Chapter

III-6

Exporting Graphics (Windows)

Overview

This chapter discusses exporting graphics from Igor graphs, page layouts, tables, and Gizmo plots to another program on Windows. You can export graphics through the clipboard by choosing Edit→Export Graphics, or through a file, by choosing File→Save Graphics.

Igor Pro supports a number of different graphics export formats. You can usually obtain very good results by choosing the appropriate format, which depends on the nature of your graphics, your printer and the characteristics of the program to which you are exporting.

Unfortunately, experimentation is sometimes required to find the best export format for your particular circumstances. This section provides the information you need to make an informed choice.

This table shows the available graphic export formats on Windows:

Export Format	Export Method	Notes
EMF (Enhanced Metafile)	Clipboard, file	Windows-specific vector format.
BMP (Bitmap)	Clipboard, file	Windows-specific bitmap format. Does not use compression.
PDF	Clipboard, file	Platform-independent and high quality. Igor PDF does not support transparency.
EPS (Encapsulated Postscript)	File only	Platform-independent except for the screen preview. Supports high resolution. EPS does not support transparency. Useful only when printing on PostScript printer, creating a PDF file or exporting to PostScript-savvy program (e.g., Adobe Illustrator, Tex).
PNG (Portable Network Graphics)	Clipboard, file	Platform-independent bitmap format. Uses lossless compression. Supports high resolution.
JPEG	Clipboard, file	Platform-independent bitmap format. Uses lossy compression. Supports high resolution.
TIFF	Clipboard, file	Platform-independent bitmap format. Supports high resolution but not compression.
SVG	Clipboard, file	Platform-independent vector graphics format. A good choice if the destination program supports SVG.

Metafile Formats

The metafile formats are Windows vector graphics formats that support drawing commands for the individual objects such as lines, rectangles and text that make up a picture. Drawing programs can decompose a metafile into its component parts to allow editing the individual objects. Most word processing programs treat a metafile as a black box and call the operating system to display or print it.

Enhanced Metafile (EMF) is the primary Windows-native graphics format. It comes in two flavors: the older EMF and a newer EMF+. Igor “dual EMF” by default. A dual EMF contains both a plain EMF and an EMF+; applications that don’t support EMF+ will use the plain EMF component. EMF+ is needed if transparency (colors with an alpha channel) is used. You can export using the older EMF format if the destination program does not work well with EMF+ - see **Graphics Technology** on page III-445 for details.

EMF is easy to use because nearly all Windows programs can import it and because it can be copied to the clipboard as well as written to a file. Some programs, notably some older versions of Microsoft Office, require that you choose Paste Special rather than Paste to paste an EMF from the clipboard.

Although drawing programs can decompose an EMF into its component parts to allow editing the individual objects, they often get it wrong due to the complexity of the metafile format. Some applications that decompose an EMF correctly are confused by the dual EMF. For such programs, you need to export a plain EMF by setting the graphics technology to GDI.

BMP Format

BMP is a Windows bitmap format. It is accepted by a wide variety of programs but requires a lot of memory and disk space because it is not compressed. A BMP is also known as a DIB (device-independent bitmap).

If the program to which you are exporting supports PNG then PNG is a better choice.

PDF Format

PDF (Portable Document Format) is Adobe’s platform-independent vector graphics format. However, not all programs can import PDF. In fact, although Igor can export PDF, it can not itself import PDF on Windows.

If a window contains drawings imported into Igor from other programs, they are rendered in the PDF as opaque bitmap images.

PDF files normally use the RGB encoding to represent color but you can also use CMYK. See **Exporting Colors** on page III-99 for details.

PDF export can and should embed TrueType fonts. See **Font Embedding** on page III-99 and **Symbols with EPS and Igor PDF** on page III-438 for details.

Igor PDF does not support transparency.

Blurry Images in PDF

When Igor exports an image plot, it exports the image as a single image object when possible. However, some PDF viewers, most notably Apple's, take it upon themselves to blur the pixels. To get around this, you can tell Igor to draw image pixels as individual rectangles using the **ModifyImage** interpolate keyword with a value of -1. You should do this only when necessary as the resulting PDF will be much larger.

Encapsulated PostScript (EPS) Format

Encapsulated PostScript was a widely-used, platform-independent vector graphics format consisting of PostScript commands in plain text form. It usually gives the best quality, but it works only when printed to a PostScript printer or exported to a PostScript-savvy program such as Adobe Illustrator. You should use only PostScript fonts (e.g., Helvetica).

Encapsulated PostScript was a widely-used platform-independent vector graphics format consisting of PostScript commands in plain text form. EPS is largely obsolete but still in use. It usually gives good quality, but it works only when printed to a PostScript printer or exported to a PostScript-savvy program such as Adobe Illustrator. You should use only PostScript fonts such as Helvetica. EPS does not support transparency.

Prior to Igor Pro 7, Igor embedded a screen preview in EPS files. This is no longer done because the preview was not cross-platform and caused problems with many programs.

EPS files normally use the RGB encoding to represent color but you can also use CMYK. See **Exporting Colors** on page III-99 for details.

Igor Pro exports EPS files using PostScript language level 2. This allows much better fill patterns when printing and also allows Adobe Illustrator to properly import Igor’s fill patterns. For backwards compatibility with old printers, you can force Igor to use level 1 by specifying /PLL=1 with the SavePICT operation.

If the graph or page layout that you are exporting as EPS contains a non-EPS picture imported from another program, Igor exports the picture as an image incorporated in the output EPS file.

Igor can embed TrueType fonts as outlines. See **Font Embedding** on page III-99 and **Symbols with EPS and Igor PDF** on page III-438 for details.

SVG Format

SVG (Scalable Vector Graphics) is an XML-based platform-independent 2D vector and raster graphics format developed by the World Wide Web Consortium. It is often used for displaying graphics in web pages and is a good choice for other uses if the destination program supports it.

Platform-Independent Bitmap Formats

PNG (Portable Network Graphics) is a platform-independent bitmap format. It uses lossless compression and supports high resolution. It is a superior alternative to JPEG or GIF. Although Igor can export and import PNG images via files and via the clipboard, some programs that allow you to insert PNG files do not allow you to paste PNG images from the clipboard.

JPEG is a lossy format whose main virtue is that it is accepted by all web browsers. However, all modern web browsers support PNG so there is little reason to use JPEG. Although Igor can export and import JPEG via the clipboard, not all programs can paste JPEGs.

TIFF is an Adobe format often used for digital photographs. Igor's implementation of TIFF export does not use compression. TIFF files normally use the RGB scheme to specify color but you can also use CMYK. See **Exporting Colors** on page III-99 for details. There is no particular reason to use TIFF over PNG unless you are exporting to a program that does not support PNG. Igor can export and import TIFF via files and via the clipboard and most graphics programs can import TIFF.

Choosing a Graphics Format

Because of the wide variety of types of graphics, destination programs, printer capabilities, operating system behaviors and user-priorities, it is not possible to give definitive guidance on choosing an export format. But here is an approach that will work in most situations.

If the destination program accepts PDF or SVG, then they are probably your best choice because of their high-quality vector graphics and platform-independence.

Encapsulated PostScript (EPS) is also a very high quality format which works well if the destination program supports it.

If SVG, PDF and EPS are not appropriate, your next choice would be a high-resolution bitmap. The PNG format is preferred because it is platform-independent and is compressed. If the application to which you are exporting does not support PNG, your next choice would be TIFF or JPEG.

Exporting Graphics Via the Clipboard

To export a graphic from the active window via the clipboard, choose Edit→Export Graphics. This displays the Export Graphics dialog.

When you click the OK button, Igor copies the graphics for the active window to the clipboard. You can then switch to another program and do a paste.

When a graph, page layout, or Gizmo plot is active and in operate mode, choosing Edit→Copy copies to the clipboard whatever format was last used in the Export Graphics dialog. For a table, Edit→Copy copies the selected numbers to the clipboard and does not copy graphics.

When a page layout has an object selected or when the marquee is active, choosing Edit→Copy copies an Igor object in a format used internally by Igor along with an enhanced metafile and does not use the format from the Export Graphics dialog.

Although Igor can export a number of different formats, not all programs can recognize them on the clipboard. You may need to export via a file.

Exporting Graphics Via a File

To export a graphic from the active window via a file, choose File→Save Graphics. This displays the Save Graphics File.

The controls in the Format area of the dialog change to reflect options appropriate to each export format.

When you click the Do It button, Igor writes a graphics file. You can then switch to another program and import the file.

If you select _Use Dialog_ from the Path pop-up menu, Igor presents a Save File dialog in which you can specify the name and location of the saved file.

Exporting a Section of a Layout

To export a section of a page layout, use the marquee tool to identify the section and then choose Edit→Export Graphics or File→Save Graphics.

If you don't use the marquee and the Crop to Page Contents checkbox is checked, Igor exports the area of the layout that is in use plus a small margin. If it is unchecked, Igor exports the entire page.

Exporting Colors

The EPS and TIFF graphics formats normally use the RGB scheme to specify color. Some publications require the use of CMYK instead of RGB, although the best results are obtained if the publisher does the RGB to CMYK conversion using the actual characteristics of the output device. For those publications that insist on CMYK, you can use the SavePICT /C=2 flag.

Font Embedding

You can embed TrueType fonts in EPS files and in PDF Files. This means that you can print EPS or PDF files on systems lacking the equivalent PostScript fonts. This also helps with publications that require embedded fonts.

In Igor Pro 7 font embedding is always on and the only option is to not embed standard fonts. For most purposes, embedding only non-standard fonts is the best choice.

Igor embeds TrueType fonts as synthetic PostScript Type 3 fonts derived from the TrueType font outlines. Only the actual characters used are included in the fonts.

Not all fonts and font styles on your system can be embedded. Some fonts may not allow embedding and others may not be TrueType or may give errors. Be sure to test your EPS files on a local printer or by importing into Adobe Illustrator before sending them to your publisher. You can also use the "TrueType Outlines.pxp" example experiment to validate fonts for embedding. You will find this experiment file in your Igor Pro 7 Folder in the "Examples:Testing:" folder.

For EPS, Igor determines if a font is non-standard by attempting to look up the font name in a table described in **PostScript Font Names** on page III-100 after doing any font substitution using that table. In addition, if a nonplain font style name is the same as the plain font name, then embedding is done. This means that standard PostScript fonts that do not come in italic versions (such as Symbol), will be embedded for the italic case but not for the plain case.

For PDF, non-standard fonts are those other than the basic fonts guaranteed by the PDF specification to be built-in to any PDF reader. Those fonts are Helvetica and TImes in plain, bold, italic, and bold-italic styles as well as plain versions of Symbol and Zapf Dingbats. If embedding is not used or if a font can not be embedded, fonts other than those just listed will be rendered as Helvetica and will not give the desired results.

PostScript Font Names

When generating PostScript, Igor needs to generate proper PostScript font names. This presents problems under Windows. Igor also needs to be able to substitute PostScript fonts for non-PostScript fonts. Here is a list of font names that are translated into the standards:

TrueType Name	PostScript Name
Arial	Helvetica
Arial Narrow	Helvetica-Narrow
Book Antiqua	Palatino
Bookman Old Style	Bookman
Century Gothic	AvantGarde
Century Schoolbook	NewCenturySchlbk
Courier New	Courier
Monotype Corsiva	ZapfChancery
Monotype Sorts	ZapfDingbats
Symbol	Symbol
Times New Roman	Times