

# **HPGLVIEW**

Viewer for HP-GL files

Version 5.40

## **User's Guide**

T. Hakulinen  
A. Matveev  
S. Mankov

CERN – Department TS  
September, 2005

# 1 General

The **CERN HP-GL** viewer is designed to display legal **HP-GL** and **HP-GL/2** files on screen and to allow printing them and converting them to other formats. The primary design criteria have been correctness, speed, and portability to various platforms. Versions of the viewer run on Microsoft **Windows** versions **95** and newer, Apple **MacOS 8** and newer, and on various **Unix** variants including IBM **AIX**, Compaq/Digital **OSF/1**, Hewlett-Packard **HP-UX**, SGI **IRIX**, **Linux**, Sun **Solaris**, and Apple **Darwin (MacOS X)**. The current version of the viewer understands most of the old **HP-GL** and the current **HP-GL/2** standards with a few exceptions (for details see the ReadMe file in the viewer distribution).

The newest version and documentation as well as other information on the viewer will always be available on the **CERN HP-GL** viewer web site at

<http://cern.ch/hpglview/hpglviewer.html>

Most examples and illustrations in this user's guide are based on the **Windows** version of the viewer but almost all functionality and interface details are similar on **Unix** and **MacOS**. When there are differences between systems, they are clearly indicated. Also, in this document, term **MacOS** used alone always refers to the classic OS version, i.e., **MacOS 8** or **9**.

## 2 Software Requirements and Installation

Installation of the viewer is quite straightforward even though the exact procedures for the different versions on **Windows**, **MacOS** and **Unix** differ slightly.

### 2.1 PC (Windows)

To run the viewer on **Windows**, only the files `hpglview.exe` and `hpglview.dat` are required. There are no special hardware requirements but, as usual, the more memory and CPU power the better.

### 2.2 Unix/Motif

For **Unix** workstations files `hpglview` and `hpglview.dat` have to be copied to a convenient place. On some systems **Motif** dynamic libraries must also be available. The **DISPLAY** environment variable must be set for the display by the appropriate **Unix** command, e.g., in C-shell:

```
setenv DISPLAY myhost:0.0
```

The above command would set the display to be the local display of the computer **myhost**. See the X-window System documentation for more details. If the **DISPLAY** variable is not set or is incorrect, you will get an error message, e.g.:

```
Can't open display!
```

and the program exits.

Environment variable **HPGLVIEW** can also be defined, which should indicate the path pointing to the `hpglview.dat` file. This is useful when you want to place the viewer in a directory in the search path or to use a link to the viewer so that it can be started without always typing the full path.

## 2.3 Classic MacOS (MacOS 8 and 9)

The classic Apple **MacOS** version comes with two binaries, `HPGLview M68K` optimized for the **68000** series and another one, `HPGLview MPPC` for the **PowerPC (PPC)** series of computers. The binary for the **68000** will run on a **PPC** machine as well, albeit more slowly. Copy the correct binary as well as the file `hpglview.dat` into the folder of your choice. The viewer should run on any version of **MacOS 8** and **MacOS 9**. This version can also be run in the **MacOS 9** compatibility mode of **MacOS X**.

## 2.4 MacOS X

The Apple **MacOS X / Darwin** version is based on the generic **Unix/Motif** version. Therefore, to run the binary, X11 support must be installed on the machine (if necessary, see Apple documentation on how to install X11 on the machine).

# 3 Starting the Viewer

You can start the viewer with or without the name of an **HP-GL** file as a parameter. If a file name is given, the file will be displayed and the name of the file will be the title of the viewer window in which the file is displayed. If a file name is omitted at startup, the viewer window will be empty except that the licence information about the viewer is displayed in it, and the window title will be **Designer viewer V5.40**.

It is also possible to start the viewer in batch mode. The following command line options are available:

```
hpglview [infile[,cfg_file[,low_x[,low_y[,size_x[,size_y[,rotate[,scale[,page]]]]]]]]]
```

Open *infile* in a window with the size defined in millimeters with parameters optionally read from *cfg\_file*.

```
hpglview -hpgl infile[,cfg_file[,page]] outfile[,rotate[,scale[,pagesize[,pagealign]]]]
```

Convert *infile* to **HP-GL** and write to *outfile*. On **Windows**, if *outfile* begins with literal string `printer:`, the rest of the name is interpreted as a **Windows** printer name, where to print the file.

```
hpglview -hpgl2 infile[,cfg_file[,page]] outfile[,rotate[,scale[,pagesize[,pagealign]]]]
```

Convert to **HP-GL/2**. On **Windows**, if *outfile* begins with literal string `printer:`, the rest of the name is interpreted as a **Windows** printer name, where to print the file.

```
hpglview -eps infile[,cfg_file[,page]] outfile[,rotate[,scale]]
```

Convert to **Encapsulated PostScript**.

```
hpglview -ps      infile[,cfg_file[,page]] outfile[,rotate[,scale[,pagesize[,pagealign]]]]
```

Convert to **PostScript**. On **Windows**, if *outfile* begins with literal string `printer:`, the rest of the name is interpreted as a **Windows** printer name, where to print the file.

```
hpglview -dxf     infile[,cfg_file[,page]] outfile[,rotate[,scale]]
```

Convert to **DXF**.

```
hpglview -cgm     infile[,cfg_file[,page]] outfile[,rotate[,scale]]
```

Convert to **CGM**.

```
hpglview -wdrv    infile[,cfg_file[,page]] outfile:printer[,rotate[,scale[,fitmode]]]
```

Only on **Windows**. Output *infile* to *outfile* or printer using the specified **Windows** printer driver with queue name *printer*. If *outfile* is the literal string `printer`, output goes to the printer, otherwise to file *outfile*. If *fitmode* is 1, the page will be printed in absolute scale (1=1), otherwise it will be scaled to fit on page.

```
hpglview -link    infile1[,cfg_file[,page]] infile2[,dx[,dy[,scale]]] ... outfile[,format]
```

Link input files *infile2*, etc. each individually scaled and shifted by *dx,dy* millimeters to *infile1* and save to *outfile* in the selected format (*format* can be `hpgl`, `hpgl2`, `eps`, `ps`, `dxf`, or `cgm`).

```
hpglview -clip    low_x[,low_y[,top_x[,top_y]]] infile[,cfg_file[,page]] outfile[,format]
```

Clip the window at coordinates *low\_x,low\_y,top\_x,top\_y* in millimeters from the bottom left corner of the drawing and save it to *outfile* in the selected format.

```
hpglview -text    infile[,cfg_file[,page]] textfile[,dx[,dy[,orig[,adj]]]] outfile[,format]
```

Add *textfile* shifted by *dx,dy* millimeters from the corner *orig* (0 = bottom left, 1 = top left, 2 = top right, 3 = bottom right) with adjustment *adj* (1 = left, c = center, r = right) to *input* with parameters from *cfg\_file* and save to *outfile* in the selected format. If *textfile* cannot be found, the string itself is added verbatim instead.

```
hpglview -size    infile[,page]
```

Output the drawing size only.

```
hpglview -paper   infile[,page]
```

Output the closest fitting standard paper sizes (ISO / ANSI).

The common optional parameters are:

*page* is the page to process (from 1 to the total number of pages).

*rotate* can be 0, 90, 180, or 270 degrees.

*scale* can be any number greater than 0.

*pagesize* can be A0 to A4, B0 to B4, ANSIA to ANSI, ARCHA to ARCHE, or AUTO meaning auto-scaling to fit the page (only works for **PostScript** output), or NAT meaning that the natural dimensions of the drawing are to be preserved, or WxH where W and H are the page width and height in mm (if scale is also given, it will be ignored).

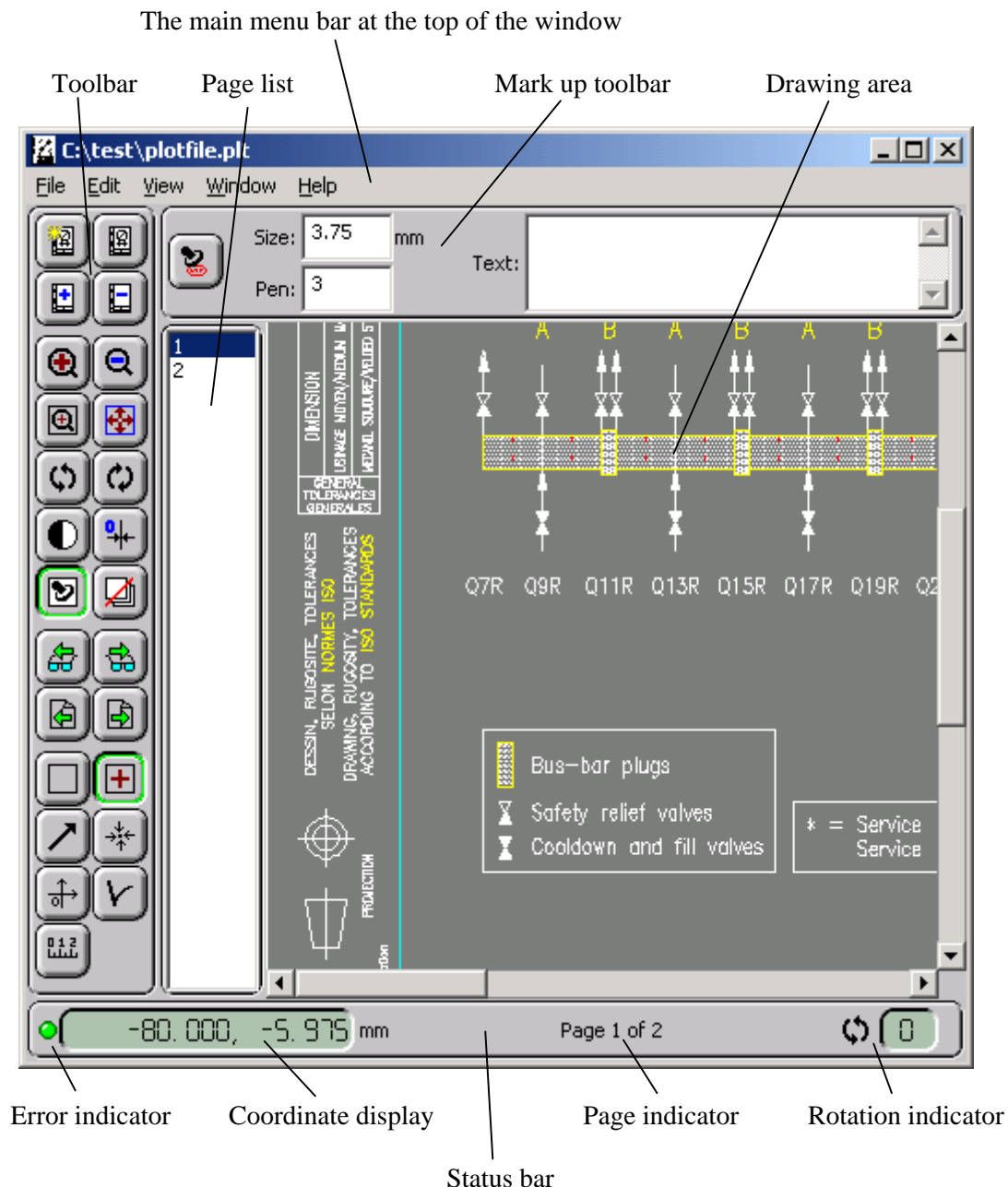
*pagealign* can be TL (top-left), TC (top-center), TR (top-right), CL (center-left), CC (center-center), CR (center-right), BL (bottom-left), BC (bottom-center), or BR (bottom-right).

Brackets `[ ]' mean that a parameter is optional. Not specifying a parameter between commas substitutes a default value for that parameter.

## 4 The Main Viewer Window

The main window of the **HP-GL** viewer consists of various graphical widgets for different functions. When the viewer is started, the main window is opened and if a plot file name was given as a parameter (on **Windows** or **Unix**), the drawing is opened in the window. Several windows can be open simultaneously displaying different drawings or parts of the same drawing (see *Lens window* menu item below).

Below is a schematic of the main window of the viewer:



**Main menu bar**

The principal menu tree where all the main functions of the viewer can be found.

**Toolbar**

A set of graphical buttons replicating the most commonly used functions of the viewer.

**Drawing area**

The actual drawing window. If only a part of the drawing is visible, horizontal and vertical scroll bars can be used to move the view around the drawing.

**Status bar**

Contains several status indicators showing the status of the drawing:

*Error indicator*

If there were no errors in reading or interpreting the drawing, the color of the error indicator is green. If errors were found, the color of the error indicator is red and the corresponding error messages are printed in the log window (see *Log File* menu item below).

*Coordinate display*

Shows the current position of the mouse cursor in the drawing using the selected coordinate units. The coordinate origin is by default the same as in the drawing coordinate system, but it can be changed if required (see *Origin set* mouse mode below).

*Page indicator*

Shows the ordinal number of the current page and the total number of pages in the plot file.

*Rotation indicator*

Indicates the rotation angle of the drawing. The initial orientation of any drawing is always such that the origin is in the lower left hand corner,  $x$ -axis horizontal increasing to the right,  $y$ -axis vertical increasing upwards. This corresponds to the rotation angle of 0 degrees. The drawing can be rotated in increments of 90 degrees, with the angle of 360 degrees corresponding to the initial rotation.

**Page list**

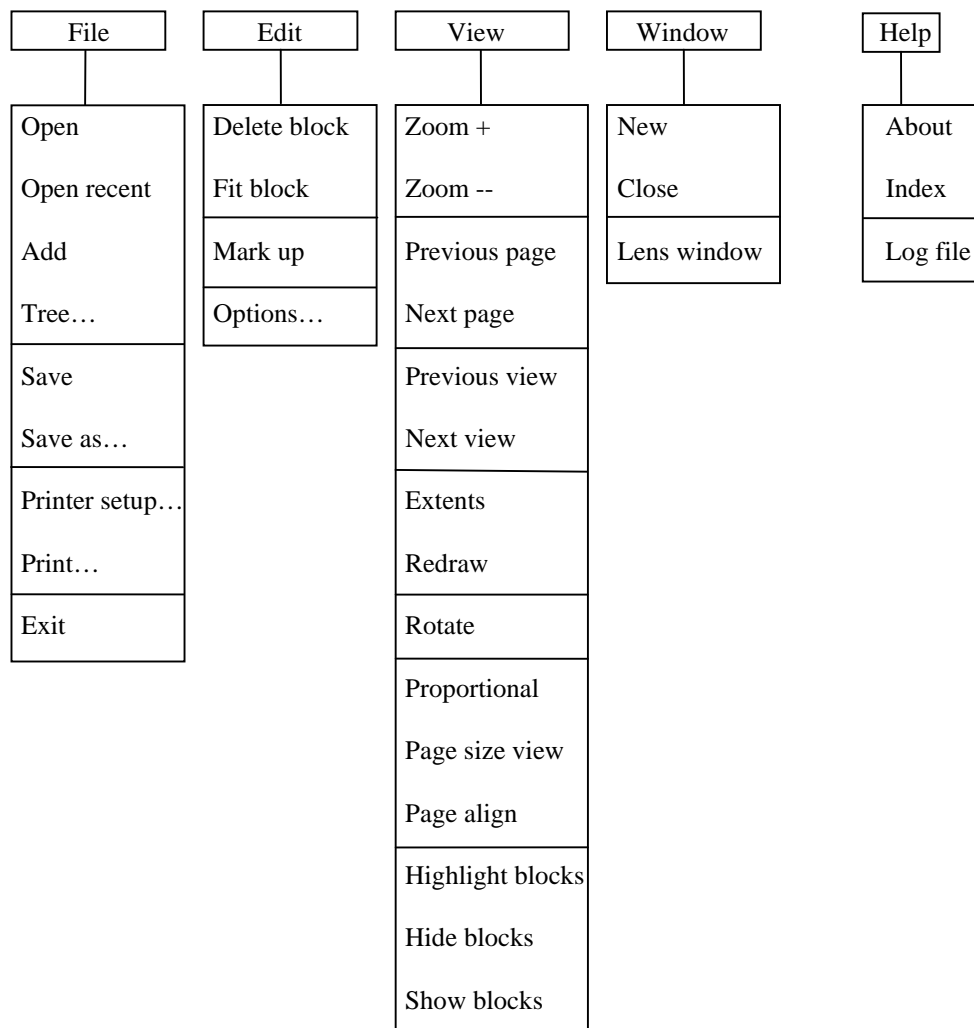
If the plot file consists of multiple pages, a page list is opened to easily navigate between the pages. There are also buttons in the toolbar, which allow traversing the pages in the numerical order (see *Next page* and *Previous page* buttons below).

**Mark up toolbar**

The mark up toolbar allows you to modify the drawing by inserting mark-ups and comments if necessary.

## 5 The Main Menu Tree


The viewer menus have the following structure:



### 5.1 File Menu

This menu contains all the functionality to manage **HP-GL** files in the viewer, to add files as blocks into existing **HP-GL** files, to save drawings in different formats, to change printer settings and print the drawing and to exit from the viewer.

#### *Open*


Open a system-specific dialog for opening a new file in the current viewer window. If a zoom window has been defined, the new file will be opened in it. Otherwise the file will be opened in the main window. Also available in the toolbar as button *Open file* (symbol  ). On **Windows**, it is also possible to drag and drop one or several plot files to the viewer main window to open them. If multiple files are dropped, a separate window is opened for each of them.



### ***Open recent***

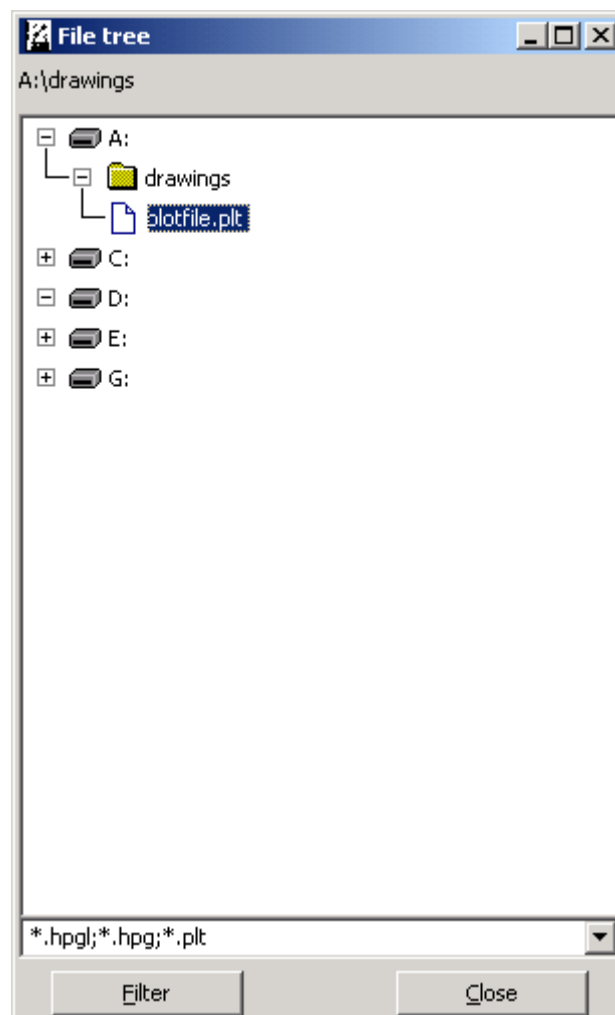
Open a recently opened file. Last 10 files opened during the current session are saved in a dropdown list, where they can be easily reselected.

### ***Add***

Add another **HP-GL** file as a block into the current page of the current drawing. If a zoom window has been defined, the file will be sized to fit in this window. Otherwise the file will be sized to fit in the current window. Also available in the toolbar as button *Add file as block* (symbol  ).

### ***Tree...***

Open the file tree window. Using file tree you can navigate through files and directories and open **HP-GL** files just by clicking on the file name (double-click on **Unix**). In the bottom part of the window there is a dialog for setting the file filter. A set of wildcard expressions can be entered separated by semicolon (‘;’) characters. The file filter is set by pressing the *Filter* button, and only files matching the filter expression will be displayed. The file filter is saved together with other configuration data when saving the configuration file from the *Edit - Options* dialog.



## Save

Save a modified file with the same name and format.

## Save as...

Save the currently open drawing in different formats. The viewer always saves whatever is displayed in the current drawing window, i.e., what you see is what you will have in the saved file. If you want to save the entire page, set the drawing window first to its extents. Also, if a color drawing is saved in monochrome mode, all color information will be lost.

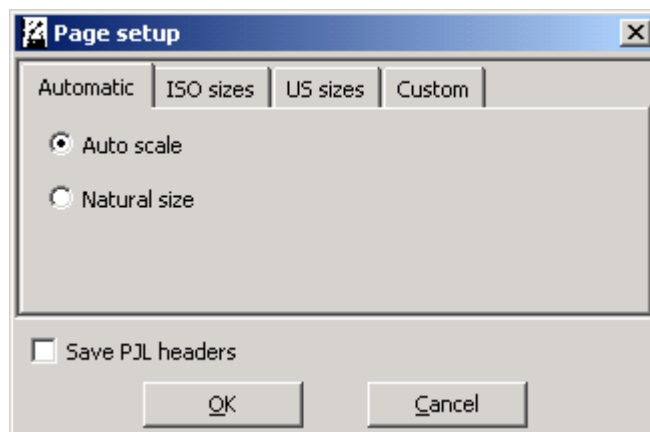
If you save the drawing in **HP-GL** or **HP-GL/2** all information about pages, visible and hidden blocks will also be saved. If you then plot the saved file, you will not see the hidden blocks on paper. However, you can later open the file in the viewer and make hidden blocks visible at any time.

If you save the drawing in **Encapsulated PostScript**, **PostScript**, **DXF**, or **CGM**, only the current page will be saved.

When saving in **HP-GL**, **HP-GL/2**, or **PostScript**, a second dialog will open where the page size can be set. There are four different sub-dialogs available:

### Automatic

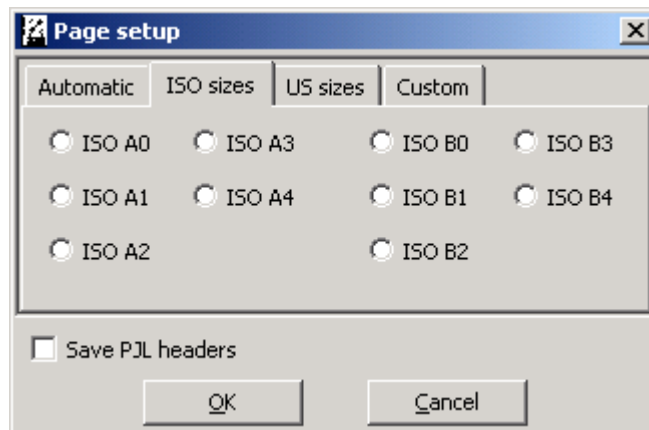
Define the drawing to be either auto scaling or to be saved in its natural size.



The *Auto scale* setting indicates that the file will be sized to fit the page according to the paper size of any **PostScript** or **HP-GL / HP-GL/2** device to which it is sent. On **HP-GL / HP-GL/2** devices the page size is defined by the hard-clip limits of the device. *Natural size* setting saves the drawing in its natural size as it appears in the plot file.

### ISO sizes

Select the paper size from ISO A or B series.

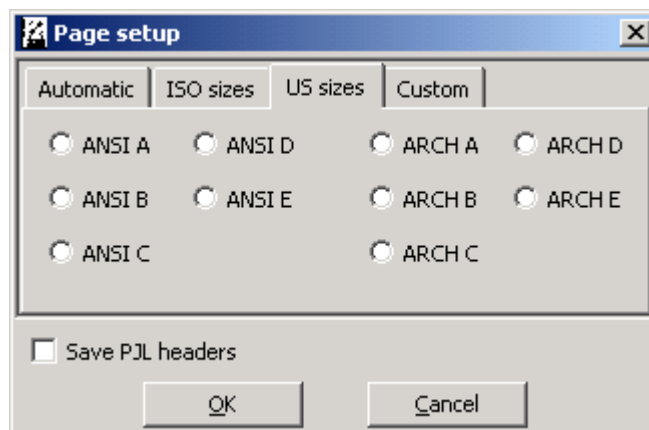


The available ISO paper sizes are:

A0	(841 x 1189 mm)
A1	(594 x 841 mm)
A2	(420 x 594 mm)
A3	(297 x 420 mm)
A4	(210 x 297 mm)
B0	(1000 x 1414 mm)
B1	(707 x 1000 mm)
B2	(500 x 707 mm)
B3	(353 x 500 mm)
B4	(250 x 353 mm)

### US sizes

Select the paper size from one of two US standard series, ANSI or Architectural.



The available ANSI paper sizes are:

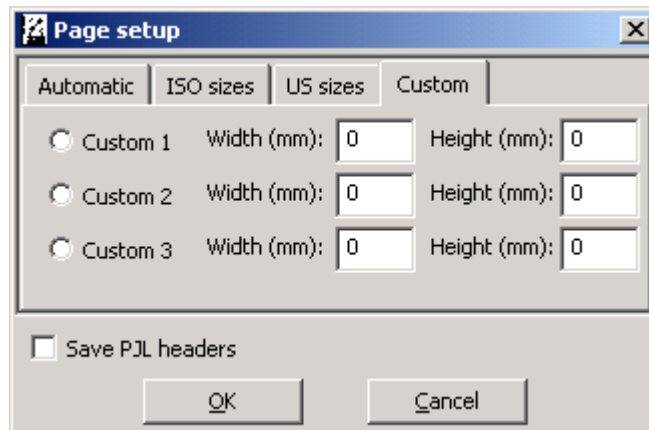
A or Letter	(8.5 x 11 in = 215.9 x 279.4 mm)
B or Tabloid or Ledger	(11 x 17 in = 279.4 x 431.8 mm)
C or Broadsheet	(17 x 22 in = 431.8 x 558.8 mm)
D	(22 x 34 in = 558.8 x 863.6 mm)
E	(34 x 44 in = 863.6 x 1117.6 mm)

The available Architectural paper sizes are:

<i>A</i>	(9 x 12 in = 228.6 x 308.4 mm)
<i>B</i>	(12 x 18 in = 308.4 x 457.2 mm)
<i>C</i>	(18 x 24 in = 457.2 x 609.6 mm)
<i>D</i>	(24 x 36 in = 609.6 x 914.4 mm)
<i>E</i>	(36 x 48 in = 914.4 x 1219.2 mm)

#### *Custom*

Define a custom paper size.



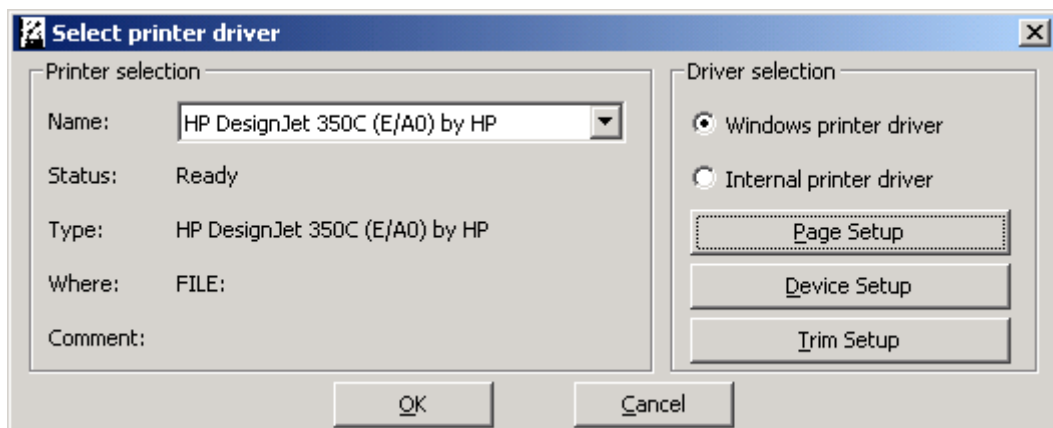
The size must be defined in a portrait orientation, i.e., where width is normally defined to be smaller than height.

#### *Save PJI headers*

Save Printer Job Language headers in the beginning of the saved file. See the description of this option in section *Edit - Options - Preferences*.

#### *Printer setup...*

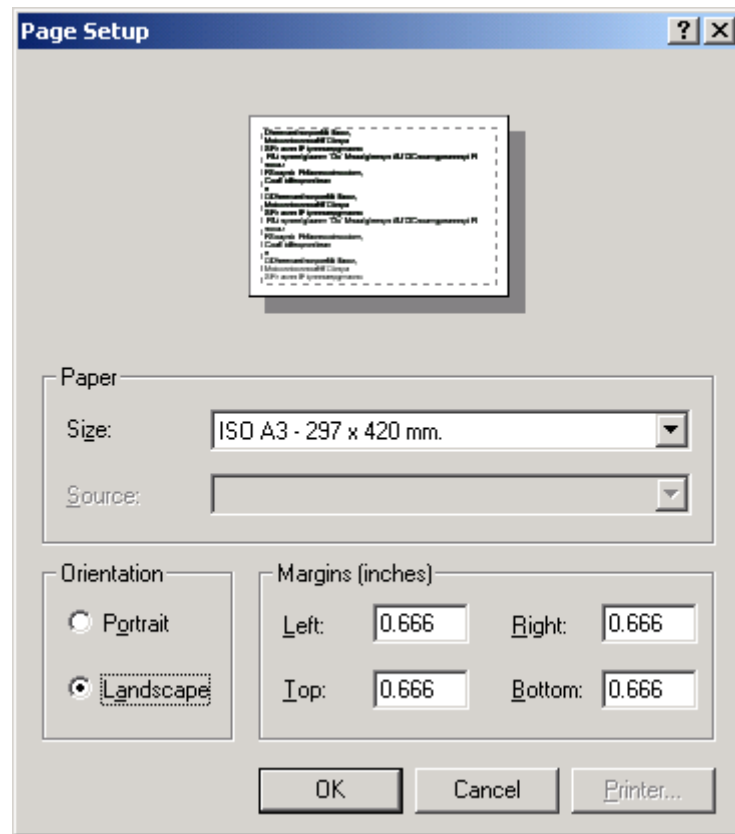
On **Windows** and **Unix**, open the printer setup dialog window. Here you can select the printer and set the appropriate parameters specific to that printer. On **Classic MacOS** this menu item is called *Page setup* and it opens the standard system page setup dialog. Below is presented the *Printer setup* dialog on **Windows**:



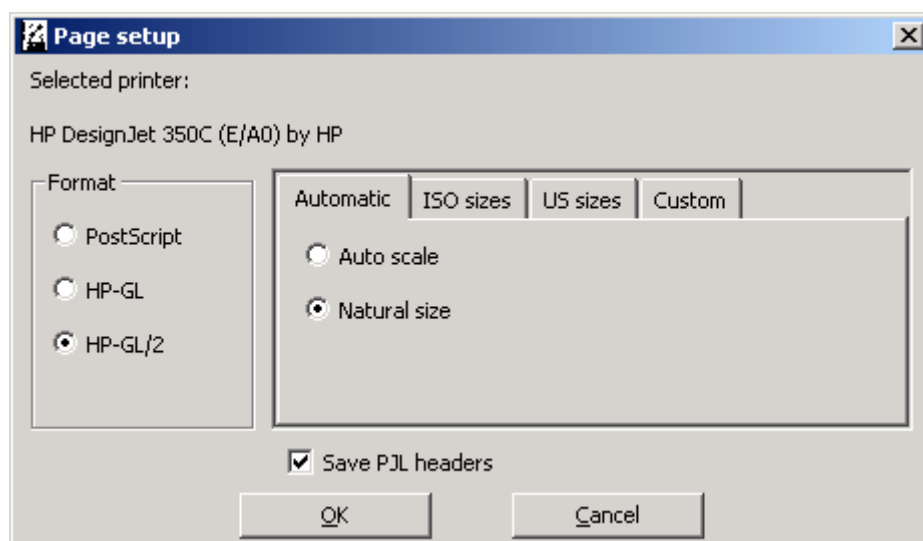
The printer can be any of the currently defined printers in the system. The printer driver can be either the standard **Windows** printer driver, which should in principle be able to print to any available printer or the internal **HP-GL** viewer printer driver, which can generate output in **PostScript**, **HP-GL**, or **HP-GL/2**. The choice of the printer driver

depends on the particular needs. Sometimes the internal driver is able to produce better and more dependable output than the standard **Windows** one, especially when exact scale correspondence is required.

When the **Windows** printer driver is selected, the *Page Setup* button opens the standard **Windows** page setup dialog, where the basic page configuration can be carried out:

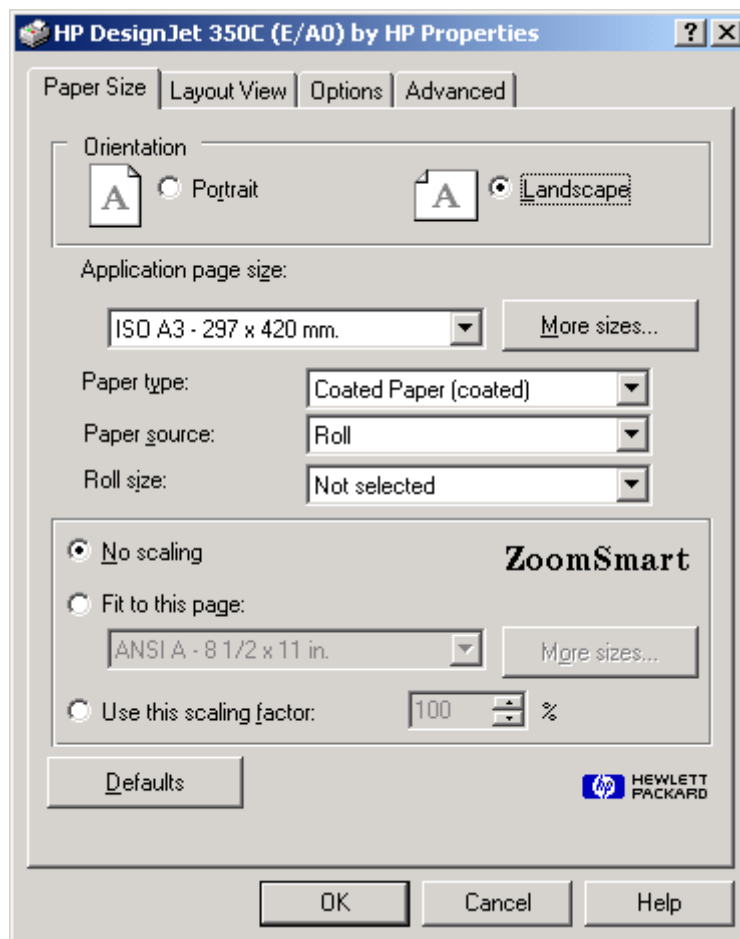


When the internal printer driver is selected, a different page setup dialog is presented:



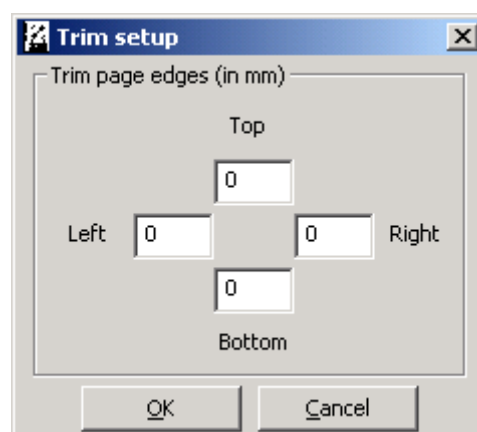
The *Format* selection defines the output language format, whereas the page size is defined using a similar dialog as described above for the *Save As* option.

When the **Windows** printer driver is selected, the *Device Setup* button opens the device setup dialog for the **Windows** printer in question. This dialog is always particular to the printer driver and usually furnished by the printer vendor. Below is presented an example of such a dialog:

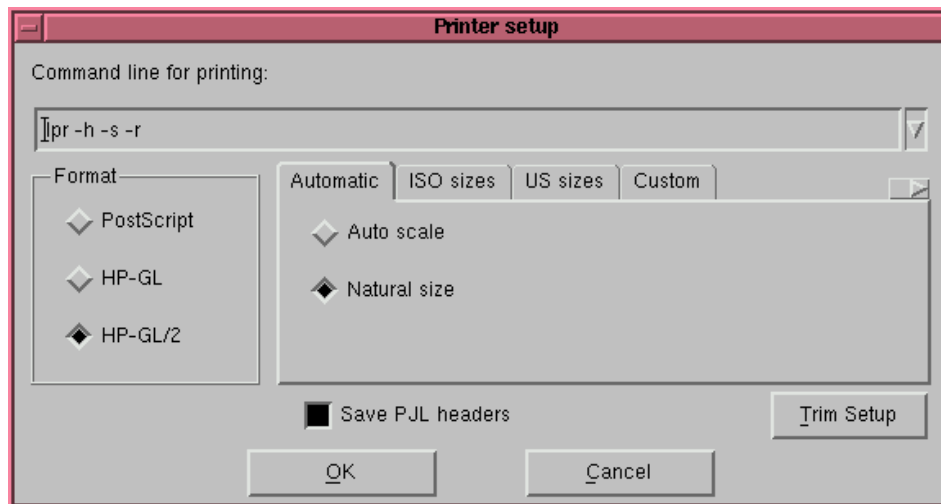


When the internal printer driver is selected, the *Device Setup* button is disabled.

The *Trim Setup* button opens a dialog for setting extra trim around the page edges. This allows you, for example, to clip away a margin around the plot or to reduce the apparent dimensions of the plot in order to better fit it on paper without affecting the scale. This dialog is not available on **Classic MacOS**.



Below is presented the *Printer setup* dialog on **Motif**:



This dialog allows you to set the command line for printing and to select the format and page size for the output. The viewer lets you define up to 5 system commands from which to choose how to print the output. Typically these commands should be valid printer commands in your system with all required command options, e.g.:

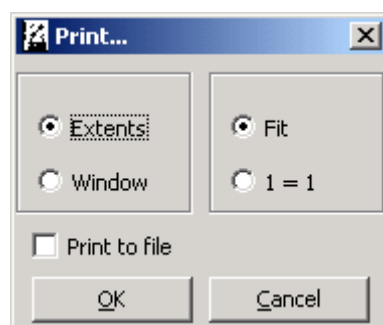
```
lpr -h -s -r
```

On most **Unix** systems the above command would send output to the default printer without a header page. The page size is defined using a similar dialog as described above for the *Save As* option. The *Trim Setup* dialog is also similar to the one described above.

On **Windows** and **Unix** platforms all selections made to the printer configurations including the internal **Windows** printers are saved together with the other viewer options (see *Options Save* below).

### ***Print...***

On **Windows**, open a print dialog, which allows you to define either the entire drawing (extents) to be printed or the current window. You can also specify whether the file should be resized to fit on the paper or whether it should be scale-correct (1=1) irrespective of the actual page size.



On **Classic MacOS**, open the standard system print dialog, which allows you to set up all the printing parameters.


On **Unix** platforms the viewer always prints the contents of the current window, i.e. what you see in viewer window is what you will get on paper (including color or monochrome mode and pen thickness on or off). You can print any part of the drawing by changing the size and position of the viewer window. If you want to print the entire drawing it is a good idea to first set the window to its true extents by choosing the *Proportional* menu item from the *View* menu.

### ***Exit***

Exit the viewer and close all viewer windows. Any modifications to files not previously saved will be lost. On **Classic MacOS** this menu item is called *Quit* in accordance with the standard **MacOS** convention.

## **5.2 Edit Menu**


### ***Delete block***

Remove all selected blocks. Note that there is no undo option in the viewer for this function. Also available in the toolbar as button *Delete block* (symbol ).

### ***Fit block***

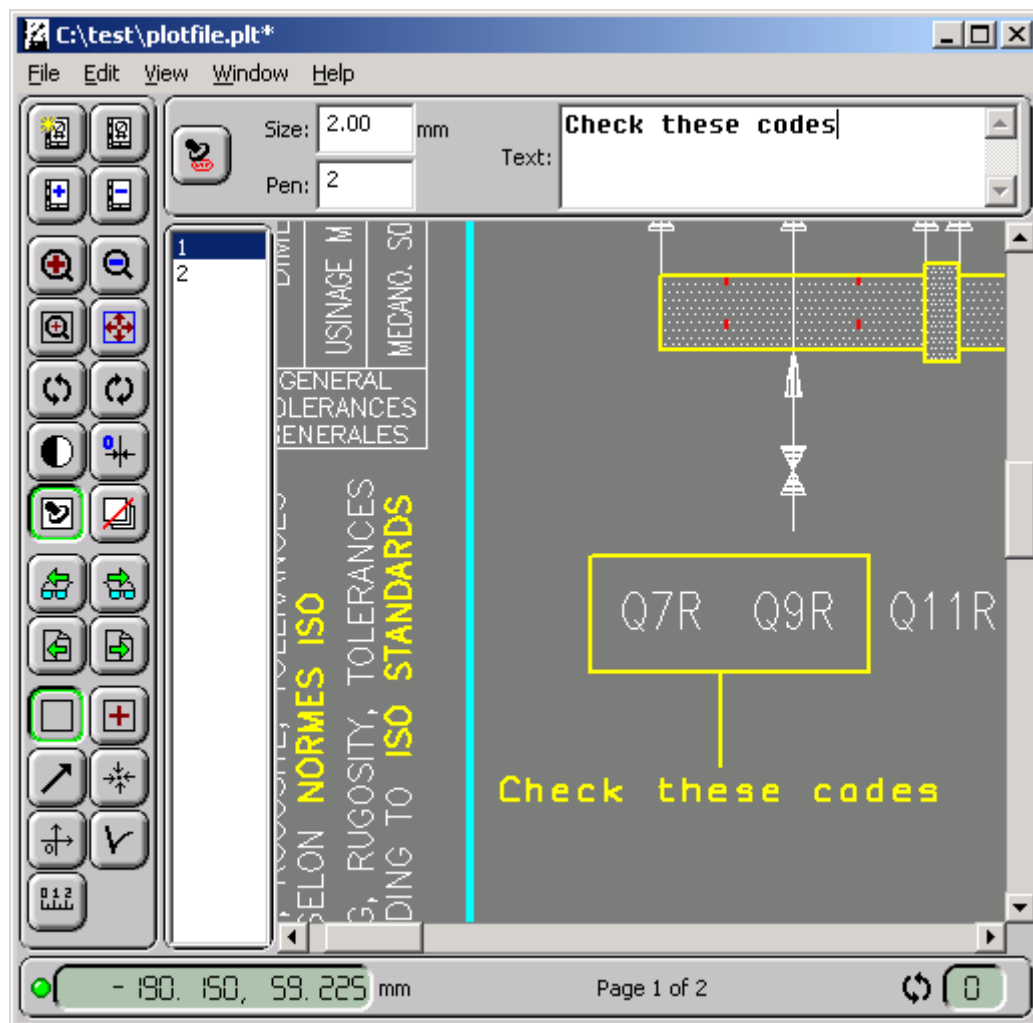
Fit all selected blocks to the zoom window if one is defined or to the current window if not.

### ***Mark up***

Switch the mark up mode on and off. In this mode you can place your own text in the drawing by clicking the left mouse button where the text should start. Also available in the mark up toolbar as button *Mark up mode on/off* (symbol ).

To be able to make mark-ups in the drawing, the mark up toolbar must be open (see *Edit - Options - Mark up* dialog below). The mark up text is written in the *Text* dialog box, and the size of the text in millimeters as well as the **HP-GL** pen to use can be set in the *Size* and *Pen* dialog boxes, respectively. To place the mark up text in the drawing, switch mark up mode on, move the mouse cursor to the position you want the text to appear, and press the left mouse button.





If a zoom rectangle (see *Zoom window* mouse mode) was defined before switching the mark up mode on, this rectangle and a line from the rectangle to the mark-up text will be drawn. The text will be adjusted with respect to this line. The easiest way to position the text in this case is to press down the left mouse button, move the mouse pointer to the intended starting point of the text, and to release the mouse button. This allows you to see a rubber band image of the connecting line while moving the mouse pointer. All mark-ups are separate blocks, which makes it possible to hide or show them again or to delete them as any other blocks.

### ***Options...***

Open the *Options* dialog for setting various configurable parameters (see below).

## 5.2.1 Options Dialog Window

The Options dialog allows you to change the basic configuration parameters of the viewer. All the settings can be saved to a configuration file, which will be read every time the viewer is started afterwards. The different dialogs available are explained below.

### Standard buttons

These buttons are visible to all sub-windows of the *Options* dialog.

#### *OK*

Apply all changes to the current configuration and close the *Options* dialog. Changes will only affect the currently running viewer instance.

#### *Apply*

Similar to the *OK* button, but keep the *Options* dialog open.

#### *Open*

Open the standard system file open dialog for choosing the configuration file to load. The current configuration will be set from this file.

#### *Save*

Save the current configuration to the default configuration file. By default this file will be named `HpglView.cfg` and it will be saved in the standard system directory on **Windows 95** and **98**, in the user's application data directory on **Windows 2000** and **XP**, in the viewer directory on **Classic MacOS**, and in the user's home directory on **Unix** platforms. When you save the configuration options, sizes and positions of the current window and the lens window will also be saved. On **Windows** and **Unix** platforms all printer configurations are saved as well.

#### *Save as...*

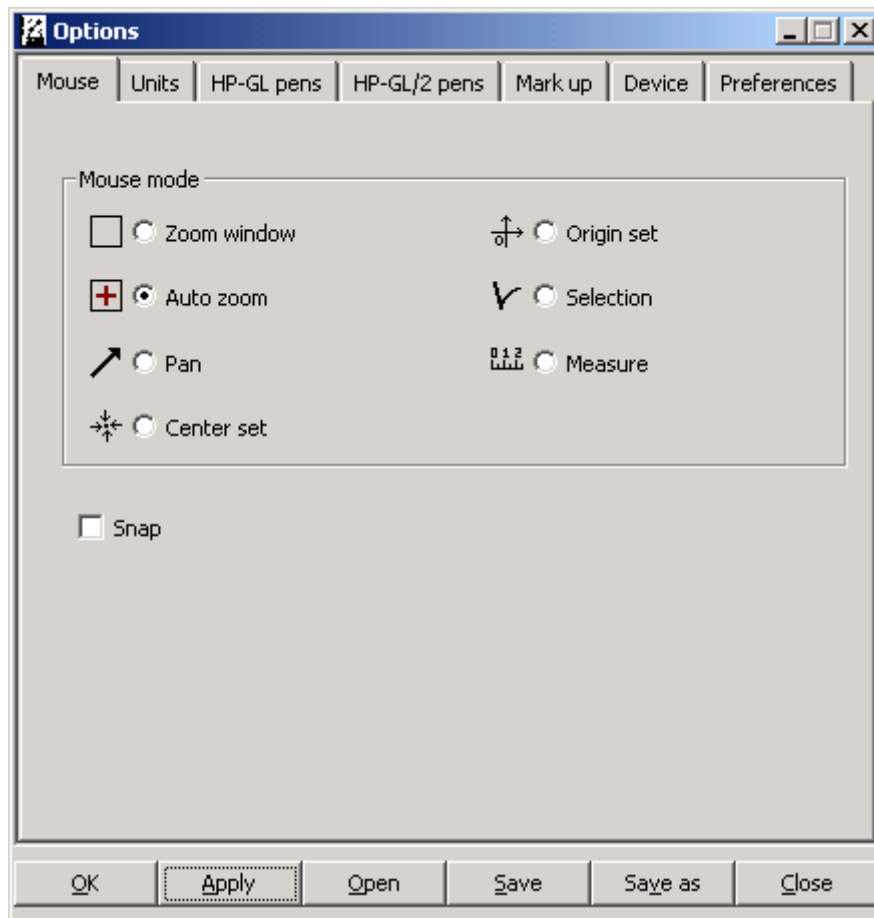
Save the current configuration but define an alternate path and name for the configuration file.


#### *Close*




Close the *Options* dialog window. If the changes made were not applied to the current viewer instance, they will be discarded.


## Mouse

Select the mouse mode. The following modes are available:




*Zoom window* (symbol )


Define a rectangle in the drawing, which can then be opened in a separate zoom window or zoomed to in the same window. To define the rectangle, press down the left mouse button in one corner of the desired area. Move the mouse pointer to the opposite corner and release the button. If you press the *Increase zoom* button (symbol ) , the area defined by the rectangle is zoomed to fit the current window. If you press the *Reduce zoom* button (symbol ) , the entire drawing will be sized to fit within the area defined by the rectangle. To open the defined area in a new zoom window, press the *Open/Close lens window* button (symbol ) .

*Auto zoom* (symbol )


Same as above, but when the button is released, the defined zoom rectangle will be automatically sized to fit to the current window.

*Pan* (symbol )


Pan the drawing using the mouse. Move the mouse pointer to the point within the drawing that you want to move. Press the left mouse button and move the mouse pointer to a new location. A rubber band pan vector will be drawn on screen indicating the direction and length of the movement. After the mouse button is released, the drawing is panned so that the starting point will be at the new location.

*Center set* (symbol )


Define the new center point within the current window. Move the mouse pointer to the point that you want to be the new center of the window and click the left mouse button.

*Origin set* (symbol )

Define the new coordinate origin. Move the mouse pointer to the point that you want to be the new coordinate origin within the drawing and click the left mouse button. Afterwards the coordinate display in the lower left corner will reflect the new coordinate system.

*Selection* (symbol )

Select or deselect blocks. Move the mouse pointer over the block that you want to select or deselect and click the left mouse button. If there are several overlapping blocks in the same area, the block with a line or point nearest to the mouse pointer will be affected.

*Measure* (symbol )

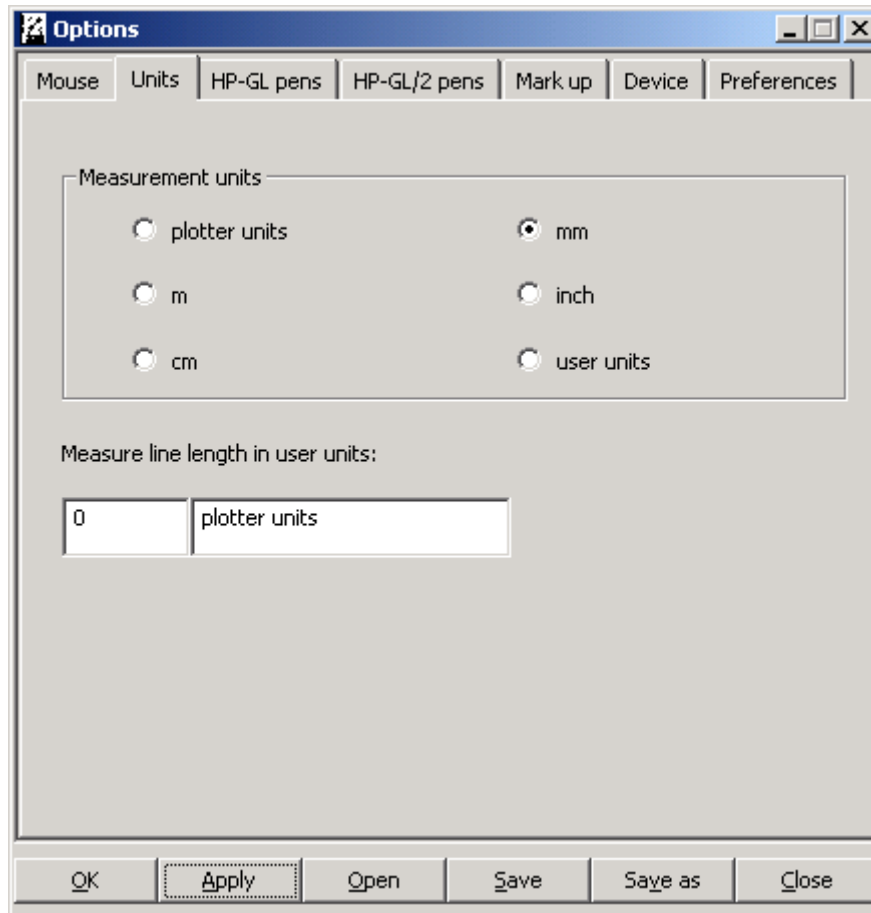
Measure distances. Press down the left mouse button at the starting point and while keeping the button pressed move the mouse pointer to the end point. A rubber band vector is drawn as the mouse pointer moves and the distance between the two points is displayed in selected units next to the vector.

*Snap*

This is a modifier of the mouse behavior for zoom window and pan modes. If snap is on, the zoom window and the pan vector will snap to the nearest drawing object, either a point or line, as the mouse pointer is moved.

## Units

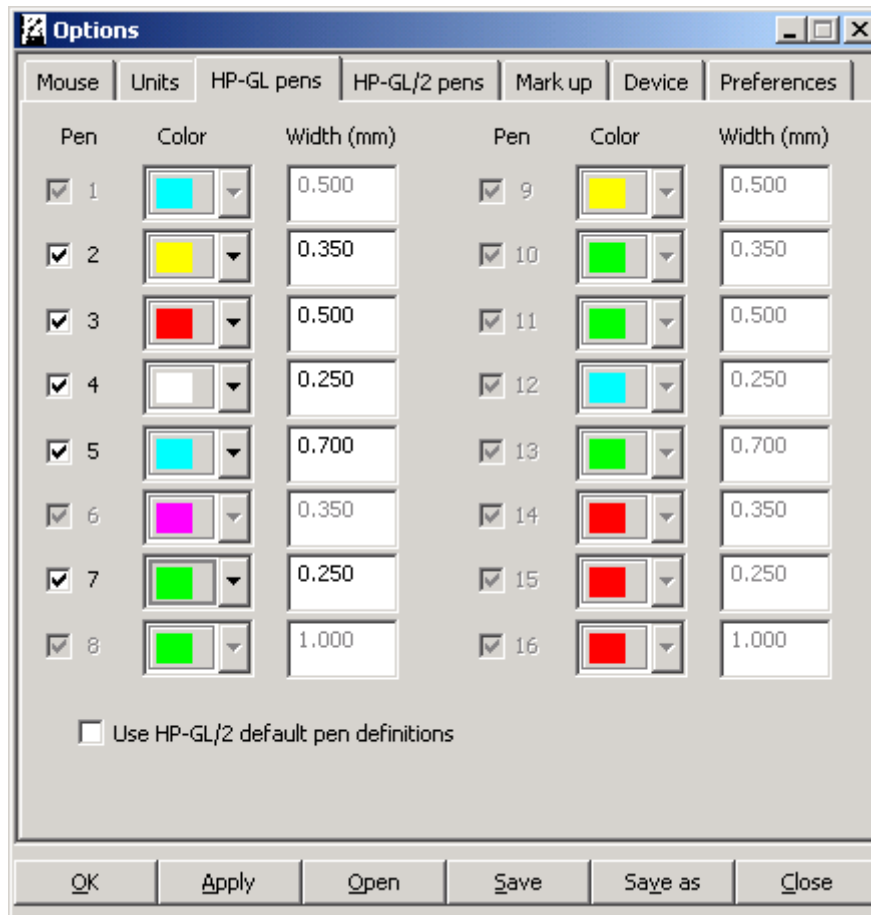
Select units for coordinate display and distance measurement. The choices are *plotter units* (1 pu = 1/40 mm), *millimeters*, *meters*, *inches* (1 in = 25.4 mm), *centimeters*, and *user units*. The choice of units affects only the viewer display and is not propagated to output files.



The *Measure line length in user units* dialog allows you to define your own units for the coordinate display and distance vector. To define your own unit, switch to *Measure* mouse mode (symbol  $\frac{0.1}{1.2}$ ) and draw a distance vector. As you draw the vector, you'll see the number field in the window change accordingly. This corresponds to the length of the vector in the currently defined user units (by default *plotter units*). To define a new correspondence, simply type in a new value in the number field. You can then also change the unit name to whatever you want.

### HP-GL pens

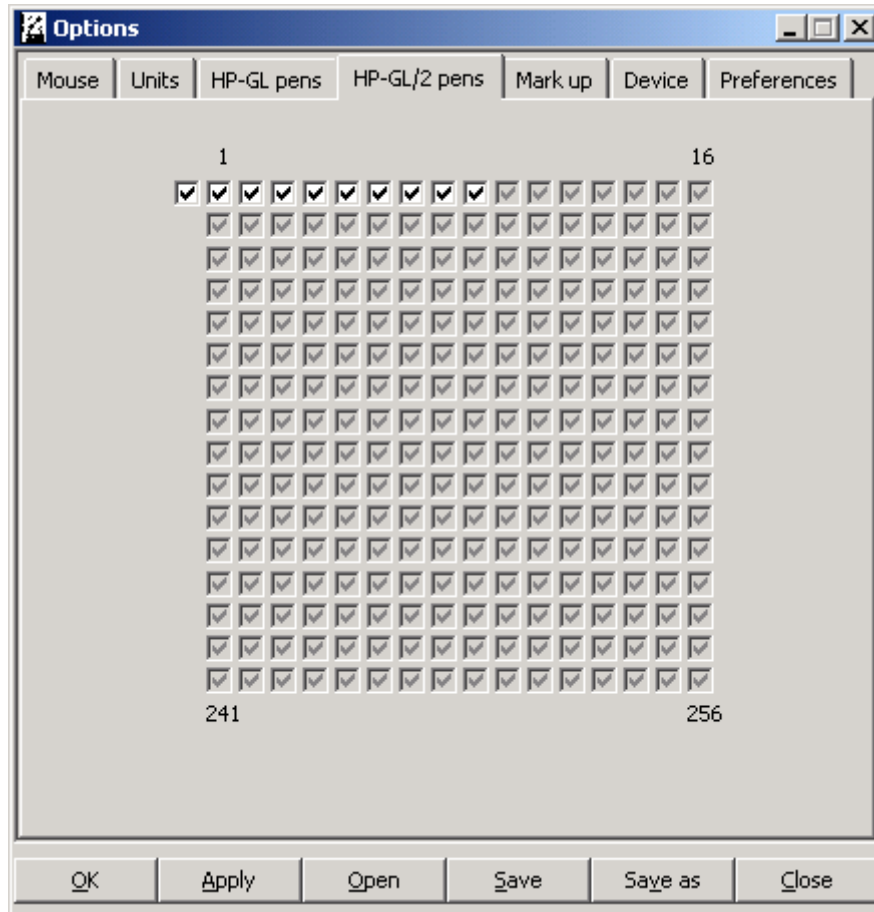
Open the dialog for setting up **HP-GL** pens. If the drawing is **HP-GL** (as opposed to **HP-GL/2**), the color and pen thickness of each pen can be set here in the same way as one would set up the pens in a real old fashioned pen plotter. The checkbox in front of every pen can be used to disable and re-enable a pen if necessary. Only those pens that are actually defined or used in the drawing can be modified, others are disabled.



If an old style **HP-GL** drawing is plotted on a modern **HP-GL/2** plotter, the resulting output will depend on how the plotter pen palette is set up because an old **HP-GL** plot file doesn't carry any information about the palette. Normally the default plotter palette is set up either programmatically in advance or from the plotter console. More often than not the plotter palette is not set specifically and the **HP-GL/2** default palette is used. The checkbox *Use HP-GL/2 default pen definitions* will reset the pen definitions to the **HP-GL/2** defaults (pen widths: 0.35 mm, pen colors: 1=black, 2=red, 3=green, 4=yellow, 5=blue, 6=magenta, 7=cyan, 8 and greater black). This will help to visualize the drawing in the viewer as it would come out on such a plotter.

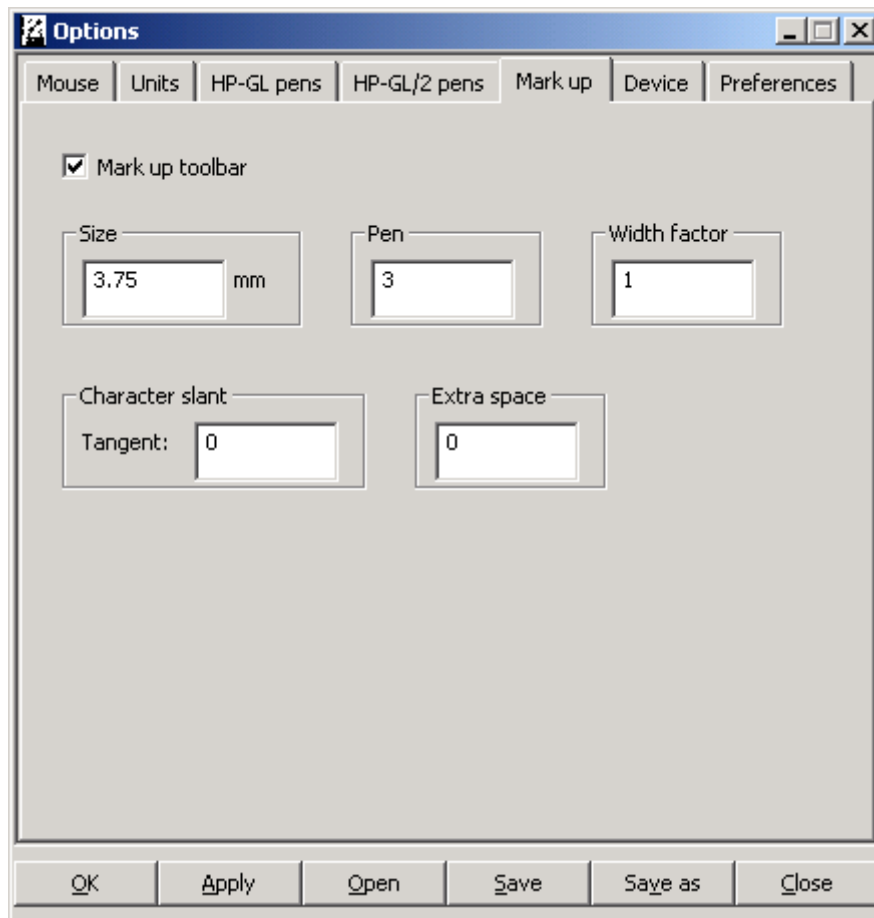
### ***HP-GL/2 pens***

Open the dialog for setting **HP-GL/2** pens. Since colors and widths of **HP-GL/2** pens are defined in the drawing or have default values, the only thing that can be done to the pens is to disable or enable them. Only those pens that are actually defined in the drawing can be modified, others are disabled. An **HP-GL/2** plot file can define up to 256 pens. However it is quite common for plot files to use only one or a few pens and keep redefining them on the fly.




## Mark up

Open the dialog for setting parameters for the mark up text.



### Mark up toolbar

Enable the mark up toolbar to be able to write and position the mark up text. Also available in the toolbar as button *Marktool open/close* (symbol ).

### Size

Set the size of the characters in millimeters. This setting is also available in the mark up toolbar.

### Pen

Set the number of the pen to use. This setting is also available in the mark up toolbar.

### Width factor

Set the factor by which the text is expanded horizontally. Both the character width and spacing are affected.

### Character slant

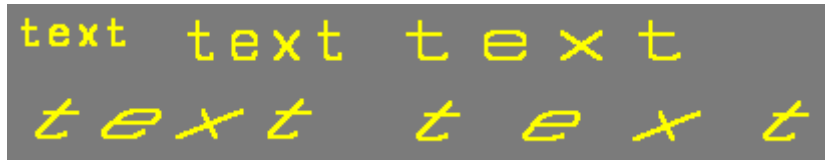
Set the character slant by defining the tangent of the slant angle. The *Tangent* parameter is defined as the ratio of the horizontal displacement of the top of the character to the height of the character. For example, *Tangent* value 1 means that the top of the character is displaced by as much as the height of the character, thereby making the slant angle 45 degrees to the right. A negative tangent value would cause characters to slant to the left.



### *Extra space*

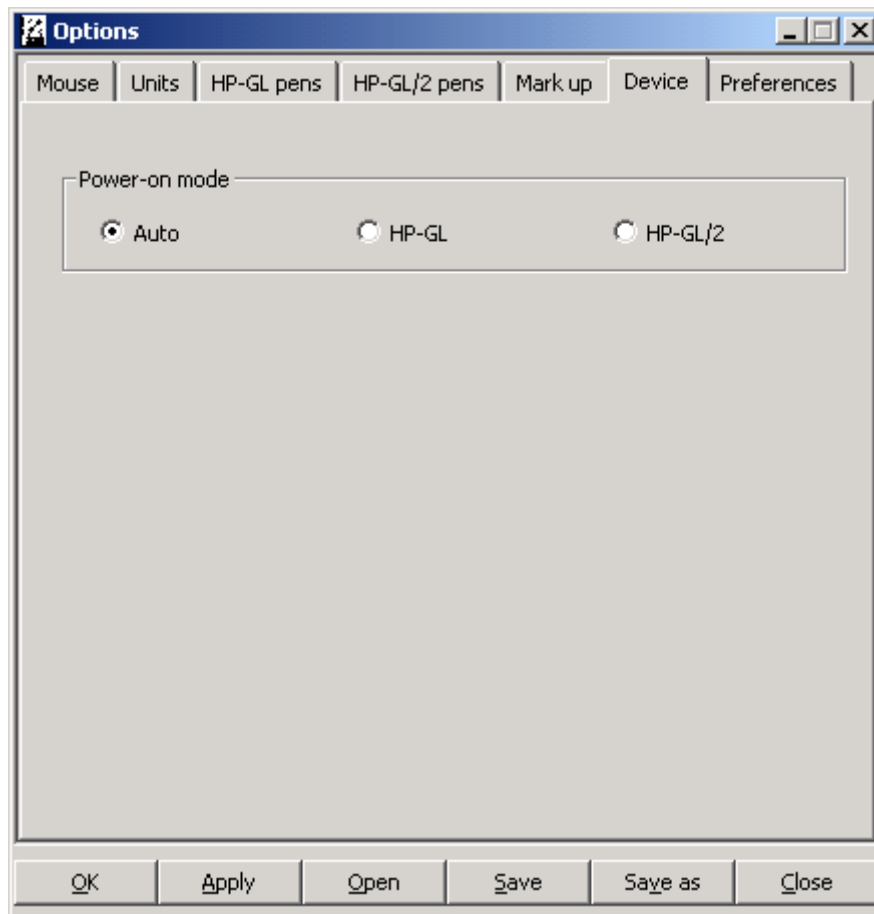
Set the amount of extra space between characters. The unit is the character width, i.e., one unit of extra space doubles the character spacing.

In the following example the first text sample is the original text using some pen. The second sample is the same but with *Size* parameter increased by 50 %, the third sample is drawn with *Width factor* set to 1.8, the fourth sample has *Tangent* set to 1.0, and the fifth sample includes 0.5 units of extra space between characters.



### *Device*

Open the dialog for setting options associated with the emulated plotter device.



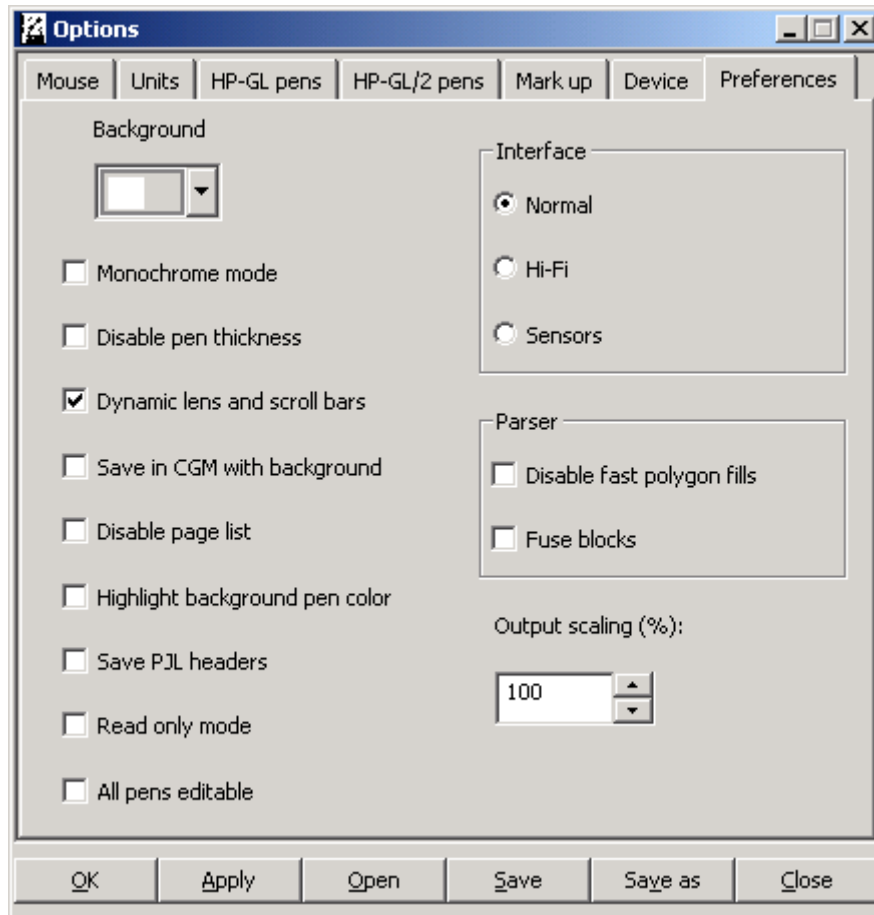
### *Power-on mode*

Select the default power-on mode of the emulated plotter device. Available options are: *Auto*, *HP-GL*, and *HP-GL/2*. The *Auto* option starts the viewer in the traditional **HP-GL** mode, but may change the mode to **HP-GL/2** if commands are found in the file, which only appear in **HP-GL/2**. In this case the plot file is reinterpreted from the beginning with the correct **HP-GL/2** default settings. The explicit *HP-GL* and *HP-GL/2* options retain the selected mode unless it is specifically changed by the appropriate **PJL** device control commands or escape

sequences in the plot file itself. This is a parser option, which means that in order to be effective, it needs to be set before the file is opened in the viewer.

## Preferences

Open the dialog for setting various miscellaneous and personal preferences.





### Background

Select the background color for the drawing window. The possible choices are *white*, *gray*, and *black*. Setting the background color may help in visualizing some drawing details especially when plotting light colors on white background. Note that this background color setting only affects the viewer, i.e., it is not propagated to the plotter. In that sense this is the only place in the viewer where what you see on screen may not appear exactly the same way on paper.



### Interface

Choose between different interface themes. There are three themes, *Normal*, *Hi-Fi*, and *Sensors*. The themes simply affect the decorations around some of the graphical elements of the viewer, like buttons and toolbars. There are no functional differences between the different interfaces.

### Monochrome mode

Switch between monochrome and color display modes. Normally all colors are rendered as they are defined in the drawing, but monochrome mode allows you to see how the drawing would look like when rendered on a monochrome plotter. Also available in the toolbar as button *Color/monochrome mode* (symbol  when in color mode,  when in monochrome mode).

#### *Disable pen thickness*

Switch on and off line thickness for all pens. This comes in handy sometimes when trying to figure out exactly how some features of the plot have been rendered. Also available in the toolbar as button *Thickness enable/disable* (symbol  when thickness enabled,  when disabled).


#### *Dynamic lens and scroll bars*

Switch the dynamic lens mode on and off. In dynamic lens mode the contents of the lens window are dynamically updated as the zoom rectangle is dragged around the main window using the mouse.

#### *Save in CGM with background*

Disable or enable saving the background color when saving in **CGM**. Normally the background is treated as transparent and not saved with the drawing.

#### *Disable page list*

Disable or enable the page list for navigating multi-page drawings. Normally the page list is shown by default whenever a multi-page drawing is opened. On the other hand it is never shown for single-page drawings. Also available in the toolbar as button *Page list hide/show* (symbol ).

#### *Highlight background pen color*

When changing the background color (see *Background* above), it may happen that some pens are defined to have the same color as the background and thus become invisible. This setting allows you to invert the pen color thereby making it visible again. On black background a black pen becomes white and on white background black.

#### *Save PJI headers*

When saving the drawing in **HP-GL**, **HP-GL/2**, or **PostScript** insert the **HP PJI** (Printer Job Language) headers in the beginning and in the end of the file. **PJI** is a meta-language, which allows one to pass directives to the printers and plotters, which understand it (most modern **HP-GL** plotters and many **PostScript** printers). The only directive that the viewer inserts into the header is the language mode (**HP-GL**, **HP-GL/2**, or **PostScript**). This may be useful when printing to some multi language plotters, which may not otherwise recognize the plot file as a file of the proper type.

#### *Read only mode*

Disable or enable interactive editing and saving of the drawing. When the read only mode is active, the following functions are disabled: *Add file as block*, *Delete block*, *Fit block*, *Mark up*, *Marktool open/close*, *Save*, and *Save As*. Visual manipulation and printing commands will remain available, however. This option also has no effect on the command line operation of the viewer.

#### *All pens editable*

Normally only **HP-GL** or **HP-GL/2** pens, which are used in the displayed plot file can be edited. Enabling this option will make all pens editable irrespective of whether they are used in the displayed plot file or not. This option exists mainly to allow creation of configuration files with certain pen parameters for future use.

#### *Disable fast polygon fills*

Disable the special polygon drawing instructions from being used in rendering the drawing. In this case the old vectorizing method is used instead. This is a parser option, which means that in order to be effective, it needs to be set before the file is opened in the viewer.

#### *Fuse blocks*

Cause all defined blocks in the drawing to be fused together. Then after saving the file as **HP-GL** or **HP-GL/2**, it contains only one monolithic data stream. This is a parser option, which means that in order to be effective, it needs to be set before the file is opened in the viewer.


#### *Output scaling (%)*

Set the scaling factor that is applied to the coordinates as the file is saved. The scaling factor is internal to the file, i.e., if the file is printed, it will appear unchanged. However, scaling the coordinates may be necessary if the output file is to be further edited or included into another application.


### **5.3 View Menu**

This menu lets you change the drawing display in the window. It contains the following items:


#### **Zoom +**

Increase zoom by a factor of 1.5 or fit the zoom rectangle to viewer window if one has been defined. Also available in the toolbar as button *Increase zoom* (symbol ).


#### **Zoom –**

Reduce zoom by a factor of 1.5 or fit the current viewer window to the zoom rectangle if one has been defined. Also available in the toolbar as button *Reduce zoom* (symbol ).


#### **Previous page**

Switch to the previous page when viewing multi page files. Also available in the toolbar as button *Previous page* (symbol ).


#### **Next page**

Switch to the next page when viewing multi page files. Also available in the toolbar as button *Next page* (symbol ).


#### **Previous view**

Switch to the previous user-defined view. Also available in the toolbar as button *Previous view* (symbol ).

#### **Next view**

Switch to the next user-defined view. Also available in the toolbar as button *Next view* (symbol ).



#### **Extents**

Fit the entire drawing to the viewer window. Also available in the toolbar as button *Draw extents* (symbol ).

#### **Redraw**

Redraw the current window.

***Rotate***

Rotate the drawing in steps of 90 degrees. A submenu gives the possible rotation angles (0, 90, 180, 270). In the toolbar two buttons exist for rotating the drawing anti-clockwise (button *Rotate +90 degrees*, symbol  ) and clockwise (button *Rotate -90 degrees*, symbol  ).

***Proportional***

Set the viewer window size and aspect so that the drawing fits the window exactly.

***Page size view***

Set the viewer window size to exact correspondence of a defined page size with respect to the internal dimensions of the viewed drawing. This will show how the drawing would fit on any particular standard paper sheet. The drawing is centered in the middle of the window but it can be moved around using the pan mouse mode. The available page sizes are: ISO A0-A4, ISO B0-B4, ANSI A-E, and Architectural A-E.

***Page align***

Align the drawing inside the defined window. There are nine predefined alignment options: Top Left, Top Center, Top Right, Center Left, Center Center, Center Right, Bottom Left, Bottom Center, and Bottom Right.

***Highlight/Conceal blocks***

The *Highlight blocks* menu item draws a gray border around each block in the drawing so that it is possible to see where all the blocks are even if they are hidden. This is often the only way of selecting hidden blocks. When highlighting is on, the menu item is named *Conceal blocks* and selecting it will then remove the highlight borders around blocks.

***Hide blocks***


Make all selected blocks invisible.

***Show blocks***

Make all selected blocks visible.

## 5.4 Window Menu


### *New*

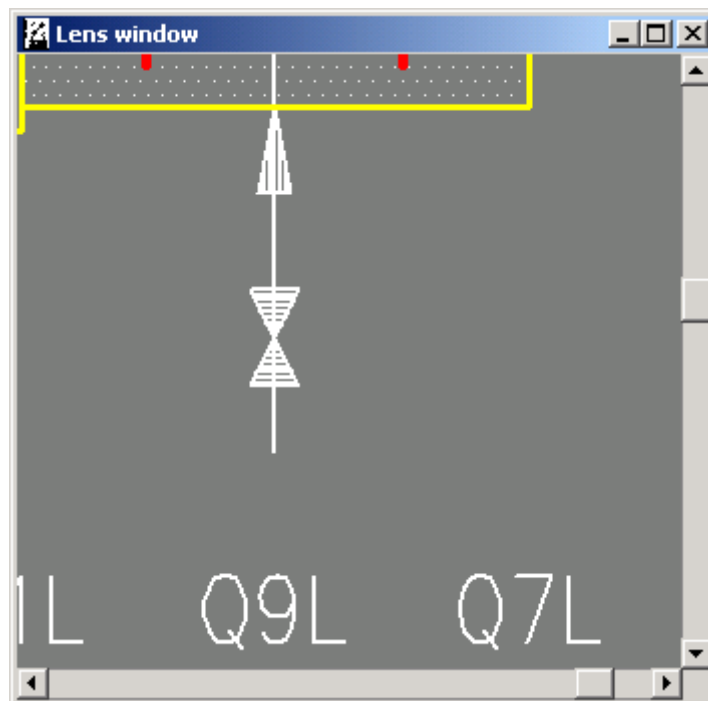
Open a new empty window. Another drawing can then be opened in this window. The number of windows and opened files is only limited by the resources of your system. Also available in the toolbar as button *Open new window* (symbol ).

### *Close*

Closes the current window. Note that if you have made modifications to the drawing and have not saved the file, all modifications will be lost.

### *Lens window*

Open and close a separate lens window if the zoom rectangle has been defined. When the lens window is open you can move the zoom rectangle with the mouse within the drawing and have the display updated in the lens window. Also available in the toolbar as button *Open/close lens window* (symbol ).



## 5.5 Help Menu

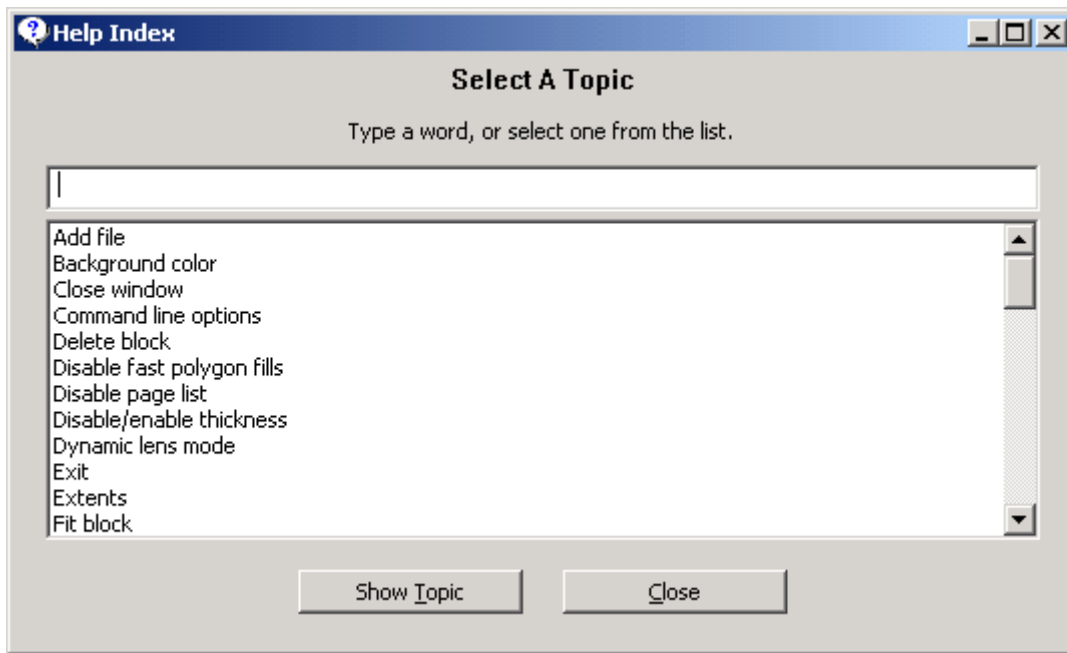
On **Classic MacOS** this menu is called *Info*, because *Help* is the standard system menu available in all **MacOS** applications.

### *About*

Show information about the program version, copyright and authors.

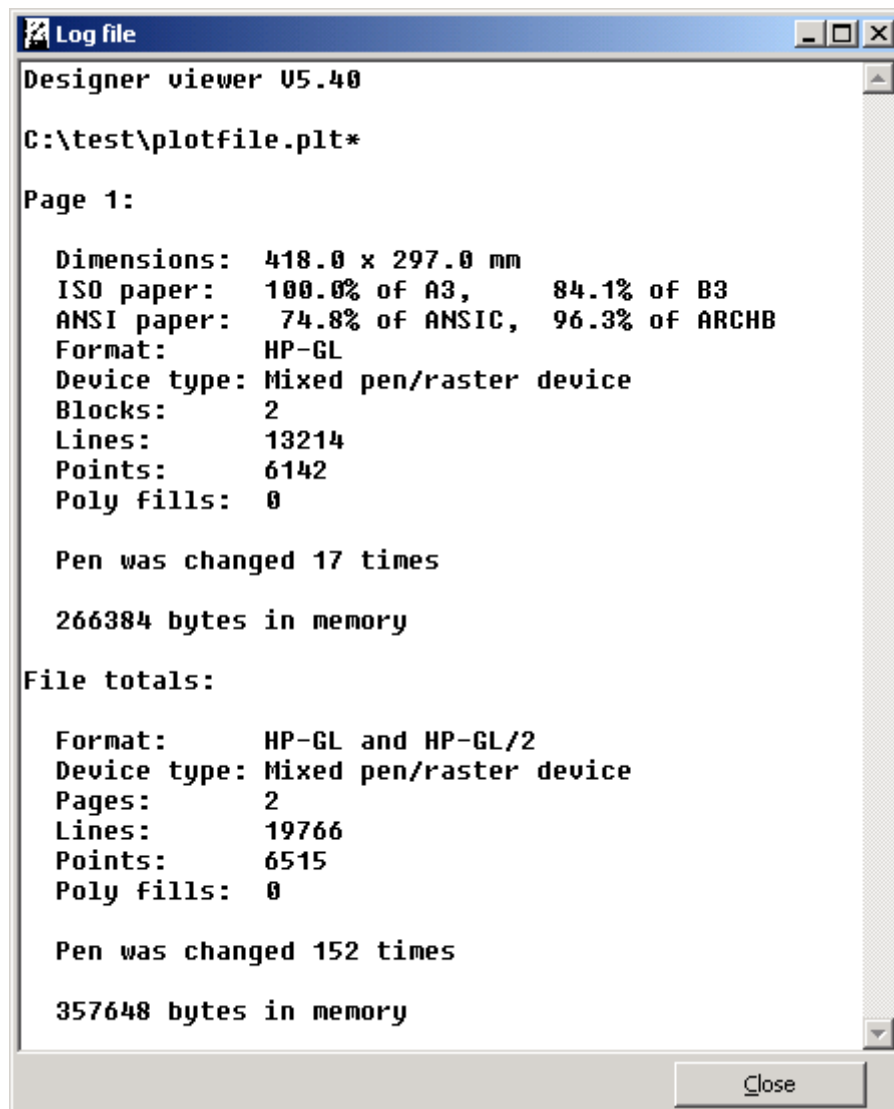
### *Index*

Open an indexed help dialog window where you can find information about the various viewer functions.



### *Log file*

Open a window with file statistics. Error messages are also printed here if there were errors when processing the plot file (the color of the error indicator in the status bar is red).



The statistics display is page specific, i.e., when displaying multi page files, the data about the currently viewed page is shown together with the file totals.



## 6 References

The **HP-GL** implementation of the viewer is based on two principal references, the first of which describes the old **HP-GL**. As implementation details of the old **HP-GL** language were largely device dependent, the viewer has chosen one reference platform to emulate (Hewlett-Packard *DraftMaster* series large format plotters). The second and third references are the current principal references for **HP-GL/2**, which define the current version of the language.

1. *HP DraftMaster Plotter Programmer's Reference*, Hewlett-Packard 1987. Manual Part Number 07595-90001.
2. *The HP-GL/2 and HP RTL Reference Guide*, Third Edition, A Handbook for Program Developers. Hewlett-Packard / Addison Wesley, 1997. ISBN 0-201-31014-7.
3. *HP DesignJet Printers Language Guide*, (i.e., Fourth Edition of ref. 2, June 14, 2000). In electronic format. Contact Hewlett-Packard for details.