulse generation mode to finite or continuous.

New FieldPoint Quad-510 Module Data Members

Refer to the FP-Quad-510 operating instructions for a detailed description of the features and functions available for this module.

Note Lookout supports channel commands, but you must still use the FieldPoint Explorer software to configure FieldPoint modules.

Data Member	Type	Read	Write	Description
CommFail	Logical	Yes	No	Object-generated signal that is ON if Lookout can not communicate with the device.
OffHook	Logical	No	Yes	When TRUE, this flag instructs the FieldPoint object to retain exclusive use of the assigned communication port.
Poll	Logical	No	Yes	When this value changes from FALSE to TRUE, Lookout polls the device.
PollRate	Numeric	No	Yes	Lookout expression that determines the device-polling frequency.
Update	Logical	Yes	No	Object-generated signal that pulses low each time Lookout polls the device.
QDPC000.00 – QDPC255.15	Numeric	Yes	No	Reports position counter input.
QDVD000.00 – QDVD255.15	Numeric	Yes	No	Velocity data registers. A positive number indicates forward motion. A negative number indicates reverse motion.
QDIS000.00 – QDIS255.15	Logical	Yes	No	Index status of channels.
QDCR000.00 – QDCR255.15	Logical	No	Yes	Control reset command. Resets the target counter.

New Phoneducer Data Member

A second data member, `Value2`, has been added to the Phoneducer object class in order to accommodate a new model of the Phoneducer that returns two values.

Lookout Function Corrections

The following changes have been made to Lookout functions.

Sum Function

The Sum function is documented in the *Statistical Functions* section of the *Lookout Developer's Manual*. It should be listed in the *Mathematical Functions* section.

Using the Round Function

The Lookout Round function has been enhanced.

`ROUND(numeric1, numeric2)` rounds the value of `numeric1` to the value of `numeric2` decimal places. For example, if Pot1 equals 234.745 and `numeric2` is set to 2, the function returns 234.75. If `numeric2` is set to zero, the function returns a 234. If `numeric2` is set to a negative number, the function rounds-off the result to the corresponding place to the left of the decimal. For example, if `numeric2` is set to -2 and Pot1 still equals 234.745, the function returns 200.

Left, Mid, and Right Functions

The text functions Left, Mid, and Right have error cases that are not documented.

Use positive integers with the Left, Mid, and Right text functions. If you use a numeric value less than 1, Lookout produces the following results.

- Left function—If you set the **numeric1** parameter to less than 1, Lookout parses the input as if you entered 1. If you set the **numeric2** parameter to less than 1, Lookout returns an empty string.
- Mid function—If you set the **numeric1** parameter to less than 1, Lookout parses the input as if you entered 1. If you set the **numeric2** parameter to less than 1, Lookout returns an empty string.
- Right function—If you set the numeric parameter to less than 1, Lookout returns an empty string.

Replace Function

The documentation for the Replace text function does not specify what happens with incorrect or unusual numeric parameters. Use positive integers with the Replace text function and set the **numeric2** parameter to an integer larger than the integer value for **numeric1**. If you use numeric values outside of these limitations, Lookout produces the following results.

- If **numeric1** is larger then the length of **text1**, Lookout appends **text2** to the end of the **text1** string.
- If **numeric1** is set to a value less than 1, Lookout parses the value as if you entered 1, and the replacement string starts at the beginning of the **text1** string.

Fixed Function

The Lookout text function Fixed now rounds-off differently when you do not enter a value or use a negative number for the **numeric2** parameter.

- If you do not enter a value for **numeric2**, Lookout rounds-off **numeric1** to the nearest whole number.
- If you set **numeric2** to a value less than 1, Lookout rounds **numeric1** to the corresponding place to the left of the decimal.

Examples:

Fixed(123.456) = 123

Fixed(123.456,0) = 123

Fixed(123.456,–1) = 120

Fixed(129.456,–1) = 130

ODBC Data Source Defaults

When you install Lookout, the installer creates an ODBC data source for the default location of your Lookout data in the Citadel database. If you install Lookout in a custom location, or log data to a database located somewhere other than the installation default location, you will have to create an ODBC data source for that location.

You can create a data source using the Windows ODBC Data Sources utility, located in the Windows Control Panel. Refer to the *Creating a Citadel ODBC Data Source* section of Chapter 8, *Structured Query Language*, of the *Lookout Developer's Manual* for more information about creating an ODBC data source.

New Modbus Advanced Configuration Option

By default, the Modbus object uses function code 6 for a single write. Check the **Use fc16 for single write** checkbox in the **Advanced Modbus** configuration dialog box to force the Modbus driver to use function code 16 for a single write.

Lookout Keyboard Driver

(Windows NT/2000) Install the Lookout keyboard driver to block users from performing certain keyboard shortcuts that interact with the computer operating system, such as <Ctrl-Alt-Delete> to reboot the computer or <Alt-Tab> to switch between applications. The Administrator can set keyboard shortcut security by selecting **Options»System** and checking the appropriate checkboxes in the **If security level is below** field.

The Lookout keyboard driver may not work with certain specialty keyboards, such as a touch screen. To avoid this problem, installing the Lookout keyboard driver is now an option in the Lookout installation process.

If you use a non-standard keyboard with a computer running Lookout, and you have keyboard difficulties, uninstall the Lookout keyboard driver. To uninstall the Lookout keyboard driver, open the Windows Control Panel (**Start»Settings»Control Panel**) and open Add/Remove Programs. Select NI Lookout from the list of programs and follow the dialog box instructions to uninstall the keyboard driver.

The Lookout installer now notifies you to install the keyboard driver to use the Lookout keyboard security feature.

Using the Lookout Web Client

It is necessary to have the a client version of Lookout or the Lookout Player and the correct version of the Lookout ActiveX control to use the web client feature. Lookout exports the ActiveX control when exporting a client process to the web.

Lookout ActiveX Control

Version compatibility problems of the ActiveX control and more recent versions of Lookout have prevented the web client feature from functioning correctly. Lookout 4.5.1 corrects version problems with using the Lookout web client feature.

When you view the web page for an imported file, version information of the ActiveX control used to read the file is included in the web page. The version information communicates the version of the ActiveX file installed in the web server directory of the server computer when Lookout exported the web client file to a web page.

When the browser of the client computer points to a Lookout-exported web page, the browser will check for the version of the ActiveX control specified by the web page. If the client computer does not have the specified version of the ActiveX control installed, you will be prompted to download the correct version. Download the correct version of the ActiveX control to view the exported file as a web page. Multiple versions of the ActiveX control can coexist on a client computer.

Lookout Player

The Lookout Player can also be a necessary component of connecting to a web server.

If the client computer from which you view the exported file as a web page does not have Lookout installed, the user must download the Lookout Player. This is a separate download, but the user will be notified by a dialog box and can link to the download site from the web page.

The default URL for the Lookout Player is on the National Instruments `ftp://ftp.ni.com/support/lookout/webclient/4.5/lk45_web.exe`. If your company intranet is not connected to the internet or if you have a slow internet connection, you can specify a local source for the Lookout Player in the **Lookout Player download location** field of the **Web server export** options dialog box. Check the **Use Default** checkbox to restore the URL address to the default location at the National Instruments `ftp://` site.

(Windows NT/2000) You must have Administrator access on your computer to run the Lookout Player installer.

To ensure that you have the most recent version of the Lookout Player available, use the Lookout Player in the `Lookout Player` directory of the Lookout installation CD, or download the `lk45_web.exe` file and place it in a local server directory.

Exporting Multiple Web Server Processes

When you export web server files, you can select more than one process to export from the process currently running in Lookout. When you connect to the resulting web page, all of the exported processes are available to the web client.

Creating Lookout Client Processes for Ease of Maintenance

Use the SymbolicLink object for connections in client processes to increase the portability and decrease the maintenance issues for the process.

In Chapter 5, *Developer Tour* of *Getting Started with Lookout,* it is suggested to use remote source URL connections in client processes. Using remote source URL connections also resolves maintenance and portability issues when modifying a client process.

If you create a SymbolicLink object or set of SymbolicLink objects in a client process that reference a server process, you can make URL connections in the Symbolic Link object. Then you can redirect all of the

connections made through that Symbolic Link with a single edit to the SymbolicLink object, instead of editing multiple URL connections.

Minimum and Maximum Settings for Digital Displays

In earlier versions of Lookout, you could enter minimum and maximum values for digital displays as well as for bar displays. Setting a minimum and maximum value is now enabled exclusively for bar displays. A digital display displays the numerical value of the corresponding data member.

Lookout Security: Exit Control

Lookout 4.5.1 has a new security setting in the **System Options** dialog box. Now there is an option to set a security level for permission to exit and close Lookout. A user with a security level below the security level set in the **System Options** dialog box can not quit Lookout. To set exit control security in Lookout, select **Options»System** and set the desired security level in the appropriate field of the **If security level is below** section. The default security setting for this feature is 8.

Maximum Latency in Lookout 4.*x*

Appendix C, *LOOKOUT.INI File*, of the *Lookout Developer's Manual* contains an entry for maximum latency. This was a setting used by Lookout 3.*x*. It is not used by Lookout 4.*x*, and can be disregarded.

Resolution Setting of the Lookout Pot Object Class

The documentation for the Lookout Pot object class indicates that the **Resolution** setting is the smallest increment of change or detent spacing the Pot object supports. This is a potentially misleading statement, because the Pot object resolution starts from zero, not the minimum value.

For example, if you create a Pot object with a resolution of 10, and a minimum value of 12, increasing the value of the Pot object one notch, or increment, results in a value of 20, not 22.

Using an expression to process the output of a Pot object is the best way to implement non-standard outputs and resolutions.

Modbus Ethernet Diagnostic file setting

Lookout 4.5.1 adds diagnostic capabilities to Modbus Ethernet objects.

To enable creation of a diagnostic file for a Modbus Ethernet object, edit the `modbus.ini`, located in the `Lookout` directory file by adding a `DiagnosticPath` key to the `[ethernet]` section. Set the key equal to the path to the location where you want to create the diagnostic file when you

run Lookout. When enabled, this file contains information you can use to debug Modbus Ethernet object problems or errors.

Known Issues

The following known issues exist for Lookout 4.5.1. Internal tracking codes are shown in parentheses for applicable issues.

FieldPoint Alias with Japanese Characters

Lookout crashes when attempting to import FieldPoint aliases using Japanese characters. Lookout does not support international character sets at this time. Attempting to import Japanese characters from FieldPoint configuration files will fail. (21K5CU6S)

Administrator Access Needed to Download Lookout Player

If you attempt to use the Lookout web client feature on a computer running Windows NT/2000, you must have administrator privileges to install the Lookout Player component of the web client.

You must have administrator privileges to install software on a Windows NT/2000 computer. To use the Lookout web client, you must have a client version of Lookout or the Lookout Player installed. If the Lookout player or a client version of Lookout is not already installed on the computer, then the Lookout ActiveX component of the web client will attempt to do this the first time you try to access a web server process on that computer.

If the person using the web client on a computer does not have administrator privileges on that computer, arrange for someone with administrator privileges to log in and access the web server the first time to download the Lookout player. After the Lookout Player is installed any person who can use the web browser can use a Lookout web client successfully.

Using ActiveX Controls in Web Client Processes

At this time the only ActiveX controls you can use in a process you intend to export for use by a Lookout web client are the National Instruments ActiveX controls provided as a part of Lookout.

Lookout Control Sources and Refresh Value Operations

Under some circumstances, a bug in an early version of Lookout 4.0 resulted in certain unpredictable or incorrect behaviors with respect to remote source URL connections. Additionally, the circumstances in which Lookout refreshed a value were not previously documented. This section describes the corrected behavior in Lookout 4.0.1 and greater.

Lookout has two basic connections: URL connections for remote source of a control, and direct connections, made with the **Edit Connections** dialog box for local source controls.

A Lookout control using a remote source URL connection to a PLC or object takes its value from that PLC or object, so you can think of the PLC or object as "owning" the value. If the PLC or object changes, the control with the remote source connection changes to reflect the new value or status. When the remote connection is made to field hardware, Lookout refreshes the control's value when the driver polls the hardware.

If you operate a Lookout control with a remote source connection, and the PLC register it is connected to cannot or does not change when Lookout writes the new value out to the PLC, the control changes back to display the actual PLC register value or status. The time this takes depends on data transmission time between your Lookout computer and the field hardware. Refer to the discussion of the `snapDelay` data member for the Switch object in the Lookout Help for more information about using the `snapDelay` data member.

A Lookout control with a local source and a direct connection (made through the **Edit Connections** dialog box) "owns" the value held by the control. The control value will not change if something acts to alter the value or status of the PLC or object it is connected to, and Lookout will restore the control's value to the PLC or object in the following circumstances:

- When the driver object is modified
- When the process is reloaded
- After communication is restored after a communications failure
- After 100 polls in some cases (this varies from driver to driver and may depend on INI settings)
- Any time I/O reconfigures (meaning any time you add or remove a read or a write to a driver)

Notice that these rules apply to Lookout client and server connections. Clients should use remote source URL connections to objects in a Lookout server, so that the server process will "own" the values, at least relative to the client (even though the server process may itself use a remote source URL connection to field hardware).

The bug fix for this problem involves both Lookout and the individual driver object CBX files. While Lookout and the most frequently used driver object classes have been fixed, the following drivers still need this fix incorporated.

AdvantechPCL	DeltaTau	Dynamic
ProfibusDP	ProfibusL2	Reliance
Sixnet	Wizdom	

There are two Lookout drivers which cannot support remote source connections because the underlying protocols themselves do not allow this sort of connection. These are the Aquatrol and the DeviceNet drivers. Aquatrol is unique because every time you polls the RTU, you write to all of the outputs. The DeviceNet Explicit Messaging driver does, however, support remote source connections, and the current DeviceNet Explicit Messaging driver does work properly with this type connection.

Check the Lookout download site at

`ftp://ftp.ni.com/support/lookout/Lookout4fixes/`

for new versions of these CBXs, which will be posted as soon the fix is implemented in each.

Security File Compatibility

Lookout security files are backward but not forward compatible. In other words, if you edit or import a `lookout.sec` or a Lookout `.lka` file from an earlier version of Lookout with Lookout 4.5 it will work properly. You will not be able to use any converted files with the earlier versions of Lookout, however.

Connection Browser Open with No Processes Visible

Occasionally the Connection Browser may open with no processes visible in the window. Close the Connection Browser and then reopen it. This will usually clear the problem. If for some reason it does not, close the Connection Browser again and make sure you select a process in your Object Explorer before reopening the Connection Browser.

Bug Fixes

The following bugs have been fixed for Lookout 4.5.1.

The Lookout 4.5 web client export feature reported the error, `Cannot find this file for exporting` if your server process includes (expression) objects. This limitation has been fixed. (23GI49L)

Some Tiway PLC models were generating an alarm stating `TIWAY host command error response 0009: Incorrect amount of data sent with request.` This alarm was stimulated by an attempt to read a certain register and created a communications failure. The underlying problem has been corrected. (23TEJJ9L)

The `WR` and `DR` data members for the Hitachi driver object had incorrect data ranges. These ranges have been corrected to `WR0-WRF1FF` and `DR0-DRF1FE`.

If you have a Switch or Pushbutton with a remote source URL connection to the Mailer object, the Mailer object sent mail twice instead of once. This was a problem with the Lookout Switch and Pushbutton objects, and has been corrected. (24BG149L)

If you pressed <Delete> while renaming an object in the Lookout Object Explorer, Lookout would delete the object. This inconvenient behavior has been corrected. If you press <Delete> while renaming an object, Lookout now only deletes characters of the object name. (244FO29L)

In versions of Lookout prior to 4.5.1, a Lookout client process could hang if it had too many real-time points (approximately 3000) connecting to the server. This was due to a message queue limit in Windows 9*x*. This limitation has been fixed. (24A9DFE9)

Under some circumstances a single-quoted data member in the Lookout `.lks` file was crashing the Lookout CBL compiler. This problem has been fixed. (244B4Q9L)

Beginning with Lookout 4.5.1, you can keep and use both Lookout 4.0 and Lookout 4.5.*x* on the same computer.

In Lookout 4.5, especially in an Aggregate object, a Pushbutton object with a remote connection to the `minimize` data member of a panel other than the panel where the Pushbutton was located failed to work. This has been corrected. (24RC41MV)

Under some circumstances, when running a process using the Lookout Recipe object in a run-time version of Lookout, you could import a new recipe without the recipe changes being preserved in the Lookout state file. This difficulty has been corrected When you load a new or corrected recipe file into a Lookout process under run-time conditions or when the operator does not have security permission to save a process, Lookout automatically preserves the new recipe values permanently. (23CE729L)

Lookout 4.5 did not always properly display the ampersand (&) character in text plates. This has been corrected for Lookout 4.5.1. (24S5S6CM)

Alarms from the DataTable object were truncated and contained non-readable characters. Alarm reporting now enables any length of alarms to report properly. (24BB5R9L)

The Lookout driver for the Siemens TI505 works for the CP1413, not the CP1431. This typographic error has been corrected in the Lookout Help. (24QB959L)

The time quality of data from the Lookout TESCO driver would read as bad after a comm failure, even when communications were restored. This problem has been fixed. Time quality now reports accurately after a comm failure. (22NH8D26)

There was a fixed alias-scaling problem. You can now configure filtering and scaling for logical and numeric data types to maintain compatibility with Lookout 3.x processes. Lookout will scale either a numeric or logical data type, based on the data type of the signal. National Instruments revised the earlier GUI change that prohibited the user from configuring logical and numeric scaling at the same time. (21KEI6MO) (266FPSMO)

The Lookout control sources and refresh values bug has been fixed for the following Lookout drivers in Lookout 4.5.1: Fisher, Phillips, RKC F series, S5_3964, Modbus Omni, Hitachi, Cutler-Hammer. Refer to the *Lookout Control Sources and Refresh Value Operations* section for more information about this bug.

The Fisher protocol driver object now monitors time quality for data.

If you used a Sequencer object that jumped from state 100 to any other state in Lookout 4.5 and some builds of 4.0.1, Lookout crashed. This problem has been fixed in Lookout 4.5.1 (26BFIRD8)

The Lookout **Expression Editor** dialog box often appeared with buttons hidden under the Windows task bar. This annoying bug has been fixed. (1QPC7FEQ)

The Gauge object did not have the graphic data member so it could be used in a Web Report page. Also, data quality reporting was not implemented for the Gauge display. Both these problems have been fixed. (25OFHABD) (25N9KVMV)

When a connection between Lookout and an ODBC source was lost, the DataTable object sometimes failed to reconnect to the ODBC data source. This problem has been fixed for Lookout 4.5.1. (24BB3S9l)

In Lookout 4.5 an (expression) object, or named expression, entered into an Aggregate object definition process, failed to display data properly when you created an Aggregate object using that definition. This limitation has been fixed for Lookout 4.5.1. (25GG0EE9)

When a connection between Lookout and an ODBC source was lost, the SQLExec object sometimes failed to reconnect to the ODBC data source. This problem has been fixed for Lookout 4.5.1. (24BB3S9l)

A problem with font size and screen redrawing developed between Lookout 4.0.1 and Lookout 4.5 that sometimes led to some indicators displaying a row of # signs when a process was opened in Lookout 4.5. This problem has been fixed for Lookout 4.5.1. (2528F4TS)

In Lookout 4.5, a DDE link between two DataTable objects did not properly update the second table when the first table was changed using ODBC or an XLS file. This problem has been fixed for Lookout 4.5.1. (24RB1H9L)

While upgrading Lookout 4.0.1 processes to Lookout 4.5, Lookout sometimes crashed if an interface object had a remote source connection to another Lookout object in non-alphabetical order. This problem has been fixed for Lookout 4.5.1 (25I99T26)

A communications error between Lookout and the model 565 Tiway PLC resulting in the error message, `0011:Invalid data sent with the command`, has been fixed for Lookout 4.5. (251HP67M)

When you had graphics with identical names placed in more than one location in the Lookout Graphics folders, compiling a Lookout process sometimes resulted in a crash. This has been fixed for Lookout 4.5.1. (25M6DJTS)

Lookout did not allow a user with a security level lower than 8 to exit Lookout. You can now configure this option in the Lookout **System Options** dialog box. (24DBPQ21)

A problem causing the Lookout web client to crash when you refreshed you browser has been fixed. (25TA83AO)

The Mitsubishi CLM driver switched the address bytes such that when you attempted to communicate with station 1, you were actually communicating with station 10 (hexadecimal; binary station 16). This error has been fixed for Lookout 4.5.1. (2608PUD8)

Applicom FMS and the Applicom board did not provide a sub0 bit. Applicom numbering starts sub1. The Lookout Applicom FMS object has been changed to reflect this numbering system for Lookout 4.5.1.

Under some circumstances, Lookout crashed when you pasted a string into a yellow expression field using <Ctrl-P>. This bug has been fixed for Lookout 4.5.1. (264GLBAO)

If you experienced difficulties with the Mailer object, you can now use the hidden data member, `tracefile`, to look for problems. Connect a filename to this data member and attempt to use the Mailer object. Messages from your mail server log in the file you designated.

Changes in Lookout 4.5

If you are upgrading form Lookout 4.0, read the following sections to learn about the features added for Lookout 4.5.

- ActiveX control object—Include any ActiveX control in your Lookout processes, including ten versatile controls from National Instruments ComponentWorks included with Lookout 4.5.
- Aggregate objects—Create Lookout processes you can then use and reuse like objects, without tedious and repetitive rebuilding of systems and processes you have built before.
- Citadel Control—Archive and maintain your Citadel databases from within a Lookout process.
- HTML reports—Generate HTML reports from Lookout that contain tables and charts, text, your own HTML code, and more.
- DataSocket object—Connect to a wide range of National instruments Hardware and software using National Instruments DataSocket technology.
- Mailer object—Distribute Lookout data or alarms through your e-mail system.
- Mouse and Joystick objects—Use mouse and joystick inputs in your processes.
- New Driver objects—Including drivers for Allen-Bradley Logix, Fatek MA/MB/MC, Mitsubishi CLM, Modbus Daniel, Modbus Omniflow, National Instruments DeviceNet, Phoneducer, ScanData, and Siemens S7-HMI.
- Image Navigator—A new, large library of graphics, from symbols to detailed drawings.

Enhancements to usability and functionality, including a new color selector with 24-bit color capability, multiline text entry and display, live data display for connections in the Connection Browser, and Object Explorer enhancements such as object copy and paste, and easy access to the **Edit Database** and **Edit Connections** dialog boxes both through objects and individual data members.

Corrections and Additions to Lookout 4.5 Documentation

This section contains information and corrections which are not in the Lookout manual set.

CitadelControl Object

You can use the CitadelControl object to archive and restore your Citadel databases. The CitadelControl object archives data from any Citadel database (the source database) to any other Citadel database (the target database). You can specify what data is archived, and set the time range for the operation.

You can automate archiving so that it takes place on a regular basis, and you can use multiple CitadelControl objects to archive data to several backup computers across the network. You can use CitadelControl to archive data from several Citadel databases into one archive database, and you can archive specific data from an archive database out to special purpose databases or back to active databases to restore data that has been corrupted or lost.

To create a CitadelControl object, select CitadelControl from the **Select Object Class** dialog box (from the **Logging** category). The Citadel object properties dialog box appears as shown in Figure 3.

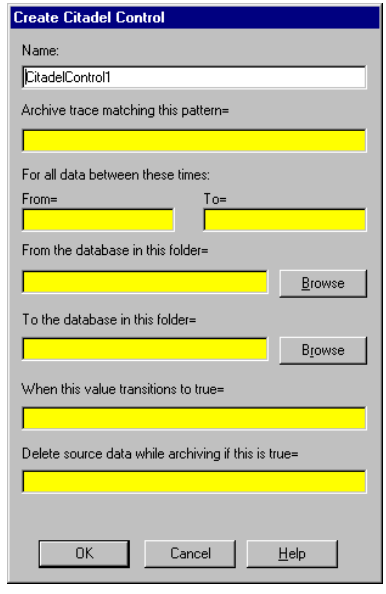

Figure 3. Citadel Object Properties Dialog Box

Note Unlike most Lookout object classes, the CitadelControl has no parameters. All the yellow expression fields on this dialog box are actually connections to CitadelControl data members. You can create a CitadelControl object without filling any of the yellow expression fields, supplying the necessary values through a direct or remote source connection later in your development efforts. If you do enter an expression in one of these fields when you create the CitadelControl object and later make a connection to that data member from an object other than the one you used when creating the CitadelControl, you will discover that your new connection replaces the original one if you edit the object's properties through the dialog box. In the same way, any entries you make when you create a CitadelControl object will appear when you edit connections to that object.

Use **Archive trace matching this pattern** (the `What` data member) to select the data you want copied. You must supply a Lookout data path (with the appropriate relativity) for the data you want to archive.

You cannot use path abbreviations for data you want to trace. You must include the computer name or use the appropriate wildcard. In other words, `"\\paroikos\PSMonitor\Modbus.1"` is a valid path for a data trace, while `"\\.\PSMonitor\Modbus.1"` is not.

You can use wildcards and pattern matching to select a large number of traces from a given Citadel database, however. Refer to the *Pattern Matching* section for more information about how to define the data you want copied.

Note As with any other string you use in a Lookout expression, you must use double quotes to delineate what you enter for the **Archive trace matching this pattern** (the `What` data member). If you connect a TextEntry object to this data member, however, you need not use double quotes because Lookout inserts them for you when you enter a string through the TextEntry object. This is a great advantage if you want to change the traces you archive. On the other hand, if you use the yellow expression field in either the **Create CitadelControl** dialog box or in the **Edit Connections** dialog box, you can browse to find the data you want to archive.

Set the **From** and **To** fields (the `FromWhen` and `ToWhen` data members) to define the time range for the data you want to archive. Setting either field to `0` extends the range to the limits available: a `0` in the **From** field means the CitadelControl will archive data from the beginning of the database. A `0` in the **To** field means that the CitadelControl will archive data to the most recent data available.

You use **From the database in this folder** (the `FromWhere` data member) to select the Citadel database you are archiving data from. The data will be copied to the database you set in **To the database in this folder** (the `ToWhere` data member). When you set these fields as parameters in the Create New CitadelControl dialog box, you can browse your directory

structure to select the Citadel directory you want to set as your source or target database.

When this value transitions to true (the `Start` data member) can contain or be connected to a logical value—either a Lookout control or a Lookout expression that evaluates to TRUE or FALSE. You can archive data regularly using a Lookout Pulse object to trigger operation of CitadelControl.

Note The CitadelControl only archives new data. It does not copy over data it has already archived in the target archive. Additionally, if you cancel archiving for some reason, the data already archived when you activate the `Cancel` data member stays in the archive. The CitadelControl will then take up archiving where it left off the next time you start archiving (assuming you have not changed the other settings).

You can set **Delete source data while archiving if this is true** (the `DeleteEnabled` data member) either as a constant or a variable. When this parameter is TRUE, Lookout deletes information from the source database when the CitadelControl copies it to the target database.

Archiving Data

The primary use for CitadelControl is for archiving and backing up data.

When you archive data with CitadelControl, it copies data from your source directory to your target directory. Your target and source directories can be on your local computer or on a computer located somewhere else on your network (as long as the remote computer is running the Citadel database).

CitadelControl copies data from the source computer to the target computer. It copies data in *pages*, units of data 4K in size. (Exactly how much data is contained in a given 4K page depends on the type of data and the resolution at which you are logging it.) When CitadelControl copies data, it copies the entire page, which may include data outside of the specified range.

If you have enabled **Delete source data while archiving** (the `DeleteEnabled` data member), and your start or stop point for archiving data is in the middle of a database page, the CitadelControl does not delete the page in order to preserve data that was not covered in the request.

You should archive data to a database that has no Lookout process logging input. This database, having no setting on how long to hold data, will keep your data perpetually. This way, when you archive data from a Citadel database configured to store data for three days to an archive database with no restrictions on how long data is held, the archived data will not be removed after the time configured for it in the original database. It is held with a perpetual lifespan.

If you move data from your archive to a database with restrictions on how long data is to be held, the data will be inserted initially with a perpetual lifespan. As soon as Lookout logs new data with that trace name to the database, however, the lifespan for that data takes effect again and your restored data will be removed if it is older than the limits set in the database configuration of the Lookout process logging that data.

Caution This represents a change of behavior in the Citadel database. Formerly, Citadel was not aggressive in enforcing data age limits, deleting data only at certain times. Citadel now removes over-age data regularly. If you had been depending on loose enforcement of data lifespans in your data backup, you should take the more timely enforcement into consideration.

Additionally, you should be aware that if you log data to your archive database, any lifespan setting on data in the process you are logging from will be enforced in your archive database. For this reason we recommend that you *never* log data directly to an archive database.

You can archive data with identical or similar names from different processes to a single archive without conflict, because the name of the data includes its source process name.

You are not limited in the number of archive databases you create. You can create special purpose report databases as well as permanent archives, depending on your needs.

At this time you can only access data in your archive database in two ways: by using SQL queries or by creating a dummy process with the same name as the process in which the archived data originated. Your dummy process has to have your archive directory set as its default data directory, and should only have a Lookout HyperTrend object in it (in order to prevent accidentally logging to the archive and changing your data lifespans). You can then view the archived data the same way you would through the original process, using a URL with the process name and trace name for the HyperTrend. Notice that you cannot run this dummy process at the same time as the actual data source process, because of the conflict of process names.

You could also restore data to the database in the original process and view it through the original process (as long as the source database is not set to remove data as old as the data you are restoring).

Restoring Data

You can restore data in an active Lookout database with the CitadelControl object. All you need to do is use an archive database as the source and the active database as the target.

Pattern Matching

If you only wanted to archive a single data trace, you could just use the path for that data for **Archive trace matching this pattern** (the `What` data member). For instance, you could use

`"\\computername\processname\Waveform1.Sinewave"`

to archive a sine wave being logged from the Waveform1 object in the server process running on the local computer.

If you wanted to archive all the waveforms being logged from that Waveform object, you could use the following for **Archive trace matching this pattern** (the `What` data member).

`"\\computername\processname\Waveform1.*"`

To archive all data in the database you are archiving from, you could use the `*` wildcard. To archive all data from a given process on a specified computer you could enter:

`"\\computername\processname\*"`

Notice that the use of the wildcard character in the computer name location will archive data from any other computer running a process with that name and logging it to the Citadel database you specified as a source.

`"*\processname\*"`

The Lookout pattern matching feature uses wildcard characters, character lists, or character ranges, in any combination, to match strings. Table 1 shows the characters allowed in a pattern and what they match:

Table 1. String Character Patterns

Character	Matches in String
?	Any single character.
*	Zero or more characters.
/#	Any single digit (0–9).
[charlist]	Any single character in charlist.
[!charlist]	Any single character not in charlist.

Use a group of one or more characters (charlist) enclosed in brackets ([ ]) to match any single character in a string. This list can include almost any character code, including digits.

Note To match the special characters left bracket ([), question mark (?), number sign (#), and asterisk (*), you enclose them in brackets. The right bracket (]) cannot be used within a group to match itself, but it can be used outside a group as an individual character.

By using a hyphen (-) to separate the upper and lower bounds of the range, a charlist can specify a range of characters. For example, [A-Z] results in a match if the corresponding character position in string contains any uppercase letters in the range A-Z. Multiple ranges are included within the brackets without delimiters.

Lookout also uses the following pattern matching rules:

- When used outside brackets, the exclamation point matches itself.
- A hyphen (-) can appear either at the beginning (after an exclamation point if one is used) or at the end of charlist to match itself. In any other location, the hyphen is used to identify a range of characters.
- When a range of characters is specified, they must appear in ascending sort order (from lowest to highest). [A-Z] is a valid pattern, but [Z-A] is not.
- The character sequence [] is considered a zero-length string.

CitadelControl Data Members

Data Member	Type	Read	Write	Description
Cancel	Logical	Yes	Yes	Stops the archive operation when this value transitions to TRUE. Data already archived remains in the archive. The next time you start archiving data, the CitadelControl begins where it stopped when `Cancel` went TRUE (assuming you have not changed the other settings).
Complete	Numeric	Yes	No	Reports the percentage of the archive operation that has been completed.
DeleteEnabled	Logical	Yes	Yes	When TRUE, CitadelControl deletes the data it is archiving from the source database.

Data Member	Type	Read	Write	Description
FromWhen	Numeric	Yes	Yes	Sets the beginning time for archiving data. Set this value using ordinary Lookout time notation, such as 01/01/00 13:30:00. Setting this value to 0 archives data all from the beginning of the database to `ToWhen` limit when the CitadelControl Start data member goes TRUE.
FromWhere	text	Yes	Yes	Path to the source directory for the database your CitadelControl object is archiving data from.
Start	Logical	Yes	Yes	When this value transitions to TRUE, CitadelControl begins archiving.
ToWhen	Numeric	Yes	Yes	Sets the end time for archiving data. Set this value using ordinary Lookout time notation, such as 01/01/00 13:30:00. Setting this value to 0 archives data up to the current time when the CitadelControl Start data member goes TRUE.
ToWhere	Text	Yes	Yes	Path to the target database for your CitadelControl archiving operation.
What	Text	Yes	Yes	The Lookout trace name of the data you want to archive. You can use wildcards and pattern matching to archive multiple traces if you want.

Citadel Logging

This section clarifies how Citadel logs continuous data, and details a change in enforcing data lifespan in Citadel.

How Citadel Logs Continuous Data with a Slow Rate of Change

If you are logging numeric data with a slow rate of change, the way Lookout logs the data is important.

If you are polling a data point with a time interval of t, where t is faster than the change in the data, you might see the following values:

(t_1, X); (t_2, X); (t_3, X); (t_4, X); (t_5, X); (t_6,Y)

What Citadel will actually have logged in this case is the value X at time t_1. Citadel then logs the values at times t_2 through t_5 temporarily (in case of system failure). When the change is detected, the value at time t_5 is logged permanently and then the value at time t_6 is logged.

Citadel Data Lifespan Enforcement

As mentioned in the documentation for the CitadelControl, data is cleared from the database according to its lifespan setting more aggressively than in earlier versions of Lookout. Formerly, Citadel deleted data only at certain times. Citadel now removes over-age data regularly. If you have been depending on loose enforcement of data lifespans in your data backup, you should take the more timely enforcement into consideration.

Using SQL Queries From Microsoft Query in Lookout

The introduction of multiline TextEntry objects in Lookout 4.5 makes it easier for you to copy an SQL query from Microsoft Query and use it in Lookout. Microsoft Query breaks lines in the SQL statement by embedding new line codes into the query. The new Lookout 4.5 multi-line edit box can handle these embedded new lines.

To cut and paste Microsoft Query generated SQL queries into Lookout, use the following procedure:

1. Before you create a DataTable or SQLExec object, create a TextEntry object.
2. Create a DataTable object or SQLExec, set the **Connect String** or **Data Source** to point to your data source and set the SQL Command to the name of your TextEntry object, as shown in Figure 4.

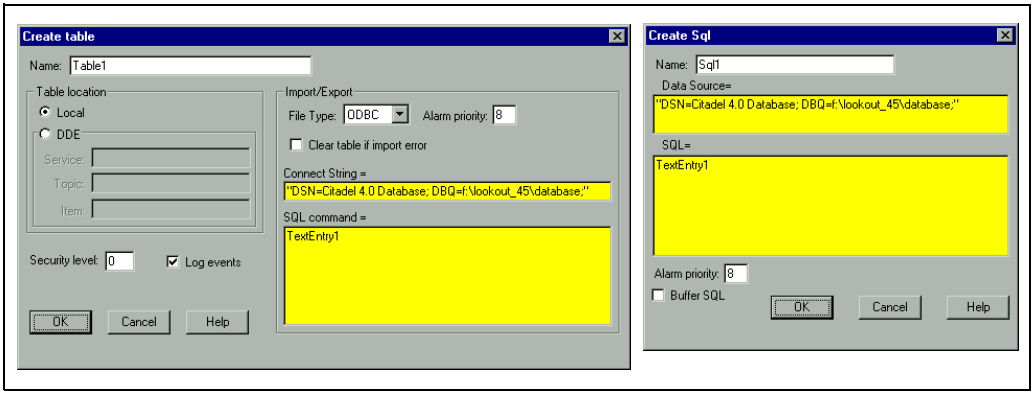

Figure 4. DataTable and SQLExec Object Properties Dialog Boxes

3. Build your query in Microsoft Query. Figure 5 shows a query created in Microsoft Query.

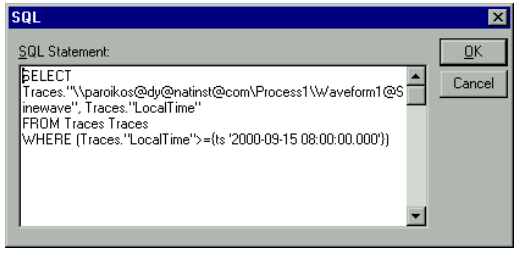

Figure 5. SQL Statement

4. Copy the SQL statement.
5. In Lookout run mode, click the TextEntry object, click the » button to make the edit box multi-line, and paste the SQL statement into the box, as shown in Figure 6. Click **OK**.

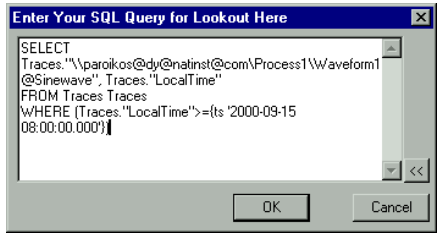

Figure 6. SQL Query

Your SQL query will now work for you in Lookout without your having to insert double quotes to denote special characters.

Mailer Object Parameters

The following table documents the parameters, listed in order, for the Lookout Mailer object class.

Number	Name	Type	Description
1	Buffer unsent messages	LogCnst	Enter yes to buffer unsent messages, no to disable buffering
2	Retry Attempts	NumCnst	Number of times Lookout will attempt to sent your message; enter an integer (such as 5)

Number	Name	Type	Description
3	Retry delay	NumCnst	Time in minutes between each attempt Lookout makes to send your message; enter an integer (such as 15)
4	SMTP server	TxtCnst	Name of your SMTP server; enter a string (such as “PO1”)
5	Timeout	NumCnst	How long in seconds Lookout waits for your mail to be sent successfully. Sending mail successfully involves a number of operations, so make sure you choose a long enough time. Enter an integer (such as 10)
6	Alarm priority	NumCnst	Alarm level 1–10
7	Generate event on successful send	LogCnst	To report that your messages was sent successfully, set to yes; otherwise set to no
8	Word wrap message	LogCnst	To wrap your message at 70 characters, set to yes; otherwise set to no

Alarm Filter Options

The **Ack User Name** and **Ack Comment** options have been removed from the Lookout Alarm Filter.

Multi-line Text Changes

In previous versions of Lookout you could not embed line returns in text strings you entered using a TextEntry object or in a Lookout expression. The multi-line capability in Lookout 4.5 makes this possible. You must remember, however, that carriage returns can interfere in certain parsing operations, so be careful how you use them.

Using Lookout as an OPC Server

Lookout functions as an OPC server, and is compliant with the OPC 2.0 specification. OPC items in Lookout take standard Lookout data names, including the computer name and process name elements of the path.

Known Issues Lookout 4.5

The following known issues exist for Lookout 4.5.

Roadmap for Experienced Users of Lookout

This section tells where you can find more detailed information on the new features and functionality of Lookout 4.5.

Refer to the *Lookout Help* for more information about new Lookout objects. Look there for information on the Aggregate, ActiveX, Report, DataSocket, Mouse, and Joystick object classes. You will also find information on the new drivers.

Lookout installs ten ActiveX controls from National Instruments ComponentWorks. Learn more about these controls through their own online help, which you can access through each control's property pages.

Detailed information on the new Image Navigator graphics is available in the *Image Navigator* section of Chapter 2, *Graphics*, of the *Lookout Developer's Manual*, as well as in the Image Navigator online help.

Using the convenient new list server and auto-detection features for registering computers is described in the *List Servers and Auto-Detection* section of Chapter 4, *Networking*, of the *Lookout Developer's Manual*, and in the Lookout online help. Access this new set of capabilities by selecting **Options»Network** in Lookout.

The usability enhancements for Lookout may best be explored by using them. Right-click objects in the Object Explorer and you will find that you can now cut, copy, and paste objects. You can also open both the Edit Connections and Edit Database box by right-clicking on an object or even on a data member in the Object Explorer, and find the box opening to your selected object or data member. For a more formal explanation of these enhancements, consult the *Object Explorer* section of Chapter 3, *Lookout Basics: Windows, Tools, and Files* in the *Getting Started With Lookout* manual.

In the Connection Browser you will notice that you can now see the data values being passed between Lookout connections. Refer to the *Connection Browser* section of Chapter 3, *Lookout Basics: Windows, Tools, and Files* in the *Getting Started With Lookout* manual for more information about the Lookout Connection Browser.

Notice that the Lookout User Manager has been renamed the National Instruments User Manager. This is because the permissions based network security of Lookout is being integrated with other National Instruments

products, such as the LabVIEW Datalogging and Supervisory Control module. The User Manager functions the same way it did in Lookout 4.0.

Object class parameters are now documented in order of their appearance in the Connection Browser and a Lookout `.lks` file. This documentation is necessary for using the Aggregate Object introduced in Lookout 4.5. You can find this documentation in Appendix A, *Object Class Parameters*, of the *Lookout Object Reference Manual*, which is installed as a PDF file in your Lookout `Documentation` directory.

Compatibility and Conversion with Lookout 4.5

To open Lookout 4.0 processes in Lookout 4.5, open the `.lks` file instead of the `.l4p` file when you load a process. As soon as the process opens, save it to update the `.l4p` file.

If you are upgrading from Lookout version 3.8, you should be able to open all your processes in Lookout 4.5 with little or no difficulty. Some minor adjustments may have to be made in some processes, and you will have to change redundancy using backup computers. This section covers compatibility and conversion issues for those upgrading form version 3.8 of Lookout.

Lookout 3.8 and Lookout 4.5 Compatibility

The following list contains known compatibility issues and fixes for Lookout 4.5.

- Lookout 4.5 does not run on Windows NT version 3.51 or earlier—you must be using Windows NT 4.0 or later.
- Earlier versions of Lookout process files used the file extension `.lkp`, and state files used the extension `.lst`. Lookout now uses the `.l4p` extension for process files and `.l4t` for state files.

  If you want to run your old 3.8 processes on a 3.8 version, save backups of the files. Otherwise, recompiling a 3.8 process in Lookout 4 or greater will create (if you save) new `.lks`, `.l4p`, and `.l4t` files. Lookout also creates new security files using the extension `.lka`.
- Avoid installing Lookout 4.5 over Lookout 3.*x*.*xx*. Either uninstall your previous version of Lookout, or put Lookout 4.5 in a new directory, transferring or copying files for conversion.
- Lookout 3.*x*.*xx* can run on the same computer with Lookout 4.5, and a process in one can be networked to a process in the other using DDE. If you intend to run both versions at one time, start the earlier version of Lookout first.

- Lookout 3.8 Citadel files cannot be used by Lookout 4.5. You should maintain old Citadel files separately from Lookout 4.5 database files.
- Events are now stored with alarms and process data in the Citadel database.
- The default location for data logged with the Spreadsheet object is the Lookout directory. The Lookout directory is also the default for any spreadsheet files you might export from the DataTable object or `.csv` (comma separated value) files you might create with the **Print Alarms** dialog box. If you do not want these files to be created in the Lookout directory, enter a complete path (including file name) for the location where you want the file created.

Converting Lookout 3.8 Processes to Lookout 4.5

To load a Lookout 3.8 process file in Lookout 4.5, select **File»Open** and set the **Files of type** to Lookout Source Files (`*.lks`). Figure 7 shows the Open Process dialog box.

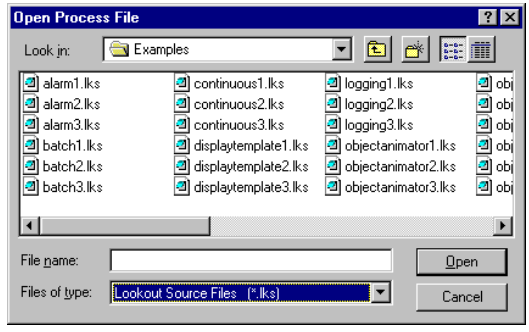

Figure 7. Open Process Dialog Box

Select the file you want to convert and click the **Open** button (or double-click the file name). Lookout uses the built-in CBL compiler to recompile the 3.8 source file into a Lookout 4.5 process file.

Converting 3.8 files may require the following adjustments:

- Lookout 3.8 versions of the Pot, Switch, Pushbutton, and TextEntry objects that are remoted convert as L3Pot, L3Switch, and so on, and continue to function as they formerly did. You cannot create new 3.8 objects like these through Lookout 4.5.

  Because remote position source connections have changed in response to the demands of networking, you may find you want to replace L3 versions of these controls with the new Lookout 4.5 versions, especially if you want these controls to operate over a network. Delete the old controls and replace them with new controls using the same

names. The Connection Browser and Object Explorer can help you track all the connections that may need to be re-established if you decide to make these replacements.

- ODBC/SQL limitations: With the introduction of networking and hierarchical arrangement of objects within processes, it is easy to exceed the 62 character limit that exists in Microsoft Access and Microsoft Query. Even fully-compliant ODBC programs are limited to 126 characters, so some consideration of these limitations when assigning computer, process, and object names is necessary, if you intend to use SQL queries.
- The old LKSwap CBX no longer works. If you used this special object class from the old Lookout Evaluation Guide, you will have to replace it with the Lookout Loader object, which is more flexible and powerful than the old LKSwap object. Remember too that Lookout 4.5 can run multiple processes, so it is not necessary to shut one process down before running another.
- Fonts are preserved when converting Lookout 3.8 process files to 4.5 using the `.lks` file, but font code page information is lost. This appears to mostly affect international users whose languages use a different code page than the US code page.
- The $Alarm object syntax in an `.lks` file has changed. The full URL for a $Alarm object will be as follows:

  `\\machine\process\$Alarm`
- Your custom built Lookout 3.8 CBXs should run in Lookout 4.5, as long as they were developed using the Lookout Object Developer Toolkit. If you have problems with a custom CBX, contact National Instruments technical support for help in resolving the problem.
- The Lookout HyperTrend object now accesses data through a URL connection. The data source for that connection must be logged to the Citadel database for the HyperTrend to display the data.
- When you convert a process with a Lookout 3.*xx* HyperTrend in it, Lookout automatically creates a named expression to make the URL connection to. Lookout automatically logs this named expression to the Citadel database. To maintain your converted HyperTrend objects, you must use or replace these named expressions.
- If you have Lookout 3.8 processes that use the NIDAQ or NISCXI objects, and you have a version 6.5.1 or earlier of NI-DAQ installed on your computer, you must upgrade to NI-DAQ 6.5.2 or better to use Lookout 4.5. You can download the current version of NI-DAQ software from the National Instruments web site, at `ni.com`.

  If your Lookout process will be running on the same computer as your National Instruments Data Acquisition or SCXI hardware, we

recommend that you edit your process to use the OPCClient object to connect to NI-DAQ, for performance reasons.

Importing Old Security Files into Lookout

You can import the user account information from your Lookout 3.8 processes into Lookout 4.5 using the National Instruments User Manager. Importing is not necessary if you are upgrading from Lookout 4.0 to Lookout 4.5 because Lookout converts the file automatically when you open it with the User Manager.

1. Select **Options»User Manager** from the Lookout menu. You must be in edit mode for the User Manager item to appear in the **Options** menu. The **User Manager** dialog box appears as shown in Figure 8.

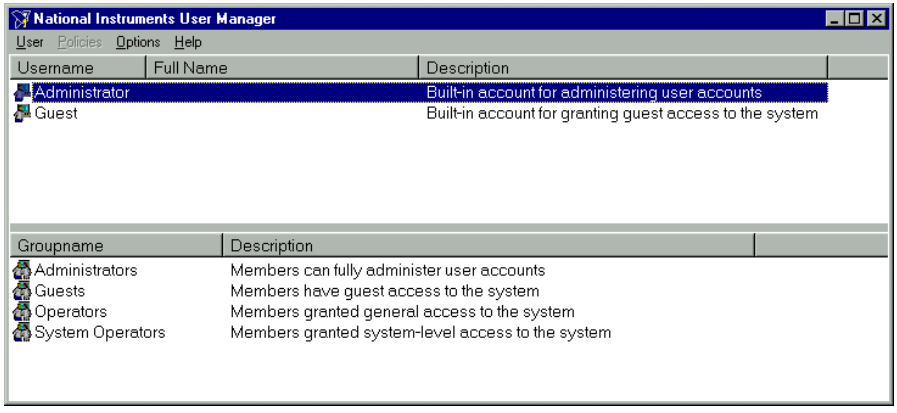

Figure 8. User Manager Dialog Box

2. Select **User»Import Security File** from the **User Manager** dialog box. The following dialog box appears as shown in Figure 9.

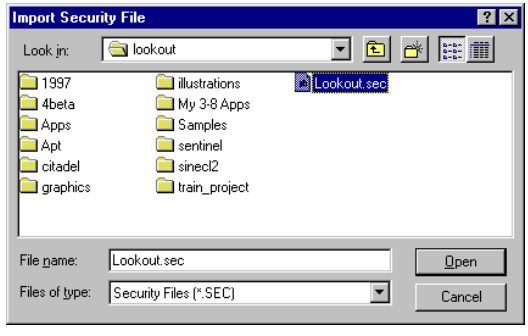

Figure 9. Import Security File Dialog Box

3. Navigate to your old Lookout 3.8 security file `Lookout.sec`, and select it. Lookout 3.8 kept the `Lookout.sec` security file in the Lookout directory.
4. Click the **Open** button.
5. If you have already created any user accounts in Lookout 4.5 that are the same as accounts you used in Lookout 3.8, you will receive a message informing you that a user account with that name already exists. You may replace your recently created account, or choose not to use the old account information.
6. Exit the User Manager.

Note Unlike Lookout 3.8, Lookout 4.5 maintains the `Lookout.sec` security file in the Windows `System` directory.

The User Manager creates a unique identification number for each user account. This means that to maintain identical accounts, you must use the same `Lookout.sec` file for each copy of Lookout running on your network. Copy your `Lookout.sec` file to the Windows `System` directory in every computer you intend to run Lookout on.