BIOINFORMATICS ORIGINAL PAPER Vol. 23 no. 3 2007, pages 344–350
doi:10.1093/bioinformatics/btl611

Systems biology

Dynamic exploration and editing of KEGG pathway diagrams

Christian Klukas and Falk Schreiber*

Leibniz Institute of Plant Genetics and Crop Plant Research, Corrensstrasse 3, 06466 Gatersleben, Germany

Received on August 1, 2006; revised on November 3, 2006; accepted on November 24, 2006
Advance Access publication December 1, 2006
Associate Editor: Alvis Brazma

ABSTRACT

Motivation: The Kyoto Encyclopedia of Genes and Genomes (KEGG) Pathway database is a very valuable information resource for researchers in the fields of life sciences. It contains metabolic and regulatory processes in the form of wiring diagrams, which can be used for browsing and information retrieval as well as a base for modeling and simulation. Thus it helps in understanding biological processes and higher-order functions of biological systems.

Currently the KEGG website uses semi-static visualizations for the presentation and navigation of its pathway information. While this visualization style offers a good pathway presentation and navigation, it does not provide some of the possibilities related to dynamic visualizations, most importantly, the creation and visualization of user-specific pathways.

Results: This paper presents methods for the dynamic visualization, interactive navigation and editing of KEGG pathway diagrams. These diagrams, given as KEGG Markup Language (KGML) files, can be visually explored using novel approaches combining semi-static and dynamic visualization, but also edited or even newly created and then exported into KGML files.

Availability: KGML-ED, a program implementing the presented methods, is available free of charge to the scientific community at http://kgml-ed.ipk-gatersleben.de

Contact: schreibe@ipk-gatersleben.de

1 INTRODUCTION

The Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa *et al*., 2006) is widely used in biology, biochemistry and medicine to study metabolic and regulatory processes. The presentation of these processes as pathway diagrams greatly helps researchers in understanding key functions of biological systems. The pathway data can be studied in a visual way and is also available as KEGG Markup Language (KGML) files. Thus it can be used as a basis for simulation models, e.g. by converting KGML files into SBML (http://systems-biology.org/001/001.html). However, the graphical presentation of pathway information in KEGG is restricted to semi-static visualization and editing KGML files is not simple.

*To whom correspondence should be addressed.

1.1 Static and semi-static visualization

Static visualization is typically characterized by the following aspects:

(1) The use of pictures which are created manually long before their use by the end-user.

(2) A view of the data (e. g. the elements shown, the level of detail), which is pre-defined by the creator of the picture and which usually cannot be changed by an end-user.

(3) Navigation is sometimes supported by links to other pictures, but the result (the new picture) either replaces the current image or is shown in an independent new view.

(4) Editing is not easily possible by the end-user.

Examples of static visualizations of biochemical pathways are pictures in text-books and on posters (Berg *et al*., 2002; Michal, 1999; Nicholson, 1997).

Some of the shortcomings of static visualization have been eliminated on the KEGG pathway website (KEGG pathway database, http://www.genome.jp/kegg/pathway.html). It can be seen as a semi-static visualization approach, where some parts of the picture can be modified. For example, it is possible to change the color of pathway objects depending on the result of search-operations. This enables highlighting of enzymes or compounds as well as the presentation of species-specific pathway diagrams. There is also an interactive pathway navigation based on links within images and the extensive search functions of the KEGG website. However, it is not possible to change the overall structure or layout of the pathway diagrams.

An advantage of static visualization is that pathway drawings can be fine-tuned by the curator. The diagram may be fitted to the available visualization space, or may be extended by additional graphical elements, which illustrate certain aspects of a pathway [e. g. localization of processes as shown in the photosynthesis pathway KEGG map00195 (http://www.genome.jp/kegg/pathway/map/map00195.html)].

The main disadvantage of static visualization is the fixed view of the pathway data. It is not possible for the end-user to remove unneeded parts of a pathway diagram or to extend the pathway with specific information from other pathways. Another restriction of static visualization is the limited usefulness in electronic

information systems. The content of the pictures is typically not accessible by computer programs.

1.2 Dynamic visualization

Dynamic visualization is characterized by the following aspects:

(1) The use of pictures, which are created by the end-user with help of a computer program based on up-to-date data at the time the drawing is needed.
(2) The view of the data and the annotation of network elements is not fixed, but node labels, links to other resources and level of detail can be modified.
(3) Navigation methods are typically supported and it is possible to extend existing drawings with new parts.
(4) Editing is usually possible. The layout and graphical representations may be changed by the end-user as needed with manual or automatic layout methods, but also the structure of pathways may be changed by adding or removing elements.

Because of these aspects, dynamic visualization is well-suited for the interactive exploration of pathway data and is the state-of-the-art method to present such information. Several methods have been proposed for the automatic computation of pathway visualizations (Becker and Rojas, 2001; Rojdestvenski, 2003; Schreiber, 2002; Sirava *et al*., 2002). However, these approaches are often based on specific pathway databases and they produce drawings which are very different to typical KEGG pathway maps. Therefore this paper presents a new approach for dynamic visualization and interactive navigation especially tailored to KEGG pathway diagrams.

1.3 The KGML format as a basis for dynamic visualization of KEGG pathways

The foundation of dynamic pathway visualization is a computer readable, well structured information resource. The KEGG system provides a XML representation of its pathway information called KGML (KEGG Markup Language, http://www.genome.jp/kegg/docs/xml/). The KGML format is widely used in life science research, and some applications, information systems and developer libraries that support KGML are VisANT (Hu *et al*., 2004), kegg2sbml (http://systems-biology.org/001/001.html), Biopathways Workbench (http://www.biopathwaysworkbench.org), BioUML (http://www.biouml.org), VANTED (Junker *et al*., 2006), GenMAPP (http://www.genmapp.org), PathwayExpert (http://ariadnegenomics.com/products/pathwayexpert) and BioRuby (http://bioruby.org).

As KGML is supported by an increasing number of software systems, the possibility of dynamically visualizing and editing KGML files becomes more and more important. The KGML format also contains layout information, and therefore it is not only possible to process the defined entries, relations and reactions for analytical purposes, but also to create visualizations with an initial layout very similar to the KEGG pathway images. Thus KGML enables the combination of the advantages of static and dynamic visualizations as presented in the next section.

2 DYNAMIC PATHWAY EXPLORATION

Several approaches for the visual exploration of KEGG pathways are discussed in this section.

(1) The exploration may start with a schematic overview of all pathway maps where each pathway is represented by a node and each link between two pathways is represented by a connection between nodes. Based on this overview it is possible to extend the drawing by replacing a node representing a particular pathway map with the full drawing of this pathway as shown in Figure 1.
(2) The exploration may start at a specific pathway and the drawing may then be extended step by step with additional linked pathways, see Figure 2.
(3) Given a set of pathways they can be arranged in a specific layout, such as a grid or a circle of pathways as shown in Figure 3.
(4) After a detailed investigation pathways may be collapsed into overview nodes.

To discuss these methods in detail we next introduce a formal description of the pathway information given by the KEGG system.

2.1 Graph representation

In general, a graph $G = (V, E)$ consists of a set of nodes V and a set of edges E, where each edge connects two nodes. Let $G_0 = (V_0, E_0)$ be the KEGG overview graph where each node $v \in V_0$ represents a KEGG pathway and each edge $e \in E_0$ represents the connection between pathways. These connections are given by links to other pathways within KEGG maps. In G_0 each node $v_i \in V_0$ represents a pathway graph G_i. A KEGG pathway graph $G_i = (V_i, E_i)$ is a graph where each node $v \in V_i$ represents a compound, enzyme or other object from KEGG and each edge $e \in E_i$ represents a relation or reaction. Further details of this representation are given in Section 3.2. A pathway graph G_i may also contain references to other pathway graphs. A reference is given by a node $v_r \in V_i$ (called map link node), which also occurs in the overview graph ($v_r \in V_0$), see also Figure 4 (upper right).

2.2 Visualization and navigation methods

2.2.1 Extending the overview The first visual exploration approach is a top-down approach which starts with the overview graph $G_0 = (V_0, E_0)$. This graph may be extended by replacing a node v_i (representing a particular pathway) with its graph representation G_0. This extension process is as follows (Fig. 4): node v_i is removed from the overview graph G_0. The pathway graph G_i may contain other nodes of G_0 representing pathways. These nodes are removed from G_i and edges which were connected to such nodes are instead connected to the corresponding nodes in G_0.

The layout of the overview graph G_0 can be given by an user or produced by layout algorithms such as force-directed (Eades, 1984; Fruchterman and Reingold, 1991) or hierarchical layout methods (Sugiyama *et al*., 1981). When v_i is replaced with G_i in the drawing the KEGG layout of G_i is used and the center of the drawing of G_i is placed at the previous x, y-coordinate of v_i. However, there may not be enough space for the drawing of G_i without overlaps with other nodes of G_0. One possible solution to this problem is to firstly extend the size of node v_i to the size the drawing of G_i will occupy and then

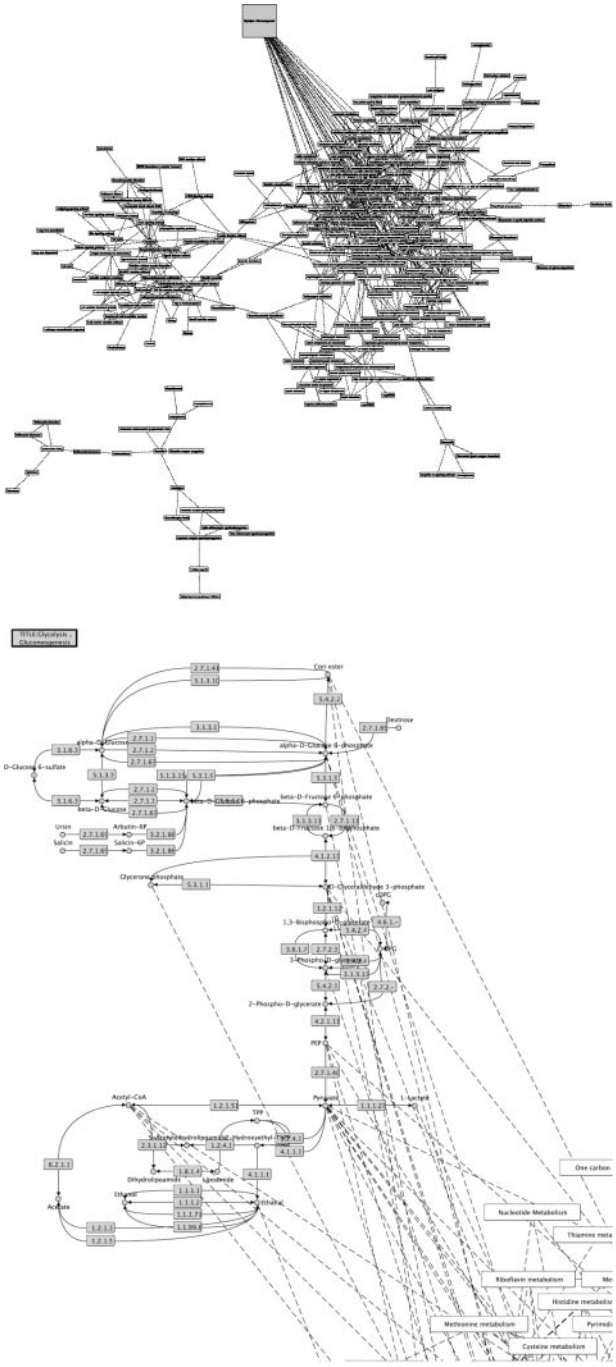

Fig. 1. KEGG pathway navigation—from overview to detail. (upper) An overview of all KEGG pathway maps with connection to at least one other pathway is computed and drawn with a force-directed layout algorithm (Eades, 1984). It shows 245 pathways and their connections, the Glycolysis pathway map is highlighted. (lower) The Glycolysis pathway map is extended (unfolded) and integrated into the view. For high-resolution versions of the pictures see http://kgml-ed.ipk-gatersleben.de/supp_material.

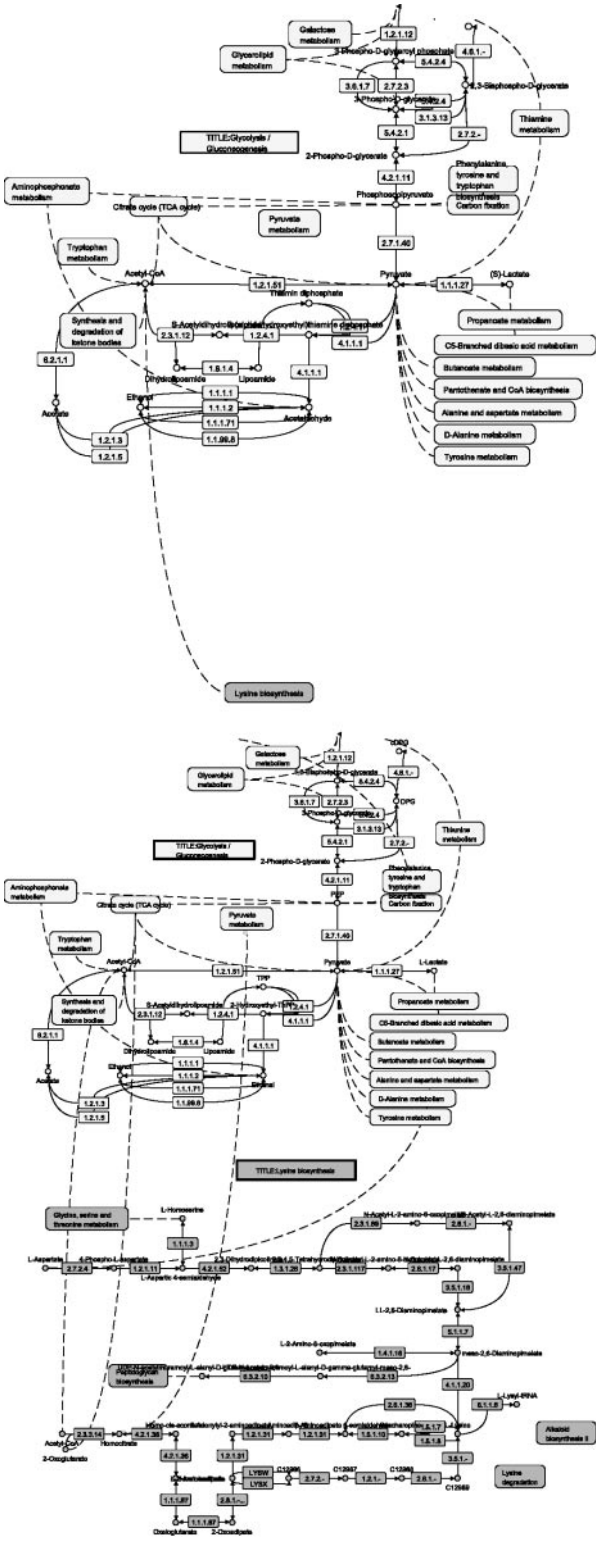

Fig. 2. KEGG pathway navigation—extending a pathway. (upper) A pathway (circle: compound, rectangle: enzyme, rectangle with round corners: link to other pathway map) and a selected link to the next pathway (dark colored). (lower) The selected pathway is integrated into the same view and extends the previous drawing. For high-resolution versions of the pictures see http://kgml-ed.ipk-gatersleben.de/supp_material.

remove node overlaps automatically (Dwyer *et al*., 2006). Another possibility is that the user moves the node v_i prior to unfolding (replacing by G_i) to an area, which provides enough space for the drawing of G_i, see Figure 1. To avoid unnecessary crossing between

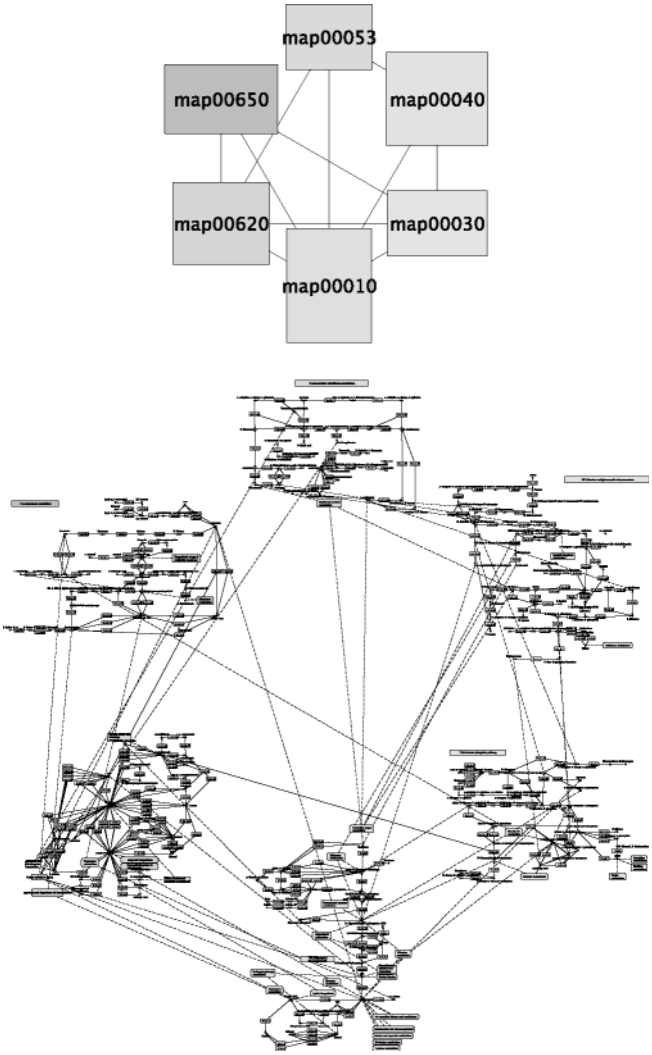

Fig. 3. KEGG Pathway navigation—arranging pathways. The combination of dynamic layout of a specific overview graph (here done with a circular layout method, see upper part of the figure) and subsequent replacement of graph nodes with complete pathways allows the user to combine individual pathways in one view (see lower). For high-resolution versions of the pictures see http://kgml-ed.ipk-gatersleben.de/supp_material.

edges and nodes an automatic algorithm [e.g. (Dobkin *et al*., 1997)] could be used.

2.2.2 Stepwise pathway extension The second visual exploration approach starts with a given pathway graph G_i and extends the drawing step by step with additional pathway graphs. For the extension of G_i by a pathway graph G_j this extension process is as follows (see also Fig. 5): node v_j is removed from the graph G_i and node v_i is removed from graph G_j. Instead, the nodes which were connected to v_i and v_j are connected by new edges. The pathway graph G_j may contain other nodes representing pathways, which are also present in G_i. These nodes are removed from G_j and edges which were connected to such nodes are instead connected to the corresponding nodes in G_i.

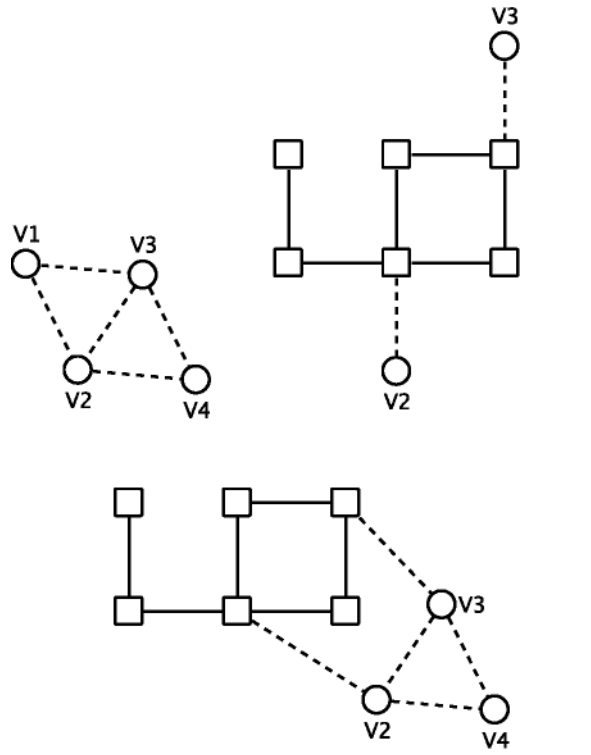

Fig. 4. Replacing a node in the overview graph by a pathway graph. (upper left) The overview graph G, each node (v_1, v_2, v_3, v_4) represents a pathway graph. (upper right) The pathway graph G_1 consisting of 8 nodes, nodes also occurring in the overview graph are shown as circles. (lower) The pathway graph G_1 replaces the node v_1 in the overview graph.

The layout for the extended graph is computed as in the above case of extending the overview graph. Either enough space is computed before the drawing of G_j automatically or the user moves the node v_j prior to extension to an area which provides enough space for the drawing of G_j, see Figure 2. Also the complete drawing could be re-layouted as shown in Figure 6.

Note that this representation preserves the previous structure of the pathways. However, based on the given KGML data the element *a* (usually a compound) would occur twice in the resulting graph (Fig. 5). This is not unusual for KEGG pathways where compounds may also occur more than once within one pathway. However, such multiple elements could be easily removed if necessary by combining them into one element as done in Figure 6. Note also that the extension of a pathway by another pathway not connected to the first pathway is easier than described above as both pathways stay unconnected after such extension step.

2.2.3 Arranging pathways An important aspect for pathway navigation is the adequate and (semi-) automatic layout of drawings combining several pathways. Because the network size increases with more and more pathways a manual layout of a network combining several pathways is insufficient. On the other hand, an automatic layout of the network with the usual layout algorithms produces pictures which are difficult to understand, see Figure 6.

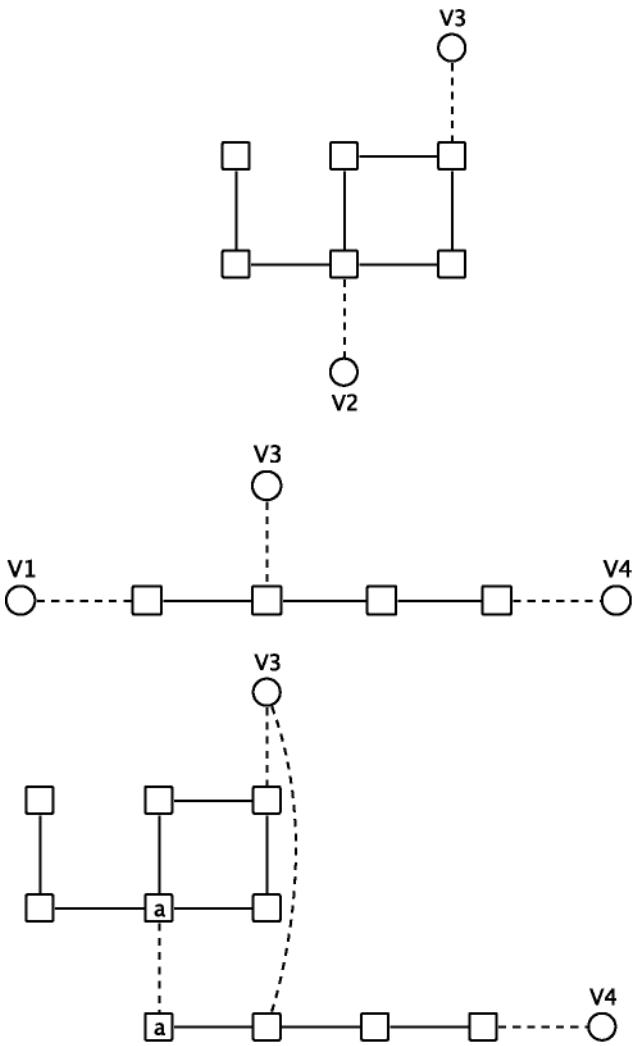

Fig. 5. Extending a pathway graph. (upper) A pathway graph G_1. (middle) A pathway graph G_2. (lower) The pathway graph G_1 is extended by the pathway graph G_2.

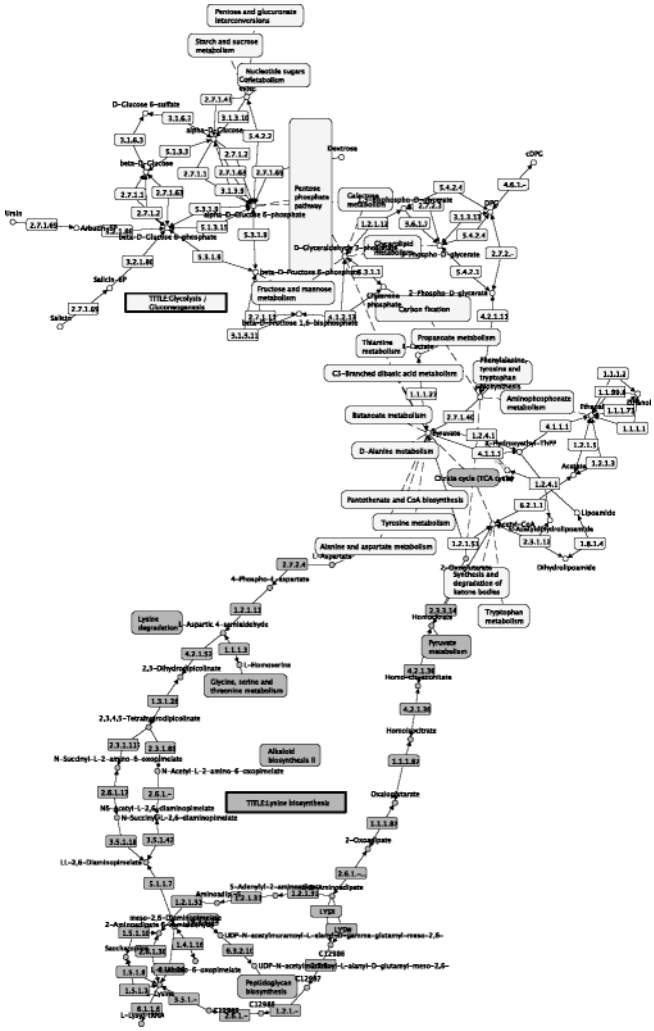

Fig. 6. The combined pathways from Figure 2 (lower) with a new automatic layout (force-directed layout). Additionally, compounds which occur more than once are combined into one element.

Again, the combination of dynamic visualization and semi-static visualization can help, see Figure 3. For a given set of pathways, which should be presented with all details a layout method could be used which works in the following way: First, a specific overview graph G is created where for each pathway graph G_i a node v_i is included in G and edges are computed as before (i. e. if there is a link from one pathway map to another the corresponding nodes are connected by an edge). The size of each node v_i is given by the size of the drawing of the corresponding pathway G_i. Second, the specific overview graph is layouted, e.g. using force-directed (Eades, 1984; Fruchterman and Reingold, 1991) or hierarchical layout methods (Sugiyama *et al*., 1981) as in Section 2.2.1 or other methods, such as grid and circular layout (Six and Follis, 1999). If node overlaps are not avoided by the layout algorithm an additional node overlap removal step can be done (Dwyer *et al*., 2006). Finally, the nodes of the specific overview graph are replaced by the pathway graphs to obtain a layout of all combined pathways, see Figure 3.

2.2.4 Collapsing pathways For interactive navigation also collapsing presented information is important and after a detailed investigation a pathway may be collapsed into an overview node. Such a collapsing (folding) operation for a given pathway G_i, which is part of a graph G is processed as follows: a new node v_i representing the pathway graph G_i is created and placed at the center of the drawing of pathway G_i. Then all edges connecting nodes from outside G_i with nodes inside G_i are reconnected to v_i (instead of the previous node inside G_i) and all nodes of G_i are removed.

3 INTERACTIVE KGML BROWSER AND EDITOR

Not only interactive visualization and exploration of pathways is desirable to study biological processes, but scientists would usually also like to change the pathway structure, e.g. to design species-specific pathways. Therefore we implemented the presented visual exploration methods in a Java based graphical editor and further included editing functionality, which will be described in this section. The KGML visualization and editing tool KGML-ED is

```
<?xml version="1.0"?>
<!DOCTYPE pathway SYSTEM "http://www.genome.jp/kegg/xml/
                          KGML_v0.6.1_.dtd">
<pathway name="path:map00010" org="map" number="00010"
         title="Glycolysis / Gluconeogenesis">
    <entry id="18" name="ec:1.2.4.1" type="enzyme" reaction="rn:R03270"
        link="http://www.genome.jp/dbget-bin/www_bget?enzyme+1.2.4.1">
        <graphics name="1.2.4.1" fgcolor="#000000" bgcolor="#FFFFFF"
            type="rectangle" x="362" y="885" width="45" height="17"/>
    </entry>
    <entry id="96" name="cpd:C00248" type="compound"
        link="http://www.genome.jp/dbget-bin/www_bget?compound+C00248">
        <graphics name="C00248" fgcolor="#000000" bgcolor="#FFFFFF"
            type="circle" x="358" y="927" width="8" height="8"/>
    </entry>
    <entry id="112" name="ec:1.1.1.140" type="enzyme" map="37"
        link="http://www.genome.jp/dbget-bin/show_pathway?
                  map00051+1.1.1.140"/>
...
    <relation entry1="54" entry2="10" type="maplink">
        <subtype name="compound" value="82"/>
    </relation>
...
    <reaction name="rn:R03270" type="irreversible">
        <substrate name="cpd:C00248"/>
        <substrate name="cpd:C05125"/>
        <product name="cpd:C00068"/>
        <product name="cpd:C01136"/>
    </reaction>
</pathway>
```

Fig. 7. Part of the KGML file for map00010, showing KGML definitions for entries, relations and reactions.

based on Gravisto (Bachmaier *et al*., 2005), a graph editor and visualization system, and supports KGML file im- and export, visualization and editing of pathway structures and attributes consistent to the KGML pathway model.

3.1 Editing operations

The KGML-ED system allows the user to modify and edit pathway structures and attributes. Entries may be modified, deleted or newly created, this includes visual attributes, such as graphical position, coloring, node sizes as well as URLs pointing to reference information. Relations and reactions, their substrates, products and the connection to enzymes, may be modified or newly defined. The visualization is automatically updated to reflect these changes.

It is possible to export a modified or user defined graph to KGML. During export to KGML the system analyzes the graph structure and its attributes and generates warning or error messages connected to incomplete or invalid graph structures. This allows an user to locate invalid network elements easily and resolve the problem. The exported KGML files can be subsequently further processed or analyzed with software supporting KGML.

3.2 Transformation between KGML and graphs

A KGML file specifies a pathway with the following entities (Fig. 7): (1) general information about a pathway, (2) entries, (3) relations and (4) reactions. These entities are transformed into a graph with graph, node and edge attributes to represent the KGML defined pathway information. To build a graph from KGML or a KGML file from a graph these entities are processed as follows:

(1) General pathway information (e.g. pathway title) is transformed to and from corresponding graph attributes.

(2) Entries are modeled as graph nodes with node attributes (e.g. for compound name). However, there are also some entries in a KGML file which are not shown in the KEGG pathway image, but which are used to indicate entries existing in other pathways. Such entries have the XML attribute *map*. The information about these entries is stored as node attributes of the corresponding map link nodes (nodes representing links to other pathways).

(3) Relations in KGML are transformed to and from graph edges. In general, a relation connects a source entry with a target entry. If there is no sub-component defined in the KGML file the corresponding graph nodes are directly connected by an edge. In case a sub-component is defined two edges are created. The first edge connects the node representing the source entry with the node for the sub-component entry. The second edge connects the node for the sub-component with the node for the target entry.

(4) A pathway given as a KGML file usually contains a number of reactions. These reactions are identified by a reaction ID, are either reversible or irreversible, and are connected to a number of substrates and products. Additionally a reaction may be catalyzed by a number of enzymes. In the KGML model enzymes contain a XML attribute, which refers to the corresponding reactions. In the graph model all nodes which represent different substrates of a reaction are connected to the corresponding enzyme nodes. From these enzyme nodes there are edges to all nodes representing different products.

The methods to extend and collapse pathways as described in Section 2.2 create a situation different to the original pathway files (where each pathway file contains a single pathway). These methods now allow the computation of combined pathways. To transform such a graph into KGML two new cases have to be considered: (1) edges between nodes in an overview graph G_O are interpreted as relations of type *maplink*, where source and target of the relation point to map entries. And (2) references between pathway elements of two different pathways where both pathways are combined are modeled as relations of type *maplink*, where source and target are not entries of type *map*, but of a different type, e.g. *compound*.

4 CONCLUSIONS

In this paper, novel methods for the dynamic exploration of KEGG pathway diagrams have been presented which support an interactive visual analysis of biological processes. These methods use a unique combination of semi-static and dynamic visualization and present pathway information in a flexible yet easily understandable way. The pathway visualization and editing system KGML-ED provides an implementation of these approaches and allows the user to navigate the KEGG pathways, to combine pathways, edit them, and finally export these pathways as KGML files or in other graph exchange formats [GML (Himsolt, 2000), DOT (Koutsofios and North, 1995), Pajek .NET (Batagelj and Mrvar, 2004), XWG (Dwyer and Eckersley, 2004)] for use in other tools. It is executable on all operating systems, which support the Java 5 runtime (Windows, Linux, Mac OS X and Sun Solaris) and can be installed and started from any web-browser with the help of the Java Web Start technology.

We believe that these novel interaction methods combining semi-static and dynamic visualization, the different levels of pathway information, and the KGML-ED tool help researchers to gain further access to the comprehensive KEGG pathway information

and further increase the usefulness of the KEGG pathway database and its export file format KGML.

5 ACKNOWLEDGEMENTS

This work was supported by the German Ministry of Education and Research (BMBF) under grant 0312706A. We would like to thank Dirk Koschützki and the reviewers for their valuable comments. Funding to pay the Open Access publication charges for this article was provided by the Leibniz Institute of plant Genetics and Crop plant Research (IPK) Gatersleben, Germany.

Conflict of Interest: none declared.

REFERENCES

Bachmaier,C., Brandenburg,F.J., Forster,M., Holleis,P. and Raitner,M. (2005) Gravisto: graph visualization toolkit. In *Proceedings of the International Symposium on Graph Drawing (GD'04),* Lecture Notes in Computer Science, Vol. **3383**, Springer, pp. 502–503.

Batagelj,V. and Mrvar,A. (2004) Pajek—analysis and visualization of large networks. In Jünger,M. and Mutzel,P. eds, *Graph Drawing Software*. Springer, pp. 77–103.

Becker,M.Y. and Rojas,I. (2001) A graph layout algorithm for drawing metabolic pathways. *Bioinformatics*, **17**, 461–467.

Berg,J.M., Tymoczko,J.L. and Stryer,L. (2002) *Biochemistry*. W H Freeman.

Dobkin,D.P., Gansner,E.R., Koutsofios,E. and North,S.C. (1997) Implementing a general-purpose edge router. In *Proceedings of the International Symposium on Graph Drawing (GD'97),* Lecture Notes in Computer Science, Vol. **1353**, Springer, pp. 262–271.

Dwyer,T. and Eckersley,P. (2004) The WilmaScope 3d graph drawing system. In Jünger,M. and Mutzel,P. eds, *Graph Drawing Software*. Springer, pp. 55–76.

Dwyer,T., Marriott,K. and Stuckey,P.J. (2006) Fast node overlap removal. In *Proceedings of the 13th International Symposium on Graph Drawing (GD'05),* Lecture Notes in Computer Science, Vol. **3843**, Springer, pp.153–164.

Eades,P. (1984) A heuristic for graph drawing. *Congressus Numerantium*, **42**, 149–160.

Fruchterman,T. and Reingold,E. (1991) Graph drawing by force-directed placement. *Soft. Prac. Exp*., **21**, 1129–1164.

Himsolt,M. (2000) Graphlet: design and implementation of a graph editor. *Soft. Prac. Exp*., **30**, 1303–1324.

Hu,Z. *et al*. (2004) VisANT: an online visualization and analysis tool for biological interaction data. *BMC Bioinformatics*, **5**, 17.

Junker,B.H., Klukas,C. and Schreiber,F. (2006) VANTED: a system for advanced data analysis and visualization in the context of biological networks. *BMC Bioinformatics*, **7**, 109.

Kanehisa,M. *et al*. (2006) From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res*., **34**, D354–D357.

Koutsofios,E. and North,S.C. (1995) Drawing graphs with dot. *Technical Report*, AT&T Bell Laboratories, Murray Hill, NJ.

Michal,G. (1999) *Biochemical Pathways*. Spektrum Akademischer Verlag.

Nicholson,D.E. (1997) *Metabolic Pathways Map (Poster)*. Sigma Chemical Co., St Louis.

Rojdestvenski,I. (2003) Metabolic pathways in three dimensions. *Bioinformatics*, **19**, 2436–2441.

Schreiber,F. (2002) High quality visualization of biochemical pathways in BioPath. *In Silico Biol*., **2**, 59–73.

Sirava,M. *et al*. (2002) BioMiner—modeling, analyzing, and visualizing biochemical pathways and networks. *Bioinformatics*, **18** (Suppl. 2), S219–S230.

Six,J.M. and Tollis,I.G. (1999) A framework for circular drawings of networks. In *Proceedings of the International Symposium on Graph Drawing (GD'99),* Lecture Notes in Computer Science, Vol. **1731**, Springer, pp. 107–116.

Sugiyama,K., Tagawa,S. and Toda,M. (1981) Methods for visual understanding of hierarchical system structures. *IEEE Trans. Syst. Man Cybern*., **11**, 109–125.