Epspdf and epspdftk User Manual

A cross-platform GUI- and command-line converter for [e]ps and pdf

Siep Kroonenberg (`n dot s dot kroonenberg at rug dot nl`)

This manual is for epspdf, version 0.6.0.

Table of Contents

What has changed?

Improvements:

- Grayscaling should now also work on bitmapped images.
- Revised multistep conversions, with less risk of losing high-level structure
- Rewritten in TeXLua, which is a component of both TeX Live and of MikTeX. Command-line use no longer requires an external scripting language.

I have been testing with Ghostscript version 9.05 and 9.06.

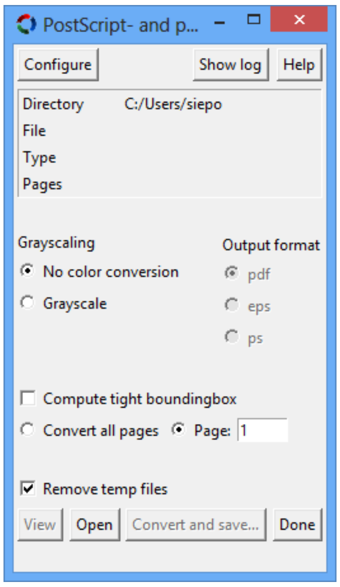
For more details, see Section 6.1 [Version 0.6], page 6.

1 Usage

1.1 Overview

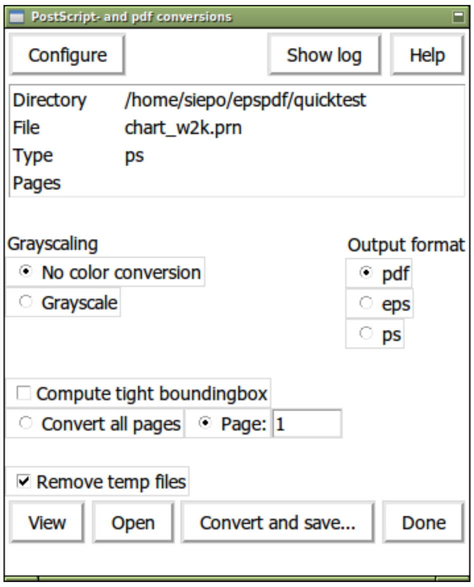
Epspdftk converts files between eps, pdf and general PostScript. The main screen lets you open a file, select some options and convert the file.

When opening a file, epspdf tries to find out file type and, in the case of a pdf file, the number of pages. This information is displayed in the box in the upper part of the screen.



Directory C:/Users/siepo
File
Type
Pages

Grayscaling: [✓ No color conversion] Grayscale
Output format: [✓ pdf] eps ps
[ ] Compute tight boundingbox
Convert all pages [✓ Page: 1]
[✓] Remove temp files



Directory /home/siepo/epspdf/quicktest
File chart_w2k.prn
Type ps
Pages

Grayscaling: [✓ No color conversion] Grayscale
Output format: [✓ pdf] eps ps
[ ] Compute tight boundingbox
Convert all pages [✓ Page: 1]
[✓] Remove temp files

1.2 Viewing

The View button at the bottom of the screen calls an external viewer.

Windows and OS X: Epspdftk simply tries to use the default Open command.

Under Linux the PostScript- and pdf viewers are configurable; see Section 1.5 [The configuration screen], page 2.

The View button is grayed if epspdf thinks that there is no previewer for the current file.

1.3 Conversion options

Grayscaling: Grayscaling is now done by Ghostscript itself and should "just work".

Compute tight boundingbox: This option is only available if a single page is converted.

Page selection: The only possibilities are selecting a single page or selecting all pages. Converting to eps implies selecting a single page.

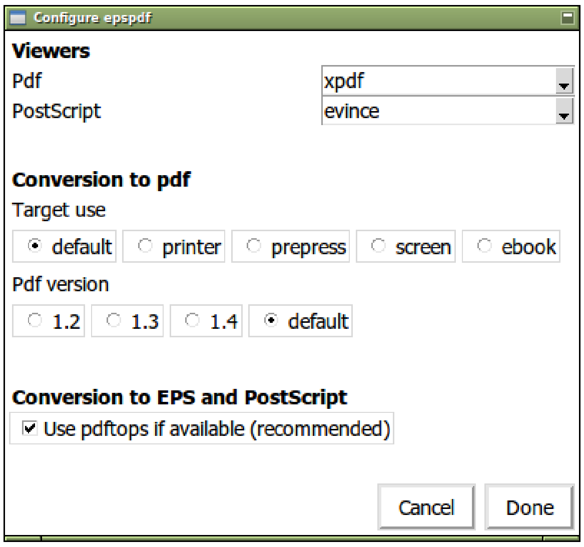
For general PostScript files, there is no quick way to determine the number of pages, so the program may not check beforehand whether you picked an existing page. If you don't like that, convert the entire document to pdf first – which will be done behind the scenes anyhow.

Specifying options such as grayscaling or page selection may require a multistep conversion.

1.4 Converting

The Convert and save... button calls up a file save dialog. After a successful conversion, the result becomes the new current file, so you can judge the result by pressing the View button again.

1.5 The configuration screen



Viewers

Pdf: xpdf

PostScript: evince

Conversion to pdf

Target use: [✓ default] printer prepress screen ebook

Pdf version: 1.2 1.3 1.4 [✓ default]

Conversion to EPS and PostScript

[✓ Use pdftops if available (recommended)]

Epspdftk also has a configuration screen. The settings here are preserved between sessions.

1.5.1 Configuring viewers

Under Unix, the preferred PostScript- and pdf viewers can be configured in this screen. Epspdf looks for a number of PostScript- and pdf viewers, from which you can select one, but you can also enter one manually.

For Windows and Mac OS X, there is no such configuration option. Epspdf will use the program associated with the file type, which can be configured outside epspdf.

1.5.2 Options for converting to pdf

Double-check the setting "Target use" under "Conversion to pdf". "prepress" is for pdfs which are going to be printed commercially. The options prepress, printer and default will try to embed all fonts. Often, printshops insist on this. With luck, Ghostscript fonts and system fonts will be used for fonts which were not originally embedded.

On the other hand, you may prefer "screen" if file size is a concern. See also the Ghostscript documentation, in particular Use.htm and Ps2pdf.htm.

Specifying anything other than "default" may cause an additional conversion step and possibly also loss of high-level structure.

As to pdf versions: this is a trade-off between more features for higher versions and better compatibility for lower versions – although even version 1.4 is far from bleeding-edge. Converting to a lower version may lead to rasterization of vector data and (much) larger file sizes. For prepress use, the printshop or publisher may require a low version number but otherwise leave this at “default”.

1.5.3 Options for converting to PostScript

For conversion the other way you may opt *not* to use pdftops even if it is available. In this case, the conversion will be done by Ghostscript. Characters may be converted to drawn shapes or little bitmaps so this is not recommended if the file uses fonts.

Under Windows and MikTeX, this is also the place to point the program to the location of pdftops.exe.

2 Command-line usage

epspdf.tlu is the backend of epspdftk, but it can also be used standalone. It shares configuration settings with epspdftk.tcl.

Below, we assume that there is a suitable wrapper or symlink for epspdf on your searchpath. This is the case if you installed epspdf as a TeX Live package.

The first parameter of epspdftk is interpreted as startup directory for the file browser.

Basic usage:

```
epspdf [options] infile outfile
```

2.1 Option summary

Typing *epspdf --help* gives you the following summary:

```
$ epspdf --help
Epspdf version 0.6.0
Copyright (c) 2006, 2008, 2009, 2010, 2011, 2013 Siep Kroonenberg

Convert between [e]ps and pdf formats
Usage: epspdf[.tlu] [options] infile [outfile]
Default for outfile is file.pdf if infile is file.eps or file.ps
Default for outfile is file.eps if infile is file.pdf

-p, --page, --pagenumber PNUM
            Page number; must be a positive integer
-g, --grey, --gray, -G, --GREY, --GRAY
            Convert to grayscale
-b, --bbox, --BoundingBox
            Compute tight boundingbox
-T, --target TARGET
            One of screen, ebook, printer, prepress or default
-N, --pdfversion VERSION
            One of 1.2, 1.3, 1.4 or default
-U          Use pdftops if available
-I          Reverses the above
-s, --save  Save some settings to configuration file
-d          Debug: do not remove temp files
-v, --version
            Display version info and exit
-h, --help  Display this help message and exit
```

3 Notes on PostScript and pdf

3.1 Bitmapped and vector

Pictures can be described either in terms of pixels, or more abstractly, in terms of geometric shapes, fonts and text.

Bitmapped or pixel-based graphics are appropriate for photographs and screenshots, but less so for diagrams and spreadsheet-generated graphics.

A file in PostScript- or pdf format can contain both types of graphic data.

Vector graphics can be freely scaled without losing sharpness or becoming pixellated. If bitmapped graphics are enlarged too much, individual pixels become apparent. With low-resolution bitmaps this happens sooner than with high-resolution bitmaps, but high-resolution bitmaps have (much) larger file sizes, and take longer to process.

So avoid converting vector to bitmap. However, converting from bitmap to vector is also best avoided, since it is very hard to do well.

Epspdf normally avoids conversion from vector to bitmap and never converts the other way. With the screen- and ebook "Target use" option, included bitmaps tend to be downsampled, i.e. reduced to a lower resolution.

3.2 Font embedding

When converting to pdf, Ghostscript handles font embedding differently depending on the "Target use" option. According to the Ghostscript documentation, it embeds all fonts without exception for prepress- and printer settings. Epspdf enforces this also for default settings. For screen- and ebook settings, standard fonts such as Times may be omitted.

3.3 Eps preview headers

Preview headers are quietly stripped from eps files. These preview headers are used by e.g. desktop-publishing software to represent eps files on screen without having to interpret the PostScript code itself. Epspdf currently has no option to preserve or add them.

3.4 Bounding- and other boxes

A PostScript file may have a page size and a boundingbox defined. A pdf file may have a mediabox, a trimbox and various other boxes. Ghostscript by itself normally converts the PostScript page – which is anchored at (0,0) – to the pdf mediabox.

Conversion from eps to pdf usually translates the bottom-left corner to the (0,0) origin and sets the page to the size of the graphic. Anything outside the boundingbox should be cut off.

I have seen Illustrator-generated pdf files with negative coordinates for the lower-left corner. However, this does not seem to cause problems.

3.5 Orientation

There appears to be no reliable way to set orientation for PostScript- and pdf files. This may result part of a graphic being cut off after conversion or even everything falling outside the page / mediabox / boundingbox.

3.6 Exporting PostScript or pdf from Windows programs

Microsoft Office 2010 can now export to pdf, either the entire document or a selection. This removes a major headache for Windows users.

As a last resort, you can "print" to a PostScript file. From some programs, you can print a selection. A suitable driver which comes with Windows is Generic / MS Publisher Imagesetter. Pay attention to the printer properties: choose "Outline" for font downloading and avoid the "Optimize for speed" setting for PostScript Output Option. In my tests, "Encapsulated PostScript" did not look very promising either. Try e.g. "Archive" instead. These options can be found under the Advanced button.

4 Bitmapped graphics for LaTeX and pdflatex

pdflatex can use graphics in .png format (best for screenshots) and .jpg format (best for photographs) directly. However, for LaTeX you are stuck with .eps format. Tips for converting to .eps:

4.1 Linux

sam2p This command-line bitmap-to-PostScript/pdf conversion utility is available from http://code.google.com/p/sam2p/ and may already be packaged for your distribution. It produces very small files: '`sam2p image.png image.eps`'

ImageMagick/convert
`convert` from the ImageMagick package is a command-line utility: '`convert image.png image.eps`'

The GIMP This is the premier open source image editing program. It is often pre-installed on Linux, and is also available for other platforms. The GIMP can save in eps- and pdf format.

4.2 Mac OS X

Mac OS X's built-in Preview application can read most bitmapped formats and save them as pdf or PostScript. In fact, in many cases it is an excellent alternative to epspdf.

4.3 Windows

Windows is not a particularly friendly environment for PostScript and pdf. A couple of command-line utilities try to fill the gap, *e.g.*

```
sam2p image.png image.eps
```

or

```
bmeps -c image.png image.eps
```

sam2p is distributed with TeX Live (Windows only), bmeps both with TeX Live and with MikTeX. Without the `-c` option, bmeps produces a grayscale image. It produces larger files than sam2p.

With TeX Live, you can convert to eps by right-clicking an image in Windows Explorer and "open" with *bitmap2eps*, which uses sam2p or bmeps in the background.

5 Troubleshooting

5.1 Part of the graphic gets cut off

If the PostScript file was generated with the Windows PostScript driver, experiment with the PostScript Output option. Don't choose Optimize for Speed.

5.2 Fonts look ugly

If Ghostscript has to do the conversion from pdf to ps then text may not remain text, but may be replaced by bitmaps. Try to get hold of a more recent Ghostscript and of either the xpdf suite or the Poppler utilities, both of which include pdftops.

5.3 The page has been converted to a bitmap

This may happen if the page contains features such as transparency which are not supported by intermediate formats.

Set pdf target use and target version both to "default" to avoid unnecessary conversions: `-T default -N default`

5.4 Resources for troubleshooting

Logfile. The GUI has a button for viewing log output. This same output is also written to a file `epspdf.log`. For Linux/Unix/Mac OS X this is in a subdirectory .epspdf of your home directory; for Windows it is in a subdirectory epspdf of `%APPDATA%`. This APPDATA directory may be `c:\Documents and Settings\`*`your user name`*`\Application Data` or `c:\Users\`*`your user name`*`\AppData\Roaming`. Otherwise, open a command prompt and type *`echo %APPDATA%`*, which will produce this information.

The logfile lists all epspdf calls and all Ghostscript- and pdftops calls plus error information.

Temporary files. The temporary files may give clues as well. Uncheck the button "Remove temp files", or for the command-line version, give a -d parameter to keep the temporary files. Check the log(file) as to which temporary files have been created.

Ghostscript- and pdftops documentation. For Ghostscript, the most important files are Ps2pdf.htm and Use.htm. For pdftops, type *`pdftops -h`*. For Unix, there is also a man page, and for Windows there is a file pdftops.txt in the distribution zip.

6 Change history

6.1 Version 0.6

The command-line backend component has been rewritten in texlua and therefore no longer needs an external scripting language.

Grayscaling is now done by Ghostscript's color options for pdf output. This also works for bitmaps.

Croppping of pdfs is now accomplished by running luatex on a suitable wrapper file (same method as Heiko Oberdiek's pdfcrop). Such a conversion preserves advanced features which might otherwise get lost during a PostScript round-trip.

I no longer own a Mac, and therefore no longer try to provide an AppleScript wrapper.

The current version has no provisions for custom Ghostscript- or pdftops parameters. I may of may not re-add those in a future update. The corresponding command-line options are accepted but have no effect.

6.2 Version 0.5

The GUI has been rewritten in Tcl/Tk, removing the dependence on the Ruby/Tk interface library.

The Windows installer now installs a small Ruby subset and the standard epspdf distribution, but with epspdftk.tcl replaced with a starpack: a single executable containing `epspdftk.tcl` and a Tcl/Tk runtime. See `http://wiki.tcl.tk/52`.

Epspdf now uses its own subdirectory for both the logfile and the configuration file. For Linux/Unix/Mac OS X this is `$HOME/.epspdf`, for Windows it is `%APPDATA%\epspdf`. On all supported platforms, settings are stored in the file `config` in this directory. Under Windows, the registry is no longer used for this.

A button has been added to view log output.

There is a second new button "Remove temp files", which is normally checked, causing temporary files to be deleted after each conversion. Unchecking this button may be useful for troubleshooting. In previous versions, temporary files were deleted at the end of the entire epspdftk session but that has become less practical now that the GUI and epspdf itself have become two separate programs.

The "Open with..." option for Windows has been dropped for technical reasons.

The "–version" option now prints the version string instead of setting the desired pdf version.

The "–info" option now also prints the number of pages for pdf files.

6.3 Version 0.4

Hi-res boundingboxes are now supported. By default, conversion from eps to pdf now uses the hires boundingbox as "page" to determine the page dimensions of the pdf file. Other conversions preserve or generate a hires boundingbox.

Under Windows, the new version looks for an installed TEX and will use its private Ghostscript if it cannot find a separately installed Ghostscript. TEX Live's pdftops, being on the searchpath, will be used unless epspdf finds another copy first.

There is now a "-v" (lowercase) option to print the version string.