W150–W154 *Nucleic Acids Research, 2010, Vol. 38, Web Server issue* *Published online 11 May 2010*
doi:10.1093/nar/gkq352

VISIBIOweb: visualization and layout services for BioPAX pathway models

Alptug Dilek[1,2], Mehmet E. Belviranli[1,2] and Ugur Dogrusoz[1,2,*]

[1]Center for Bioinformatics and [2]Computer Engineering Department, Bilkent University, Ankara 06800, Turkey

Received February 2, 2010; Revised April 5, 2010; Accepted April 22, 2010

ABSTRACT

With recent advancements in techniques for cellular data acquisition, information on cellular processes has been increasing at a dramatic rate. Visualization is critical to analyzing and interpreting complex information; representing cellular processes or pathways is no exception. VISIBIO*web* is a free, open-source, web-based pathway visualization and layout service for pathway models in BioPAX format. With VISIBIO*web*, one can obtain well-laid-out views of pathway models using the standard notation of the Systems Biology Graphical Notation (SBGN), and can embed such views within one's web pages as desired. Pathway views may be navigated using zoom and scroll tools; pathway object properties, including any external database references available in the data, may be inspected interactively. The automatic layout component of VISIBIO*web* may also be accessed programmatically from other tools using Hypertext Transfer Protocol (HTTP). The web site is free and open to all users and there is no login requirement. It is available at: http://visibioweb.patika.org.

INTRODUCTION

Available knowledge on cellular processes has been increasing at a rapid rate. However, the incomplete, fragmented and incompatible nature of pathway information makes the representation and integration of pathways rather complicated. Thus, constructing a knowledge base of cellular pathways to effectively use such data requires a strong representation at the model and presentation levels. Such a knowledge base can then act as a blueprint for simulations and other analysis methods, improving our ability to understand and predict behavior of cell (1).

Within the past decade, some standards and notations (1–7) have been created in this direction. A recent estimation of the number of pathway databases stands at more than 300 (8). Still, current bioinformatics infrastructure lacks software tools for effectively querying these databases and for visualizing and analyzing the resulting pathways.

Among these standards, a continuing community effort called BioPAX (2) has made great progress in building a standard format for exchanging biological pathway data. Many prominent pathway databases (3,5,7,9–12) have already made their data available in this standard. Another community effort, the Systems Biology Graphical Notation (SGBN) (6), has been recently published as the first standard notation to present pathways in a graphically rich format.

Both of these standards use the concept of 'compound graphs' extensively. Compound graphs are an extension of graph-based representation, where a member of a biological network may recursively contain a subnetwork of other pathway elements. These structures can be used for representing molecular complexes and subcellular locations as well as subpathways, and help to manage the inherent complexity of pathways by interactively decomposing them into distinct components or modules (1,13).

A number of pathway visualization tools with a variety of foci (14–17) that have been built in recent years, have limited support for the aforementioned modeling and viewing standards. VISIBIO*web*, one of the few software applications supporting the new standard notation SBGN, is a free, open-source pathway visualization and layout service for pathway models in BioPAX format. It is also one of the rare tools, if not the only tool, that can handle compound structures and work within a web browser without the need for extra plug-ins or excessive downloads. Furthermore, to our knowledge, it is the only public programmatically accessible layout service for on-demand creation of well-organized maps of biological pathways.

VISIBIO*web* ARCHITECTURE

The VISIBIO*web* user side is a thin-client JavaScript application based on Google Maps API (http://code.google.com/apis/maps) and the server side is made up of a

*To whom correspondence should be addressed. Tel: +90 312 290 1612; Fax: +90 312 266 4047; Email: ugur@cs.bilkent.edu.tr

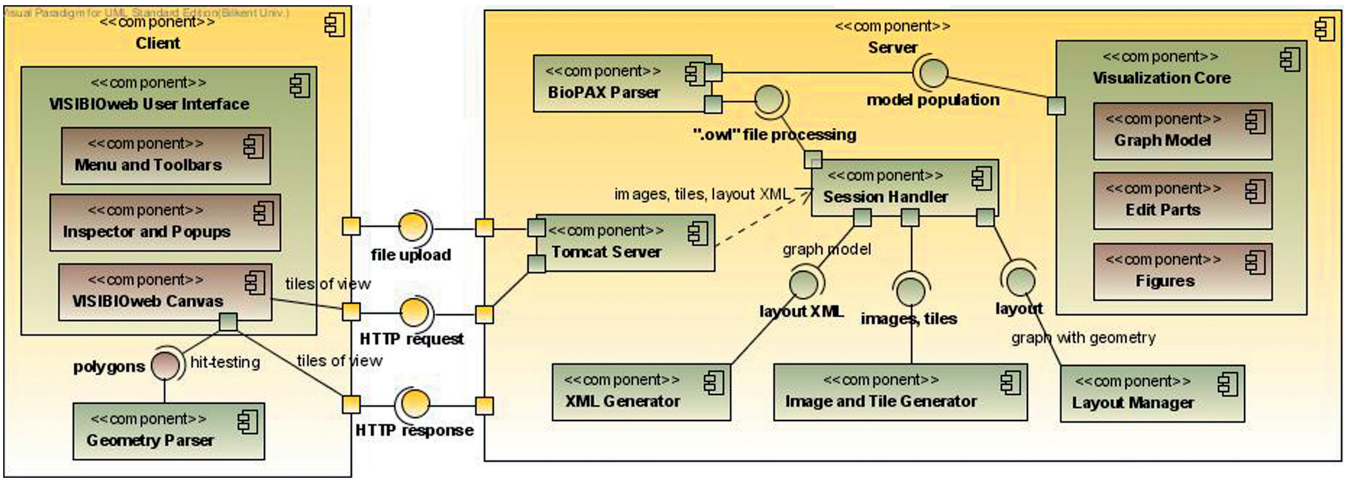

Figure 1. High-level architecture of VISIBIO*web* depicted with a UML component diagram.

component based on the Eclipse Graphical Editing Framework, version 3.1 (http://www.eclipse.org/gef). It does not require any particular application or plug-in installed other than a standard browser (e.g., Firefox 3.0, IE 8.0, and Safari 4.0 or later versions) running JavaScript and an Internet connection.

Figure 1 shows the architecture of VISIBIO*web* at a high-level. Communication between the client and the server is initiated with a file upload event through a web browser. Requests from the client side arrive at an Apache Tomcat server. Tomcat executes requested Java Server Pages (JSP) files and assigns the tasks to the Session Handler, which is responsible for managing all server-side logic by delegating these tasks to other server components for a user until the end of that session. These components are implemented in Java and JSP.

The client side is mainly composed of user interface components, the most important one being a canvas, which is essentially a customized Google Map. The VISIBIO*web* canvas is responsible for displaying the view constructed on the server side properly. It also detects various user actions and events. The Geometry Parser unit parses the geometry of the XML file sent from the server. The output of this component is a list of polygons, which is added on top of the VISIBIO*web* canvas and facilitates hit-testing for proper pathway object inspection.

USING VISIBIO*web*

The graphical user interface (GUI) of VISIBIO*web* (Figure 2) is organized into a menu bar, a toolbar and a canvas with controls. The canvas is where the pathway maps are displayed. On the lower right corner of the canvas is a collapsible overview window. An object-properties window pops up as the user clicks on a particular pathway object for inspection, and typically contains the object's name, ID, data source, external links to related web pages and a description.

Input

A pathway model stored in a BioPAX (level 2) formatted file may be loaded into VISIBIO*web* . The content and appearance of the resulting views may be configured by the user through various display (see below) and layout options (Figure 3).

Compound Visualization. Along with molecular complexes, compound nodes may be used for complexity management to visually represent either 'Pathways' (default) or 'Compartments' in a pathway view.

Degree of Separation. Ubiquitous molecules, which typically participate in many different biological activities, have a relatively constant concentration and do not transmit a signal. For instance, a ubiquitous molecule such as ATP might be involved in hundreds, if not thousands, of reactions. Such high-degree nodes can easily clutter drawings if not cloned for each involvement in a pathway. This option specifies the degree after which a molecule is considered to be 'ubiquitous'. Thus, a molecule with degree 20 will be drawn 20 times (as 20 separate degree-one nodes) if this option has a value ≤ 20.

Allow partial model view. A pathway model often consists of a very large number of molecules and reactions. The user might want to manage this complexity by inspecting one subpathway (or a union of subpathways) of interest at a time. When this option is enabled (the default), the upload operation shows the list of pathways in the input file.

Sample models are available under the File menu to get quickly acquainted with the VISIBIO*web* user interface. Note that since the cleaved states of proteins cannot be represented using BioPAX level 2, certain sample pathways may not be modeled correctly. We hope this will be fixed with level 3 of BioPAX.

Interpreting BioPAX models

Here, we discuss a number of issues coming up during the conversion of a BioPAX model into a view using SBGN.

A view of a BioPAX model in VISIBIO*web* shows either the entire set of objects defined in that model, or the contents of one `pathway` instance (or a union of

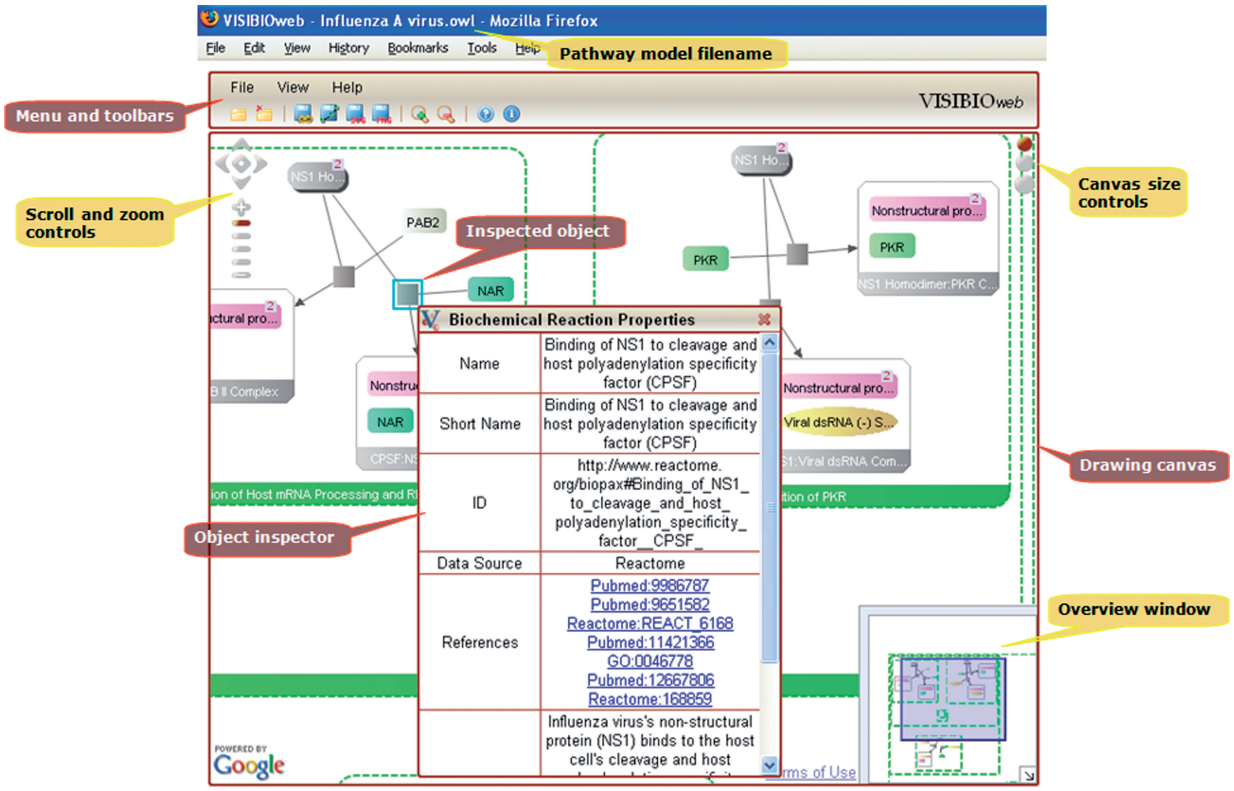

Figure 2. A sample screenshot from VISIBIO*web*. The GUI is organized into a menu bar, a toolbar, and a canvas with various controls. An object inspection window is opened upon demand.

`pathway` instances) in BioPAX as interactively selected by the user. When a pathway is nested within another, the subgrouping is realized through compound nodes unless the user prefers to use compound structures for compartments (i.e., subcellular location information). Using compound structures for both constructs seem to present great difficulties for visualization, and thus has been avoided.

Even though expressing a molecular complex using another molecular complex is not allowed in BioPAX, many BioPAX models do contain such constructs. In such cases, VISIBIO*web* simply 'flattens' such complexes by ignoring subgroupings.

Instances of `interactions` in a `pathway` may be ordered using the `NEXT_STEP` property of `pathwayStep` in BioPAX. This property is particularly useful for representing flows in pathways, but unfortunately, there does not seem to be a consensus on how it should be visualized in the SBGN community at the moment, and VISIBIO*web* currently ignores it. However, we believe that a natural ordering is implied anyway in most pathways through molecules linking reactions.

For brevity, VISIBIO*web* tries to keep the appearance of each pathway object simple, displaying most information via the object inspection window.

Automatic layout

Before a pathway view is presented to the user, it needs to be laid out in an easy-to-understand manner.

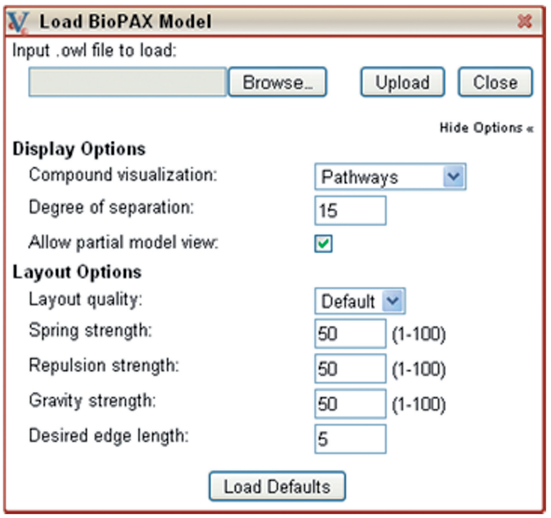

Figure 3. Display and layout options in load dialog.

Pathway views are automatically laid out in VISIBIO*web*, using a customized version of the algorithm described in (18,19). Layout operation may also be configured through user options such as quality and desired edge (link) length.

Output

Generated pathway maps mostly comply with SBGN for Process Diagrams (level 1) (6). In the case of complexes

and compartments, we opted to put labels at the bottom, as opposed to upper-right corner of the associated shape as suggested by SBGN, for ease of implementation and clarity. The views may be interactively inspected, and can be saved as static images (in SVG or PNG format). Finally, the geometry of these maps could be output in a simple native XML-based format (Figure 5). A formal description of the format is available in '.xsd' format in the public distribution of the software.

Persisting views

A pathway view generated by VISIBIO*web* is discarded as the session for that particular user ends unless the user explicitly persists a particular view. When a view is successfully persisted, the user will see the Persist BioPAX Model View Dialog (Figure 4). This dialog not only provides the user with the URL to use when embedding a VISIBIO*web* view of the particular pathway but also helps to configure the user interface of the associated view.

Notice that the provided view does not have to occupy the whole browser window but could be embedded into a subwindow (e.g. a frame), with other useful information about that particular pathway provided in the rest of the web page and formatted as desired by the user.

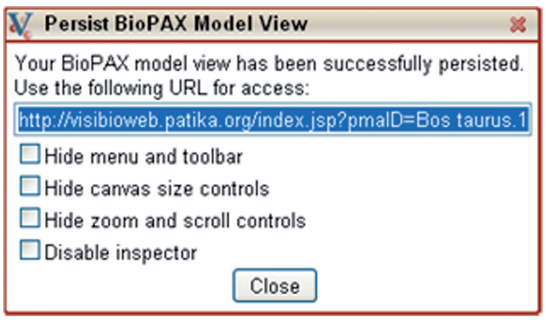

Figure 4. A sample view of the Persist BioPAX model view dialog.

LAYOUT AS A SERVICE

For the user who has his or her own rendering utilities but needs an elegant solution for the layout of the pathway maps, VISIBIO*web* provides a layout service through Hypertext Transfer protocol (HTTP). Thus, a user can programmatically access the layout services of VISIBIO*web* in his or her own application.

The topology of a pathway graph to be laid out is passed to the server using the native XML-based format mentioned earlier (Figure 5). The resulting layout is passed back to the client using the same format, with geometry information added.

An example layout client showing the use of this service written in Java is included in the distribution, and such clients in other platforms should be straightforward to build.

AVAILABILITY

The VISIBIO*web* website (http://visibioweb.patika.org) is free and open to all users without any login requirement. The site includes a full user's guide in addition to a quick help page, and the sources are available through a sourceforge project at https://sourceforge.net/projects/visibioweb.

ACKNOWLEDGEMENTS

We thank Dr. Ozgun Babur and Dr. Emek Demir for their valuable comments and suggestions during the development of VISIBIO*web*.

FUNDING

The Scientific and Technological Research Council of Turkey (105E065).

Conflict of interest statement. None declared.

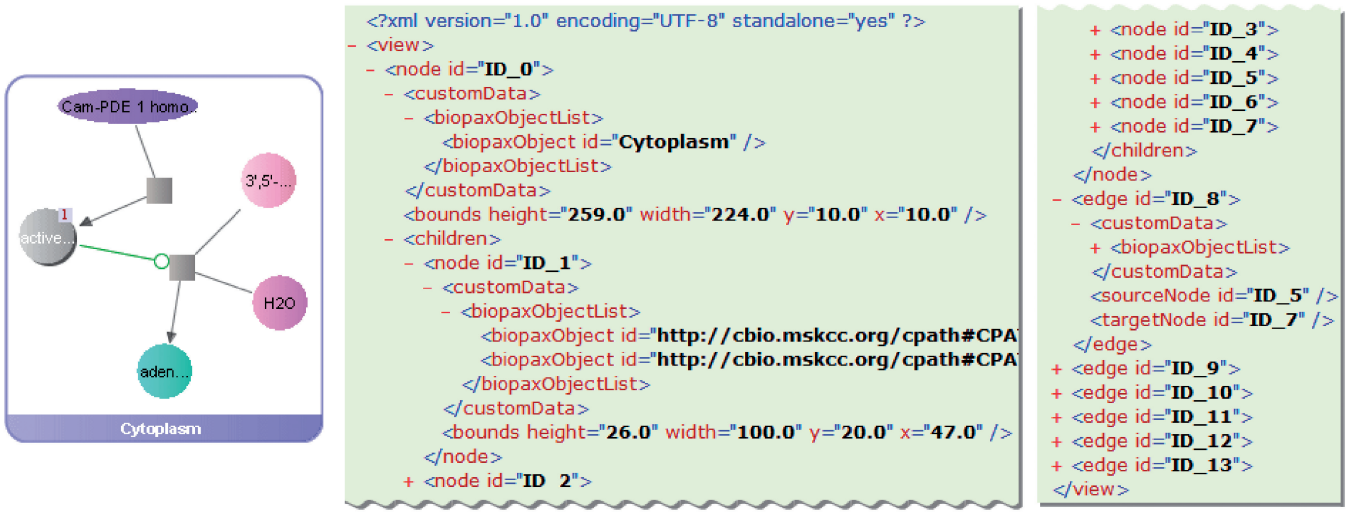

Figure 5. A sample pathway view (Cam-PDE 1 activation) and its geometry in native VISIBIO*web* format (certain parts hidden for brevity).

REFERENCES

1. Demir,E., Babur,O., Dogrusoz,U., Gursoy,A., Ayaz,A., Gulesir,G., Nisanci,G. and Cetin-Atalay,R. (2004) An ontology for collaborative construction and analysis of cellular pathways. *Bioinformatics*, **20**, 349–356.
2. BioPAX. Biological pathways exchange. http://www.biopax.org (4 May 2010, date last accessed).
3. Caspi,R., Foerster,H., Fulcher,C.A., Kaipa,P., Krummenacker,M., Latendresse,M., Paley,S., Rhee,S.Y., Shearer,A.G., Tissier,C. *et al.* (2008) The MetaCyc Database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases. *Nucleic Acids Res.*, **36**, D623–D631.
4. Hucka,M., Finney,A., Sauro,H.M., Bolouri,H., Doyle,J.C., Kitano,H., Arkin,A.P., Bornstein,B.J., Bray,D., Cornish-Bowden,A. *et al.* (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, **19**, 524–531.
5. Kanehisa,M., Araki,M., Goto,S., Hattori,M., Hirakawa,M., Itoh,M., Katayama,T., Kawashima,S., Okuda,S., Tokimatsu,T. *et al.* (2008) KEGG for linking genomes to life and the environment. *Nucleic Acids Res.*, **36**, D480–D484.
6. Le Novère,N., Hucka,M., Mi,H., Moodie,H., Schreiber,F., Sorokin,A., Demir,E., Wegner,K., Aladjem,M.I., Wimalaratne,S.M. *et al.* (2009) The systems biology graphical notation. *Nat. Biotechnol.*, **27**, 735–741.
7. Matthews,L., Gopinath,G., Gillespie,M., Caudy,M., Croft,D., de Bono,B., Garapati,P., Hemish,J., Hermjakob,H., Jassal,B. *et al.* (2009) Reactome knowledgebase of human biological pathways and processes. *Nucleic Acids Res.*, **37**, D619–D622.
8. Bader,G.D., Cary,M.P. and Sander,C. (2006) Pathguide: a pathway resource list. *Nucleic Acids Res.*, **34**, D504–D506.
9. INOH. Inoh pathway database. http://www.inoh.org (4 May 2010, date last accessed).
10. The Cancer Cell Map. http://cancer.cellmap.org/cellmap (4 May 2010, date last accessed).
11. Le Novère,N., Bornstein,B.J., Broicher,A., Courtot,M., Donizelli,M., Dharuri,H., Li,L., Sauro,H., Schilstra,M., Shapiro,B. *et al.* (2006) Biomodels database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res.*, **34**, 689–691.
12. Schaefer,C.F., Anthony,K., Krupa,S., Buchoff,J., Day,M., Hannay,T. and Buetow,K.H. (2009) PID: the Pathway Interaction Database. *Nucleic Acids Res.*, **37**, 674–679.
13. Fukuda,K. and Takagi,T. (2001) Knowledge representation of signal transduction pathways. *Bioinformatics*, **17**, 829–837.
14. Elliott,B., Kirac,M., Cakmak,A., Yavas,G., Mayes,S., Cheng,E., Wang,Y., Gupta,C., Ozsoyoglu,G. and Ozsoyoglu,Z.M. (2008) PathCase: Pathways Database System. *Bioinformatics*, **24**, 2526–2533.
15. Funahashi,A., Matsuoka,Y., Jouraku,A., Morohashi,M., Kikuchi,N. and Kitano,H. (2008) CellDesigner 3.5: a versatile modeling tool for biochemical networks. *Proc. IEEE*, **96**, 1254–1265.
16. Hu,Z., Hung,J.H., Wang,Y., Chang,Y.C., Huang,C.L., Huyck,M. and DeLisi,C. (2009) Visant 3.5: multi-scale network visualization, analysis and inference based on the Gene Ontology. *Nucleic Acids Res.*, **37**, 115–121.
17. Yeung,N., Cline,M.S., Kuchinsky,A., Smoot,M.E. and Bader,G.D. (2008) Exploring biological networks with Cytoscape software. *Curr. Protoc. Bioinform.*, Chapter 8, Unit 813.
18. Dogrusoz,U., Giral,E., Cetintas,A., Civril,A. and Demir,E. (2009) A layout algorithm for undirected compound graphs. *Inf. Sci.*, **179**, 980–994.
19. Genc,B. and Dogrusoz,U. (2006) A layout algorithm for signaling pathways. *Inf. Sci.*, **176**, 135–149.