

Comprehensive model for formalized description, visualization and simulation of biological systems

Fedor Kolpakov

Biosoft.Ru, Novosibirsk, Russia.

Design Technological Institute of Digital Techniques SB RAS, Novosibirsk, Russia.

e-mail: fedor@biosoft.ru

Keywords: systems biology, graphic notation, meta model, graph, Java

BioUML is integrated extensible workbench for systems biology (<http://www.biouml.org>). Its core is BioUML meta model that provides a formalism for comprehensive description, visualization and simulation of wide range of biological and other complex systems. The suggested model is problem domain neutral and splits the system description into three interconnected levels: 1) graph structure - the system structure is described as compartmentalized graph; 2) database level - each graph element can contains reference to database object, additionally arbitrary attributes can be associated with graph element; 3) executable model - any graph element can be element of executable (mathematical) model, for example it can be variable or equation.

Taking into account problem domain specificity we introduce concept of diagram type, diagram view builder to visualize the graph structure as diagram using problem domain specific visual syntax and semantic controller to provide semantic integrity during the diagram editing. The module concept allows to developer define new diagram types and incorporate other databases on biological pathways into BioUML framework.

Introduction

Sydney Brenner, 2002 Nobel Prize winner says: "We now have unprecedented ability to collect data about nature but there is now a crisis developing in biology, in that completely unstructured information does not enhance understanding. We need a framework to put all of this knowledge and data into — that is going to be the problem in biology. We've reached the stage where we can't talk to each other — we've all become highly specialized. We need a framework, a framework where people can come back to us and say, 'Yes, I understand.' Driving toward that framework is really the big challenge." (Bradford, 2003).

BioUML – Biological Universal Modeling Language – is a step in this direction. Main challenge of bioinformatics is producing of a “book of life” i.e. model of the world in biological terms. BioUML is imagined as a language to write this book, it suggests an approach for comprehensive formalized description of wide range of biological and other complex systems.

The core of BioUML workbench is meta model that provides an abstract layer for comprehensive formal description of wide range of biological and other complex systems. Content of databases on biological pathways or SBML models can be expressed in terms of meta model and then can be visualized and edited as diagram by BioUML diagram editor, simulated using MATLAB or BioUML simulation engine, etc.

Meta model levels

BioUML meta model splits model description to three levels:

1. **Graph structure.** This level is used to present structure of biological system as a clustered graph that further can be visualized as a diagram.
2. **Database level.** Each diagram element contains 'kernel' that is an object from a concrete database. By this way we wrap database content into a set of objects that are associated with diagram elements. Additionally arbitrary attributes can be associated with graph element, it can be useful for dynamic linking of annotations with diagram elements.
3. **Executable model.** Any diagram element can be element of executable (mathematical) model, for example it can be variable or equation. Each element of executable model should implement `Role` interface and should be associated with some diagram element. Now we have implemented support for simulating biological systems using system of ordinary differential equations. Latter we plan implement support for Petri nets, Boolean networks and Markov chains.

Diagram markup language

Special BioUML diagrams markup language (DML) is developed to store BioUML meta model instance in XML format. Diagram structure description is divided into two parts:

1. diagram structure model - it describes the graph structure, location of diagram elements and database objects ('kernels') that are associated with diagram elements.
2. executable model - stores mathematical model associated with graph.

Its detailed description is available at: <http://www.biouml.org/dml.shtml>.

Problem domain issues

The suggested meta model is domain neutral and can be used for formalized description, visualization and simulation not only biological but for other complex systems: ecological systems, financial systems, business process, etc. Problem domain is taken into account by following way:

- database content - diagram elements are associated with problem domain specific database content.
- diagram view builder – is used to build graph view that take into account problem domain peculiarity, for example biological pathway diagram view builder uses specific views for different pathway elements: proteins are shown as circles and substances as squares.
- semantic controller - provides semantic integrity of the diagram during its editing. It takes into account problem domain constraints, for example if some specie is removed on biological pathway diagram, all related reactions should be removed too.
- filters – hide or highlight diagram elements according to some selection criteria, for example to according gene expression specificity or expression level.

Diagram type concept

To take into account different diagram types and problem domain specificity we have introduced the `DiagramType` concept that defines `DiagramViewBuilder` to generate specific diagram view and `SemanticController` to provide semantic control during the diagram editing (for example, if some substance is removed from the diagram, all related reactions should be removed too). `DiagramType` also defines what classes (database objects) can be added to the diagram as nodes or edges.

Module concept

The module concept allows to developer define new diagram types and incorporate other databases on biological pathways into BioUML framework. The module defines mapping of database content into diagram elements and diagram types that can be used with the database.

BioUML workbench includes standard module type intended for modeling of metabolic pathways, signal transduction pathways and gene networks. The module defines most common biological data types (gene, protein, RNA, substance, reaction, etc.) and they mapping into simple text database as well as three diagram types that allows user to describe his biological system on several semantic levels:

1. Ontology diagram type is used to describe semantic relationships between system components, system states, and related problem domain concepts.
2. Pathway diagram type is used for formalized description of biological pathway structure. This information is presented in the form of diagram similar with GeneNet database.
3. Pathway simulation diagram type is extension of pathway structure diagram, where variables are associated with graph nodes and differential equations with graph edges. This allows to BioUML workbench automatically generate mathematical model of the system and simulate its dynamics.

Now we are developing modules for GeneNet database (Kolpakov et al., 1998), KEGG/Pathways (Kanehisa et al., 2002) and TRANSPATH (Schacherer et al., 2001) databases. Special module (http://www.biouml.org/plugin_sbml.shtml?overview) was developed for SBML models. SBML models have not information about species layout, thus some very simple layout algorithm is used by SBML plug-in for initial species and reactions layout. Further we plan to provide models with hand made layout and optimize them for BioUML workbench.

Discussion

There are a number of ongoing efforts with similar goals as those of BioUML. The most similar approach was implemented in SBML (Hucka et al., 2003) and CellML (Physiome Sciences, 2001). SBML and CellML are XML-based markup languages designed for storing and exchanging computer-based biological models. They include facilities for representing model structure, mathematics and additional information for database storage and search. The CellML constructs tend to be more abstract and general than those in SBML and describes the structure and underlying mathematics of cellular models in a very general way. By contrast, SBML is closer to the internal object model used in model analysis software.

General approach implemented in BioUML, SBML and CellML is quite similar, so SBML models can be mapped into BioUML models and we are planning to develop module that will map CellML models into BioUML models, however this is more complicated task. Nevertheless there are some differences between BioUML meta model and SBML and CellML approach that are summarized in table 1.

Table 1.

Feature	BioUML meta model	SBML, CellML
Implementation	Java class library. Diagram Model Language is only one of the possible means to store the model.	XML languages
Biological object description	Diagram element contains reference to corresponding object in the database.	Diagram contains complete description for each element. If several diagrams contain the same object then its description will be duplicated.
Integration with existing biological databases	module concept	not implied
Visual modeling	was designed for visual modeling	complicated
Mathematical model	facultative	implied

While the suggested approach suites well for modeling of biological pathways and many other biological systems, it has some limitations: it does not take into account geometric properties of the system that is essential for some biological models, for example the heart model. Now we are investigating the possibilities how we can extend BioUML model to overcome this limitation. Current versions of SBML and CellML have the same limitation that they plan to overcome too. It is planned to add geometric information in SBML level 3. CellML refers to FieldML that appropriate for storing spatial distribution of parameters inside compartments in CellML.

Acknowledgments

Part of this work was supported by the grant of Volkswagen-Stiftung (I/75941). Author is grateful to Sergey Zhatchenko and Alexander Kel for useful comments and discussions, as well as to Igor Tyazhev, Vlad Zhvaleev and Oleg Onegov for technical support.

References

- Bradford R. 2003. *A Man, A Worm, and A Nobel*. Salk Signals, 5(2):12-17.
- Hucka M., Finney A., Sauro H. M., Bolouri H., et al. (2003) *The Systems Biology Markup Language (SBML): A Medium for Representation and Exchange of Biochemical Network Models*. Bioinformatics, **19(4)**:524-531.
- Kanehisa M., Goto S., Kawashima S., and Nakaya, A. (2002) *The KEGG databases at GenomeNet*. Nucleic Acids Res., **30**, 42-46.
- Kolpakov F.A., Ananko E.A., Kolesov G.B. and Kolchanov N.A. (1998) *GeneNet: a database for gene networks and its automated visualization*. Bioinformatics, **14(6)**, 529-537.
- Physiome Sciences, I. (2001). CellML™ home page. Available via the World Wide Web at <http://cellml.org/>.
- Schacherer F., Choi C., Gotze U., Krull M., Pistor S., Wingender E. (2001) *The TRANSPATH signal transduction database: a knowledge base on signal transduction networks*. Bioinformatics, **17(11)**, 1053-1057.