BIOINFORMATICS APPLICATIONS NOTE Vol. 18 no. 9 2002
Pages 1280–1281

An ontology driven architecture for derived representations of macromolecular structure

Douglas S. Greer[1,*], John D. Westbrook[2] and Philip E. Bourne[1]

[1]The Research Collaboratory for Structural Bioinformatics at the San Diego Supercomputer Center, 9500 Gilman Drive, University of California, San Diego, La Jolla CA 92093-0505, USA and [2]at Rutgers, the State University of New Jersey, Department of Chemistry and Chemical Biology, 610 Taylor Road, Piscataway, NJ 08854-8087, USA

Received on January 23, 2002; revised on February 20, 2002; accepted on March 25, 2002

ABSTRACT

Summary: An object metamodel based on a standard scientific ontology has been developed and used to generate a CORBA interface, an SQL schema and an XML representation for macromolecular structure (MMS) data. In addition to the interface and schema definitions, the metamodel was also used to generate the core elements of a CORBA reference server and a JDBC database loader. The Java source code which implements this metamodel, the CORBA server, database loader and XML converter along with detailed documentation and code examples are available as part of the OpenMMS toolkit.

Availability: http://openmms.sdsc.edu

Contact: dsg@sdsc.edu

The speed and utility of bioinformatics software depends to a great extent on the underlying data access method and organization. CORBA, SQL and XML expressions of MMS data were designed to address the needs of a wide range of applications and to provide efficient, high performance access to these data for several common use cases. These cases include many types of services and applications that require fast and flexible access to molecular structure information provided by the Protein Data Bank (PDB; Berman *et al.*, 2000).

The Object Management Group (OMG) has recently voted to adopt the CORBA IDL Macromolecular Structure specification derived from the MMS metamodel. This specification, which is available at http://www.omg.org/technology/documents/formal/macro_molecular.htm, defines an open, standardized object-oriented Application-Programming Interface (API) that allows direct access by remote programs to the binary data structures of the PDB.

OpenMMS

The overall view of the OpenMMS toolkit is shown in Figure 1. In the upper portion of this figure, the flow of information that describes the structure of the MMS data is shown using thin black arrows. In the lower portion, the flow of actual MMS data is shown using thick gray arrows.

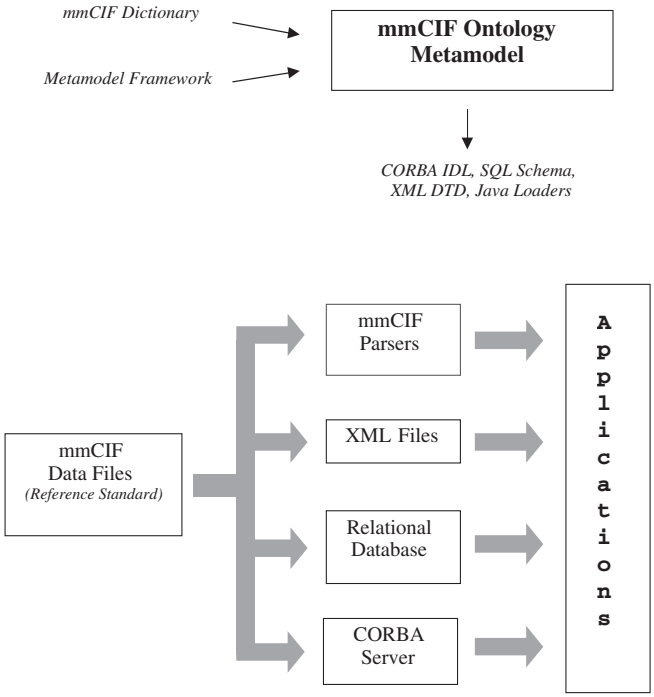

Fig. 1. The OpenMMS metamodel and dataflow.

The OpenMMS ontology metamodel is derived from the scientific ontology developed by the International Union of Crystallography (IUCr) and based on the macromolecular Crystallographic Information File (mmCIF) standard, which encompasses both the mmCIF dictionary and the mmCIF data files (Westbrook and Bourne, 2000; Bourne *et al.*, 1997). Although the dictionary was written in STAR format (Hall, 1991), OpenMMS extracts the needed information from the dictionary such that the ontology and its

*To whom correspondence should be addressed.

derivations are independent of STAR or any other particular file format.

The 'glue' that ties together the CORBA, XML and SQL representations of the MMS data is the data dictionary which defines the semantics for the mmCIF data files. Discrepancies in the semantics of the derived models are resolved by consulting this data dictionary. An advantage of using a metamodel approach is that any corrected or updated data can be automatically propagated to the derived representations.

With a flat file representation, users are required to retrieve and parse the complete file in order to use even a small portion of it. However, implementations of the CORBA and SQL interfaces allow applications to retrieve a single MMS data element from a remote server and import it for local use. Moreover the data is already in binary form, so the time consuming parsing and ASCII to binary conversions that are performed repeatedly by each application, are not necessary.

Representations of macromolecular structure data

Among the four methods for obtaining MMS data that are shown in Figure 1, the mmCIF parser is the most general purpose and provides the most direct access to the underlying data. The OpenMMS parser uses the 'Builder' design pattern (Gamma *et al.*, 1994) that allows a separation between the parsing of a file and its representation so that the same construction process can be used to create different representations. Using this pattern, the application creates a subclass of the parsers Builder class and overrides its methods with code that examines and stores data values of interest into a data structure that it defines.

A relational database that supports an SQL-92 compatible interface provides an appropriate API for many applications, particularly ones that require extensive string searches.

XML is a simple, powerful and widely used standard for interchanging data. However, the use of opening and closing tags around each data value results in a file size approximately ten times larger than the corresponding mmCIF data file which uses a single white space separator between data values in a loop structure. Nevertheless, XML files can serve as the concrete exchange syntax between applications that are able to handle the large file sizes.

The CORBA server potentially provides the highest performance access to MMS data. This object-oriented interface is used to define structures independent of platform and programming language, and may be optimized to copy binary data quickly and efficiently across the network. Additional applications using this server are currently under development and will be included in future OpenMMS releases.

Applications are always limited by the computational power and I/O bandwidth of the available hardware and by the time available for software development and optimization. An ontology driven architecture can transform and present scientific data in ways that reduce the burden on software developers while at the same time increasing software efficiency, functionality and performance.

ACKNOWLEDGEMENTS

The authors wish to thank David Benton, Helen Berman, Ward Fleri, Gary Gilliland, Karl Konnerth, Martin Senger, Lynn Ten Eyck, Helge Weissig and Sheila Alessi for their suggestions and moral support. The members of the Research Collaboratory for Structural Bioinformatics (htp://www.rcsb.org), which manage the Protein Data Bank, Rutgers University, the San Diego Supercomputer Center and the National Institute of Standards and Technology. Support for this effort was provided through the Protein Data Bank (Grant DBI-9814284) and the National Partnership for Advanced Computational Infrastructure (Grant ACI-9619020).

REFERENCES

Berman,H.M., Westbrook,J.D., Feng,Z., Gilliland,G., Bhat,T.N., Weissig,H., Shindyalov,I.N. and Bourne,P.E. (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

Bourne,P.E., Berman,H.M., McMahon,B., Watenpaugh,K., Westbrook,J. and Fitzgerald,P.M.D. (1997) The macromolecular CIF dictionary. *Meth. Enzymol.*, **227**, 571–590.

Gamma,E., Helm,R., Johnson,R. and Vlissides,J. (1994) *Design Patterns, Elements of Reusable Object-Oriented Software*. Addison-Wesley, Reading, MA.

Hall,S.R. (1991) The STAR file: a new format for electronic data transfer and archiving. *J. Chem Inf. Comput. Sci.*, **31**, 326–333.

Westbrook,J.D. and Bourne,P.E. (2000) STAR/mmCIF: an extensive ontology for macromolecular structure and beyond. *Bioinformatics*, **16**, 159–168.