

The `epstopdf` package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/04/26 v1.4

Abstract

This packages adds support of handling eps images to package `graphics` or `graphicx` with option `pdftex`. If an eps image is detected, `epstopdf` is automatically called to convert it to pdf format.

Contents

1	Documentation	1
1.1	Requirements	1
2	Usage	2
2.1	Configuration	2
2.2	Other image formats	2
3	Implementation	3
3.1	Relead check and identification	3
3.2	Help macros for miniltx mode	4
3.3	Checks	4
3.4	Adding conversion support	5
4	Installation	5
4.1	Download	5
4.2	Bundle installation	6
4.3	Package installation	6
4.4	Refresh file name databases	6
4.5	Some details for the interested	6
5	History	7
	[2001/01/06 v1.0]	7
	[2001/02/04 v1.1]	7
	[2006/02/20 v1.2]	7
	[2006/08/26 v1.3]	7
	[2007/04/26 v1.4]	7
6	Index	7

1 Documentation

1.1 Requirements

- The program `epstopdf` for the conversion from EPS to PDF. However, other programs can be used and configured by `\DeclareGraphicsRule`.

- The feature `\write18` must be enabled. This allows the running of external programs during \TeX 's compile run. Keep in mind that this is a security risk. The feature is an addition to \TeX . Mik \TeX , $\text{te}\TeX$, \TeX Live support it. In Web2C based \TeX distributions ($\text{te}\TeX$, \TeX Live) it can be enabled in the configuration file `texmf.cnf`:

```
shell_escape = 1
```

Because of the security risk, it is better to do it on the command line only:

```
-shell-escape ( $\text{te}\TeX$ ,  $\TeX$  Live)
--enable-write18 (Mik $\TeX$ )
```

Example:

```
pdflatex -shell-escape test.tex
```

2 Usage

The package is loaded after `graphic{s,x}`, e.g.:

```
\usepackage[pdftex]{graphicx}
\usepackage{epstopdf}
```

Now images with extension ‘.eps’ are detected and supported:

- Implicitly: `\includegraphics{bild}`
If `bild.eps` is present, but `bild.pdf` is missed, then `bild.eps` is converted to `bild.pdf` that is then included by $\text{pdf}\TeX$. On the next occurrences or on the next $\text{pdf}\TeX$ run, the PDF file is already available, so the conversion step is skipped.
- Explicitly: `\includegraphics{bild.eps}`
Each time the conversion program is called.

2.1 Configuration

Package `epstopdf` implements the forth argument of `\DeclareGraphicsRule` for $\text{pdf}\TeX$. There conversion programs can be configured and package `epstopdf` declares:

```
\DeclareGraphicsRule{.eps}{pdf}{.pdf}{'epstopdf #1}
```

Additionally the extension `.eps` is added to the list of extensions that are recognized by package `graphics`, see `\GraphicsExtensions`.

Instead of `epstopdf` other programs can be used, for example `ghostscript`:

```
\DeclareGraphicsRule{.eps}{pdf}{.pdf}{'ps2pdf -dEPSCrop #1}
```

2.2 Other image formats

The support that package `epstopdf` implements is not limited to EPS files. Other image conversions can be declared. The following example shows it for GIF images under Unix with ImageMagick's `convert`:

```
\DeclareGraphicsRule{.gif}{png}{.png}{%
  'convert #1 'basename #1 .gif'.png%
}
```

Or without `basename`:

```

\makeatletter
\DeclareGraphicsRule{.gif}{png}{.png}{%
  'convert #1 \noexpand\Gin@base.png%
}
\makeatother

```

The file extension `.gif` can be added to the extension list that package `graphics` searches if the file extension is not given in `\includegraphics`. The list can be set by `\GraphicsExtensions`. The following example shows another way by adding `.gif` to the list:

```

\makeatletter
\g@addto@macro\Gin@extensions{.gif}
\makeatother

```

3 Implementation

1 (*package)

3.1 Relead check and identification

Reload check, especially if the package is not used with L^AT_EX.

```

2 \begingroup
3   \catcode44 12 % ,
4   \catcode45 12 % -
5   \catcode46 12 % .
6   \catcode58 12 % :
7   \catcode64 11 % @
8   \expandafter\let\expandafter\x\csname ver@epstopdf.sty\endcsname
9   \ifcase 0%
10    \ifx\x\relax % plain
11    \else
12      \ifx\x\empty % LaTeX
13      \else
14        1%
15      \fi
16    \fi
17  \else
18    \expandafter\ifx\csname PackageInfo\endcsname\relax
19      \def\x#1#2{%
20        \immediate\write-1{Package #1 Info: #2.}%
21      }%
22    \else
23      \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
24    \fi
25    \x{epstopdf}{The package is already loaded}%
26  \endgroup
27  \expandafter\endinput
28 \fi
29 \endgroup

```

Package identification:

```

30 \begingroup
31   \catcode40 12 % (
32   \catcode41 12 % )
33   \catcode44 12 % ,
34   \catcode45 12 % -
35   \catcode46 12 % .
36   \catcode47 12 % /
37   \catcode58 12 % :
38   \catcode64 11 % @
39   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
40     \def\x#1#2#3[#4]{\endgroup

```

```

41     \immediate\write-1{Package: #3 #4}%
42     \xdef#1{#4}%
43 }%
44 \else
45     \def\x#1#2[#3]{\endgroup
46     #2[#3]}%
47     \ifx#1\relax
48     \xdef#1{#3}%
49     \fi
50 }%
51 \fi
52 \expandafter\x\csname ver@epstopdf.sty\endcsname
53 \ProvidesPackage{epstopdf}%
54 [2007/04/26 v1.4 Conversion with epstopdf on the fly (H0)]

```

3.2 Help macros for miniltx mode

```

55 \@ifundefined{PackageWarning}{%
56   \def\PackageWarning#1#2{%
57     \begingroup
58     \newlinechar=10 %
59     \def\MessageBreak{%
60       ^^J(#1)\@spaces\@spaces\@spaces\@spaces
61     }%
62     \immediate\write16{^^JPackage #1 Warning: #2\on@line.^^J}%
63   \endgroup
64 }%
65 }{}
66 \@ifundefined{PackageWarningNoLine}{%
67   \def\PackageWarningNoLine#1#2{%
68     \PackageWarning{#1}{#2\@gobble}%
69   }%
70 }{}
71 \@ifundefined{on@line}{%
72   \def\on@line{ on input line \the\inputlineno}%
73 }{}
74 \@ifundefined{@spaces}{%
75   \def\@spaces{\space\space\space\space}%
76 }{}

```

3.3 Checks

Check, whether package graphics is loaded (also graphicx loads graphics). Because miniltx.tex does not know \@ifpackageloaded we test for \Gin@setfile instead.

```

77 \begingroup\expandafter\expandafter\expandafter\endgroup
78 \expandafter\ifx\csname Gin@setfile\endcsname\relax
79   \PackageWarningNoLine{epstopdf}{%
80     No graphics package \string'graphic{s,x}\string' found%
81   }%
82   \expandafter\endinput
83 \fi

```

Check, whether pdftex.def is loaded. \ver@pdftex.def is not available with miniltx.tex, thus we test for \Gin@driver.

```

84 \begingroup
85   \def\x{pdftex.def}%
86   \ifx\Gin@driver\x
87   \else
88     \PackageWarningNoLine{epstopdf}{%
89       Graphics driver file \string'pdftex.def\string' not found%
90     }%
91   \expandafter\endgroup\expandafter\endinput
92 \fi

```

```

93 \endgroup
    Check, whether the shell escape feature is enabled.
94 \begingroup
95 \expandafter\ifx\csname pdfshellescape\endcsname\relax
96 \else
97 \ifnum\pdfshellescape>0 %
98 \else
99 \PackageWarningNoLine{epstopdf}{%
100     Shell escape feature is not enabled%
101 }%
102 \fi
103 \fi
104 \endgroup

```

3.4 Adding conversion support

Patch `\Gin@setfile` to execute #3, if it contains a command.

```

105 \let\ETEorg@Gin@setfile\Gin@setfile
106 \def\Gin@setfile#1#2#3{%
107 \if'\@car #3\relax\@nil
108 \immediate\write18{\@cdr #3\@empty\@nil}%
109 \ETEorg@Gin@setfile{#1}{#2}{\Gin@base #2}%
110 \else
111 \ETEorg@Gin@setfile{#1}{#2}{#3}%
112 \fi
113 }

```

Adding .eps at the end of the list of extensions, defined by `\GraphicsExtensions`.

```

114 \ifundefined{Gin@extensions}{%
115 \def\Gin@extensions{.eps}%
116 }{%
117 \expandafter\ifx\expandafter\indent\Gin@extensions\indent
118 \def\Gin@extensions{.eps}%
119 \else
120 \expandafter\def\expandafter\Gin@extensions\expandafter{%
121 \Gin@extensions,.eps%
122 }%
123 \fi
124 }
    \DeclareGraphicsRule for .eps
125 \@namedef{Gin@rule@.eps}#1{{pdf}}{.pdf}{'epstopdf #1}}
126 \endpackage

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/epstopdf.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/epstopdf.pdf](#) Documentation.

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

TDS refers to the standard “A Directory Structure for \TeX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

¹[ftp://ftp.ctan.org/tex-archive/](http://ftp.ctan.org/tex-archive/)

4.2 Bundle installation

Unpacking. Unpack the `oberdiek-tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek-tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain-`TeX`:

```
tex epstopdf.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
epstopdf.sty → tex/latex/oberdiek/epstopdf.sty
epstopdf.pdf → doc/latex/oberdiek/epstopdf.pdf
epstopdf.dtx → source/latex/oberdiek/epstopdf.dtx
```

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

4.4 Refresh file name databases

If your `TeX` distribution (`teTeX`, `mikTeX`, ...) relies on file name databases, you must refresh these. For example, `teTeX` users run `texhash` or `mktexlsr`.

4.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk epstopdf.pdf unpack_files output .
```

Unpacking with \LaTeX . The `.dtx` chooses its action depending on the format:

plain-`TeX`: Run `docstrip` and extract the files.

\LaTeX : Generate the documentation.

If you insist on using \LaTeX for `docstrip` (really, `docstrip` does not need \LaTeX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{epstopdf.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfL^AT_EX:

```
pdflatex epstopdf.dtx
makeindex -s gind.ist epstopdf.idx
pdflatex epstopdf.dtx
makeindex -s gind.ist epstopdf.idx
pdflatex epstopdf.dtx
```

5 History

[2001/01/06 v1.0]

- First public version, published in the pdfT_EX mailing list.

[2001/02/04 v1.1]

- Minor documentation update.
- CTAN.

[2006/02/20 v1.2]

- DTX framework.
- Compatibility for `miniltx.tex`.

[2006/08/26 v1.3]

- Check for `\write18` if available and print a warning if the feature is not enabled.

[2007/04/26 v1.4]

- Documentation rewritten and extended.

6 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols		5, 6, 7, 31, 32, 33, 34, 35, 36, 37, 38
\@car	107	\csname 8, 18, 39, 52, 78, 95
\@cdr	108	
\@empty	108	E
\@gobble	68	\empty 12
\@ifundefined	55, 66, 71, 74, 114	\endcsname 8, 18, 39, 52, 78, 95
\@namedef	125	\endinput 27, 82, 91
\@nil	107, 108	\ETEorg@Gin@setfile 105, 109, 111
\@spaces	60, 75	G
C		\Gin@base 109
\catcode	3, 4,	\Gin@driver 86
		\Gin@extensions 115, 117, 118, 120, 121

\Gin@setfile	105, 106	P	
I		\PackageInfo	23
\if	107	\PackageWarning	56, 68
\ifcase	9	\PackageWarningNoLine .	67, 79, 88, 99
\ifnum	97	\pdfshellescape	97
\ifx ..	10, 12, 18, 39, 47, 78, 86, 95, 117	\ProvidesPackage	53
\immediate	20, 41, 62, 108	S	
\indent	117	\space	75
\inputlineno	72	T	
M		\the	72
\MessageBreak	59	W	
N		\write	20, 41, 62, 108
\newlinechar	58	X	
O		\x	8, 10, 12, 19, 23, 25, 40, 45, 52, 85, 86
\on@line	62, 72		