

version

1.1

BILKENT UNIVERSITY

Bilkent Center for Bioinformatics

VISIBIO*web*
User's Guide

BILKENT CENTER FOR BIOINFORMATICS

VISIBIO *web* User's Guide



© Bilkent Center for Bioinformatics 2009

Bilkent University

Ankara 06800, TURKEY

Phone +90 312.290.1401 • Fax +90 312.266.4047

Table of Contents

Overview	2
Uses.....	2
Architecture.....	2
How to Start VISIBIO <i>web</i>	3
Tool Sources.....	4
Basics	5
Pathway Model	5
Pathway View	5
Ontology.....	6
Notation.....	7
Loading BioPAX Models.....	8
Loading a Model	8
Display Options.....	9
Layout Options.....	10
Closing a Model	11
Sample Models	11
Navigating Over and Inspecting Pathway Views	12
Zooming.....	12
Scrolling.....	13
Changing Canvas Size.....	13
Object Inspection	13
Persisting Pathway Views	14
Embedding a View on Your Own Web Page.....	14
Saving Geometry and Layout Service	16
Saving Geometry in Native Format.....	16
Saving Geometry in SBGN-ML Format.....	17
Using the Layout Service	18
Exporting as Image	19
Saving as SVG.....	19
Saving as PNG	19
Third-Party Software License Agreements	20
References.....	21



Overview

In this chapter, we introduce the tool VISIBIOweb, and explain how to start the tool.

VISIBIOweb is a free, open-source, web-based pathway visualization tool for BioPAX (Level 2) models [1]. The user side is a thin-client JavaScript application based on Google Maps API [5], whereas the server side is made up of a component based on the Eclipse Graph Editing Framework (GEF) version 3.1 [6].

Uses

VISIBIOweb can be mainly used to produce interactive views of pathway models represented with BioPAX Level 2 format. It can also be used as a pathway layout service through http protocol.

Architecture

Figure 1 shows the architecture of VISIBIOweb at a high-level. The communication between the client and the server is initiated with a file upload event through your web-browser. Requests from the client side arrive at an Apache Tomcat server. Tomcat executes requested JSP (Java Server Pages) files and delegates the tasks to the Session Handler. Session Handler is responsible to manage all server side logic (with the help of the following components: BioPAX Parser, Core, Layout Manager, Tile and Image Generator, and XML Generator) 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 component on the client side is a canvas, which is a customized Google Map. VISIBIOweb canvas is responsible to display the view constructed on the server side properly. It also detects various user actions and events. The Geometry Parser unit is responsible for parsing the geometry of the XML file sent from the server. The output of this component is polygons, which are provided by Google Maps API, added on top of VISIBIOweb canvas, and facilitates hit-testing for proper pathway object inspection.

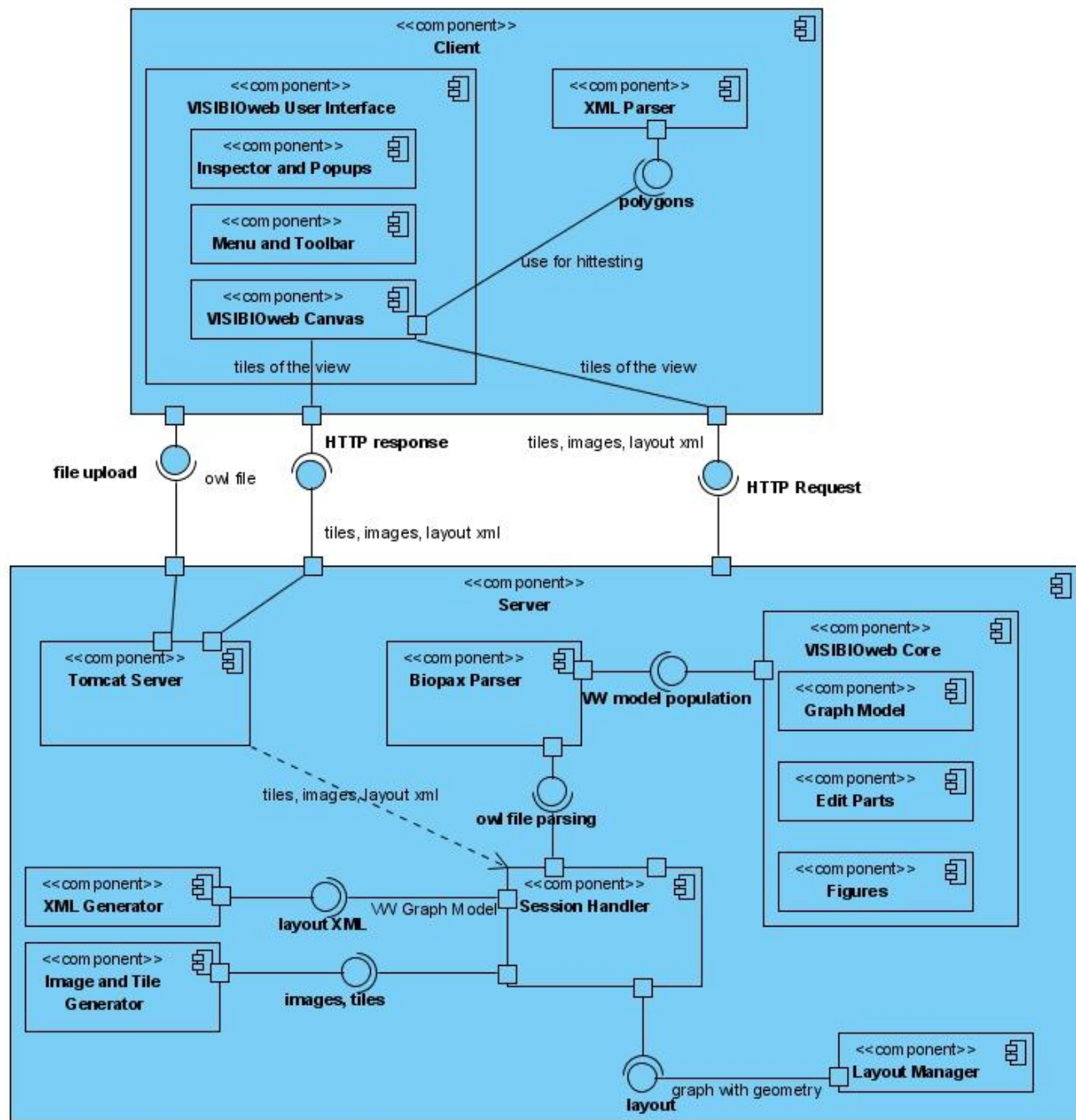


Figure 1 High-level architecture of VISIBIOweb

How to Start VISIBIOweb

VISIBIOweb runs on any computer with access to the Internet (needed to access Google Maps). It was tested with the following browsers but should run on any browsers on any operating system:

- Mozilla Firefox version 3.0 or later,

- Internet Explorer version 8.0 or later, and
- Safari 4.0 or later

Just make sure JavaScript version 1.6 (or later) is installed with your browser.

To start using VISIBIOweb, simply point your browser to

<http://visibioweb.patika.org/>

Tool Sources

VISIBIO*web* sources are available for distribution under Eclipse Public License version 1.0. Please contact us if you would like a copy of the sources to build a local server or use them otherwise under the terms.

Basics

In this chapter, we define the basic concepts and terminology used.

VISIBIO*web* is a tool for visualization of biological pathway models through an interactive view for effective analysis. Each pathway model represented in BioPAX Level 2 format is potentially composed of other pathways, defined recursively. VISIBIO*web*'s ontology is based on that of PATIKA [9], and its notation complies with that of Process Diagrams of SBGN (Systems Biology Graphical Notation) [2][3].

Pathway Model

We call the entire set of biological information loaded from a BioPAX file a *pathway model*, or simply a *model*. Each BioPAX file or each pathway model is potentially composed of one or more pathways, which may be defined in a hierarchical manner. To be exact, we use the notion of a pathway as defined by BioPAX as follows:

A pathway is a set or series of interactions, often forming a network, which biologists have found useful to group together for organizational, historic, biophysical or other reasons.

Pathway View

VISIBIO*web* allows users to choose the exact set of pathways to be visualized upon load (see Chapter titled Loading BioPAX Models on page 8 for more details). A drawing canvas is used to draw a combination of these pathways forming a *pathway view*, or simply a *view*. A menu and a toolbar are provided for convenient navigation and inspection of pathway views (Figure 2).

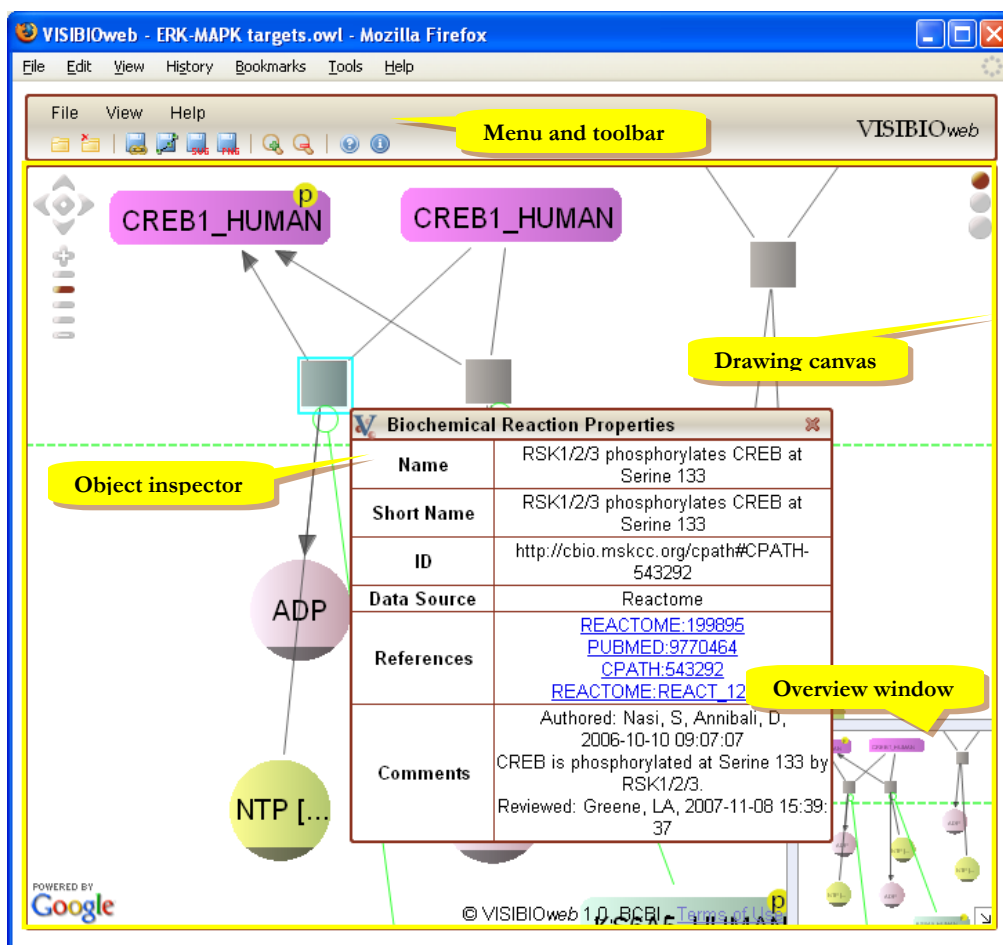


Figure 2 Sample screenshot from VISIBIOweb

Ontology

A pathway view is composed of pathway objects (nodes and compound nodes) and their interactions. Compound nodes are exclusively used to represent molecular complexes and pathway abstractions or cellular compartments, as determined by a user option.

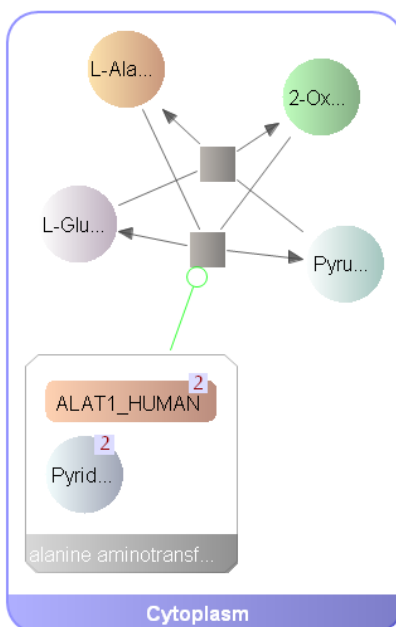
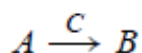


Figure 3 Sample notation for a pathway view; grey colored compound node in this view represents a molecular complex and blue colored one is a cellular location (Cytoplasm).

Non-compound (or simple) nodes are of two types: process nodes (transitions) and entities (states). Transitions are represented with small gray boxes (biochemical reaction, transport) or circles (complex assembly), whereas states of a biological entity are represented with either rounded rectangles (macromolecules) or circles (small molecules).

A transition represents group additions or removals, complex formations and disassociations as well as transportations and other cellular events. This model is very similar to the chemical equations of the form



where A is a substrate, B is a product and C is an effector. The states associated with a transition are connected to the corresponding transition box through substrate, product and effector edges, respectively. Substrate and product edges of a transition are drawn in black, whereas effectors are drawn in green and red for activation and inhibition, respectively.

You may refer to [9] for more details of the ontology used for VISIBIO*web*'s pathway views.

Notation


The notation used in VISIBIO*web* complies with that of Process Diagrams of SBGN (Systems Biology Graphical Notation) [2][3].

Loading BioPAX Models

In this chapter, we show you how to load your BioPAX models into VISIBIOweb to produce pathway drawings.

Pathway models stored in BioPAX formatted files may be loaded into VISIBIOweb, one at a time. The content and appearance of these views may be configured by the user through some options.

Loading a Model

Pathway models stored in BioPAX formatted files may be loaded into VISIBIOweb by using “File | Load Model ...” menu item or by clicking on  on the toolbar. This opens up the “Load Dialog”. Choose the BioPAX file (“.owl” file) to be uploaded and click “Upload”.

If you would like to change any of the user options, click on “Show Options”. This will open up two sets of options: Display Options and Layout Options (Figure 4).

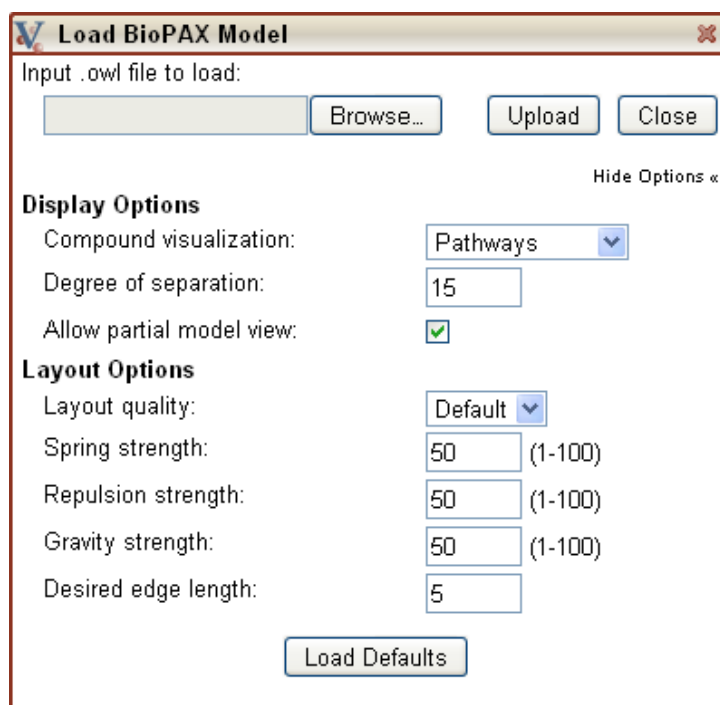


Figure 4 Display and Layout Options in Load Dialog

Display Options

Below is a list of display options that the user can configure:

- **Compound Visualization:** Along with molecular complexes, compound nodes may be used for visually representing either “Pathways” or “Compartments” in a pathway view (Figure 5). By default, pathways are represented with compound nodes.
- **Degree of Separation:** Ubiquitous molecules, which typically participate in many different biological activities have 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 at the mechanistic level. 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” as a rule of thumb. Thus, ATP with degree 20 will be drawn 20 times (as 20 separate degree-one nodes) if this option has a value less than or equal to 20.
- **Allow Partial Model View:** This option specifies whether or not the user should be prompted for the set of pathways to be displayed. Often times, a pathway model consists of a very large number of molecules and reactions. The user might want to manage this complexity by inspecting a sub-pathway of interest at a time. When this option is enabled (the default), the upload operation returns a list of pathways in the input file (Figure 6).

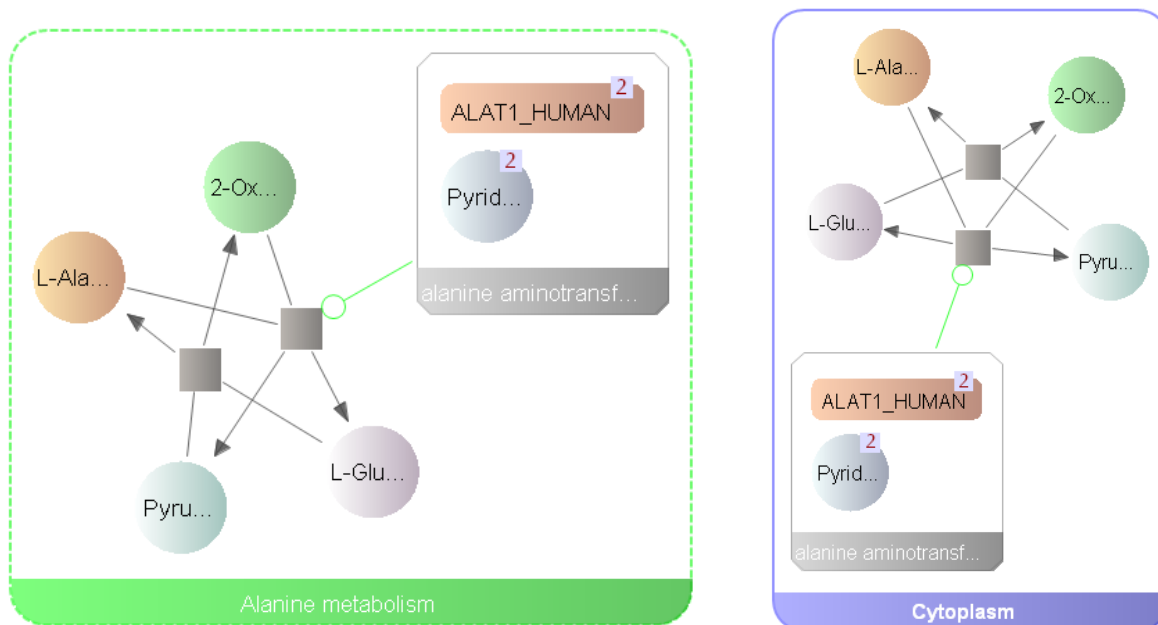


Figure 5 The same pathway (Alanine metabolism) displayed with Pathways (left) and Compartments (right) represented with compound nodes. In either case, compound nodes are used for molecular complexes as well.

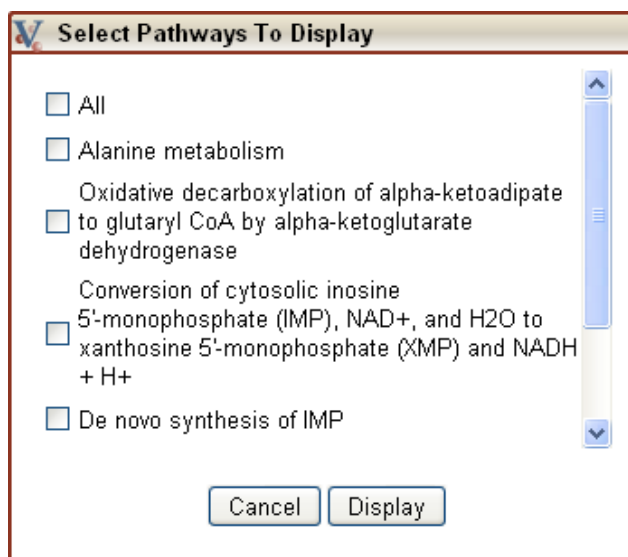


Figure 6 An example of the list of pathways for the user to select to include in the view; to show them all is always an option.


Layout Options

Before a pathway view is presented to the user, it needs to be laid out in an easy-to-understand manner. Pathway views are automatically laid out in VISIBIONeb, using the algorithm by Dogrusoz et al. [7][8] before they are presented to the user. Below is a list of things that can be used to configure this layout operation:

- **Layout Quality.** Quality of layout; usually the higher the quality, the slower the layout will be.

- **Spring Strength.** Constant for spring forces corresponding to edges; the higher this value is the stronger spring forces will be.
- **Repulsion Strength.** Constant for repulsion forces that are applied to node pairs; the lower this value is, the closer the nodes will be.
- **Gravity Strength.** Factor for the gravity of each graph; the higher this value is, the closer disconnected parts of the graph will be.
- **Desired Edge Length.** Desired length of an edge; the higher this value is, the longer the edges will be.

Closing a Model

You may close the current model you are working on by choosing “File | Close Model” from the menu or by clicking on  on the toolbar. This unloads the pathway model from both the client and the server.

Sample Models

Sample pre-loaded models that are ready to be viewed can be found under “File | Sample Models”.

Navigating Over and Inspecting Pathway Views

In this chapter, we describe how you can navigate over a pathway view and inspect its contents.

Frequently, a pathway view displayed with VISIBIO*web* will not entirely fit in the canvas in a readable manner. Thus scrolling and zooming are essential capabilities of such a tool. In addition, details of a particular biological object or relation needs to be explored on demand. VISIBIO*web* inspector facilitates users to do exactly that.

Zoom, scroll, and canvas size controllers are located on the upper-left and right corners of the canvas, respectively (Figure 7).

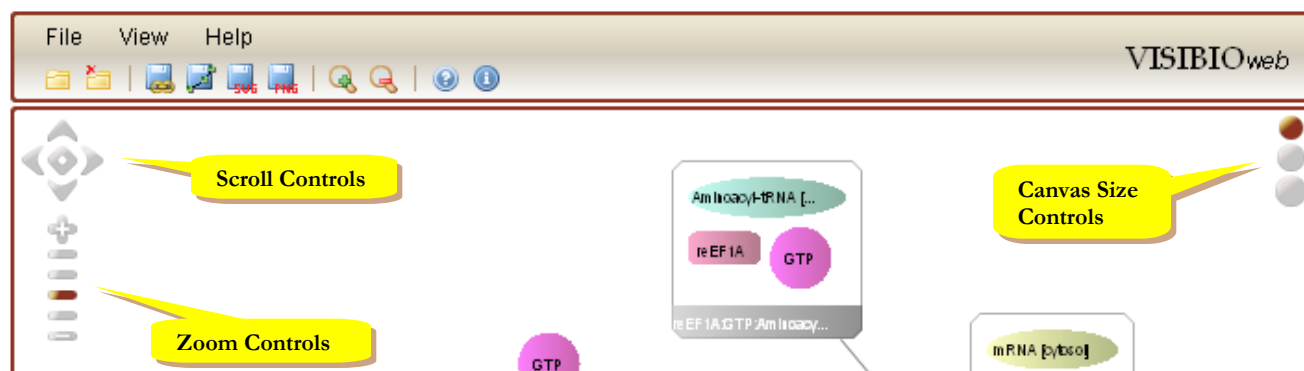




Figure 7 Zoom, scroll, and canvas size controls used to navigate over a pathway view

Zooming

Pathways can be viewed at four different zoom levels with VISIBIO*web*. The calculation of particular zoom levels is performed with respect to the number of tiles needed to cover the pathway drawing. In other words, unless the graph bounds are unusually big, we use the following number of tiles in increasing order of zoom levels: 1, 4, 16, and 64. For larger drawings we end up creating more tiles at the expense of performance.

Zoom levels may be changed using the zoom controls located at the upper-left corner of the canvas as well as the “View | Zoom” menu. In addition, the toolbar has buttons  and  for zooming in and out, respectively. Yet another fast way to zoom in (out) is by left (right) double-clicking on the canvas. However, this operation also centers the drawing at the click point.

Scrolling

You may scroll the view by using the scroll controls located at the upper-left corner of the canvas as well as the “View | Scroll” menu. In addition, a convenient way to pan the view is by left-clicking anywhere on the canvas and dragging.

Changing Canvas Size

Size of the canvas may be changed using the canvas size controls located at the upper-right corner of the canvas as well as the “View | Canvas” menu. For larger pathway drawings, you may choose to work with a larger canvas.

Object Inspection

All nodes, edges and compound nodes of a pathway view in VISIBION^{web} have distinct properties and UIs. These properties may be inspected using the Object Properties dialog by left-clicking on that object (Figure 8).

Biochemical Reaction Properties	
Name	alanine + alpha-ketoglutarate <=> pyruvate + glutamate
Short Name	alanine + alpha-ketoglutarate <=> pyruvate + glutamate
ID	http://cbio.mskcc.org/cpath#CPATH-541178
Data Source	Reactome
References	REACTOME:70523 CPATH:541178 REACTOME:REACT_196 PUBMED:9119391
Comments	At the beginning of this reaction, 1 molecule of 'L-Alanine', and 1 molecule of '2-Oxoglutarate' are present. At the end of this reaction, 1 molecule of 'L-Glutamate', and 1 molecule of 'Pyruvate' are present. This reaction takes place in the 'cytosol' and is mediated by the 'alanine transaminase activity' of 'alanine aminotransferase, holoenzyme'.

Figure 8 Sample view of an Object Properties (Inspector) Dialog


Among the properties displayed, users may find the external references on the particular object most useful. Left-clicking on a particular reference will take you to the corresponding web page.

Persisting Pathway Views

In this chapter, we describe how you can persist a pathway view for later use.

Pathway views generated by VISIBIOweb are normally discarded as the session for that particular user ends. You can however embed a particular view in your own web page using the URL provided.

Embedding a View on Your Own Web Page

You may embed a particular VISIBIOweb pathway view on your own web page by choosing “File | Persist View ...” from the menu or directly clicking on the toolbar button . When a view is successfully persisted, you should get the Persist BioPAX Model View Dialog (Figure 9).

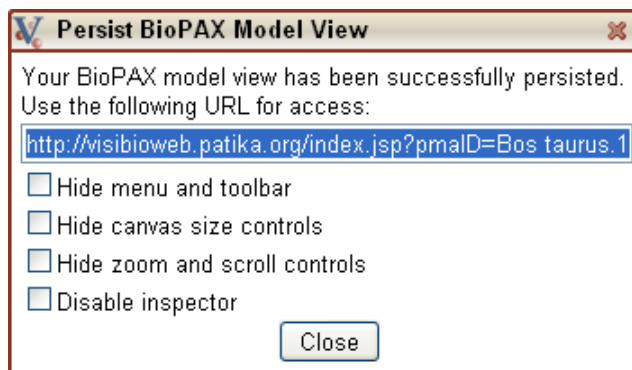


Figure 9 Sample view of the Persist BioPAX Model View Dialog

This dialog not only provides you the URL that you should be using when embedding a VISIBIOweb view of the particular pathway but also helps you configure the associated view with the following options:

- ***Hide menu and toolbar***: when checked, the menu and the toolbar are not shown;
- ***Hide canvas size controls***: when checked, the canvas size controls are not shown;
- ***Hide zoom and scroll controls***: when checked, zoom and scroll controls are not shown;

- ***Disable inspector.*** when checked, inspector or object properties dialog is not shown on left-click on a pathway object.

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

Saving Geometry and Layout Service

In this chapter, we show you how to save the geometry of a pathway view in XML as well as how to use VISIBIOweb as a layout service through http protocol.

The geometry of the pathway view may be saved in XML format. VISIBIOweb also provides a service for layout through http protocol, in case you would like to use your own rendering facilities for pathways but use VISIBIOweb for its layout.

Saving Geometry in Native Format

You may save the geometry of the current pathway view by using “File | Save Geometry | Native” menu item. VISIBIOweb will then construct an XML file on the server and return it for your use. This file has a simple format, where a view is composed of a list of nodes and edges (Figure 10). Nodes, in turn, can be either simple or compound. For each compound node, a list of children is to be listed. Both nodes and edges also know about their associated BioPAX objects (mapped according to the input “.owl” file). Note that often times, a single node or edge is associated with multiple BioPAX objects.

```


<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<view>
  <node id="ID_0">
    <customData>
      <biopaxObjectList>
        <biopaxObject id="http://cbio.mskcc.org/cpath#CPATH-594065" />
      </biopaxObjectList>
    </customData>
    <bounds height="310.0" width="405.0" y="10.0" x="10.0" />
    <children>
      <node id="ID_1">
        <customData>
          <biopaxObjectList>
            <biopaxObject id="http://cbio.mskcc.org/cpath#CPATH-LOC" />
            <biopaxObject id="http://cbio.mskcc.org/cpath#CPATH-598" />
            <biopaxObject id="http://cbio.mskcc.org/cpath#CPATH-LOC" />
          </biopaxObjectList>
        </customData>
        <bounds height="40.0" width="40.0" y="198.0" x="180.0" />
      </node>
      ...
    </children>
  </node>
  <edge id="ID_17">
    <customData>
      <biopaxObjectList>
        <biopaxObject id="http://cbio.mskcc.org/cpath#CPATH-597883" />
        <biopaxObject id="http://cbio.mskcc.org/cpath#CPATH-LOCAL-76122" />
      </biopaxObjectList>
    </customData>
    <sourceNode id="ID_7" />
    <targetNode id="ID_12" />
  </edge>
  ...
</view>

```

Figure 10 Sample XML file used for storing the geometry of a pathway view in VISIBIO*web*

The “.xsd” file describing the formal structure of the file format containing the geometry is available for distribution upon demand.

Saving Geometry in SBGN-ML Format

You may save the geometry of the current pathway view in SBGN-ML format by using “File | Save Geometry | SBGN-ML” menu item or by simply clicking on  on the toolbar. VISIBIO*web* will then construct an SBGN-ML file on the server and return it for your use. This file is constructed via libSBGN [11] utilities. The SBGN-ML format is very similar to graph formats such as GraphML. Distinctively, nodes are represented with *ghpbs*, while edges are represented with *arcs* (Figure 11). As with the native XML format, nodes can be compound structures such as complexes and cellular locations. For each compound node, a list of children is listed.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sbgn xmlns="http://sbgn.org/libsbgn/pd/0.1">
  <map>
    <glyph id="glyph1" class="macromolecule"...>
    <glyph id="glyph2" class="macromolecule"...>
    <glyph id="glyph3" class="simple chemical">
      <label text="ATP"/>
      <bbox y="196.0" x="128.0" h="40.0" w="40.0"/>
    </glyph>
    <glyph id="glyph4" class="simple chemical"...>
    <glyph id="glyph5" class="macromolecule"...>
    <glyph id="glyph6" class="process">
      <label text="Phosphorylation of p53 at ser-15 by ATM kinase"/>
      <bbox y="133.0" x="143.0" h="18.0" w="18.0"/>
      <port y="142.0" x="133.0" id="glyph6.1"/>
      <port y="142.0" x="171.0" id="glyph6.2"/>
    </glyph>
    <glyph id="glyph7" class="macromolecule"...>
    <glyph id="glyph8" class="macromolecule"...>
    <glyph id="glyph9" class="process"...>
    <glyph xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
    <arc target="glyph6.1" source="glyph3" class="consumption">
      <start y="196.0" x="143.0"/>
      <end y="142.0" x="133.0"/>
    </arc>
    <arc target="glyph5" source="glyph6.2" class="production"...>
    <arc target="glyph4" source="glyph6.2" class="production"...>
    <arc target="glyph6" source="glyph1" class="catalysis"...>
    <arc target="glyph8" source="glyph9.2" class="production"...>
    <arc target="glyph6.1" source="glyph2" class="consumption"...>
    <arc target="glyph9.1" source="glyph7" class="consumption"...>
  </map>
</sbgn>

```

Figure 11 Sample SBGN-ML file for storing the geometry of the p53-Dependent G1 DNA Damage Response pathway in SBGN format

Using the Layout Service

For those who have their own rendering utilities but need an elegant layout for their pathway views, VISIBIO*web* provides a layout service through http service. Thus, one can programmatically access the layout services of VISIBIO*web* in their own applications.


An example layout client showing the use of this service written in Java is included in the sources distribution. Please contact us if you are interested in a copy.

Exporting as Image


In this chapter, we show you how to export pathway views as static images.

VISIBIO*web* can generate static images of a pathway view created as well. These images may be downloaded locally for later use.

Saving as SVG

VISIBIO*web* pathway views can be saved in SVG format by choosing “File | Export as SVG” from the menu or by simply clicking on  from the toolbar. Most browsers will prompt you for an action by popping up their open / save dialog when this facility is used.

Saving as PNG

VISIBIO*web* allows pathway views to be saved in PNG format as well, by choosing “File | Export as PNG” from the menu or by simply clicking on  from the toolbar. Most browsers will prompt you for an action by popping up their open / save dialog when this facility is used.

Third-Party Software License Agreements

In this chapter, we list any third-party software license agreements.

VISIBIO*web* is distributed under Eclipse Public License, version 1.0 (<http://www.eclipse.org/org/documents/epl-v10.php>). Notwithstanding the terms and conditions of this license and any agreement you have with Bilkent Center for Bioinformatics, the third-party software code, products, or files identified below are "Excluded Components" and are subject to the following terms and conditions.

Component	Version	By	License
Google Maps API	2.166	Google	Google Maps Terms of Service
Paxtools	1.0	MSKCC, INOH	LGPL
Batik	1.6	The Apache Software Foundation	Apache Software License, v 2.0
Java WSDP	1.5	Sun Microsystems, Inc.	Sun Microsystems Inc. Software License Agreement
Apache log4j	1.2	The Apache Software Foundation	Apache Software License, v 2.0
Menus	-	Ilya S. Lyubinskiy	License
Popup Dialogs	-	Ilya S. Lyubinskiy	License
Silk Icons	2.5	Mark James	Creative Commons Attribution License, v 3.0
libSBGN	-	-	The Apache License, Version 2.0

References

- [1] BioPAX, <http://www.biopax.org>
- [2] SBGN, <http://www.sbgm.org>
- [3] N. Le Novère, M. Hucka, H. Mi, S. Moodie, F. Schreiber, A. Sorokin, E. Demir, K. Wegner, M.I. Aladjem, S.M. Wimalaratne, F.T. Bergman, R. Gauges, P. Ghaza, H. Kawaji, L. Li, Y. Matsuoka, A. Villéger, S.E. Boyd, L. Calzone, M. Courtot, U. Dogrusoz, T. Freeman, A. Funahashi, S. Ghosh, A. Jouraku, S. Kim, F. Kolpakov, A. Luna, S. Sahle, E. Schmidt, S. Watterson, G. Wu, I. Goryanin, D.B. Kell, C. Sander, H. Sauro, J.L. Snoep, K. Kohn, H. Kitano, "The Systems Biology Graphical Notation", Nature Biotechnology, vol. 27, pp. 735-741, 2009.
- [4] Paxtools, <http://www.biopax.org/paxtools>
- [5] Google Maps API, <http://code.google.com/apis/maps/>
- [6] Eclipse GEF (Graph Editing Framework) version 3.1, <http://www.eclipse.org/gef>
- [7] U. Dogrusoz, E. Giral, A. Cetintas, A. Civril, and E. Demir, "A Compound Graph Layout Algorithm for Biological Pathways", In Graph Drawing (Proc. GD '04), LNCS, vol. 3383, pp. 442-447, 2005.
- [8] U. Dogrusoz, E. Giral, A. Cetintas, A. Civril, and E. Demir, "A Layout Algorithm For Undirected Compound Graphs", Information Sciences, 179, pp. 980-994, 2009.
- [9] E. Demir, O. Babur, U. Dogrusoz, A. Gursoy, A. Ayaz, G. Gulesir, G. Nisanci, and R. Cetin-Atalay, "An Ontology for Collaborative Construction and Analysis of Cellular Pathways", Bioinformatics, 20(3), pp. 349-356, 2004.
- [10] PATIKA, <http://www.patika.org>
- [11] libSBGN, <http://libsbgn.sourceforge.net>