Using Grafix

Ulrich K. Deiters

Institute of Physical Chemistry, University of Cologne,
Köln, Germany

Grafix is a simple, yet very efficient and highly portable tool for creating diagrams. This text describes its usage; the installation is explained in the INSTALL file.

1 Starting

This is the simple part. Make sure that you have $GRXHOME set to the Grafix installation directory and exported, e.g.,

```
GRXHOME=/home/ukd/grafix; export GRXHOME
```

Then just enter

```
$GRXHOME/grafix
```

The script gives you the choice of creating a new diagram from scratch, creating a new diagram as a copy of an existing one, or editing an existing diagram.

When you have answered, two windows will appear — XEmacs and a PostScript viewer (e.g., gv or GhostView, depending on your installation). XEmacs is needed for editing the diagram, the PostScript viewer for viewing the results. The XEmacs window will have (at least) two subwindows, diagram.xml and global.xml, each one belonging to one of the components of a Grafix document. You can activate one for editing just by clicking on its tab (if your XEmacs does not support tabs, you can set it to a split-window display).

2 The concept of Grafix

Before we continue, a few words about the structure of a diagram document are in order: A diagram is represented by a directory containing (usually) three files:

- diagram.xml, which contains your data and most of the formatting instructions,
- global.xml, which contains global settings of style parameters,
- diagram.ps, which is your output file.

XML, like HTML, is a so-called mark-up language. Its purpose is to declare the contents of a document to the computer; the declaration is made by placing ”tags” around the contents, e.g.,

<italic>some text</italic>

tells the computer that ”some text” should be printed in italics. Of course, this tag is permissible for texts only; it would not make sense to apply it to a curve or a line. Fortunately, XML has a built-in guidance system: Because the computer “understands” the document contents, it will tell you which tags are meaningful at a given location and which not.

Example: Place the cursor behind the <cartesian> tag of diagram.xml and depress the right mouse button: A menu will show that only an x axis tag or a comment may be placed here.

Some tags have options which control how the contents are treated. These options can be global — then they apply to all tags of the same kind — or local — i.e., they apply to one tag instance only.

Example: Place your cursor on the <xaxis> tag and depress the right mouse button: A menu will show all options for an x axis. If you do this in diagram.xml, you see the *locally* settable options. If you do this in global.xml, you see the *globally* settable options.

You can easily “import” a diagram style (size, fonts used, line widths, etc.) by copying global.xml from one diagram to another. This may be useful if you have to deal with a large number of similar diagrams.

3 Output

Select ”Grafix: translate” in the File menu as often as you wish to see the results of the changes you made. Grafix will then save your XML code, translate it to PostScript and display the result.

The output format is EPS, Encapsulated PostScript. It is real outline PostScript, not the pixel raster format that a certain major unenlightened company tries to substitute for the real thiell. Grafix diagram files are remarkably small, yet you can scale diagrams to arbitrary sizes without loosing accuracy.

4 The tags

4.1 <cartesian> — Grafix coordinate systems

This is the outermost tag of a diagram. It has no local options. The global options are:

- diagr-llx, diagr-lly, diagr-urx, diagr-ury: the coordinates of the lower left and upper right corner of the diagram frame

  These coordinates are measured in *absolute* units (cm, mm, in, pt) from the lower left corner of the page (see Fig. 1). Enter data in TEX style (e.g., 25mm) without spaces between numbers and units!

- graph-llx, graph-lly, graph-urx, graph-ury: the coordinates of the lower left and upper right corner of the graph area (the rectangle that will contain the graph)

  These coordinates are measured in *relative* units; the width of the diagram frame (diagr-urx – diagr-llx) corresponds to 1.00 .

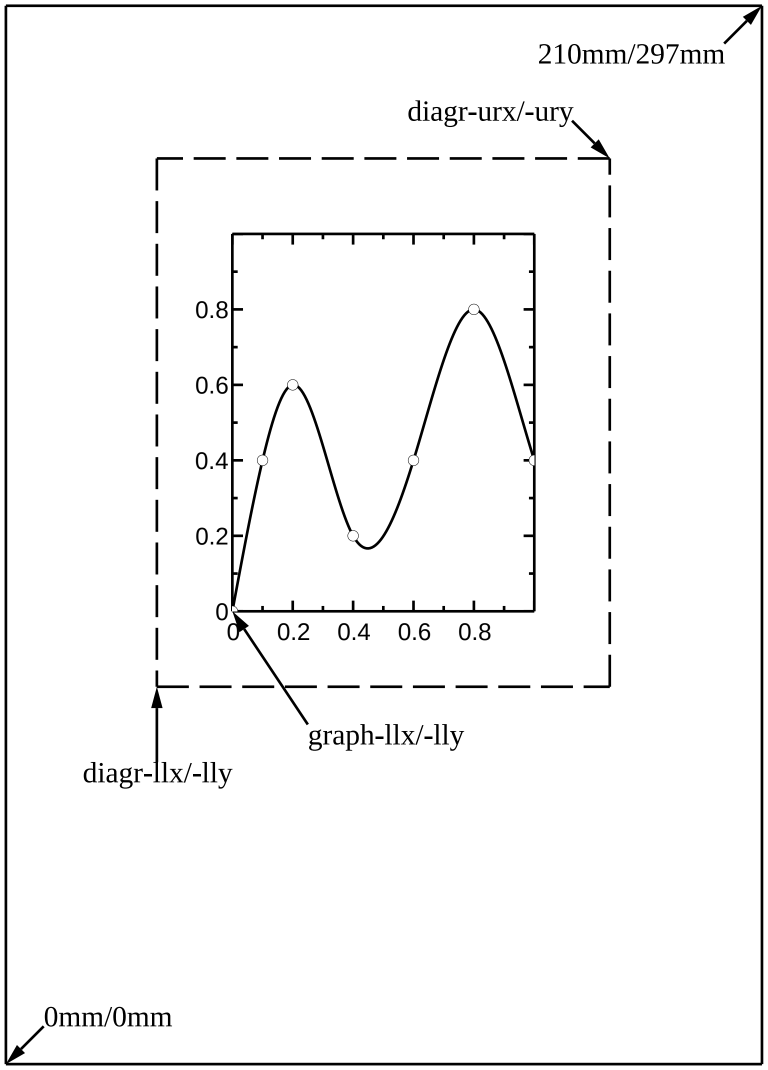

Figure 1: The drawing area of Grafix and its coordinates. The outermost frame represents a sheet of paper, the dashed frame the diagram area, and the innermost frame the graph area.

- scale-llx, scale-lly, scale-urx, scale-ury: the scaling parameters

  These are the coordinate values you assign to the corners of the graph area. Enter them here without units.

4.2 <area>

With this tag it is possible to create a coloured background or quite arbitrary coloured objects. Grafix will interpret the enclosed coordinates as vertices of a polygon. If desired, a (parametric) spline or Akima interpolation is performed on the polygon. Then it is plotted and filled with the desired colour. As the fill colour is opaque, the area tag should be placed before all other tags.

The line width, pattern and line colour options are the same as for the line tag.

4.3 <arrow>

Use this tag to plot an arrow. The coordinates of the arrow point and the tail are required local parameters; you will be prompted for values when you create an arrow tag. The other options can be set locally or globally.

The coordinates can be specified with any unit; use mm, cm, in, or pt for absolute positioning, no unit for relative positioning within the diagram frame, or xu, yu to indicate the user-defined units of the x and y axis.

The line pattern and line width options (for the arrow shaft) are the same as for the line tag.

4.4 <curve>

This tag is used to plot data into the graph area. The data points can be joined by straight lines, by smooth interpolation curves, or not at all.

The data — coordinate pairs — must be placed between the two curve tags. You can do this manually, import a file via the File/Insert File menu, or extract two columns of a table via the File/Grafix: read table menu.

By setting the option “error” to “dy” it is possible to plot vertical error bars. In this case data triples must be provided for each point, $x, y, \Delta x$.

Interpolation functions are either *natural cubic splines*, Akima polynomials, or user-defined functions (see below); in the case of splines and Akima polynomials you furthermore have the option of interpolating $y(x)$ or $x(y)$, or to perform a parametric spline, $x(t), y(t)$. Note that the x values have to be in ascending order for an interpolation along x. Likewise, the “y” interpolation mode requires that the y values are in ascending order

For an explanation of marker symbols and line styles see the sections about the corresponding tags below.

The options “xmathops” and “ymathops” define mathematical operations to be performed on the data. Specify your wishes in RPN (Reverse Polish Notation), with a comma as separator between operators, and NO spaces in the option, e.g.,

″x,2.5,+″	increase all values by 2.5
″x,1.8,*,32.0,+″	convert from °C to °F
″x,2,\ˆ,neg,exp″	$\exp(-x^2)$
″x,o,*″	multiply with the “other” coordinate

Note that “x” is the value to be transformed, and “p0”, “p1”, . . . ″p19″ are parameters set with the <parameters> tag.

It is possible to do some parameter fitting by specifying the indices of the adjustable parameters in the ″fit″ option (comma-separated list, e.g., “0,1” means “optimize p0 and p1”). The fitting routines use the parameter set stored with the <parameters> tag as initial values and overwrite them. The parameters are also written to the buffer *Shell Command Output*.

4.5 <line>

This tag creates a straight line. The endpoints of the line are required local parameters. You will be prompted for them when you select this tag. All units are allowed.

The “linewidth” option controls the width of the line; the line “pattern” option is self-explaining once you know that the menu shows the repeat pattern, and that “d” stands for a dot, “s” for short dash, and “l” for a long dash. E.g., the pattern “ld” creates a dot-dashed line.

4.6 <marker>

Use this tag to place a single marker symbol into the diagram. The coordinates of the symbol are required local parameters; you will be prompted for them. The other parameters can be set globally or locally.

The tag menu lists the available symbols. Some of them can appear as open or as filled symbols; this is controlled by the "version" option.

4.7 <parameters>

This tag pair lets you enter parameters for function plotting or curve fitting. It has no user-settable options. Enter the parameters between the tags; they can then be referred to as "p0", "p1", . . . "p19" in the xmathops, ymathops, and function arguments of <curve>.

4.8 <text>

This tag pair lets you plot text into the diagram. The coordinates of the text anchor point are required local parameters, the other parameters can be set locally or globally.

The text that you want to plot must appear between the tags. Note that this text may contain further tags for changing the font or setting subscripts or superscripts. Furthermore, you can use the Symbols menu for generating a number of symbols outside the ASCII range. Please refer to Section 5 for some hints and caveats.

The "xalign" and "yalign" options control how the text is placed with respect to the specified coordinates (the anchor point). See Fig. 2 for a demonstration.

4.9 <xaxis>, <yaxis>

These two tags generate the axes of the coordinate system. The first option specifies the axis location, e.g., "y = ymin" in <xaxis> generates an x axis along the lower border of the graph area, and "y = 0.0" generates one along the zero of the user-defined coordinate system.

The axes will be plotted with little tic marks. The "minorincr" option controls their distance (in user-defines units); "majorincr" gives the distance of the major tic marks, which will be labelled with coordinate values. Both options can be set to "auto", in which case Grafix will provide some estimates.

The coordinate values can be suppressed in a single instance (with the "omit" option) or everywhere (by selecting "numbering = none").

4.10 <bold>, <bold-italic>, <italic>, <roman>, <symbol>

These tags can appear only inside text and let you change the font.

4.11 <sub>, <super>

These tags can appear only inside text. Their contents will be printed as subscripts or superscripts.

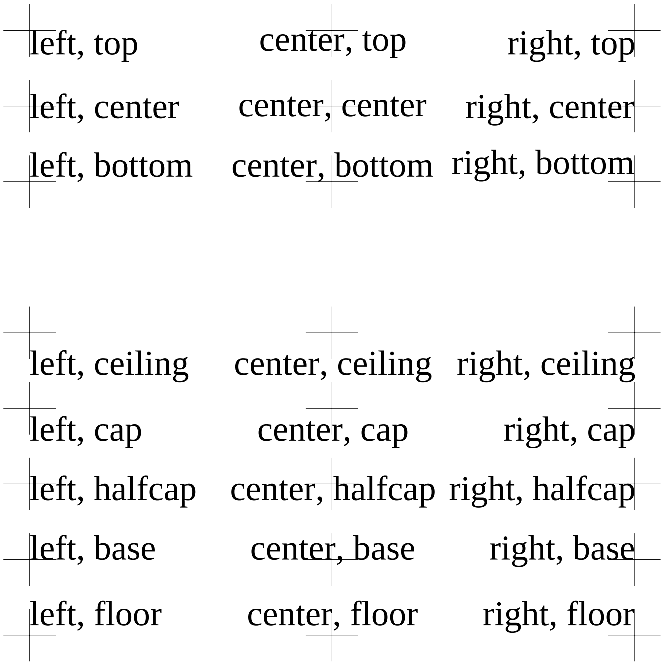

Figure 2: Text alignment options. The y-alignment options *top*, *center* and *bottom* derive their displacements from the actual text outlines, the other options from properties of the font.

5 Useful hints

5.1 Which coordinates are best?

For arrows, lines, markers, and texts the placement can be specified using absolute coordinates, relative coordinates, or user-defined x- or y-coordinates. All choices lead to the same diagram, so why make so many distinctions?

The difference appear if you, at a later time, change the options of the diagram tag. For example, you might want to shift the diagram to the center of the page, or to change the scaling parameters.

1. Objects specified with absolute coordinates will maintain their position on the page. If you move the diagram, they may even slide off the diagram area! This is usually not want you want, but there are exceptions: folding marks, reference marks for reproduction, or filing labels (“Fig. X of manuscript Y”) can be placed conveniently with absolute coordinates.
2. Objects specified with user-defined coordinates will move when the scaling parameters are changed. This is sometimes useful, e.g., and arrow pointing at a data point will continue to point there.
3. On the other hand, axis labels should remain attached to their axis, irrespective of the scaling parameters. Such objects are best specified with relative units.

5.2 How can I find the coordinates of a location in the diagram?

Ghostview displays two coordinates whenever the pointer is over the diagram. These coordinates are absolute ones, measured in PostScript points (pt).

You can invoke a “translation program” via the File/Show Location menu. If you enter the Ghostview coordinates, it reports the location in all other coordinate systems that Grafix uses.

5.3 Does the sequence of objects matter?

Eventually yes: The objects are plotted in the sequence you specify. If a curve runs over a marker sysmbol, the sequence of the tags decides whether the curve is visible across the marker or blotted out by the marker.

5.4 How to print special symbols?

We use the term “special symbol” for everything that is not a letter, a number, or a common interpunction mark.

1. First, a warning: Do not place <, >, or & into texts! These characters have a special meaning to XML. If you need to print them, use the XML transscription available via the Symbols menu!
2. Some characters with diacritical marks are treated correctly by Grafix: vowels with a dieresis (umlauts), vowels with an acute accent, å, ç, ñ. You can type them on your keyboard as usual. On saving they will be replaced by XML transscriptions (e.g., ä→&uml;). You can use the Modify/Decode Character Entities menu entry to switch back to the normal form.
3. Some characters usually not found on an ISO-Latin1 keyboard are accessible via the Symbols menu. For some of them we have to switch to the PostScript symbol font, e.g., for Greek characters.