

**NAME**

epstopdf, repstopdf – convert an EPS file to PDF

**SYNOPSIS**

**epstopdf** [*options*] [*epsfile*]

**DESCRIPTION**

By default, **epstopdf** converts the input PostScript file to PDF, using Ghostscript.

**Epstopdf** transforms the Encapsulated PostScript file *epsfile* (or standard input) so that it is guaranteed to start at the 0,0 coordinate, and it sets a page size exactly corresponding to the BoundingBox. Thus, the result needs no cropping, and the PDF MediaBox is correct.

PJL commands at the start of a file are removed. DOS EPS binary files (TN 5002) are supported.

If the bounding box in the input is incorrect, of course there will be resulting problems.

**OPTIONS**

Options may start with either "-" or "--", and may be unambiguously abbreviated. It is best to use the full option name in scripts, though, to avoid possible collisions with new options in the future.

General script options:

**--help** display help message and exit

**--version**  
display version information and exit

**--outfile=*file***  
write result to *file*. If this option is not given, and **--nogs** or **--filter** is specified, write to standard output; otherwise, the default is to construct the output file name by replacing any extension in the input file with '.pdf'.

**--[no]debug**  
write debugging info (default: off).

**--[no]exact**  
scan ExactBoundingBox (default: off).

**--[no]filter**  
read standard input and (unless **--outfile** is given) write standard output (default: off).

**--[no]gs**  
run Ghostscript (default: on). With **--nogs**, output (to standard output by default) the PostScript that would normally be converted; that is, the input PostScript as modified by **epstopdf**.

**--[no]hires**  
scan HiresBoundingBox (default: off).

**--restricted=*val***  
turn on restricted mode (default: [on for repstopdf, else off]); this forbids the use of **--gscmd** and **--gsopt** and imposes restrictions on the input and output file names according to the values of `openin_any` and `openout_any` (see the Web2c manual, <http://tug.org/web2c>).

Options for Ghostscript (more info below):

- gscmd=val**  
pipe output to *val* (default: **gswin32c** on Windows, else **gs**)
- gsopt=val**  
include *val* as one argument in the **gs** command (can be repeated),
- gsopts=val**  
split *val* at whitespace, include each resulting word as an argument in the **gs** command (can be repeated)
- autorotate=val**  
set `AutoRotatePages` (default: `None`); recognized *val* choices: `None`, `All`, `PageByPage`. For EPS files, `PageByPage` is equivalent to `All`.
- [no]compress**  
use compression in the output (default: on).
- device=dev**  
use `-sDEVICE=dev` (default: `pdfwrite`); not all devices are allowed in restricted mode.
- [no]embed**  
embed fonts (default: on).
- pdfsettings=val**  
use `-dPDFSETTINGS=/val` (default is 'prepress' if **--embed**, else empty); recognized *val* choices: `screen`, `ebook`, `printer`, `prepress`, `default`.
- [no]quiet**  
use `-q`, a.k.a. `-dQUIET` (default: off).
- res=dpi, dpixdpi**  
set image resolution (default: [use **gs** default]); ignored if **--debug** is set.
- [no]safer**  
use `-d(NO)QUIET` (default: true).

In addition to the specific options above, additional options to be used with **gs** can be specified with either or both of the two cumulative options **--gsopts** and **--gsopt**.

**--gsopts** takes a single string of options, which is split at whitespace, each resulting word then added to the **gs** command line individually.

**--gsopt** adds its argument as a single option to the **gs** command line. It can be used multiple times to specify options separately, and is necessary if an option or its value contains whitespace.

In restricted mode, options are limited to those with names and values known to be safe; some options taking booleans, integers or fixed names are allowed, those taking general strings are not.

## EXAMPLES

Examples all equivalently converting 'test.eps' to 'test.pdf':

```
epstopdf test.eps
cat test.eps | epstopdf --filter >test.pdf
cat test.eps | epstopdf -f -o=test.pdf
```

Example for using `HiResBoundingBox` instead of `BoundingBox`:

```
epstopdf --hires test.eps
```

Example for producing **epstopdf**'s attempt at corrected PostScript:

```
$program --nogs test.ps >testcorr.ps
```

In all cases, you can add **--debug (-d)** to see more about what **epstopdf** is doing.

## BUGS

The case of "%BoundingBox: (atend)" when input is not seekable (e.g., from a pipe) is not supported.

Report bugs in the program or this man page to [tex-k@tug.org](mailto:tex-k@tug.org). When reporting bugs, please include an input file and the command line options specified, so the problem can be reproduced.

## SEE ALSO

**gs(1)**, **pdfcrop(1)**.

The **epstopdf** LaTeX package, part of the oberdiek bundle, which automates running this script on the fly under TeX: <http://ctan.org/pkg/epstopdf-pkg>.

## AUTHOR

Originally written by Sebastian Rahtz, for Elsevier Science, with subsequent contributions from Thomas Esser, Gerben Wierda, Heiko Oberdiek, and many others. Currently maintained by Karl Berry.

Man page originally written by Jim Van Zandt.

**epstopdf** home page: <http://tug.org/epstopdf>.

You may freely use, modify and/or distribute this file.