Available online at www.sciencedirect.com

ScienceDirect

Carbohydrate Research 343 (2008) 2162–2171

Carbohydrate
RESEARCH

GlycoCT—a unifying sequence format for carbohydrates

S. Herget,[a,*] R. Ranzinger,[a] K. Maass[b] and C.-W. v. d. Lieth[a]

[a]*German Cancer Research Center, Molecular Structure Analysis (W160), Molecular Modeling Group, Im Neuenheimer Feld 280, D-69120 Heidelberg, Germany*

[b]*Institute of Biochemistry, Faculty of Medicine, University of Giessen, Friedrichstrasse 24, D-35392 Giessen, Germany*

Received 5 November 2007; received in revised form 27 February 2008; accepted 6 March 2008
Available online 13 March 2008

Presented at Eurocarb 14th Lübeck, Germany, September 2007

Abstract—As part of the EUROCarbDB project (www.eurocarbdb.org) we have carefully analyzed the encoding capabilities of all existing carbohydrate sequence formats and the content of publically available structure databases. We have found that none of the existing structural encoding schemata are capable of coping with the full complexity to be expected for experimentally derived structural carbohydrate sequence data across all taxonomic sources. This gap motivated us to define an encoding scheme for complex carbohydrates, named GlycoCT, to overcome the current limitations. This new format is based on a connection table approach, instead of a linear encoding scheme, to describe the carbohydrate sequences, with a controlled vocabulary to name monosaccharides, adopting IUPAC rules to generate a consistent, machine-readable nomenclature. The format uses a block concept to describe frequently occurring special features of carbohydrate sequences like repeating units. It exists in two variants, a condensed form and a more verbose XML syntax. Sorting rules assure the uniqueness of the condensed form, thus making it suitable as a direct primary key for database applications, which rely on unique identifiers. GlycoCT encompasses the capabilities of the heterogeneous landscape of digital encoding schemata in glycomics and is thus a step forward on the way to a unified and broadly accepted sequence format in glycobioinformatics.

Keywords: XML; Carbohydrate sequence format; Glycobioinformatics; Structure encoding

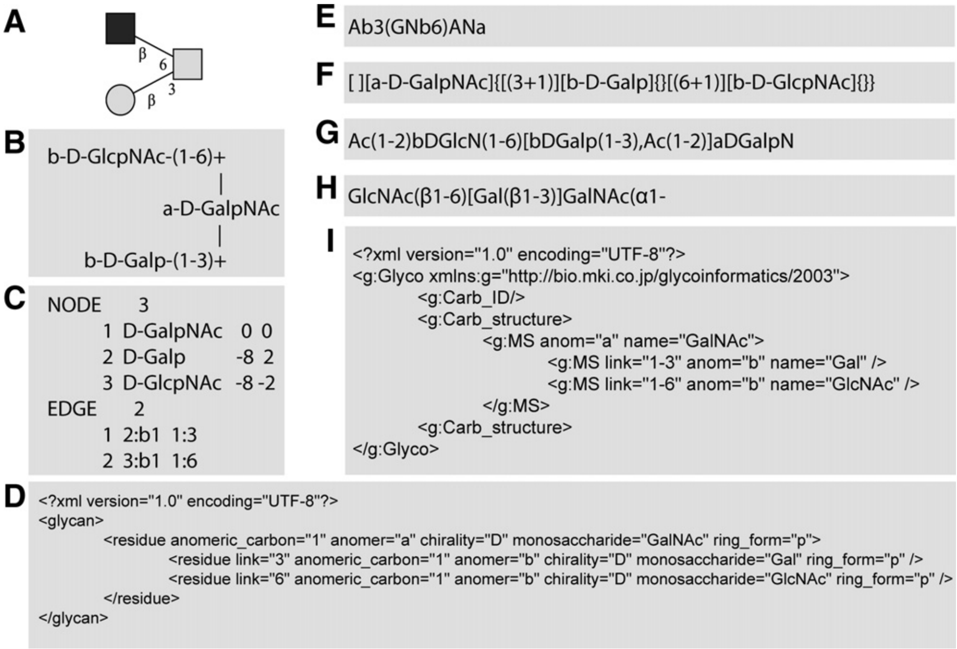
1. Introduction

The task of storing complex carbohydrate sequences in a structured digital format was first explicitly addressed by the pioneering designers of the Complex Carbohydrate Structure Database (CCSD or often also called CarbBank),[1,2] by defining an intuitive, yet powerful way of storing carbohydrate sequence topologies as two-dimensional sketches (Fig. 1). These ASCII 2-D plots, closely resembling IUPAC recommendations,[3] were directly stored in the database. Subsequent initiatives using relational databases opted for storing the carbohydrate sequences as linear strings, similar to those used in protein or nucleotide databases. These strings were generated through an ordered traversal of the carbohydrate sequences and could thus serve as primary keys in database systems (e.g., string representations such as those used in LINUCS,[4] GlycoSuiteDB,[5,6] LinearCode[7] and the bacterial carbohydrate structure database (BCSDB) encoding[8]). Later the connectivity information in carbohydrate sequences was stored using connection table-like representations (KCF[9]). Also, XML encodings exist, applying a tree-like representation for the saccharide topologies (Glyde,[10] CabosML[11]).

Sequences of DNA or proteins can be handled bioinformatically as simple linear strings, whereas carbohydrate sequences present special informatics challenges caused by their property of branching. They can be described in computational terms as graphs with the monosaccharide residues as the vertices (nodes) and the glycosidic linkages as edges (lines) (Fig. 2). As carbohydrate sequences contain a preferred direction, they

[*] Corresponding author. Tel.: +49 6221 424541; fax: +49 6221 42454; e-mail: s.herget@dkfz.de

0008-6215/$ - see front matter © 2008 Published by Elsevier Ltd.
doi:10.1016/j.carres.2008.03.011



```
b-D-GlcpNAc-(1-6)+
                 |
          a-D-GalpNAc
                 |
b-D-Galp-(1-3)+
```

```
NODE     3
     1 D-GalpNAc   0  0
     2 D-Galp     -8  2
     3 D-GlcpNAc  -8 -2
EDGE     2
     1 2:b1  1:3
     2 3:b1  1:6
```

```
<?xml version="1.0" encoding="UTF-8"?>
<glycan>
        <residue anomeric_carbon="1" anomer="a" chirality="D" monosaccharide="GalNAc" ring_form="p">
                <residue link="3" anomeric_carbon="1" anomer="b" chirality="D" monosaccharide="Gal" ring_form="p" />
                <residue link="6" anomeric_carbon="1" anomer="b" chirality="D" monosaccharide="GlcNAc" ring_form="p" />
        </residue>
</glycan>
```

```
Ab3(GNb6)ANa
```

```
[ ][a-D-GalpNAc]{[(3+1)][b-D-Galp]{}[(6+1)][b-D-GlcpNAc]{}}
```

```
Ac(1-2)bDGlcN(1-6)[bDGalp(1-3),Ac(1-2)]aDGalpN
```

```
GlcNAc(β1-6)[Gal(β1-3)]GalNAc(α1-
```

```
<?xml version="1.0" encoding="UTF-8"?>
<g:Glyco xmlns:g="http://bio.mki.co.jp/glycoinformatics/2003">
        <g:Carb_ID/>
        <g:Carb_structure>
                <g:MS anom="a" name="GalNAc">
                        <g:MS link="1-3" anom="b" name="Gal" />
                        <g:MS link="1-6" anom="b" name="GlcNAc" />
                </g:MS>
        <g:Carb_structure>
</g:Glyco>
```

Figure 1. An overview of the sequence formats used in glycobioinformatics, using as an example the O-glycan core 2 motif. Normally, database users will be confronted with graphical representations similar to A and B. (A) Graphical representation as suggested by the Consortium for Functional Glycomics (http://glycomics.scripps.edu/CFGnomenclature.pdf). (B) CarbBank two-dimensional graph. (C) KCF, an application of the connection table approach, as used by the Kyoto Encyclopedia of Genes and Genomes (KEGG). (D) Glycan data exchange format (Glyde), an XML variant. (E) GlycoMinds encoding, as employed by the CFG[15] (sequences are read from right to left, AN is D-Gal*p*NAc, GN is D-Glc*p*NAc, A is D-Gal*p*. An online introduction is available[16]). (F) LINUCS sequence format as applied in GLYCOSCIENCES.de. (G) Bacterial carbohydrate structure database (BCSDB) encoding. (H) GlycoSuiteDB format. (I) CabosML, another XML-variant.

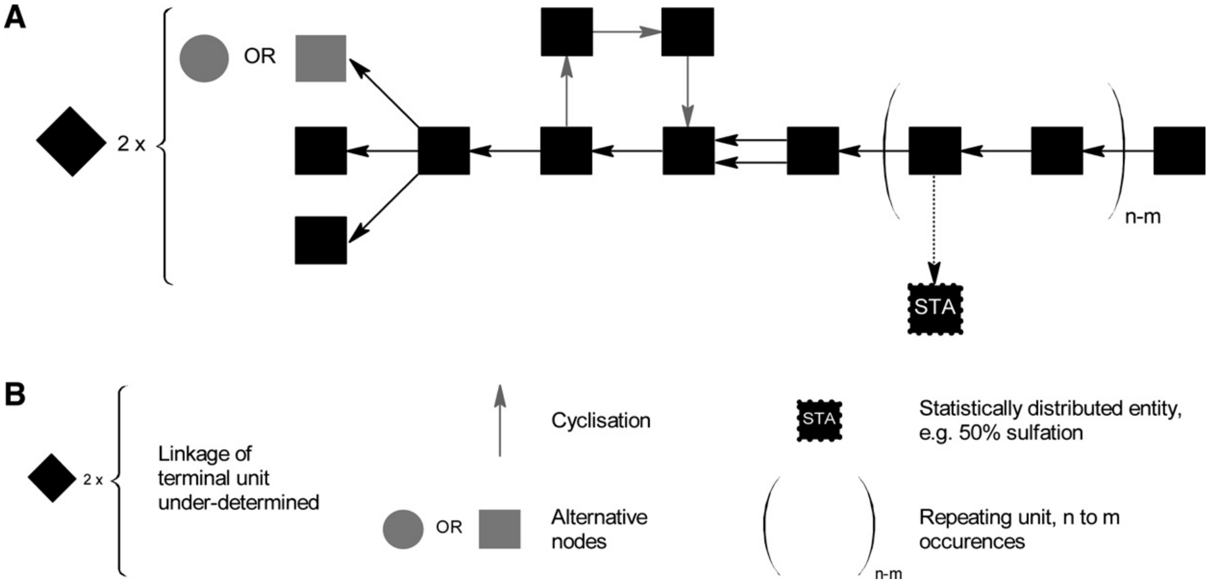


Figure 2. An abstract scheme of a hypothetical complex carbohydrate sequence. (A) Squares and circles represent monosaccharide residues (graph vertices), arrows symbolize glycosidic linkages (directed edges). Carbohydrate sequences may contain repeating units with non-stoichiometric modifications ('STA', e.g., glycosaminoglycans), multiple connections between vertices (e.g., lactonized sialic acids, symbolized here with doubled arrow) and cyclic substructures (e.g., cyclodextrins, grey arrows). Alternative residues and fuzzily defined terminal unit locations may be caused by an incomplete structure elucidation. (B) Legend for part A.

can be viewed as directed graphs (digraphs). The existence of potential multiple connections between two residues can degenerate the graph to a multigraph (e.g., in lactonized sialic acids). The rare cyclization of carbohydrate structures (e.g., cyclodextrins) can lead to cyclic graphs. To avoid a combinatorial expansion of identical substructures, so-called repeating units (e.g., polylactosamines or bacterial O-antigens) are frequently encoded as special entities. Limitations of analytical techniques to determine all structural features, resulting in partial structures, can produce uncertainties in the sequences, especially regarding the location of terminal residues. Some secondary modifications (e.g., sulfation in glycosaminoglycans) are only present on a fraction of the residues of repeating units, leading to non-stoichiometric modification patterns. Finally, a list of alternative residues at a certain position can be contained in a reported structure.

2. GlycoCT

As part of the EUROCarbDB project we were searching for a suitable structure encoding solution. As a first step, we have performed an analysis of the existing structural encoding schemata already used in glycomics. The existing formats as described above have different capabilities to store the complex information potentially present in carbohydrate sequences (Table 1).

Additionally, further tasks have to be fulfilled by the sequence format to be used. A central requirement was a unique encoding of all structures. This is achieved by applying strict sorting rules, which determine the order of appearance of the different elements in the sequence, thus leading to a unique string representation for each potential sequence. Together with machine readability the unique encoding is mandatory for its role as primary key in databases. A unique identifier is also beneficial for the implementation of exact structure searches. Therefore, a controlled vocabulary for monosaccharides had to be defined. Encoding of uncertain linkages and unspecified monosaccharides have to be supported as well as the possibility to store sugar compositions or sugars fragments. Finally, a seamless gateway to the atomic details of the encoded structures is highly requested, facilitating the translation to molecule description formats used in chemoinformatics.

Unfortunately, none of the existing solutions covered all crucial aspects. This gap motivated us to develop a comprehensive solution to capture the full complexity present in carbohydrate sequence data. The general concept of GlycoCT is a connection table (CT) based approach, and hence its name (Fig. 3).

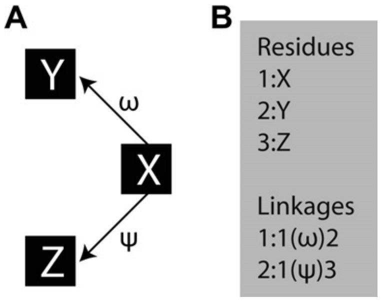


Figure 3. The general idea of GlycoCT is a connection table approach. (A) A schematic, branched trisaccharide with residues X, Y, Z and the linkages ψ and ω. (B) The connection table with its two main sections, the entity and the linkage list. The entity list contains all occurring residues, assigning a number to each. The linkages section specifies the connectivities using these numbers to address the residues.

The format exists in two variants, a condensed form and a more verbose XML variant. The $\text{GlycoCT}_{\{\text{condensed}\}}$ variant can serve as a unique identifier for glycan structures, even in the case of ambiguities in the structure. The $\text{GlycoCT}_{\{\text{XML}\}}$ variant facilitates computational handling and data exchange. The format is divided into different sections, structuring the sequence information (Fig. 4).

The subsequent examples will use the condensed variant. These examples can be easily transformed to the XML-variant, as it uses the same definitions. We will now explain the central features of GlycoCT, the nomenclature conventions and the topological aspects and conclude with a discussion of the sorting algorithm and the XML variant. The interested reader can find an online version of the GlycoCT handbook on the EUROCarbDB webpages, which is updated continuously and contains more examples (http://www.eurocarbdb.org/recommendations/encoding).

2.1. Basic monosaccharide namespace

Since freetext identifiers for chemical entities can result in long-term database consistency problems, a controlled, machine-readable vocabulary for monosaccharides is a prerequisite for the GlycoCT definition. As

Table 1. A comparison of the capabilities of the major sequence formats to store special structural features ('+' can be handled, '–' cannot be encoded, 'O' can partially be encoded)

Sequence format	Multiple connections	Cyclization	Repeating units	Terminal unit underdetermined	Non-stoichiometric modification	Alternative residues
CCSD	–	+	+	O	O	O
LINUCS	–	+	+	–	–	–
BCSDB	–	–	O	–	–	+
LinearCode	–	+	+	+	–	+
KCF	+	+	+	–	–	–
CabosML	–	+	+	–	–	–
Glyde	–	+	+	–	–	–

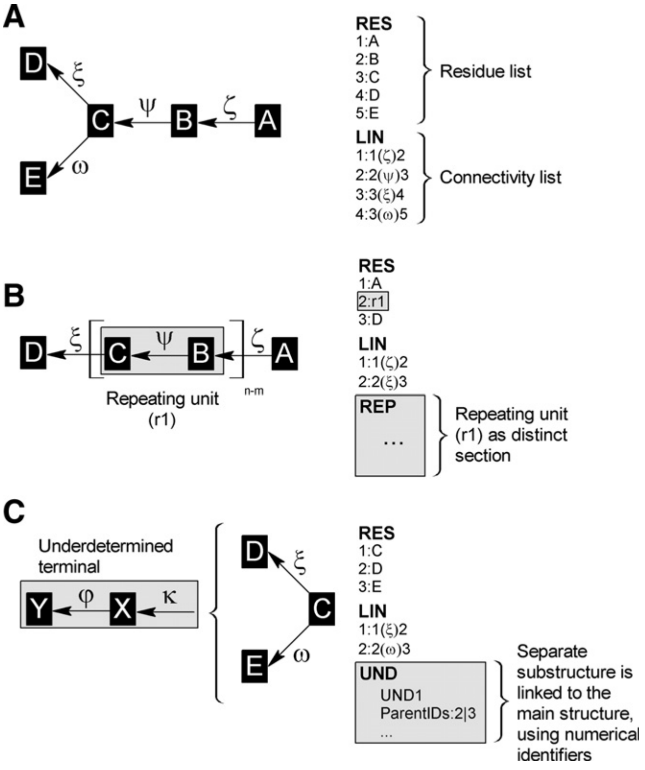
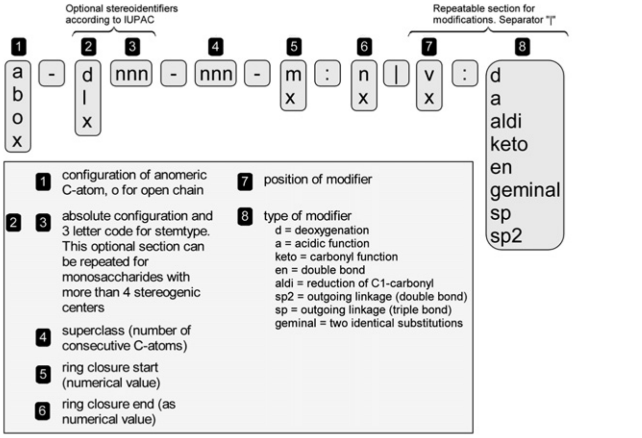


Figure 4. Sections in GlycoCT and block concept. (A) A branched schematic pentasaccharide in simplified GlycoCT code, using symbols for residues and linkages. The RES-section contains the entities, the LIN-section lists the connectivities. (B) An example of an additional block is the repeating unit, which is referenced in the RES-section and further specified in the REP-section. (C) Using the numerical addresses of the residues, it is possible to link to further sections. Shown here is an underdetermined terminal unit linked either to residues D or E of the main graph via the ParentIDs.

GlycoCT should not be restricted to a limited set of monosaccharides (e.g., the ten 'classical' mammalian monosaccharides), a universal approach was needed. The rules defined by IUPAC provide a robust and widely accepted naming convention for monosaccharides and we have adopted them for computational purposes. Principally five attributes are sufficient to describe a basic monosaccharide (basetype): Anomeric configuration, (composite) stem type with configurational prefix, chain length indicator, ring forming positions and further modification designators. We use these attributes in a formatted string to identify the basic monosaccharides (Fig. 5). Trivial names, like fucose, quinovose or rhamnose, are not allowed in GlycoCT, as they introduce a potential source of inconsistencies. The trivial names though can easily be generated by exporter programmes designated for user interaction (e.g., GlycanBuilder[12]).

2.2. Substituents

The basic monosaccharides occurring in natural oligo- and polysaccharides are frequently substituted with small molecules. An analysis of the substituted monosaccharides deposited in recent databases has revealed a small set of 37 substituents (Fig. 6 and Supplementary data). The members of this substituent library are connected via linkages to the basetypes yielding the traditional monosaccharides.


Optional stereoidentifiers according to IUPAC
Repeatable section for modifications. Separator "|"
1
2
3
4
5
6
7
8
a b o x
-
d l x
nnn
-
nnn
-
m x
:
n x
|
v x
:
d a aldi keto en geminal sp sp2
1 configuration of anomeric C-atom, o for open chain
2 3 absolute configuration and 3 letter code for stemtype. This optional section can be repeated for monosaccharides with more than 4 stereogenic centers
4 superclass (number of consecutive C-atoms)
5 ring closure start (numerical value)
6 ring closure end (as numerical value)
7 position of modifier
8 type of modifier
d = deoxygenation
a = acidic function
keto = carbonyl function
en = double bond
aldi = reduction of C1-carbonyl
sp2 = outgoing linkage (double bond)
sp = outgoing linkage (triple bond)
geminal = two identical substitutions

Examples

GlycoCT	IUPAC
b-dglc-HEX-1:5	β-D-Glc*p*
a-lgal-HEX-1:5|6:d	α-L-Fuc*p*
a-drib-HEX-1:5|3:d|6:d	3,6-dideoxy-α-D-*ribo*-hexose-pyranosyl (Paratose)
x-dgro-dgal-NON-2:6|1:a|2:keto|3:d	3-deoxy-D-*glycero*-D-*galacto*-non-2-ulopyranosonic acid (KDN)

Figure 5. Monosaccharide naming in GlycoCT resembles IUPAC and CarbBank definitions, but is more strictly ordered. Trivial names are not used on the sequence level, but can be generated during the conversion to graphical formats. The names used in GlycoCT are easily readable for humans and facilitate parsing with computer programs. The value '*x*' stands for unknown property.

2.3. Other entities

Apart from the basic monosaccharides and the substituents, other entities have been defined in GlycoCT (Table 2). Repeating, alternative and underdetermined terminal

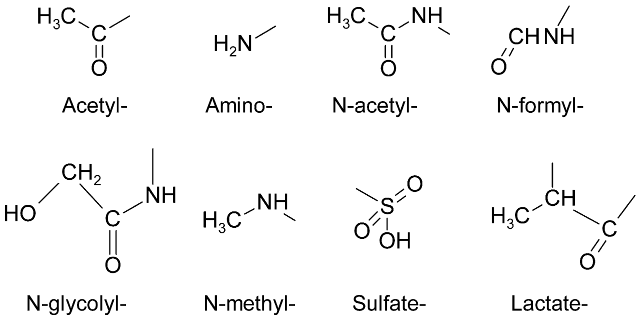


Figure 6. Selection of substituents used in GlycoCT. For a full list see the Supplementary data.

Table 2. Basic residue (RES) entities found in GlycoCT

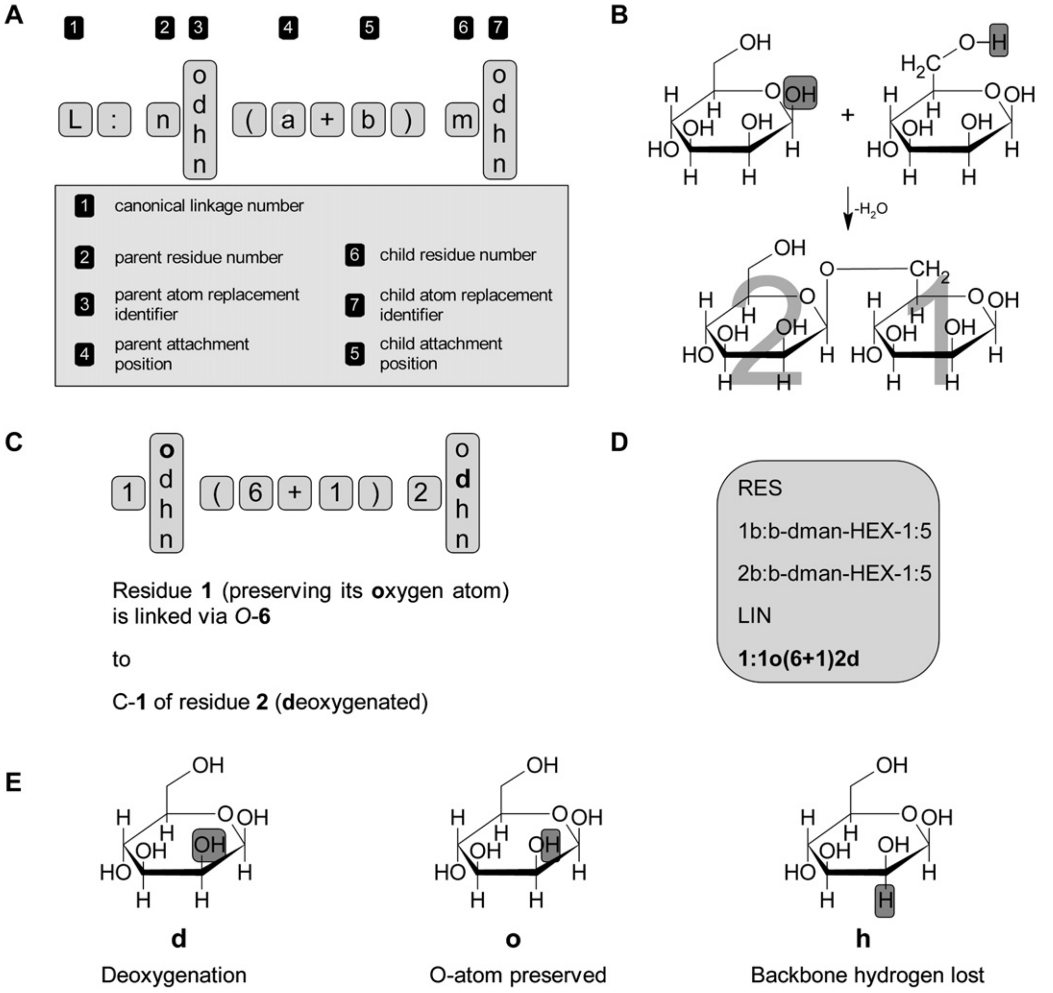
Abbreviation	Entity
b	Basetype
s	Substituent
r	Repeating unit
a	Alternative unit

These four classes constitute the RES-section of GlycoCT. For explanation refer to the main text below.

units are used to model specific topological arrangements. Additional non-carbohydrate units are reserved for non-sugar residues connected covalently to the carbohydrate sequence. The encoding of these non-carbohydrate units is principally beyond the scope of a carbohydrate sequence format, and we strongly recommend taking advantage of existing initiatives and approaches to reference these entities in separate datastructures (e.g., links to protein database identifiers, usage of INCHI[13] encodings for small molecules). Nevertheless, for backward compatibility with older sequence formats, we have introduced a separate block for freetext identifiers of small molecules. Because of uniqueness concerns for the primary key we have refrained from using this non-carbohydrate block in our own database installations.

2.4. Topology

The residue entities in GlycoCT as listed in the RES-section are connected covalently to form saccharides. Each bond is modelled in the LIN-section of GlycoCT including the atomic details by using an atom replacement schema (Fig. 7). This replacement mechanism is


A
1
2 3
4
5
6 7
L : n o d h n (a + b) m o d h n
1 canonical linkage number
2 parent residue number
3 parent atom replacement identifier
4 parent attachment position
6 child residue number
7 child atom replacement identifier
5 child attachment position
B
OH
O
OH
H
H
OH
OH
HO
H
H
H
+
H₂C
O–H
O
OH
-H₂O
CH₂
2
1
C
1 o d h n (6 + 1) 2 o d h n
D
E
OH
d
o
h
Deoxygenation
O-atom preserved
Backbone hydrogen lost

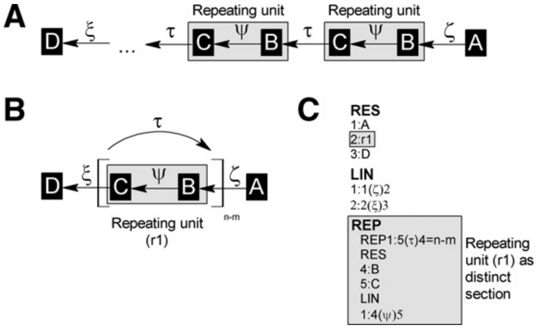
Residue **1** (preserving its **o**xygen atom) is linked via *O*-**6**

to

C-**1** of residue **2** (**d**eoxygenated)

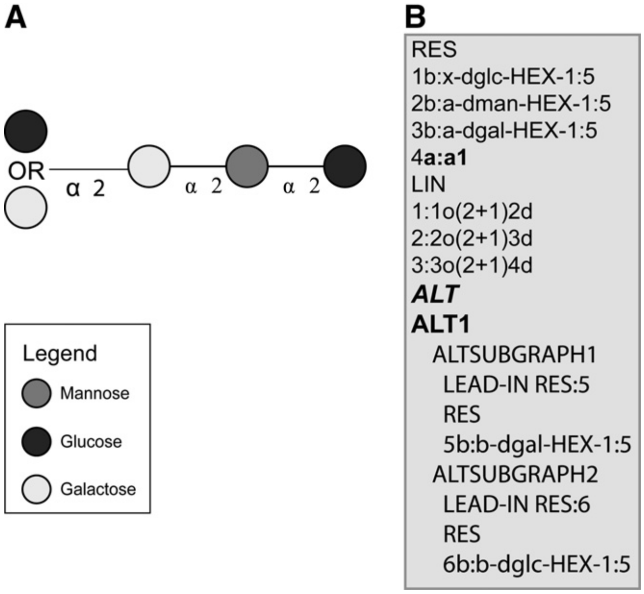
```
RES
1b:b-dman-HEX-1:5
2b:b-dman-HEX-1:5
LIN
1:1o(6+1)2d
```

Figure 7. The linkage information in GlycoCT. An atom replacement scheme specifies the structural modifications of the connected entities caused by the formation of the chemical bond. (A) Schematic illustration of a linkage in GlycoCT. The parent residue is in the direction of the reducing end. (B) The glycosidic bond formation between two mannoses is a formal elimination of a water molecule. (C) This linkage is expressed in GlycoCT terms. The linkages are encoded in the order of the linkage comparison algorithm (see below). (D) The full sequence of the disaccharide fragment in B. (E) Illustration for the major atomic replacement operators as found in the linkages of GlycoCT. Grey boxes indicate the replaced atom(s). For a full list of operators, refer to the main sequence description handbook (http://www.eurocarbdb.org/recommendations/encoding).



```
RES
1:A
2:r1
3:D
LIN
1:1(ζ)2
2:2(ξ)3
REP
  REP1:5(τ)4=n-m
  RES
  4:B
  5:C
  LIN
  1:4(ψ)5
```

Figure 8. Repeating units can be encoded in GlycoCT. They are numbered and further decomposed in the REP-section. (A) A repeating unit as typically found in polysaccharides (grey box). The repeating units are connected to each other with the internal repeat linkage (here abbreviated as τ). (B) The same structure as in A, in a more condensed graphical representation. (C) The GlycoCT code for this polysaccharide. The repeating unit (r1) is listed in the RES-section and specified in its own block, the REP-section.



```
RES
1b:x-dglc-HEX-1:5
2b:a-dman-HEX-1:5
3b:a-dgal-HEX-1:5
4a:a1
LIN
1:1o(2+1)2d
2:2o(2+1)3d
3:3o(2+1)4d
ALT
ALT1
  ALTSUBGRAPH1
    LEAD-IN RES:5
    RES
    5b:b-dgal-HEX-1:5
  ALTSUBGRAPH2
    LEAD-IN RES:6
    RES
    6b:b-dglc-HEX-1:5
```

Figure 9. Alternative units, describing a choice between two potential substructures, are encoded in the ALT-section. (A) Graphical representation of a terminal alternative residue. (B) GlycoCT code for this structure. Each alternative is encoded as a distinct substructure with a defined connecting residue (LEAD-IN). Should the alternative unit be in the middle of a longer sequence, also LEAD-OUT residues have to be defined for each substructure (not shown, see documentation on http://www.eurocarbdb.org/recommendations/encoding).

described by a single letter code (Fig. 7E). We have incorporated this mechanism to facilitate the future generation of exact molecular descriptors on the atomic level directly from GlycoCT.

Incomplete carbohydrate sequences and compositions can be produced by the partial omission of linkages, multiconnected residues by additional linkages and cyclizations by adding appropriate linkage(s).

A frequently found characteristic of natural carbohydrate sequences are repeating units. GlycoCT handles the repeating units in a distinct section, the REP-section. This block is embedded in the RES-section with a numerical identifier and declared subsequently in the GlycoCT code. The recurrences of the repeating unit and the connection between the repeated elements (the internal linkage) must be specified (Fig. 8).

A less common feature found in digital carbohydrate sequences is alternative declarations of specific residues or bigger units (e.g., saccharide contains at a certain position either a galactose or a glucose). The ALT-section of GlycoCT handles this case in a distinct block (Fig. 9).

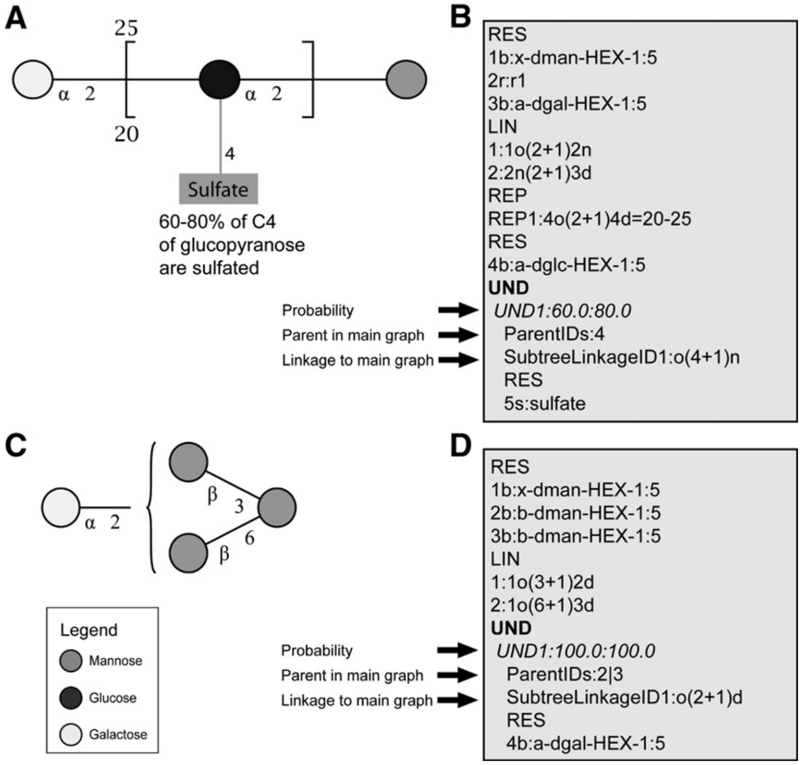
Another special feature of carbohydrate sequences is the existence of structural elements, whose attachment positions to the main sequence are not fully defined. These underdetermined terminal units in carbohydrate sequences are due to two different mechanisms. Whereas the first is caused by the limitations of the analytical techniques (Fig. 10C and D), the second is a result of biosynthetic non-stoichiometric modifications (Fig. 10A and B). The UND-section is used to model both of these cases. Concurrent substructures are declared inside the UND-section, which are then linked to the main structure via the canonical residue numbering. To indicate the prevalence of the UND-substructures, a range of percentage values must be given.

3. Sorting

One central aim of the GlycoCT$_{\{condensed\}}$ encoding is to generate a unique representation for all carbohydrate structures deposited in databases, even if they include some incomplete assignments or fuzziness in their structural description. The existing encoding schemata with the ability to generate unique representations (e.g., LINUCS, LinearCode) can handle such structures only to a limited extent. However, unique sorting of carbohydrate sequences is beneficial for database applications, as the sorted string can be used directly as a primary key in databases.

To ensure unique representations, we have introduced a set of hierarchical rules by which the ordering of residues, linkages and special structural features is unambigiuously defined. This system of hierarchical rules is described as follows:

The mandatory RES-section appears first in every GlycoCT code, directly continued by the facultative LIN-section, which can be omitted if only a composition is encoded. Subsequently, all the other sections can be appended in the order of REP, UND, ALT and NON. If multiple starting points exist, then the residue comparison algorithm is applied (Fig. 11A and text below). The internal ordering of the RES and LIN-sections follows the results of the linkage comparison algorithm (Fig. 11B and text below). For the additive sections



```
RES
1b:x-dman-HEX-1:5
2r:r1
3b:a-dgal-HEX-1:5
LIN
1:1o(2+1)2n
2:2n(2+1)3d
REP
REP1:4o(2+1)4d=20-25
RES
4b:a-dglc-HEX-1:5
UND
 UND1:60.0:80.0
  ParentIDs:4
  SubtreeLinkageID1:o(4+1)n
  RES
  5s:sulfate
```

```
RES
1b:x-dman-HEX-1:5
2b:b-dman-HEX-1:5
3b:b-dman-HEX-1:5
LIN
1:1o(3+1)2d
2:1o(6+1)3d
UND
 UND1:100.0:100.0
  ParentIDs:2|3
  SubtreeLinkageID1:o(2+1)d
  RES
  4b:a-dgal-HEX-1:5
```

Figure 10. Examples of underdetermined terminal units. (A) Graphical representation of a repeating unit with an incomplete sulfatation. (B) This structure in GlycoCT. The UND-section contains a percentage value range indicating its occupancy, the canonical number of the parent residue and a subtree including its linkage to the main graph. (C) Graphical representation of a terminal galactose with incomplete parent-relationship. (D) This structure in GlycoCT. The list of parents is given via their canonical residue numbers in the UND-section. The percentage values are set to 100 in this case.

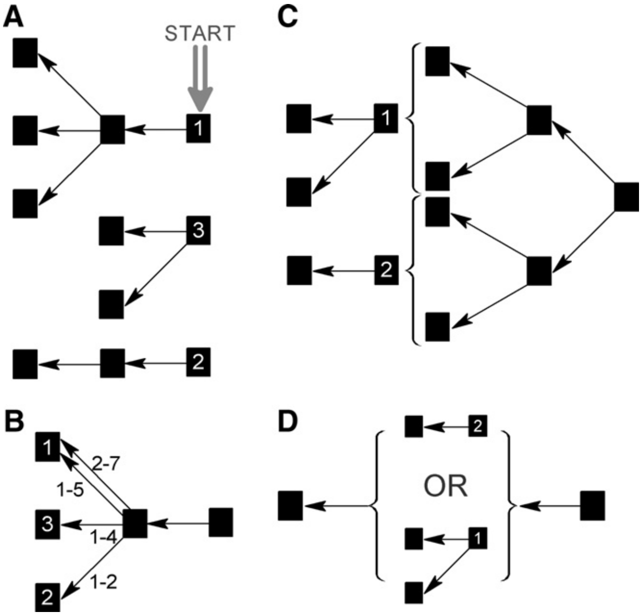


Figure 11. Sorting strategy of GlycoCT. The squares with numbers represent the entities to be sorted. (A) As structures with missing connectivity information can be described with one GlycoCT code, a precedence rule for isolated substructures is needed. (B) The linkage comparison is applied to branching points. (C) The correct order of underdetermined terminal units has to be maintained by another comparison method. (D) The different substructures of alternative units need to be listed in a defined order.

(ALT, UND) further comparison and sorting steps are implemented (Fig. 11C, D and text below).

3.1. Starting points: the residue comparison algorithm

GlycoCT is principally capable of encoding a saccharide consisting of several unconnected fragments (Fig. 11A). The sequence of the starting points is determined by the following rules:

(a) Number of child residues.
(b) Length of the longest branch.
(c) Number of terminal residues.
(d) Number of branching points.
(e) Lexical order.

For the residues at the reducing end of each fragment the characteristic numbers are calculated in the order as given above and the values are compared until a difference is found.

The highest priority is assigned to the fragment with the highest characteristic number. In case the rules a–d provide no distinction, the GlycoCT$_{\{\text{condensed}\}}$ code for each fragment is lexically compared. Should no difference be found, both fragments are identical and they can be encoded in an arbitrary order.

A composition can be regarded as a special case of this unique encoding of fragments. As no connections between residues are given, the characteristic numbers will be all identical and the lexical order of the residues is used as the single criterion.

3.2. Ordering of branches: linkage comparison algorithm

Carbohydrate sequences normally contain a preferred direction from the reducing end to the non-reducing end. For a simple linear carbohydrate, the preferred direction is defined by the direction from the reducing to the non-reducing end, and the linkages are ordered in the same way. For each branching point in the sequence, the order of the branches has to be defined (Fig. 11B). The order of the branches is determined through the application of the following hierarchy of rules:

(a) Number of bonds between parent and child residues (cardinal number of bonds).
(b) Atom linkage position at the parent residue.
(c) Atom linkage position at the child residue.
(d) Linkage type at the parent residue.
(e) Linkage type at the child residue.
(f) Comparison of child residues with residue comparison algorithm.

The characteristic numbers for all key features are calculated in the order as given above and the values are compared until a difference is found. Should the rules a–e be insufficient to prioritise the linkages at a given branching point, then they must be identical (e.g., only unknown linkages for each branch). Therefore, the sequence of the attached branches is taken as an additional criterion. Formally, each child residue of a branching point is regarded as a new reducing end residue and the set of rules from the residue comparison algorithm are applied to each branch. In this way a unique ordering of the branches can be achieved except for totally identical branches.

3.3. Undefined terminal subsequences: underdetermined subtree comparison algorithm

The encoding of undefined terminal subsequences—here called underdetermined terminal units (UND)—is handled separately from the description of the other topological features (Fig. 11C). Each UND is sorted by applying the set of rules from the residue and linkage comparison algorithm. Afterwards the reducing residues of all UND are compared with the residue comparison algorithm to define their order. If no decision is possible, then the compared underdetermined terminal units are identical. Consequently, the topology and linkages of the chains to which the UND are connected—the parent residues—have to be evaluated. Two additional rules to establish a correct order are defined:

(a) Comparison of the list of parent residues from each UND.
(b) Linkage comparison of parent linkages (linkage between UND and main graph) from each UND.

3.4. Alternative subtree comparison algorithm

Each alternative subtree of an alternative unit can consist of one or more residues (Fig. 11D) The residues contained in each alternative subtree are ordered using the residue comparison hierarchy described above. To encode alternative blocks uniquely, a comparison of the reducing residues is needed. The reducing end residues of the ALT-section are compared with the residue comparison algorithm.

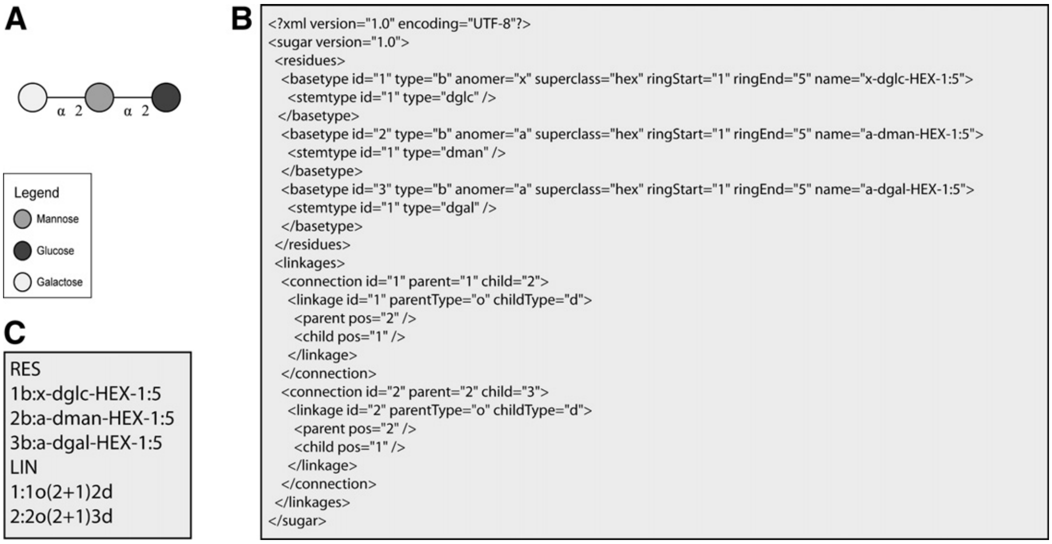
4. GlycoCT$_{\{XML\}}$

We have, mainly for reasons of saving figure space, concentrated on the condensed variant of GlycoCT. Equally important is the GlycoCT$_{\{XML\}}$ variant, as it facilitates computational handling of the sequences (Fig. 12). All properties encoded in GlycoCT are atomized in this variant into elements and attributes, and are thus readily available for extraction with standard methods for handling XML documents. The XML variant has been used in various applications in the EUROCarbDB project and has proven its value. The interested reader may refer to the XML schema in the Supplementary data.

5. First applications and results of GlycoCT

We have extracted and translated the monosaccharide namespace of CarbBank. A total of 49.897 entries contain 1.439 different names for monosaccharides. These structures contain a total amount of 241.280 monosaccharides. In the GlycoCT namespace this results in 474 different basetypes and 29 different substituents, reducing the number of distinct residues by 65% (Table 3).

There are two main reasons for the reduction in number of distinct residues: first of all the separation of monosaccharides into basetypes and substituents reduces the encoding space and secondly the unique encoding for monosaccharides eliminates redundancies. CarbBank did not ensure a strict naming for monosaccharides, resulting in different names for the same monosaccharide, especially when information is missing or trivial names are in use. For example, missing anomeric configurations could be directly encoded, or left out. The basetype 'x-dglc-hex-1:5' has been named in CarbBank as either '?-D-Glcp' or 'D-Glcp'. In addition



```
<?xml version="1.0" encoding="UTF-8"?>
<sugar version="1.0">
 <residues>
  <basetype id="1" type="b" anomer="x" superclass="hex" ringStart="1" ringEnd="5" name="x-dglc-HEX-1:5">
   <stemtype id="1" type="dglc" />
  </basetype>
  <basetype id="2" type="b" anomer="a" superclass="hex" ringStart="1" ringEnd="5" name="a-dman-HEX-1:5">
   <stemtype id="1" type="dman" />
  </basetype>
  <basetype id="3" type="b" anomer="a" superclass="hex" ringStart="1" ringEnd="5" name="a-dgal-HEX-1:5">
   <stemtype id="1" type="dgal" />
  </basetype>
 </residues>
 <linkages>
  <connection id="1" parent="1" child="2">
   <linkage id="1" parentType="o" childType="d">
    <parent pos="2" />
    <child pos="1" />
   </linkage>
  </connection>
  <connection id="2" parent="2" child="3">
   <linkage id="2" parentType="o" childType="d">
    <parent pos="2" />
    <child pos="1" />
   </linkage>
  </connection>
 </linkages>
</sugar>
```

```
RES
1b:x-dglc-HEX-1:5
2b:a-dman-HEX-1:5
3b:a-dgal-HEX-1:5
LIN
1:1o(2+1)2d
2:2o(2+1)3d
```

Figure 12. GlycoCT$_{\{XML\}}$. (A) Graphical representation of a linear trisaccharide. (B) The same structure in GlycoCT$_{\{condensed\}}$. (C) GlycoCT$_{\{XML\}}$ encoding of this structure. The basic definitions of GlycoCT are used, but every structural aspect is atomized using attributes of the main residue and linkage elements defining the sequence.

Table 3. All monosaccharides from CarbBank have been transcoded to the naming as introduced in GlycoCT

(A) Basetype	% of all basetypes
b-dglc-HEX-1:5	22.70
b-dgal-HEX-1:5	16.77
a-dman-HEX-1:5	12.56
a-dglc-HEX-1:5	6.66
a-dgro-dgal-NON-2:6|1:a|2:keto|3:d (sialic acid core structure)	4.65
a-lgal-HEX-1:5|6:d (L-fucose)	4.47
b-dman-HEX-1:5	4.36
a-dgal-HEX-1:5	3.08
x-dglc-HEX-x:x	2.98
a-lman-HEX-1:5|6:d (L-rhamnose)	2.56
(B) Substituent	% of all substituents
n-Acetyl	70.41
Sulfate	8.71
Amino	5.01
Methyl	4.56
Acetyl	3.46
n-Sulfate	1.85
Phosphate	1.56
Anhydro	1.23
n-Glycolyl	0.90
Phospho-ethanolamine	0.64

(A) Ten most frequently occurring basetypes in CarbBank structures.
(B) Ten most frequently occurring substituents.

CarbBank used trivial names inconsistently, leading to the expansion of monosaccharide names (e.g., 'B-D-6-DEOXY-GLCP4NAC' and 'B-D-QUIP4NAC'). Through a fully defined GlycoCT monosaccharide namespace, these problems are alleviated.

In order to test the comprehensiveness of the encoding scheme, a set of 10,000 randomly chosen carbohydrate sequences originating from CarbBank were translated to GlycoCT$_{\{XML\}}$ and GlycoCT$_{\{condensed\}}$. These structures are now publicly available at the homepage of the EUROCarbDB project (http://www.eurocarbdb.org/recommendations/encoding).

6. Conclusion and outlook

After the termination of funding for the Complex Carbohydrate Structure Database, the glycobiology area has experienced an era of more than a decade of fragmentation regarding digital description formats, with multiple initiatives developing incompatible structural encoding schemata tailored to their specific needs and applications. To overcome this unfavourable situation characterized by isolated knowledge, the first step towards data integration in glycomics is the definition of a 'glue' language, capable of encapsulating all structural features of glycans. With the GlycoCT encoding scheme we have defined such a language, which is a superset of the capabilities of all known sequence formats in glycobioinformatics. Furthermore, a number of enlargements for structurally underdetermined sequences, caused by both biological heterogeneity and analytical limitations, complete the GlycoCT format to a broadly usable tool in carbohydrate research. The block orientated approach combined with the canonical labelling will make GlycoCT capable of future development. Each residue and linkage in the carbohydrate sequence is addressable with a unique numerical identifier. This number is used for internal addressing purposes, but can also be used by external applications and has been already applied with the block concept. Another major improvement compared to the existing formats used in glycobioinformatics is the adoption of a consistent naming scheme based on IUPAC recommendations for the monosaccharides, which is of crucial importance for the long-term consistency of database projects. This controlled vocabulary can be easily maintained with automatic checking routines due to its complete machine readability. Present work in our group is aiming towards complete data integration in glycomics by merging all digital resources available.

The importance of a comprehensive and flexible encoding scheme to establish a global carbohydrate structure database has been emphasized by a whitepaper initiated by the NIH at the NIH meeting 'Frontiers in Glycomics' in September 2006.[14] All major database projects dealing with carbohydrate structures agreed to prospective use of the XML-based carbohydrate structure exchange format Glyde II developed by Will York of the Complex Carbohydrate Research Center, Athens, Georgia, USA. The group of Will York was interested in adopting the EUROCarbDB definitions of GlycoCT described in this paper. Finally, the collaboration, which improved our work, led to the integration of our definitions (controlled vocabulary for monosaccharides, connection table based description of the topology and definition of uncertainties) into the Glyde II standard. In such a way, an optimal conversion between the global exchange format Glyde II and the database optimized structure encoding format GlycoCT is guaranteed.

Following the overwhelming international affirmation to use one generally agreed data exchange format for glycan structures, it can be anticipated that both encoding schemata will be widely accepted by the community of glycoscientists.

Finally, we want to emphasize, that there is no need for glycoscientists to learn writing a GlycoCT description of a structure on their own. Actual software developments, such as GlycanBuilder (http://www.eurocarbdb.org/applications/structure-tools), provide graphical interfaces for rapid and intuitive structure drawing, editing and displaying. These tools can translate their internal encoding to the GlycoCT format and vice versa to guarantee a data transfer to databases and other software.

Acknowledgements

The authors like to thank all developers from the EUROCarbDB initiative for the fruitful discussions in the starting phase of this project, all database developers from the KEGG, BCSDB and CFG projects for their willingness for answering questions and the EU (6th Research Framework Programme, RIDS Contract No. 011952) and the German Research Foundation (DFG BIB 46 HDdkz 01-01) for financial support.

Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.carres.2008.03.011.

References

1. Doubet, S.; Albersheim, P. *Glycobiology* **1992**, *2*, 505.
2. Doubet, S.; Bock, K.; Smith, D.; Darvill, A.; Albersheim, P. *Trends Biochem. Sci.* **1989**, *14*, 475–477.
3. McNaught, A. *Carbohydr. Res.* **1997**, *297*, 1–90.
4. Bohne-Lang, A.; Lang, E.; Forster, T.; von der Lieth, C. W. *Carbohydr. Res.* **2001**, *336*, 1–11.
5. Cooper, C.; Harrison, M.; Wilkins, M.; Packer, N. *Nucleic Acids Res.* **2001**, *29*, 332–335.
6. Cooper, C.; Joshi, H.; Harrison, M.; Wilkins, M.; Packer, N. *Nucleic Acids Res.* **2003**, *31*, 511–513.
7. Banin, E.; Neuberger, Y.; Altshuler, Y.; Halevi, A.; Inbar, O.; Dotan, N.; Avinoam, D. *Trends Glycosci. Glycotechnol.* **2002**, *14*, 127–137.
8. Toukach, F.; Knirel, Y. In *New database of bacterial carbohydrate structures*, XVIII International Symposium on Glycoconjugates, Florence, Italy, 2005; pp 216–217.
9. Aoki, K.; Yamaguchi, A.; Ueda, N.; Akutsu, T.; Mamitsuka, H.; Goto, S.; Kanehisa, M. *Nucleic Acids Res.* **2004**, *32*, 267–272.
10. Sahoo, S.; Thomas, C.; Sheth, A.; Henson, C.; York, W. *Carbohydr. Res.* **2005**, *340*, 2802–2807.
11. Kikuchi, N.; Kameyama, A.; Nakaya, S.; Ito, H.; Sato, T.; Shikanai, T.; Takahashi, Y.; Narimatsu, H. *Bioinformatics* **2005**, *21*, 1717–1718.
12. Ceroni, A.; Dell, A.; Haslam, S. M. *Source Code Biol. Med.* **2007**, *2*, 3.
13. Stephen, E.; Stein, S. R. H. Dmitrii Tchekhovskoi. In *An Open Standard for Chemical Structure Representation: The IUPAC Chemical Identifier*. In *Proceedings of the 2003 International Chemical Information Conference, Nimes, 2003*; Infonortics: Nimes, 2003; pp 131–143.
14. Packer, N. H.; von der Lieth, C. W.; Aoki-Kinoshita, K. F.; Lebrilla, C. B.; Paulson, J. C.; Raman, R.; Rudd, P.; Sasisekharan, R.; Taniguchi, N.; York, W. S. *Frontiers in glycomics: Bioinformatics and biomarkers in disease*. In An NIH White Paper prepared from discussions by the focus groups at a workshop on the NIH campus, Bethesda, MD, September 11–13, 2006. *Proteomics* **2008**, *8*, 8–20.
15. Raman, R.; Venkataraman, M.; Ramakrishnan, S.; Lang, W.; Raguram, S.; Sasisekharan, R. *Glycobiology* **2006**, *16*(5), 82–90.
16. Glycominds Online introduction to LinearCode. http://www.glycominds.com/index.asp?menu=Research&page=glycoit (10.01.2007).