Journal of Biomolecular NMR, **15:** 169–172, 1999.
KLUWER/ESCOM

StarDOM: From STAR format to XML

Jens P. Linge*, Michael Nilges** & Lutz Ehrlich***
European Molecular Biology Laboratory, Meyerhofstrasse 1, D-69012 Heidelberg, Germany

Received 22 June 1999; Accepted 10 August 1999

Key words: BioMagResBank, document object model, mmCIF, STAR, XML

Abstract

StarDOM is a software package for the representation of STAR files as document object models and the conversion of STAR files into XML. This allows interactive navigation by using the Document Object Model representation of the data as well as easy access by XML query languages. As an example application, the entire BioMagResBank has been transformed into XML format. Using an XML query language, statistical queries on the collected NMR data sets can be constructed with very little effort. The BioMagResBank/XML data and the software can be obtained at http://www.nmr.embl-heidelberg.de/nmr/StarDOM/

In order to share the results of their experiments, NMR spectroscopists must encode their data in a form that can be understood and analyzed by computers. Typically, many hours are wasted by manually transforming data from a storage format into one that is easily queried or visually represented. It would be highly desirable if every scientist could use readily available, easy-to-use tools to accomplish these tasks.

Recently, Self-Defining Text Archive and Retrieval (STAR) (Hall, 1991) has emerged as a standard data format for structural biology data. Several scientific databases (e.g. the Protein Databank, BioMagResBank or the Cambridge Structural Database) use the STAR format to store structural, crystallographic diffraction and NMR data. A growing number of programs (e.g. CNS (Brünger et al., 1998), NMRView (Johnson and Blevins, 1994), MODELFREE (Palmer, 1999)) can utilize the STAR format for their respective data output.

To extract information from STAR data files, specific library routines (e.g. STARLIB (Mading, 1999)) or the STAR_BASE query program (Spadaccini and Hall, 1994) are provided. However, no software package exists to extract STAR information from a scripting language (so-called parsers).

The eXtensible Markup Language (XML) (The World Wide Web Consortium, 1999a) is a standard for semantic markup of data independent of a particular application domain. This independence implies that many parties develop different parsers; software is thoroughly tested across specific problem domains. Besides, parser implementations exist in a wealth of programming languages (including scripting languages), which means more freedom for the scientist wishing to analyze certain data.

Where could the use of XML ease data access and analysis compared to STAR? One reason lies in the use of *standard XML parsers.* As XML is used in a broad spectrum of application domains, many different parser implementations are available. This enables the programmer to choose a well-tested parser for a given problem.

Another advantage of XML is the existence of *standard XML viewers/editors*. With the advent of the next generation of Web browsers, XML will be supported as a standard format for data exchange over the web. Hierarchical information contained in a web file can be displayed and edited in general-purpose XML viewers (such as Microsoft's Internet Explorer 5) or editors (such as IBM's Xeena (Ifergan

*E-mail: linge@embl-heidelberg.de
**E-mail: nilges@embl-heidelberg.de
***To whom correspondence should be addressed. E-mail: ehrlich@embl-heidelberg.de

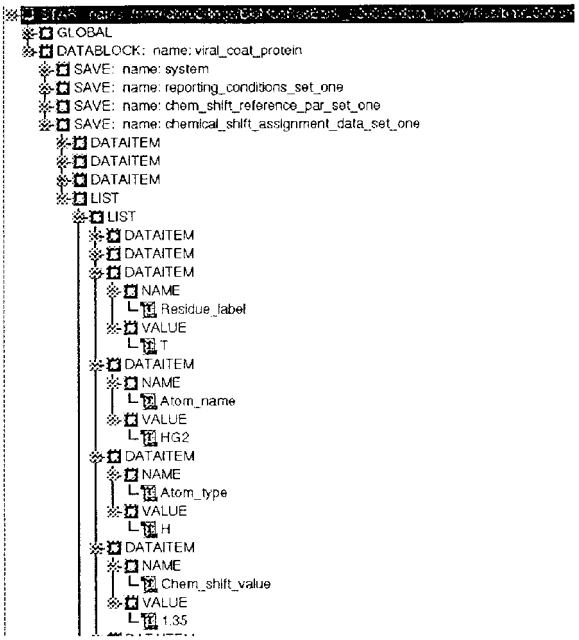

Figure 1. Viewing and editing of BioMagResBank files with Xeena.

```
<!ELEMENT STAR (GLOBAL|DATABLOCK|SAVE|LIST|DATAITEM)*>
<!ATTLIST STAR name CDATA #REQUIRED>
<!ELEMENT GLOBAL (LIST|SAVE|DATABLOCK|DATAITEM)*>
<!ELEMENT DATABLOCK (SAVE|LIST|DATAITEM)*>
<!ATTLIST DATABLOCK name CDATA #REQUIRED>
<!ELEMENT SAVE (LIST|DATAITEM)*>
<!ATTLIST SAVE name CDATA #REQUIRED>
<!ELEMENT LIST (LIST|DATAITEM)*>
<!ELEMENT DATAITEM (NAME, VALUE)>
<!ELEMENT NAME (#PCDATA)*>
<!ELEMENT VALUE (#PCDATA)*>
```

Figure 2. Document Type Definition (DTD).

et al., 1999)). Figure 1 illustrates the visual display of BioMagResBank (BMRB) data.

Additionally, *XML query languages* make it easier to formulate ad hoc queries in a structured way. Two proposed standards are XQL (Robie, 1999) and XML-QL. For both proposals, working prototype implementations are available.

Finally, the *validity of documents* can be checked. XML documents can be validated against a document type definition (DTD). In this way, the integrity of the data can be checked as the document is generated.

We have created a software package that transforms STAR files into XML. This translation makes it possible to utilize standard tools for data extraction, navigation, querying and validation. The first application of the codes consisted in transforming the entire BMRB data set into XML. After describing the transformation procedure, we will demonstrate the use of XQL, an XML query language for extracting biological data by ad hoc queries.

```
global_
 _Submission_form_date      1996 04 02
 loop_                  # Data types included in the entry
     _Data_type
     _Data_value_count
     '1H chemical shifts'     1241
     '13C chemical shifts'    827
 stop_
```

a

```
<STAR name = 'example'>
    <GLOBAL>
        <DATAITEM>
            <NAME>Submission_form_date</NAME>
            <VALUE>1996 04 02</VALUE>
        </DATAITEM>
        <LIST>
            <LIST>
                <DATAITEM>
                    <NAME>Data_type</NAME>
                    <VALUE>1H chemical shifts</VALUE>
                </DATAITEM>
                <DATAITEM>
                    <NAME>Data_type_number</NAME>
                    <VALUE>1241</VALUE>
                </DATAITEM>
            </LIST>
            <LIST>
                <DATAITEM>
                    <NAME>Data_type</NAME>
                    <VALUE>13C chemical shifts</VALUE>
                </DATAITEM>
                <DATAITEM>
                    <NAME>Data_type_number</NAME>
                    <VALUE>827</VALUE>
                </DATAITEM>
            </LIST>
        </LIST>
    </GLOBAL>
</STAR>
```

b

Figure 3. A short STAR document (a) and its XML counterpart (b).

Conversion of STAR files to XML is performed in a two-stage process. First, the tree structure inherent in the STAR data files is unrolled into an object tree structure. In a second stage, this tree is mapped to a Document Object Model (DOM (The World Wide Web Consortium, 1999b)), which describes a tree representation of XML documents. This DOM is then rendered as XML. The underlying DTD is shown in Figure 2. So far, no NMR specific information is contained in the tag set. As soon as a complete data dictionary is available, an NMR-specific DTD will be developed in close collaboration with the BMRB.

The first stage of the mapping process is illustrated in Table 1, which shows equivalent fragments of STAR and XML code. Figure 3 shows a comparison of corresponding STAR and XML data.

The complete transformed BMRB data set can be accessed via the StarDOM homepage at http://www.nmr.embl-heidelberg.de/nmr/StarDOM.

The extraction of subsets of information contained in an NMR data file can be a time consuming task, particularly if the data requires preprocessing or many such queries have to be performed. XML query languages can alleviate the need of writing elaborate query programs to extract complicated information from STAR files. In the following example, an XML query language (XQL (Robie, 1999)) will be used to extract the hydrogen chemical shifts of all aspartic acid residues. XQL filters certain XML elements

Table 1. Mapping scheme used to convert STAR files into XML

STAR tag	Corresponding XML tag
global_	<GLOBAL></GLOBAL>
data_*blockcode*	<DATA name='*blockcode*'></DATA>
save_*framecode*	<SAVE name='*framecode*'></SAVE>
loop_	<LIST>
_var1	<LIST>
_var2	<DATAITEM>
1 2	<NAME>*var1*</NAME><VALUE>*1*</VALUE>
3 4	</DATAITEM>
stop_	<DATAITEM>
	<NAME>*var2*</NAME><VALUE>*2*</VALUE>
	</DATAITEM>
	</LIST>
	<LIST>
	<DATAITEM>
	<NAME>*var1*</NAME><VALUE>*3*</VALUE>
	</DATAITEM>
	<DATAITEM>
	<NAME>*var2*</NAME><VALUE>*4*</VALUE>
	</DATAITEM>
	</LIST>
	</LIST>
_dataname datavalue	<DATAITEM>
	<NAME>*dataname*</NAME><VALUE>*datavalue*</VALUE>
	</DATAITEM>

from a document. The declarative query syntax describes how the qualifying elements should look. To gradually build up the query string, we first ask for all LIST elements which contain aspartic residue data. The respective XQL substring is

```
//LIST[/NAME='Residue\_label'
        and DATAITEM/VALUE='D']
```

This expression will return list elements that contain a DATAITEM element whose NAME element has the value Residue_label and a DATAITEM element whose VALUE element has the value D. This will filter out all aspartic acid sublists. As we only want to extract hydrogen chemical shifts, we filter the returned lists further by appending

```
[DATAITEM/NAME='Atom\_type'
        and DATAITEM/VALUE='H']
```

to the previous query string. Only those sublists which contain hydrogen information remain. Extraction of the chemical shift information can be done by extracting the DATAITEM elements which contain chemical shifts (i.e. they have a NAME element with the value Chem_shift_value). From those elements the numerical value of the chemical shift can be accessed by appending

```
/DATAITEM[NAME='Chem_shift_value']/VALUE
```

to the above query. The entire query string looks like

```
//LIST[DATAITEM/NAME='Residue_label'
        and DATAITEM/VALUE='D']
[DATAITEM/NAME='Atom_type'
        and DATAITEM/VALUE='H']
/DATAITEM[NAME='Chem_shift_value']/VALUE
```

This query can be executed via the command line by typing

```
%>perl xql.pl
"//LIST[DATAITEM/NAME='Residue_label'
        and DATAITEM/VALUE='D']
[DATAITEM/NAME= 'Atom_type'
        and DATAITEM/VALUE='H']
/DATAITEM[NAME= 'Chem_shift_value']/VALUE"
        bmr749.xml
```

which will return the respective chemical shifts. In this way, statistical queries against the complete BMRB data can be developed in very little time. Much more complicated queries can be performed by concatenating the respective XQL query strings. Future implementations could include a graphical querying front-end integrated with an information display as in Figure 1.

A current prototype for a revised StarDOM tag set utilizes the STAR keywords directly. The use of a CHEM_SHIFT keyword, for example, will further ease query formulation as well as visual representation of the data.

With the use of XML as the data format for structural biology, new ways of interacting with the data become possible. XML editors and query languages make the harvesting of biological data much easier than before. New developments in connecting XML documents (XLink) as well as new database technology to store and serve XML content will lead to further improvement in how we archive and disseminate our data.

Acknowledgements

We thank E. Ulrich, J. Markley (BioMagResBank) and G. Barton (EBI) for discussions. J.P.L. thanks the Boehringer Ingelheim Fonds for financial support through a Ph.D. fellowhip.

References

Brünger, A.T., Adams, P.D., Clore, G.M., Delano, W.L., Gros, P., Grosse-Kunstleve, R.W., Jiang, J.S., Kuszewski, J., Nilges, M., Pannu, N.S., Read, R.J., Rice, L.M., Simonson, T. and Warren, G.L. (1998) *Acta Crystallogr.*, **D54**, 905–921.

Hall, S.R. (1991) *J. Chem. Inf. Comput. Sci.*, **31**, 326–333.

Ifergan, S.S., Maarek, Y.S. and Ur, S. (1999) http://www.alphaworks.ibm.com/tech/xeena.

Johnson, B.A. and Blevins, R.A. (1994) *J. Biomol. NMR*, **4**, 603–614.

Mading, S. (1999) http://www.bmrb.wisc.edu/sb_lib/starlib/index.html.

Palmer, A.G. (1999) MODELFREE, manuscript in preparation.

Robie, J. (1999) http://www.texcel.no/whitepapers/xql-design.html.

Spadaccini, N. and Hall, S.R. (1994) *J. Chem. Inf. Comput. Sci.*, **34**, 509–516.

The World Wide Web Consortium (1999a) http://www.w3.org/TR/REC-xml.

The World Wide Web Consortium (1999b) http://www.w3.org/TR/WD-DOM.