GoVisual

AUTOMATIC DIAGRAM LAYOUT

Diagram Editor

User Guide

Version 1.2

oreas GmbH

Köln, Germany

oreas

OPTIMIZATION RESEARCH AND SOFTWARE

Product and Design

Stiftung caesar

center of advanced European studies and research

Ludwig-Erhard-Allee 2

53175 Bonn

www.caesar.de

Distributor

OREAS GmbH

Optimization Research and Software

Kirschgarten 4

50999 Köln

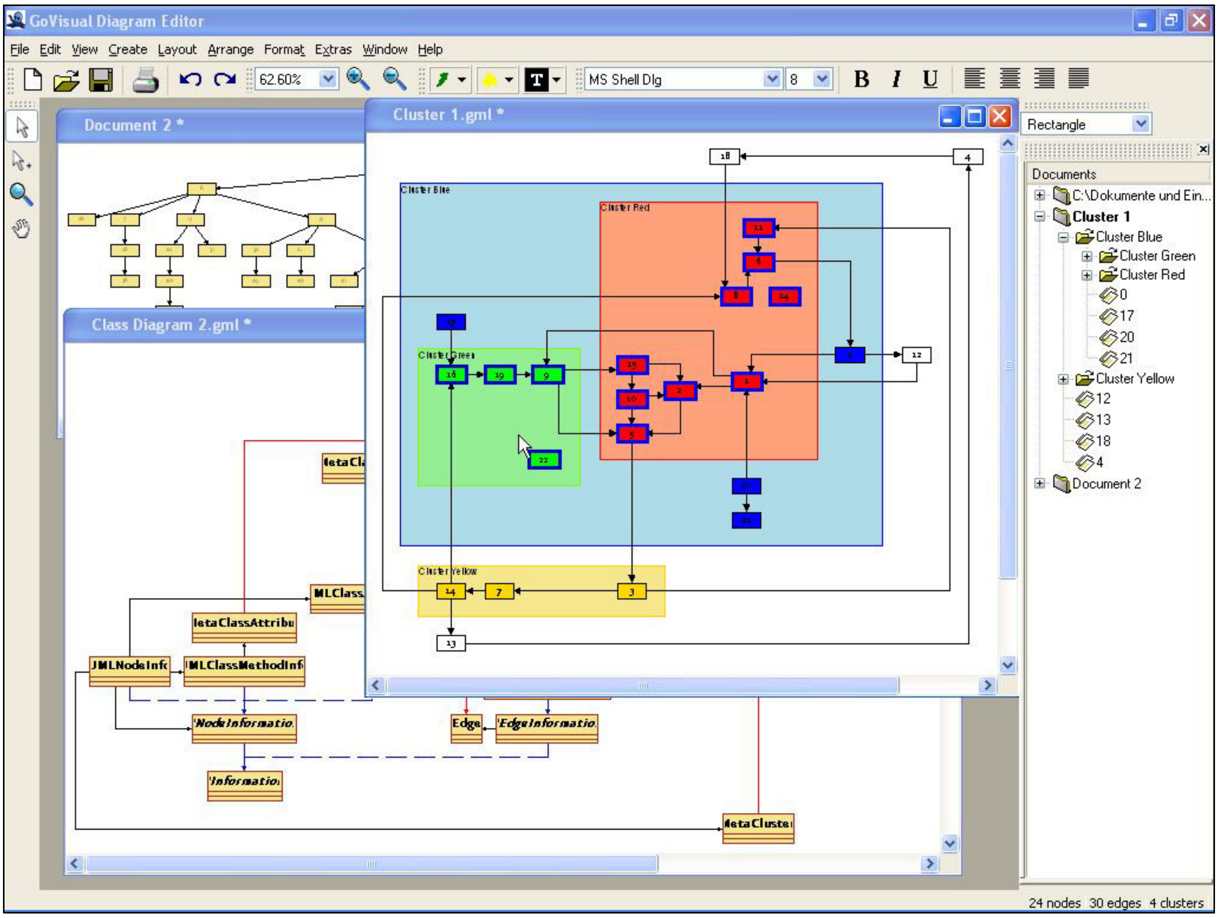
www.oreas.com

gde@oreas.com

Content

1 Introduction

The **GoVisual® Diagram Editor** (GDE) provides sophisticated *diagram editing* and *automatic diagram layout facilities* which help the user to model and understand complicated data. Diagrams are represented as *graphs* or *clustered graphs*. A graph is an abstract mathematical structure consisting of *nodes* (objects) and *edges* (connections between objects). A clustered graph extends the graph structure by the notion of *clusters* which hierarchically group the nodes of the graph. This grouping is represented by the *cluster tree*.



The main window of GoVisual Diagram Editor.

GoVisual Diagram Editor provides the following capabilities:

- Opening and saving data.
- Graph editing capabilities: creating and deleting nodes, edges and clusters, maintaining clusters.
- Layout editing capabilities: moving nodes, edges, bend points and clusters; inserting and deleting bend points; resizing nodes and clusters; aligning nodes and clusters; orthogonal routing of edges.
- Formatting of line style, fill, and text font.
- Undo and redo.
- Zooming and panning.
- Automatic diagram layout in different layout styles.

- Printing.

1.1 GDE Community Edition

The GDE Community Edition is available free of charge. It allows users to create and edit diagrams, as well as to evaluate the automatic diagram layout algorithms contained in the GoVisual API.

The GDE Community Edition license does not permit to modify the software in any way, to decompile, disassemble or reverse-engineer the software, or to sell or rent the software (as a stand-alone product or bundled with other software). Please refer to the license agreement for further details.

2 Getting Help

GoVisual Diagram Editor provides help in various ways.

- The first source of information is the **GoVisual** ***Diagram Editor User Guide*** (this document) which introduces basic concepts of the application. The User Guide can quickly be opened from within the GDE application by pressing <F1>.
- GDE displays information in its **status bar** located at the bottom of the application window.
- Control elements like combination boxes, line edits, or push buttons display **tool tips** that are short descriptions of the functionality associated with the control elements. Tool tips appear in a small rectangular area when the cursor is moved over the control element.
- **Context sensitive help** is a more comprehensive description of the functionality of menu items and other control elements. Users can access context sensitive help by pressing <SHIFT>+F1 or selecting **Context Help** on the tool bar. The application will launch context sensitive help mode as indicated by an arrow with a question mark in place of the mouse cursor. Context sensitive help also works for menu items. Within dialog boxes, context sensitive help is provided by a small question mark button in the upper right corner of the dialog frame. Clicking that button enters context sensitive help mode for this dialog box.

3 Getting Started

3.1 Installation

3.1.1 Microsoft Windows Platforms

GDE comes with a standard installation program. Make sure that all Windows applications are closed and execute **SETUP.EXE** on the GoVisual Diagram Editor CD (or your hard disk if you received GDE in electronic form). Then, follow the instructions of the installation wizard. After successful installation, a new program group called **GoVisual Diagram Editor** will appear in the Windows Start menu.

3.1.2 Linux Platforms

GDE comes a packed tar archive gde-linux.tgz. Unpack the archive into the desired target directory, e.g.,

```
mkdir GDE
cd GDE
tar xvfz gde-linux.tgz
```

3.2 Starting the Application

3.2.1 Microsoft Windows Platforms

After installation, GDE creates a new program group named **GoVisual Diagram Editor**. This can be renamed at the installer's discretion. GDE is launched by selecting the **GoVisual Diagram Editor** application (e.g. **Start** → **Programs** → **GoVisual Diagram Editor** → **GoVisual Diagram Editor**).

3.2.2 Linux Platforms

GDE is started by calling the start script GDECommunityEdition in the installation directory of GDE.

3.2.3 Getting Started

If GDE is started for the first time, a dialog box appears that welcomes the new user.

The main window consists of a *menu bar*, several *tool bars* and a *status bar*. The status bar, located at the bottom of the window displays messages, and shows the number of nodes, edges and clusters in the current graph. If no document is opened, an empty workspace will be seen within this window. The next step is to open or create a graph. A new empty document can be created by **New** from the **File** menu. For opening a document saved on disk, or using a graph generator to create a random graph, please refer to the sections 4, Opening and Saving Documents and 12, Graph Generators for further details.

3.3 Exiting the Application

To exit from GDE, select **Exit** from the **File** menu or click the Close button of the application window. If there are documents open that contain unsaved modifications, GDE will ask you to save them.

4 Opening and Saving Documents

When opening or importing a file, GDE creates a new window for the document within its workspace. The standard file format for opening and saving diagrams is the GML file format. Is it also possible to import from and export to the CSV (Comma Separated Values) file format which is also supported, e.g., by Microsoft Excel. Moreover, drawings may be saved to graphics file formats like JPEG and SVG. In this case, it is not possible to load the diagram back into GDE since the graph structure is not stored in a graphics file.

4.1 Opening and Saving Graphs (GML)

A document can be opened several ways. On Windows platforms, GDE associates GML files with the GDE application. This allows to open a GML file with a double click. If GDE is already open, the file will be opened in the running GDE application. It is also possible to open a GML file by dragging the file onto the GDE application icon.

Within the GDE application, you open a file by selecting the menu item **Open** from the **File** menu. Alternatively, click on the respective icon on the standard tool bar or use the accelerator, <CTRL>+O, to open a new document. A dialog box will open allowing for the selection of a file. Files opened this way will be stored in GDE's proprietary file format. Currently, GDE uses the GML graph file format extended by the notion of a cluster tree. Later releases will be changed to an XML based file format. GDE will still support GML in future releases so that graphs stored in the GML file format can be imported.

The current graph that is displayed in the active workspace window can be saved several ways. First, select **Save** on the menu bar. Alternatively, press <CTRL>+S or the **Save** icon from the standard tool bar.

Another option is to select **Save As** from the **File** menu. Once **Save As** has been selected, a dialog box appears allowing the user to choose the file name. If no file extension is specified the standard file extension (e.g. "`.gml`") is appended. Choosing **Save** for a document which has not yet been assigned a file name will also display this dialog. Saving all currently opened documents can be done by selecting **Save All** from the **File** menu.

GDE maintains a "*most recently used files*" list for user convenience. This list contains the names of the files that have been opened or saved most recently. It appears at the end of the **File** menu just before the **Exit** item. Clicking on the file name opens the file.

If a document is closed that contains unsaved modifications, GDE will prompt the user to save the document before closing it. The same will occur when GDE is exited. A modified document can easily be recognized by the asterisk "*" appended to the document name in the caption of the window.

4.2 Comma separated values (CSV)

An easy way to import data from other applications is to use the CSV file format. These files are text files that simply list the edges in the graph. A CSV file can be exported from Microsoft Excel. A CSV file represents a table consisting of rows and columns. Each line of the file represents one row of the table (also called record). The columns are separated using the so-called list separator character. This character is depends on the region settings of Windows and can be adjusted in the Windows Control Panel. The standard for the United States is a comma ',' and for Germany a semicolon ';'. Note that Excel as well as GDE use the current list separator character settings on Windows platforms. On Linux platforms, a comma is always used as separator. An example CSV file representing an edge list might look as follows (U.S. regional settings, so the list separator is ',').

```
1, Node 1, Node 2, orange
2, Node 3, Node 4, blue
3, Node 1, Node 3, "white, blue and orange"
4, Node 5, Node 2,
5, Node 3, Node 5, whatever
6, Node 4, Node 5,
```

This file represents a table with 6 rows and 4 columns. GDE actually considers only two columns which are interpreted as edge list. In this case, we choose column 2 to represent the source node of an edge, and column 3 to be the target node of an edge. Thus, the file above represents a graph consisting of 5 nodes and 6 edges (the nodes are given implicitly by the edges). Column 1 and column 4 are simply ignored. Since nodes are identified by their names, it is essential that different nodes are also given different names.

CSV files can be imported by selecting **Import** from the **File** menu (or using the accelerator <CTRL>+I). The Import dialog box allows to select a file format from the file type combination box. The default is CSV, however GML can also be selected. Next, select the file name and click **OK**. GDE will ask for the column of the source node of an edge and the column for the target node. Adjust the settings according to requirements of your import file and click **OK**. Since graphs stored in CSV format do not contain any layout information, all nodes will be placed on top of each other after importing is complete. The easiest way to get an overview of the graph is to apply automatic layout after importing. Please refer to the section Automatic Layout for more information.

Graphs can also be exported to the CSV file format by selecting **Export** from the **File** menu (or using the accelerator <CTRL>+E). Chose the CSV format in the file type combination box and select the file name. Click **OK** and GDE exports the current graph as CSV file.

4.3 Exporting graphics formats (JPEG, PNG, BMP, SVG)

GDE can export a diagram to various graphics formats. It supports the pixel formats JPEG, PNG (Portable Network Graphics) and BMP as well as the vector format SVG (Scalable Vector Graphics).

For exporting the current diagram to a graphics format, select **Export** from the **File** menu (or use the accelerator <CTRL>+E). The choose the respective file format in the file type combination box, select the file name and press OK. If a pixel format is chosen, GDE will ask for the size of the exported image in pixel, and the quality of the image if supported by the file format.

5 The Workspace

The GDE application complies with the **multiple document interface (MDI)** standard meaning that several documents (graphs) can be opened at one time. Each document appears in its own window. These windows are arranged in the application's *workspace*. A document window can be maximized in which it occupies the complete workspace. Each window can also be minimized so that only its title bar is shown at the bottom of the workspace. A window can be normalized so that several documents can be seen at once within the workspace.

If there are open documents, only one of them is *active*. Most operations performed will only affect the active graph.

By pressing <CTRL>+<TAB> the user can move back and forth between different documents. The **Window** menu maintains the list of all currently opened documents. This list is sorted according to latest usage, so that the most recently used document appears in first place. Selecting a document from that list activates the respective window. The **Window** menu contains options for closing document windows, or arranging them in a tile or cascade pattern.

Apart from document windows, GDE provides also docking windows and toolbars, which can be docked at any of the four sides of the workspace area, or which can be undocked so that they appear in a small freely movable window.

6 Dockable Tool Bars and Windows

GDE offers several *dockable tool bars*. The **Standard** tool bar contains actions for opening, saving and printing. The **View** tool bar contains actions for zooming. The **Format** tool bar contains actions for formatting line styles and object fills, and the **Format Text** tool bar contains actions for formatting the text font and paragraph alignment. The **Templates** tool bar provides a combination box for selecting the node type that is used for creating a new node. By default, these tool bars appear at the top of the workspace area below the menu bar. The **Tools** tool bar is located at the left side by default. It provides different tools for working within document windows. Depending on the tool activated, the appearance of the mouse cursor changes and mouse operations like clicking and dragging within a document window are interpreted differently.

There are currently four tools available.

- The **Select** tool indicated by a usual arrow cursor allows to select nodes, edges and clusters, and to move nodes, edges, bend points of edges and clusters.
- The **Create** tool is used for creating nodes and edges, and for inserting bend points into existing edges. It also performs some of the actions provided by the Select tool for convenience.
- The **Zoom** tool which is shown by a magnifying glass cursor is used for zooming in and out, as well as for zooming to a specific area.
- Finally, the **Scroll** tool turns the cursor into a hand symbol and serves for easy two-dimensional scrolling.

The most common tool is the Select tool. It can quickly be activated by pressing the <SPACE> bar. If it was already activated, the previously used tool is activated instead, which allows to easily switch between the Select tool and another tool. The different tools are discussed in detail in the sections 7, ***The Document Window*** and 9, ***Navigation***.

Besides the various tool bars, there are also dockable windows. They essentially behave like tool bars, but they provide more functionality than simply clicking a tool button. Currently, the **cluster explorer** is the only dockable window. It is located at the right side of the application window by default and provides functionality for displaying and managing the cluster tree of a clustered graph.

Tool bars and dockable windows can be docked at any of the four sides of the workspace area. They can also be undocked so that they appear in a small freely movable window called a *floating window*.

To change the position of a tool bar or a dockable window, click the handle on the left (or top) side of the tool bar and drag it to the desired area. When a tool bar or a dockable window is drug to the border of the workspace area, it will be docked at that side. If it is drug within this area, it will change to a floating window.

Once the tool bar or dockable window is turned into a floating window, the handle disappears and it can be moved by clicking in the title bar. Double clicking the title bar will dock the window again. Similarly, double clicking the handle of a docked window will undock it. A single click on the handle will minimize the tool bar or dockable window, so that only the handle is shown. Clicking the handle again normalizes the window and places it in its original location.

If a tool bar or dockable window is displayed as a floating window, a close button appears at the top right corner. If this button is clicked, the tool bar or dockable window disappears completely. To make it reappear right click in the menu or docking area and select the desired tool bar or dockable window. You can also close tool bars or dockable windows by un-checking them. Alternatively, select or deselect tool bars in the **View→Toolbars** submenu, or **Cluster Explorer** in the **View** menu.

Undocking and repositioning of tool bars can be forbidden by checking the item **Lock the Toolbars** in the **View** menu. In this case, docked tool bars will not have a handle anymore. Floating tool bars can still be repositioned and docked, but once they are docked they will be locked as well. To allow repositioning and undocking again, uncheck **Lock the Toolbars**. This does not affect other dockable windows.

The layout of tool bars, including their position and if they are open or closed, docked or floating, is stored when GDE is closed. When launching GDE again, the layout is restored.

7 The Document Window

A document window displays a graph and allows the modification of the graph and its layout as well as the navigation within the displayed graph. Navigation is covered in Section 9, ***Navigation*** below. Nodes are usually represented as rectangles containing a text label. Edges may be straight-lines, or polylines containing bend points. Optionally, edges can be directed or undirected, that is they are drawn with arrowheads or not. This is controlled by the menu item **Show Edge Directions** in the **View** menu. If this item is checked, arrowheads are drawn, otherwise not. Clusters are shown as rectangles with a label in the upper left corner. Clusters are only shown if the **Show Clusters** item in the **View** menu is checked.

If the Select or Create tool is active, right clicking on a node, edge or the border of a cluster displays a context menu for the respective object. Right clicking on the background of a document window also shows a context menu with some frequently used actions from the **File**, **Edit** and **View** menu. All context menus can be closed without selecting an item by pressing <ESC>.

7.1 Selecting Nodes, Edges and Clusters

Mouse actions within the drawing area are handled according to the selected tool. Objects like nodes, edges and clusters are selected using the **Select tool**. Clicking on an object selects the object. If the <SHIFT> button is pressed while clicking, the object is added to the current selection if it was not selected before, or removed from the selection if it was already selected. For ease of usage, a cluster can only be selected by clicking on the surrounding border of the cluster. Clicking in the interior part of the cluster behaves like clicking on the background.

Several nodes and edges can easily be selected by clicking on the background and dragging a selection rectangle. This will select all nodes and edges contained in the rectangle. If <SHIFT> is pressed, all nodes and edges not yet selected will be added to the selection, and all nodes and edges already selected will be removed from the selection. Dragging the selection area near the edge of the document window automatically scrolls the drawing area, which allows to include parts of the graph that are not yet visible. Dragging a selection rectangle can be aborted by pressing <ESC>.

Clicking on the background clears the selection again. All nodes and edges are selected by selecting **Select All** from the **Edit** menu, or alternatively by pressing <CTRL>+A. Selecting all objects of the same type (e.g. all nodes) can easily be done by selecting the respective item from the submenu **Select** in the **Edit** menu. The respective objects are always added to the current selection. If exactly those objects shall be selected, make sure that the current selection is empty before by clicking on the background. Similarly, all objects of a specific type can be deselected by choosing the respective item from the submenu **Deselect** in the **Edit** menu.

7.2 Positioning Nodes and Clusters, Routing Edges

The position of nodes and clusters as well as bend points of edges can be modified using the **Select tool**. Some operations are also provided by the Create tool. Nodes can be moved by clicking the node with the left mouse button and dragging the mouse to the desired position. Similarly, bend points can be moved by clicking and dragging.

Clusters can be moved by clicking on the surrounding border and dragging. Moving a cluster will only move the graphical object, and not the nodes and clusters contained in it. If the complete subgraph consisting of the cluster, all its descendants in the cluster tree and all edges between those nodes shall be moved, proceed as follows:

1. Make sure that no objects are selected by clicking on the background.

2. Right click on the cluster and choose **Select Pertinent Subgraph** from the context menu that appears.

3. Click on any object that is now selected and drag.

The start and end points of edges are called *anchor points*. They lie within the area occupied by the adjacent node. By default, anchor points lie in the center of that node, but they can be positioned freely within their allotted area. In order to move anchor points, it is recommended to select only the edge where the anchor point is attached. Once the edge is selected, anchor and bend points are clearly highlighted by small markers. Otherwise, the position of an anchor point is hard to see, since an edge is cut at the borders of their adjacent nodes.

Pressing the <CTRL> key while moving an object restricts the move operation to move only horizontally or vertically. When the mouse cursor is moved near the edge of the window while dragging, the drawing area is automatically scrolled. This allows to easily move objects to positions that are currently not visible. New bend points are inserted using the Create tool (see ***Creating and Deleting Nodes, Edges and Bend Points*** below).

An edges can also be moved by clicking on the edge (but not a bend point) and dragging, that is all bend points are moved and the anchor points at nodes remain at their position. I.e., moving a straight-line edge will not move anything. If an object is moved that is currently selected, the whole selection will be moved at once.

Some more tools are provided in order to make layout editing easier. Frequently, some nodes in the diagram shall be aligned, e.g. horizontally. Alignment of nodes is provided by the **Align Objects** entry in the **Arrange** menu. The dialog box that appears allows to align the selected nodes in almost any way. It is also possible to align clusters, but this is rarely useful. It is also hard to achieve an orthogonal routing of edges by just moving line points by hand. The **Beautify Orthogonal** tool provided by the context menu of an edge transforms an almost orthogonal routing into a truly orthogonal routing consisting of only horizontal and vertical line segments. The anchor points of the edge remain fixed and the bend points are used to find the desired orthogonal routing.

7.3 Resizing Nodes, Clusters and Edges

The size of graphical objects can be adjusted using the **Select tool**. When an object is selected, eight small rectangles (*markers*) appear around the object indicating that it is selected. Clicking and dragging such a marker resizes the selected object. Dragging one of the middle markers adjusts only the width or height, respectively. Dragging a corner marker adjusts the width and height. The modifier keys <CTRL> and <SHIFT> provide further options:

- If the <CTRL> key is not pressed while dragging a corner marker, the object is resized such that the width-to-height ratio is preserved, otherwise the object can freely be resized.
- If the <SHIFT> key is pressed while dragging, the center of the object remains at the same position.

When an edge is resized, only the bend points are affected and the anchor points remain at their position. If several objects are selected, the complete selection is resized at once.

7.4 Creating and Deleting Nodes, Edges and Bend Points

Nodes and edges as well as bend points of edges are created with the **Create tool**. Clicking on the background creates a new node. There are several types of nodes (called **templates**) available. The template that is used for creating a new node is displayed by the combination box in the **Templates** tool bar. The template used for creating a new node determines the graphical appearance and behavior of the node. The following templates are currently provided:

- **Simple Rectangle**: A rectangular node with a single line text label. The font and color of the label text can only be changed for the complete label. The node can be resized arbitrarily.
- **Rectangle**: A rectangular node with a rich text label. The text can be formatted in several paragraphs and lines are marked up according to the current width of the node. The text can be formatted individually, that is the font or color can be changed for any piece of the text if desired. The node can be resized arbitrarily.
- **UML Class**: A rectangular node used for displaying a class in a UML class diagram. The node shows sections for the class name, the attributes and the operations of the class. The sections can be easily edited with a specially designed dialog box. The size of the node is adjusted automatically, so that is not necessary (or possible) to resize the node.

Depending on the template used to create a node, different actions may be provided by its context menu.

Edges are created as follows:

1. Click on the node at which the new edge shall start. When moving the cursor, you will notice a new line from the starting point to the mouse cursor.
2. For every bend point the new edge shall have click on the background. The creation of the edge can be aborted by a right click or by pressing <ESC>.
3. Click on the node at which the new edge shall end.

New bend points can be inserted into an existing edge by clicking on an edge at a position where there is no bend point and dragging.

Nodes and edges can be deleted several ways. The most common way is to select all the nodes and edges that shall be deleted, and then pressing <DEL> (or selecting **Delete** from the **Edit** menu). This works with all tools. Alternatively, context menus can be used if the Create or Select tool is active. Right click on the node (or edge) and select **Delete** from the context menu that appears. **Delete** will delete the complete selection, so if the node (or edge) and some other objects had been selected before, all these objects will be deleted.

Bend points can also be removed using the context menu for edges. Right click on a bend point and select **Remove Bend** from the context menu that appears. If all bend points shall be removed from the edges, right click on the edge (anywhere) and select **Remove All Bends** from the context menu. This operation will result in a straight line.

The graphical appearance of a newly created node or edge can be adjusted by the user. Please refer to Section 13.1, ***Preferences for Drawing Nodes, Edges and Clusters***.

7.5 Managing Clusters

GoVisual Graph Editor is able to handle clustered graphs in which the nodes of the graph are hierarchically grouped according to a cluster tree. Each node and each cluster except for the root cluster are contained in exactly one cluster. The root cluster is on top of the hierarchy and will not be graphically displayed in the editor. Clusters will only be shown, if the item **Show Clusters** in the **View** menu is checked. If it is not checked, menu entries dealing with clusters are disabled or omitted.

The cluster structure of the graph can also be displayed and managed using the **cluster explorer** dock window. It displays the nodes and clusters in a document in a tree view and allows to manipulate the hierarchical structure using drag & drop. Please refer to Section 8, ***The Cluster Explorer***, for detailed information on the usage of the cluster explorer. In this section, we will only consider the functionality provided by the document window for managing clusters.

A new cluster can be created in two ways. In order to create a new cluster without using the explorer, proceed as follows:

1. Select all nodes and clusters in the drawing area that shall be contained in the new cluster.
2. Right click on a selected object and choose **Create Cluster** from the context menu that appears.

The new cluster will contain exactly the selected nodes and clusters. The parent of the new cluster will be the least common ancestor of the selected nodes and clusters in the cluster tree. Its graphical representation is a rectangle that surrounds all nodes and clusters contained in it.

Clusters can be deleted in the same way as nodes and edges are deleted in the drawing area, e.g. by selecting the cluster and pressing <DEL>. Deleting a cluster will not delete the nodes and clusters contained in it. Instead, those nodes and clusters are assigned a new parent which is the parent of the deleted cluster. In order to remove a cluster and all its descendants in the cluster tree, proceed as follows:

1. Make sure that no objects are selected by clicking on the background.
2. Right click on the cluster and choose **Select Pertinent Subgraph** from the context menu that appears. This will select all descendants of the cluster in the cluster tree as well as all edges between nodes that are descendants of the cluster.
3. Press <DEL>.

GDE allows a cluster to be drawn such that it does not surround all nodes and clusters contained in it. This is usually not desired for a finished drawing, but might be useful as intermediate stage of the interactive drawing process. The size of clusters can be adjusted so that clusters correctly surround all nodes and clusters contained. Select all clusters that should be adjusted and right-click on a selected clusters. Then choose **Adjust Size** from the context menu that appears. The size of a cluster is chosen such that there is a margin between the line of the cluster rectangle and objects contained. The size of this margin can be changed at the **Canvas** topic page of the options dialog box (choose **Extras** → **Options**, and click on **Canvas** in the list view). In order to apply this process recursively to all descendants of a cluster, all these clusters must be selected, i.e. only the size and position of selected clusters is adjusted, and the size and position of all other clusters is used as is. As a simple example, the size and position of one cluster and all of its descendant clusters can be adjusted as follows:

1. Make sure that no objects are selected by clicking on the background.
2. Right click on the cluster and choose **Select Pertinent Subgraph** from the context menu that appears.
3. Right click on any selected cluster and choose **Adjust Size** from the context menu.

7.6 Formatting Line Style and Fill

The line style of a node or cluster refers to the color, style and width of the line drawn as outline around the graphical object representing the node or cluster. The line style of an edge refers to the color, style and width used to draw the line representing the edge as well as the style of the line ends, e.g. the arrow heads. The fill of a node or cluster refers to the color and fill pattern used to draw the node or cluster object.

GDE provides various ways for formatting line style and fill. Formatting is always applied to the selected objects in the active document window. Selecting **Line Style** or **Fill** from the **Format** menu or context menu opens a dialog which allows to adjust all settings for line style or fill, respectively. If only the color of the line style or fill shall be changed, actions from the Format tool bar can be applied. This tool bar provides combination boxes for selecting a color for line style or fill from a collection of predefined colors.

7.7 Formatting Text

The label text of nodes can be formatted in the following way. If one or more node objects are selected, the format of the whole text of all selected nodes can be adjusted by selecting the **Text** item in the **Format** or context menu or by using actions from the Format Text tool bar. The **Format Text** dialog allows to adjust the font by choosing the font family, style and point size from the list of installed fonts as well as the text color and some additional effects like underline and strike through. Some of these font attributes can be adjusted separately using the controls in the **Format Text** tool bar. This tool bar also allows to adjust the paragraph alignment to left-, right-, center or justified alignment. The text color can be changed using the text color tool in the **Format** tool bar.

The text displayed within a node object can be edited depending on the node template. Node editing is initiated by double-clicking the node or selecting **Edit** from its context menu. Simple Rectangle and UML Class nodes provide dialog boxes for editing. Font and alignment can only be changed for the whole node by selecting the node and changing the font or text color.

The label of a Rectangle node can be edited and formatted in a word processor like fashion. Double clicking a Rectangle node with the Select tool opens a text editor for editing and formatting the text within a node object in almost any way. When editing the text in a node object, the size of the node object can be adjusted by dragging the selection markers around the node and the text is formatted according to the given node size. When the text editor is active, all actions for formatting text font, alignment and color are applied to the current text selection in the editor, or change the settings of the current font, text color or alignment, respectively, if no text is selected. These settings are used when new text is typed.

7.8 Formatting Semantic Information

Edges can be assigned a special type which is observed by some layout styles. In particular, edges can have the following types:

- association
- generalization
- dependency

The default is association. The type generalization is used in UML class diagrams for specifying inheritance relations. Currently, only the UML Orthogonal and UML Hierarchical layout styles consider edge types and distinguish between generalization edges and other edges.

The type of edges can be set by selecting these edges and choosing **Edge Semantics** from the **Format** menu or the context menu of an edge. The dialog that appears shows a list box with the available types.

7.9 Undo and Redo

GDE provides *Undo* and *Redo* for all operations that modify the graph, its cluster structure or its layout. This includes creation and deletion of nodes, edges and clusters, repositioning of nodes and clusters, rerouting of edges by moving, inserting or removing bend points, applying automatic layout, and showing or hiding edge directions or clusters. Undo and Redo is always applied to the active graph. That is, applying Undo will undo the last edit operation of the active document, which is not necessarily the last edit operation performed. Each document maintains its own Undo and Redo buffer.

Undo is applied by selecting the **Undo** menu entry from the **Edit** menu, pressing the **Undo** button in the Standard tool bar, or using the accelerator <CTRL>+Z. The size of the undo and redo buffer, that is the maximal number of operations that can be undone or redone, can be adjusted on the **Edit** page in the option dialog box (choose **Extras** → **Options**, and click on **Edit** in the list view). Undone edit operations can be redone by selecting the **Redo** menu entry from the **Edit** menu, pressing the **Redo** button in the Standard tool bar, or using the accelerator <CTRL>+Y.

For both Undo and Redo, a short description of the operation that will be affected is displayed in the menu entry, the tool tip of the tool bar button, and the status bar. Scrolling and Zooming are not covered by the undo/redo mechanism, since they do not modify the graph. Therefore, an undo or redo operation might not be directly visible, if the area affected by the respective operation is currently not visible on the screen.

8 The Cluster Explorer

The cluster structure of the graph can be displayed and managed using the **cluster explorer** dock window. If it is not visible, make sure that the **Cluster Explorer** item in the **View** menu is checked. By default, the cluster explorer appears docked at the right border of the application window, but it can freely be docked at another place or be undocked and freely moved as a floating window. The cluster explorer displays the nodes and clusters in a document in a tree view. The tree view contains all open documents. The active document is highlighted in bold face. The document item also represents the root cluster of the respective graph. When the mouse is moved over an item, the cluster explorer displays some useful information about the object.

8.1 Navigation and Selection

The cluster explorer list view can be navigated in the usual manner. Double clicking on a document or cluster item, or clicking the plus or minus sign next to the item, expands or collapses a the item, making the nodes and clusters contained in the associated cluster visible or not. The list view can also be navigated with the keyboards as shown in the following table:

key	*action*
<HOME>	Makes the very first item the new current item.
<END>	Makes the very last item the new current item.
<PAGE UP>	Makes the item above the top visible item the new current item. This can be used to scroll page wide up.
<PAGE DOWN>	Makes the item below the bottom visible item the new current item. This can be used to scroll page wide down.
< UP>	Makes the item above the current item the new current item.
< DOWN>	Makes the item below the current item the new current item.
< LEFT>	If the current item is collapsed or has no children, makes its parent item the new current item. If the current item is expanded, it will be collapsed.
< RIGHT>	If the current item is collapsed and has children, the item will be expanded. If the current item is expanded and has children, the first child of this item is made the new current item.

The tree view allows to freely select nodes and clusters in a document and apply operations on the current selection. It is not allowed to select nodes or clusters from different documents. If an item from a different document is selected, all previously selected items will be deselected. The list view supports the usual operations for multiple selection, i.e. pressing <CTRL> while selecting adds the item to the current selection, and pressing <SHIFT> allows to select a range of items.

The selection in the cluster explorer is not synchronized with the selection within the document window, that is selecting a node or cluster in the list view does not select it in the document window as well and vice versa. If you want to select the item in the cluster explorer that belongs to an object in the document window, right-click on this object and select **Show in Explorer** from the context menu that appears.

The list view can also be used to a assign a new label to a node or cluster. Clicking on a selected item, or right clicking on an item and choosing **Rename** from the context menu that appears, allows to edit the item label. Pressing <RETURN> will assign the new name to the label of the respective node or cluster. Pressing <ESC> will cancel renaming and leave the original name unchanged.

8.2 Locating Objects in the Document Window

Finding the graphical object in the document window that corresponds to a node or cluster item in the cluster explorer list view is quite easy. Double-clicking the explorer item will activate the document window containing the object and will select and center the corresponding object. Alternatively, it is also possible to press <RETURN> in which case the current list item will be shown, or to right-click on the explorer item and to select **Show in Document** from the context menu that appears. If the item that shall be located in the document window is a cluster, the zoom factor and scroll bars will be adjusted, such that the cluster is completely visible and centered.

A document can be activated by double-clicking the respective explorer item, or by selecting **Activate** from the context menu. The context menu of a document also contains the menu item **Remove**, which closes the document.

The items in the list view are always in lexicographical order such that cluster items appear before node items within a cluster or document. The list view supports searching for an item by name as follows. If the user starts typing letters or digits while the list view has the focus, an incremental search for an item that begins with these letters is started, that is the next item after the current item that begins with these letters is selected and becomes the new current item.

8.3 Assigning Nodes and Edges to Clusters

The cluster explorer provides drag & drop functionality in order to a assign nodes and clusters to a new parent cluster.

1. Select all nodes and clusters in the cluster explorer that shall be assigned to a new parent cluster.
2. Click on a selected item and drag it to the desired cluster item, or the document item if the selection shall be moved to the root cluster. The current target cluster will be highlighted while dragging. Moving the cursor over a cluster item that is not yet expanded for a short period of time will expand the cluster item. This allows to navigate to a desired target cluster that is not yet visible since one of its ancestor clusters is expanded.
3. If the cursor is over the target cluster, release the mouse button. All selected items will now be contained in this cluster. Notice that it is not possible to move the selection to a descendant of a selected cluster since this would introduce a cyclic dependency in the cluster tree.

The assignment of a new parent cluster will not affect the drawing of the graph, so it might be the case that a cluster is drawn in a way so that it does not surround all clusters and nodes it contains.

8.4 Creation of New Clusters

Creating a new cluster using the cluster explorer works as follows:

1. Within the cluster explorer, select all nodes and clusters that shall be contained in the new cluster.
2. Right click on a selected item and choose **New Cluster** from the context menu that appears. This will create a new cluster containing all selected nodes and clusters. The parent of the new cluster will be the least common ancestor of the selected nodes and clusters in the cluster tree.

The new cluster will now be selected.

8.5 Deleting Nodes and Clusters

The cluster explorer also allows to delete nodes and clusters. An important difference to the deletion operation provided by the document window is that deleting a cluster in the cluster explorer will also delete all objects contained in the cluster, i.e. the complete subtree rooted at this cluster. Though edges cannot be selected within the cluster explorer, deleting a node will also delete all adjacent edges. Objects are deleted with the cluster explorer as follows:

1. Within the cluster explorer, select all nodes and clusters that shall be deleted. Be aware that the deletion of a cluster will remove the complete subtree.
2. Press <DEL> or right click on a selected item and choose **Delete** from the context menu that appears.

9 Navigation

Navigational features are provided in various ways: by the **View** menu, the *View* tool bar, the *Zoom* and *Scroll* tools, the scroll bars, and some accelerators. All actions concerning navigation are applied to the active document only.

9.1 Scrolling

If the graph is larger than the window's size, utilize the horizontal and vertical scroll bars to move to a specific area of the graph. Alternatively, activate the **Scroll tool** and scroll the drawing area by clicking with the left mouse button and dragging the mouse. Additionally, the item **Center Graph** located in the **View** menu, or accelerator <F6>, adjusts the scroll bars so that the graph is centered within the window (as far as this is possible), and the item **Center Selection**, or accelerator <SHIFT>+<F6>, adjusts the scroll bars so that the current selection is centered.

9.2 Zooming

The zoom factor for the active document is displayed in the combination box in the **View** tool bar. The zoom factor can be adjusted by selecting an item from the drop-down list of the combination box, or by entering a zoom factor directly.

There are also three special items in the drop-down list: **selection**, **graph**, **width** and **height**.

- By selecting selection, the zoom factor is adjusted such that the whole selection can be displayed within the window. The item is only available if the selection is not empty. The same action is also listed in the **View** menu under the item **Zoom to Selection** (accelerator <SHIFT>+<F5>).
- By selecting graph, the zoom factor is adjusted such that the whole graph can be displayed within the window. The same action is also listed in the **View** menu under the item **Zoom to Graph** (accelerator <F5>).
- By selecting **width**, the zoom factor is adjusted such that the whole width of the graph can be displayed within the window. In particular, the whole graph can be viewed just by scrolling up and down.
- By selecting **height**, the zoom factor is adjusted such that the whole height of the graph can be displayed within the window. In particular, the whole graph can be viewed just by scrolling left and right.

Zooming in and out is performed by selecting the respective items from the **View** menu or by clicking the respective icons in the **View** tool bar. **Zoom In** (accelerator <+>) doubles the zoom factor and **Zoom Out** (accelerator <−>) halves it. The operations for zooming in and out, as well as zooming to a specific zoom factor adjust the scroll bars. If no node or edge is selected, they make sure that the same point remains in the center of the document window. If the current selection is not empty, the center of the selection is scrolled to the center of the window.

More accurate control is provided by the **Zoom tool**. When activated, the cursor turns into a magnifying glass with a little plus sign, and clicking with the left mouse button zooms into this point. In particular, the point where you clicked the mouse remains at the same position. Similarly, clicking with the right button performs zooming out. The Zoom tool also allows to zoom to a particular area. Clicking with the left button and dragging opens a rectangle that specifies the area to zoom in. The drawing area is automatically scrolled, when dragging the mouse near the edge of the window. This allows to extend the zoom area beyond the visible part of the drawing area.

Zooming operations with the Zoom tool can be aborted with the <ESC> key. Zooming in and out will be aborted when the <ESC> key is pressed while the mouse button is still hold down, and zooming to area will be aborted when the <ESC> key is pressed while dragging the mouse.

Moreover, navigation is also provided by the context menu of the document's background. It will be opened when right clicking to the background with the Select or Create tool. It provides actions for zooming to the graph, zooming in and out, and centering the graph. The zoom in and zoom out actions behave like the respective operations with the Zoom tool. They assure that the point within the drawing area where you opened the context menu remains at the same position.

10 Automatic Layout

GDE provides the following automatic layout styles:

- Orthogonal
- Cluster-Orthogonal
- UML-Orthogonal
- UML-Hierarchical
- Hierarchical
- Tree
- Symmetric
- Circular

The different layout styles are available in the **Layout** menu. Applying a layout style repositions the nodes in the graph and reroutes the edges. Currently, there is only one layout style that also repositions clusters (Cluster-Orthogonal Layout). Layout styles are applicable independent of the fact that the graph contains clusters or not, but layout styles that do not handle clusters will hide clusters by un-checking the **Show Clusters** item in the **View**.

Most layout styles allow to adjust various parameters in the dialog box that appears when calling the layout style. GDE stores all option settings when the dialog is confirmed with **OK**. The default values of the parameters can be restored by clicking the **Default** button. All layout styles provide an option **zoom to graph** which adjusts the zoom factor so that the complete graph fits into the document window, after the layout process has been completed.

Some layout styles might not be applicable for a particular graph, e.g. only few layout styles supports self-loops (edges that connect a node with itself) at the moment. This is checked before calling the layout style. If the layout style is not applicable, this will be reported to the user with a message box.

10.1 The Orthogonal Layout Style

The **Orthogonal** layout style draws edges as lines consisting of horizontal and vertical segments. It aims to minimize the number of edge crossings. If a graph can be drawn without edge crossings in theory, the displayed graph will not contain any edge crossing at all. This is the only layout style that can guarantee this. Orthogonal layout is best suited if

- orthogonal edge routing is preferred.
- the graph can be drawn with few edge crossings. Note that this might be possible even if a layout style like Symmetric requires many crossings.
- the graph is not too dense and not too large.

Orthogonal layout is not applicable if the graph contains self-loops.

When the Orthogonal layout style is selected, a dialog box will appear where the following settings can be adjusted. The parameter **crossing minimization** determines the quality of the crossing minimization procedure. For small or sparse graphs, the algorithm is usually very quick, but for large or dense graphs, the running time will be longer. When laying out a large graph, it is recommended that fast crossing minimization is tried first in order to assess the running time. The density of the graph can also have an impact on the running time. Density means the ratio of the number of edges divided by the number of nodes. For example, a graph with 1,000 nodes and 3,000 edges has a density of 3, which is quite dense. A graph with 2,000 nodes and 3,000 edges has a density of 1.5, which is sparse. The parameter **OBJECT DISTANCE** determines the minimal distance between nodes and edges. Orthogonal layout also provides special treatment of cliques. A clique is a subgraph in which each node is connected with every other node. Usually, cliques are very hard to recognize in an orthogonal drawing and can slow down the running time drastically. If **EMPHASIZE CLIQUES** is selected, the nodes in a clique will be arranged on a circle and the connections within the clique are drawn as straight-line. This makes it very easy to recognize a clique, and speeds up the computation of the layout. If the special clique handling is turned on, the parameter **MINIMUM CLIQUE SIZE** controls, how many nodes a clique must contain so that it is considered.

10.2 The Cluster-Orthogonal Layout Style

The **Cluster-Orthogonal** layout style is an extension of the Orthogonal layout style to clustered graphs. It layouts the graph in an orthogonal fashion similar to the Orthogonal style, and arranges clusters such that all the descendants of a cluster are contained within the cluster and no two clusters intersect. Cluster-Orthogonal layout is best suited if

- the graph contains clusters.
- orthogonal edge routing is preferred.
- the graph can be drawn with few edge crossings. Note that this might be possible even if a layout style like Symmetric requires many crossings.
- the graph is not too dense and not too large.

Cluster-Orthogonal layout is not applicable if the graph contains self-loops, empty clusters, or if the root cluster contains only a single cluster and no nodes.

When the Cluster-Orthogonal layout style is selected, a dialog box will appear where the settings of the layout style can be adjusted. There is currently only one parameter **OBJECT DISTANCE** which determines the minimal distance between nodes, edges and the borders of clusters.

10.3 The UML-Orthogonal Layout Style

The **UML-Orthogonal** layout style is an extension of the Orthogonal layout style which also considers inheritance trees in UML class diagrams. Edges can be assigned special types like ***association*** and ***generalization*** using the **FORMAT→EDGE SEMANTICS** menu.

Subgraphs formed by generalization edges can be interpreted as inheritance hierarchies in UML class diagram. This layout style guarantees that each inheritance tree is drawn such that the tree edges point in a common direction (e.g. upwards), and all other edges are arranged in order to minimize certain criteria like edge crossings, bends and edge lengths. This enhances the readability of UML class diagrams. UML-Orthogonal layout is best suited if

- the graph consists of hierarchical parts (formed by generalizations) and non-hierarchical parts (formed by the other edges).
- orthogonal edge routing is preferred.
- the graph can be drawn with few edge crossings. Note that this might be possible even if a layout style like Symmetric requires many crossings.
- the graph is not too dense and not too large.

UML-Orthogonal layout is not applicable if the graph contains self-loops.

When the UML-Orthogonal layout style is selected, a dialog box will appear where the following settings can be adjusted. The parameter **CROSSING MINIMIZATION** determines the quality of the crossing minimization procedure. For small or sparse graphs, the algorithm is usually very quick, but for large or dense graphs, the running time will be longer. When laying out a large graph, it is recommended that fast crossing minimization is tried first in order to assess the running time. The density of the graph can also have an impact on the running time. Density means the ratio of the number of edges divided by the number of nodes. For example, a graph with 1,000 nodes and 3,000 edges has a density of 3, which is quite dense. A graph with 2,000 nodes and 3,000 edges has a density of 1.5, which is sparse. The parameter **object distance** determines the minimal distance between nodes and edges. The parameter **preferred direction** refers to the direction in which most of the generalization edges shall point, e.g. **North** means upwards and so on.

10.4 The Hierarchical Layout Style

The **Hierarchical** layout style takes edge directions into account. The direction of edges is only visible if the menu item **Directed** in the **View** menu is checked. In this case, arrowheads are drawn indicating the direction of an edge. Hierarchical layout draws graphs such that nodes are arranged on horizontal levels and edges are drawn as polylines so that most are pointed in a common direction (top to bottom). The Hierarchical layout style shall be applied if

- flow of information shall be displayed.
- the hierarchical structure of the graph shall be visualized.

When the Hierarchical layout style is selected, a dialog box will be opened which allows to adjust the following settings. The parameter **crossing minimization** affects the running time spent for crossing minimization (similar to the crossing minimization parameter of the Orthogonal layout style). Parameter **object distance** determines how far apart nodes on the same level are drawn, **level distance** determines the distance between the levels. If **fixed level distance** is checked, all adjacent levels will have the same distance to each other. Otherwise, this distance is sometimes enlarged to improve the readability of the diagram by avoiding small angles. Hierarchical Layout provides two alternative implementations for the coordinate assignment phase of the algorithm: a fast heuristics and an optimal implementation. This is controlled by the checkbox **optimal coordinate assignment**. The optimal implementation also provides an optional balancing of the layout, which means that the algorithm tries to place each node in the center of its neighbors. The influence of balancing can be adjusted with the **balancing** slider. If balancing is set to none, no balancing at all is applied which also speeds up the running time.

10.5 The UML-Hierarchical Layout Style

The **UML-Hierarchical** layout style is an extension of the Hierarchical layout style which also considers inheritance trees in UML class diagrams. It is an alternative to the UML-Orthogonal layout style which provides some additional alignment options for inheritance hierarchies. However, the UML-Orthogonal layout style usually leads to much more readable drawings and should be the first choice. The UML-Hierarchical layout style shall be applied if

- the graph consists of hierarchical parts (formed by generalizations) and non-hierarchical parts (formed by the other edges).
- all inheritance hierarchies shall point upwards.
- base classes or siblings in inheritance trees shall be aligned.

When the UML-Hierarchical layout style is selected, a dialog box will be opened which allows to adjust the following settings. If **align base classes** is checked, all base classes (nodes with no outgoing but incoming generalization edges) will be aligned at the top of the drawing. If **align siblings** is checked all siblings in an inheritance tree (that is all classes derived from the same class) will be aligned. The parameter **crossing minimization** affects the running time spent for crossing minimization (similar to the crossing minimization parameter of the Orthogonal layout style). Parameter **OBJECT DISTANCE** determines how far apart nodes on the same level are drawn, **LEVEL DISTANCE** determines the distance between the levels. If **FIXED LEVEL DISTANCE** is checked, all adjacent levels will have the same distance to each other. Otherwise, this distance is sometimes enlarged to improve the readability of the diagram by avoiding small angles. UML-Hierarchical Layout provides two alternative implementations for the coordinate assignment phase of the algorithm: a fast heuristics and an optimal implementation. This is controlled by the checkbox **OPTIMAL COORDINATE ASSIGNMENT**. The optimal implementation also provides an optional balancing of the layout, which means that the algorithm tries to place each node in the center of its neighbors. The influence of balancing can be adjusted with the **BALANCING** slider. If balancing is set to none, no balancing at all is applied which also speeds up the running time.

10.6 The Tree Layout Style

The **Tree** layout style can only be applied to a special kind of graphs called *trees* (or more general *forests*). A forest is a graph that contains no undirected cycle and a tree is a connected forest, i.e. a forest consists of one or more trees. The Tree layout style draws rooted trees which give trees a hierarchical structure. One node in the tree is called the root of the tree, and all edges are considered to be directed such that the root is the only source. Typically, trees are drawn with the root on top and all edges directed downwards. The nodes are placed on levels such that nodes with the same distance from the root are on the same level.

When calling the Tree layout style, a dialog is shown offering various options. The distance parameters define the minimal distances between **SIBLINGS**, **SUBTREES**, **LEVELS** and **TREES**. The **ORTHOGONAL** checkbox allows to draw the edges in an orthogonal style (if checked) or straight-line. The drawing direction can be either **TOP TO BOTTOM** (the root is on top, its children below and so on), **BOTTOM TO TOP**, **LEFT TO RIGHT** or **RIGHT TO LEFT**.

The root of the tree (or the root of each tree if the graph is a forest) can be determined using the directions of the edges or the coordinates of the nodes. If **SOURCE** (**SINK**) is selected, the root will be a source (sink) of the graph. If there are several sources (sinks), an arbitrary one is chosen. If **COORDINATES** is selected, the determination of the root depends on the coordinates of the input drawing and the selected drawing direction. If the drawing direction is **TOP TO BOTTOM**, then the root must be moved to the *topmost* position in the drawing before calling the layout style. Similarly, the root must be moved to the *leftmost* position if the direction is **LEFT TO RIGHT** and so on. If the input graph consists of several trees, the root selection as described above is applied to each tree of the forest.

10.7 The Symmetric Layout Style

There are two algorithms for **Symmetric** layout. They are **SYMMETRIC (FR)** and **SYMMETRIC (GEM)**. The aim of the Symmetric algorithms is to achieve short, straight-line edges and equally distribute nodes. Symmetric layout should be applied if

- symmetric structures shall be visualized.
- sparse networks shall be laid out with short straight-line edges.
- a quick overview of the graph structure shall be shown.
- the graph is too complex for applying other layout styles, i.e. Orthogonal.

When one of the two Symmetric layout styles is selected, a dialog box will appear that allows for the adjustment of settings. These settings are similar for both Symmetric algorithms. The parameter **OPTIMIZATION QUALITY** determines how much time is spent for optimization. The better the quality, the more running time is required. The parameter **DESIRED EDGE LENGTH** allows the adjustment of the length of edges.

Both algorithms start their optimization procedure with an initial layout. If the **USE CURRENT LAYOUT** box is checked, the initial layout will be the current layout of your graph and the algorithms can be used for fine-tuning. If this box is not checked, an initial layout is pre-computed and the resulting layout is independent of the current layout of the graph.

10.8 The Circular Layout Style

The **Circular** layout style is suited for graphs containing many cut-vertices. A *cut-vertex* is a node whose deletion cuts the graph into several unconnected pieces. The Circular layout style places each biconnected component (that is a maximal sub-graph that does not contain a cut-vertex) on a circle and arranges all these circles. If the graph is biconnected, the Circular layout style will place all nodes on a circle, which should be avoided. Applying a different layout style in this case should be considered. The Circular layout style should be applied if

- the biconnectivity structure of a graph shall be visualized.
- ring / star networks shall be laid out.

When Circular layout is selected, a dialog box appears allowing the adjustment of a single option **OBJECT DISTANCE** which determines how far apart the nodes are drawn. The shorter the distance, the smaller the area to be occupied by the displayed graph.

11 Printing

GDE allows to adjust print output in various ways. Selecting **Page Setup** from the File menu displays a dialog box for adjusting the print layout of the active graph. The **Page** group shows the current printer settings for the page, i.e. the size of the printable area of the page, and allows to select the orientation of the page (portrait or landscape). The **Scaling for Printing** group determines the scaling of the output. In order to scale to a specific scale factor, select the **scale factor** radio button and type the appropriate scale factor in the combination box to the right. For adjusting the output such that it is a given number of pages wide and tall, select the **adjust** radio button and set the spin boxes to the right to the appropriate values.

The print output is always scaled such that it preserves the aspect ratio. Changing the value of the spin box that determines how many pages wide or tall the output will be, automatically adjusts the value of the other spin box. The actual size of the print output is displayed at the bottom of the **Scaling for Printing** group. The Page Setup dialog is confirmed by either pressing **Ok** or **Print**. **Print** will also display the **Print** dialog, which is convenient if you directly want to print according to the new page settings.

The Print dialog can be invoked by selecting **Print** from the **File** menu, clicking the print icon in the **Standard** tool bar, pressing <CTRL>+P, or selecting **Print** from the document window's context menu. This dialog allows you to select your printer and adjust its settings. Multiple copies can be printed if required, and collation of multiple copies can be enabled or not. Pressing **Ok** will print the active graph according to its page settings.

12 Graph Generators

The menu **Create** contains some graph generators, which can be used to quickly create test graphs of different types. The graph generators adjust the **Show Edge Directions** and **Show Clusters** items in the **View** menu according to the nature of the generated graph. If the generated graph represents a hierarchical structure, edge directions will be shown, otherwise not. Graph generators that do not create clusters turn the **Show Clusters** item of.

12.1 Random Graphs

The **Random Graph** generator creates graphs that can be drawn with few crossings. When selected, a dialog box appears offering four options. The generator creates a biconnected planar graph with **number of nodes** nodes and **number of planar edges** edges. Planar means that the graph can be drawn without edge crossings. Next, randomly chosen edges are deleted. Finally, new edges are added by random. The number of these edges is determined by **number of edges randomly deleted** and **number of edges randomly added**.

To begin, choose the number of planar edges to be about twice as much as the number of nodes, the number of deleted edges to be about half as much as the number of nodes, and set the number of added edges to a small value. The number of edges randomly added has the biggest impact on the number of edge crossings that will be necessary to draw the graph. Use the Orthogonal layout style for graphs created this way. This generator can also demonstrate the superiority of the Orthogonal layout style with respect to crossing minimization compared to all other layout styles.

12.2 Random Cluster Graphs

The **Random Cluster Graph** generator creates a random graph using the generator **Random Graph** and assigns a cluster structure to it. The parameters are the same as discussed for the Random Graphs generator, plus the parameter **maximal number of clusters**, which is the upper limit for the number of created clusters.

12.3 Random UML Graphs

The **Random UML Graph** generator creates random graphs consisting of generalization edges and association edges. These graphs can be seen as UML class diagrams in which the generalization edges form inheritance trees.

The parameter **number of components** refers to the number of connected components the generated graph will have, **hierarchies per component** is the maximal number of inheritance trees for each connected component, **nodes per hierarchy** is the maximal number of nodes (classes) within an inheritance tree, and **max. degree** is the maximal degree (number of incoming generalizations) of a node within an inheritance tree. The last two parameters determine how many associations edges are inserted by giving the minimal and maximal density (number of association edges divided by number of nodes). Useful values range between 0.0 and 1.5.

12.4 Random Hierarchical Graphs

The **Random Hierarchical Graph** generator creates random directed acyclic graphs. The number of nodes and edges can be specified with the first two parameters. **If single source** is checked, there will be a single node with only outgoing edges. If **upward planar** is checked, the created graph can be drawn (in theory) in an upward fashion without any edge crossings. If simple hierarchy is checked, the nodes of the graph can be assigned to horizontal layers, so that each edge connects nodes on neighbored layers and points upwards.

We remark that an upward planar hierarchical graph will not always be laid out without edge crossings using the Hierarchical layout style though this is possible in theory. The reason is the complexity of the crossing minimization problem for hierarchical graphs. However, choosing high quality for crossing minimization should yield drawings with few crossings.

12.5 Random Trees

The Random Tree generator creates trees, that is connected graphs without undirected cycles. The options of the generator allow to set the **number of nodes**, the **maximal degree** of each node (that is the number of children within the tree), and the **maximal level width** which is the maximal number of nodes on each level of the tree. The tree is constructed such that the root of the tree is the unique source of the graph and all edges are directed from parent to children.

12.6 Circular Graphs

The **Circular Graph** generator creates graphs consisting of many biconnected components. These graphs are interesting for testing the Circular layout style. When selected, a dialog box appears allowing for the adjustment of four parameters. The **number of BC's** parameter determines the number of biconnected components the automatically created graph will have. The **maximum number of nodes / BC** is the upper bound for the number of nodes in each biconnected component. The last two parameters determine the lower and upper bound for the density of each biconnected component. Useful values are, e.g., 1.0 as **min. density BC** and 1.4 as **max. density BC**. If **max. density BC** is set to 1.0, then all biconnected components will be simple circles.

13 Customization

13.1 Preferences for Drawing Nodes, Edges and Clusters

The initial graphical appearance of newly created nodes, edges and clusters can be adjusted by selecting **Preferences** from the **Options** menu. The dialog box that will be displayed shows several tab pages: **Node**, **Node Font**, **Edge** and **Cluster**. These tab pages allow to adjust the following initial settings:

- **Node page**: line style, fill and size of a new node.
- **Node Font page**: text font and color of a new node.
- **Edge page**: line style of a new edge.
- **Cluster page**: line style and fill of a new cluster.

The default settings for each page can be restored by pressing the **Default** button. Pressing this button will only change the settings in the currently visible tab page. After confirming the Preferences dialog by pressing **Ok**, the settings will be used for newly created objects, that is objects created by mouse clicks or graph generators. They will not affect objects that have been created before.

13.2 Global Options

GDE provides various optional settings that can be adjusted by the user to fit its particular needs. The default settings provided are useful and sufficient for most users, but an experienced user might want to adjust some or all of these settings to his personal needs or taste. A dialog box for adjusting all options is opened when choosing **Options** from the **Extras** menu. The settings are placed on different topic pages which can be chosen by selecting the respective item in the tree view located at the left side. Some topics also contain sub-topics which appear if the plus sign next to the item is clicked. A description of each option setting is shown by clicking on the question mark in the title bar and then clicking on the respective control box.

All options are applied when the dialog is closed by clicking **Ok**, no settings are applied when the dialog is closed by clicking **Cancel** or closing it with the close button in the title bar. In particular, changing option settings at one topic page and then choosing another topic page will not automatically apply the changes. Only when clicking **Ok** the changes will be applied. On the other hand, the Default button only sets options on the current topic page to their default values, the settings on all other pages remain unchanged.

14 Further Information

14.1 Legal Information

14.2 GDE Community Edition License

This copy of GoVisual Diagram Editor Community Edition ("the Software Product") and accompanying documentation is licensed and not sold. This Software Product is protected by copyright laws and treaties, as well as laws and treaties related to other forms of intellectual property. Center of Advanced European Studies and Research (Stiftung caesar) or its subsidiaries, affiliates, and suppliers (collectively "caesar") own intellectual property rights in the Software Product. The Licensee's ("you" or "your") license to download, use, copy, or change the Software Product is subject to these rights and to all the terms and conditions of this End User License Agreement ("Agreement").

Acceptance

YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT BY INSTALLING THE SOFTWARE PRODUCT OR USING, OR COPYING THE SOFTWARE PRODUCT. YOU MUST AGREE TO ALL OF THE TERMS OF THIS AGREEMENT BEFORE YOU WILL BE ALLOWED TO INSTALL THE SOFTWARE PRODUCT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, YOU MUST NOT INSTALL, USE, OR COPY THE SOFTWARE PRODUCT.

License Grant

This Agreement entitles you to install and use one copy of the Software Product. In addition, you may make any archival copy of the Software Product.

Restrictions on Transfer

Without first obtaining the express written consent of OREAS GmbH, you may not assign your rights and obligations under this Agreement, or redistribute, encumber, sell, rent, lease, sublicense, or otherwise transfer your rights to the Software Product. Restrictions on UseYou may not decompile, "reverse-engineer", disassemble, or otherwise attempt to derive the source code for the Software Product.

Restrictions on Alteration

You may not modify the Software Product or create any derivative work of the Software Product or its accompanying documentation. Derivative works include but are not limited to translations. You may not alter any files or libraries in any portion of the Software Product.

No Warranty

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NOWARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. UNLESS OTHERWISE EXPLICITLY AGREED TO IN WRITING BY OREAS GMBH OR CAESAR, OREAS GMBH OR CAESAR PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, IN FACT OR IN LAW, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

oreas GmbH or caesar make no warranty that the Software Product will meet your requirements or operate under your specific conditions of use. oreas GmbH or caesar make no warranty that operation of the Software Product will be secure, error free, or free from interruption.

YOU MUST DETERMINE WHETHER THE SOFTWARE PRODUCT SUFFICIENTLY MEETS YOUR REQUIREMENTS FOR SECURITY AND UNINTERRUPTABILITY. YOU BEAR SOLE RESPONSIBILITY AND ALL LIABILITY FOR ANY LOSS INCURRED DUE TO FAILURE OF THE SOFTWARE PRODUCT TO MEET YOUR REQUIREMENTS. OREAS GMBH OR CAESAR WILL NOT, UNDER ANY CIRCUMSTANCES, BE RESPONSIBLE OR LIABLE FOR THE LOSS OF DATA ON ANY COMPUTER OR INFORMATION STORAGE DEVICE. UNDER NO CIRCUMSTANCES SHALL OREAS GMBH, ITS DIRECTORS, OFFICERS, EMPLOYEES OR AGENTS, OR CAESAR, ITS DIRECTORS, OFFICERS, EMPLOYEES OR AGENTS BE LIABLE TO YOU OR ANY OTHER PARTY FOR INDIRECT, CONSEQUENTIAL, SPECIAL, INCIDENTAL, PUNITIVE, OR EXEMPLARY DAMAGES OF ANY KIND (INCLUDING LOST REVENUES OR PROFITS OR LOSS OF BUSINESS OR LOST DATA, COST OF SUSTITUTE GOODS, TECHNOLOGY OR SERVICES) RESULTING FROM THIS AGREEMENT, OR FROM THE FURNISHING, PERFORMANCE, INSTALLATION, OR USE OF THE SOFTWARE PRODUCT, WHETHER DUE TO A BREACH OF CONTRACT, BREACH OF WARRANTY, OR THE NEGLIGENCE OF OREAS GMBH OR CAESAR OR ANY OTHER PARTY, EVEN IF OREAS GMBH OR CAESAR ARE ADVISED BEFOREHAND OF THE POSSIBILITY OF SUCH DAMAGES. TO THE EXTENT THAT THE APPLICABLE JURISDICTION LIMITS OREAS GMBH'S OR CAESARS ABILITY TO DISCLAIM ANY IMPLIED WARRANTIES, THIS DISCLAIMER SHALL BE EFFECTIVE TO THE MAXIMUM EXTENT PERMITTED.

Governing Law and Jurisdiction

This Agreement is governed and construed in accordance with the laws of the Federal Republic of Germany. Place of jurisdiction is Köln, Germany.

Severability

If any provision of this Agreement shall be held to be invalid or unenforceable, the remainder of this Agreement shall remain in full force and effect. To the extent any express or implied restrictions are not permitted by applicable laws, these express or implied restrictions shall remain in force and effect to the maximum extent permitted by such applicable laws.

14.3 Contact Information

For further information about GoVisual Diagram Editor, please contact

OREAS GmbH
Kirschgarten 4
50999 Köln, Germany

phone: +49 (228) 96 56 - 251
email: gde@oreas.com

Information about GoVisual software products are also available at http://www.oreas.com.

Please send bug reports, problems and feature requests to gde@oreas.com. If you report a bug, please make sure that you also add the following information: Platform (which operating system?), GDE version, detailed description of the bug, example data for recreating the bug (if necessary).

14.4 System Requirements

The following table lists the minimal and recommended system requirements:

	minimal	*recommended*
Processor:	300 MHz	2.4 GHz
Operating System Platforms:	Windows® 98, Me, NT 4.0, 2000 or XP	Windows® XP
	Linux x86	
Memory:	128 MB RAM	256 MB RAM
Hard Drive:	10 MB available hard disk space	
Display:	Super VGA (800×600) with 256 colors	XVGA (1024×768) with 65,536 colors
Other Peripherals:	Mouse	

Appendix A: Action Table

The following table lists the actions that can be performed with the mouse in a document window in detail. The behavior depends on the currently active tool (Select, Create, Zoom, Scroll) and the type of graphical object (node, edge, bend point, cluster border, screen background) under the mouse cursor. The operations shown in gray are mirrored actions provided for convenience.

	object	*Select*	*Create*	*Zoom*	*Scroll*
Click	Node	Select Node	Start Create Edge	Zoom In	Zoom In
	Edge	Select Edge	Create Node	Zoom In	Zoom In
	Bend	Select Edge	Create Node	Zoom In	Zoom In
	Cluster	Select Cluster	Create Node	Zoom In	Zoom In
	Screen	Deselect All	Create Node	Zoom In	Zoom In
Click <SHIFT>	Node	Toggle Select Node	Start Create Edge	Zoom In	Zoom In
	Edge	Toggle Select Edge	Create Node	Zoom In	Zoom In
	Bend	Toggle Select Edge	Create Node	Zoom In	Zoom In
	Cluster	Toggle Select Cluster	Create Node	Zoom In	Zoom In
	Screen		Create Node	Zoom In	Zoom In
Right Click	Node	Context Node	Context Node	Zoom Out	Zoom Out
	Edge	Context Edge	Context Edge	Zoom Out	Zoom Out
	Bend	Context Bend	Context Bend	Zoom Out	Zoom Out
	Cluster	Context Cluster	Context Cluster	Zoom Out	Zoom Out
	Screen	Context Screen	Context Screen	Zoom Out	Zoom Out
Click & Drag	Node	Move Node + Select Node	Move Node + Select Node	Zoom To Area	Scroll
	Selected Node	Move Selection	Move Selection	Zoom To Area	Scroll
	Edge	Move Edge + Select Edge	Insert Bend	Zoom To Area	Scroll
	Selected Edge	Move Selection	Insert Bend	Zoom To Area	Scroll
	Bend	Move Bend	Move Bend	Zoom To Area	Scroll
	Cluster	Move Cluster + Select Cluster	Move Cluster + Select Cluster	Zoom To Area	Scroll
	Selected Cluster	Move Selection	Move Selection	Zoom To Area	Scroll
	Selection Marker	Resize Selection	(not recognized)	Zoom To Area	Scroll
	Screen	Select Area	Select Area	Zoom To Area	Scroll

	object	*Select*	*Create*	*Zoom*	*Scroll*
Click & Drag <SHIFT>	Node	Toggle Select Node	Toggle Select Node	Zoom To Area	Scroll
	Edge	Toggle Select Edge	Toggle Select Edge	Zoom To Area	Scroll
	Bend	Toggle Select Edge	Toggle Select Edge	Zoom To Area	Scroll
	Cluster	Toggle Select Cluster	Toggle Select Cluster	Zoom To Area	Scroll
	Screen	Toggle Select Area	Toggle Select Area	Zoom To Area	Scroll
Right Click & Drag	Node	Context Node	Context Node	Zoom Out	Zoom Out
	Edge	Context Edge	Context Edge	Zoom Out	Zoom Out
	Bend	Context Bend	Context Bend	Zoom Out	Zoom Out
	Cluster	Context Cluster	Context Cluster	Zoom Out	Zoom Out
	Screen	Context Screen	Context Screen	Zoom Out	Zoom Out

Appendix B: Accelerators

The following table lists all accelerators and their associated operations in alphabetical order.

accelerator	*operation*	*accelerator*	*operation*
<CTRL>+A	Select All	**+**	Zoom In
<CTRL>+C	Copy	**–**	Zoom Out
<CTRL>+E	Export	**<SPACE>**	Switch Select Tool
<CTRL>+I	Import	**<DEL>**	Delete
<CTRL>+N	New	**<F1>**	User Guide
<CTRL>+O	Open	**<SHIFT>+<F1>**	Context Help
<CTRL>+P	Print	**<F2>**	Create Tool
<CTRL>+S	Save	**<F5>**	Zoom To Graph
<CTRL>+V	Paste	**<SHIFT>+<F5>**	Zoom To Selection
<CTRL>+X	Cut	**<F6>**	Centre Graph
<CTRL>+Y	Redo	**<SHIFT>+<F6>**	Center Selection
<CTRL>+Z	Undo	**<F11>**	Zoom Tool
		<F12>	Panning Tool
<ALT>+o	Cluster Explorer		

Appendix C: Terminology

This appendix defines and explains some commonly used graph-theoretical terms.

Biconnected Components: A *cut-vertex* in a connected graph *G*=(*V*,*E*) is a node *v* in *V* such that deleting *v* in *G* results in a non-connected graph. A connected graph *G* is called *biconnected* if *G* does not contain a cut-vertex. A *biconnected component* of a graph *G* is a maximal biconnected subgraph of *G*.

Clique: A *clique* is a subgraph *C* in which each node in *C* is connected with every other node in *C*. The size of a clique *C* is the number of nodes in *C*.

Cluster: see **Clustered Graph**

Cluster Tree: see **Clustered Graph**

Clustered Graph: Clustered graphs are graphs with a recursive clustering structure over the nodes. A *cluster graph* *C*=(*G*,*T*) consists of a graph *G* and a rooted tree *T* such that the leaves of *T* are exactly the nodes of *G*. Each node ν of *T* represents a cluster *V*(ν) of the nodes of *G* that are leaves of the subtree rooted at ν. The tree *T* describes an inclusion relation between clusters and is called the *inclusion* (or *cluster*) *tree* of *C*.

The inclusion tree of a cluster graph is visualized in the tree view explorer of GDE. The purpose of a cluster graph is to group nodes. Since it is also possible to group clusters, clustered graphs can also represent nested groupings.

Connected Components: A graph *G*=(*V*,*E*) is called *connected*, if there is an undirected path from *u* to *v* for any pair *u*, *v* of nodes in *V*. A *connected component* of *G* is a maximal connected subgraph of *G*. In especially, for two nodes *x*, *y* in two different connected components, there is no unconnected path in *G* leading from *x* to *y*.

Cut-Vertex: A *cut-vertex* is a node in a graph *G=(V,E)* such that the deletion of *v* in *G* increases the number of connected components in *G*. E.g., if *G* is a connected graph and *v* a cut-vertex in *G*, then the graph resulting from deleting *v* in *G* is disconnected.

Density: The *density* of a graph *G=(V,E)* is defined as the number of edges *E* divided by the number of nodes *V*.

Edge: see **Graph**

Graph: A *graph G=(V,E)* consists of a set *V* of *nodes* and a multi-set *E* of *edges*, such that each edge is a pair of nodes. Since E is a multi-set, it is allowed that an edge *e=(v,w)* appears several times in a graph. In a this case, we say *e* is a multiple edge. It is also allowed that an edge connects a node with itself and such an edge is called a self-loop.

In a diagram, nodes are drawn as two-dimensional objects, e.g. rectangles or ellipses, and edges as one-dimensional lines, e.g. straight-lines or polylines. The line representing an edge *e=(v,w)* starts at the object representing *v* and ends at the object representing *w*.

Multiple Edge: A multiple edge *e*=(*u*,*v*) in a graph *G*=(*V*,*E*) is an edge that appears several times in the multi-set *E*. If *G* is considered as undirected graph, *e* is called a multiple edge if there is another occurrence of (*u*,*v*) or any occurrence of (*v*,*u*) in *E*.

Node: see **Graph**

Planar: A drawing of a Graph *G* is called planar if there are no edge crossings (and overlapping or touching nodes and edges) in the drawing. A graph *G* is called planar if there exists a planar drawing of *G*.

Root: see **Tree**

Rooted Tree: see **Tree**

Self-Loop: A *self-loop* is an edge *e=(v,v)* connecting a node *v* with itself.

Subgraph: A *subgraph S* of a graph *G*=(*V*,*E*) is a graph *S*=(*V'*,*E'*) such that *V'* is a subset of *V* and *E'* is a subset of *E*. Hence a subgraph consists of some of the nodes and edges of *G* and does not contain any nodes or edges that are not contained in *G*.

Subtree: : see **Tree**

Tree: A *tree T* is a connected graph *T*=(*V*,*E*) that does not contain any undirected cycle or multiple edge. An *undirected cycle* is an undirected path that connects a node with itself. Hence, a tree with *n* nodes always has exactly *n*-1 edges. A connected subgraph in a tree is always a tree and is called a *subtree*.

A *rooted tree* is a tree in which one node is called the root. A tree can be rooted at any node in the tree. The edges in a tree can be directed such that each node in T except for the root has exactly one incoming edge, and the root has no incoming edge. In this case, all nodes with no outgoing edges are called *leaves*, and the other nodes are called *internal nodes*. If *v* is a node in *T*, the *subtree rooted* at *v* refers to the rooted tree consisting of all nodes that can be reached from *v* by traversing a directed path and has root *v*.