

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 (teTeX, TEX Live)
--enable-write18 (MiKTEX)
```

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   \catcode44 12 % ,
32   \catcode45 12 % -
33   \catcode46 12 % .
34   \catcode58 12 % :
35   \catcode64 11 % @
36   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
37     \def\x#1#2#3[#4]{\endgroup
38       \immediate\write-1{Package: #3 #4}%
39       \xdef#1{#4}%
40     }%

```

```

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

```

3.2 Help macros for miniltx mode

```

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

```

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.

```

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

```

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

```

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

```

Check, whether the shell escape feature is enabled.

```

91 \begingroup
92   \expandafter\ifx\csname pdfshellescape\endcsname\relax
93   \else
94     \ifnum\pdfshellescape>0 %
95     \else
96       \PackageWarningNoLine{epstopdf}{%
97         Shell escape feature is not enabled%
98       }%
99     \fi
100   \fi
101 \endgroup

```

3.4 Adding conversion support

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

```

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

```

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

```

111 \ifundefined{Gin@extensions}{%
112   \def\Gin@extensions{.eps}%
113 }{%
114   \expandafter\ifx\expandafter\indent\Gin@extensions\indent
115   \def\Gin@extensions{.eps}%
116   \else
117     \expandafter\def\expandafter\Gin@extensions\expandafter{%
118       \Gin@extensions,.eps%
119     }%
120   \fi
121 }
122 \DeclareGraphicsRule for .eps
123 \@namedef{Gin@rule@.eps}#1{{pdf}}{.pdf}{'epstopdf #1}}
124 \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](#)

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			
<code>\@car</code>	104	<code>\csname</code>	8, 18, 36, 49, 75, 92
<code>\@cdr</code>	105	E	
<code>\@empty</code>	105	<code>\empty</code>	12
<code>\@gobble</code>	65	<code>\endcsname</code>	8, 18, 36, 49, 75, 92
<code>\@ifundefined</code>	52, 63, 68, 71, 111	<code>\endinput</code>	27, 79, 88
<code>\@namedef</code>	122	<code>\ETEorg@Gin@setfile</code> ...	102, 106, 108
<code>\@nil</code>	104, 105	G	
<code>\@spaces</code>	57, 72	<code>\Gin@base</code>	106
C		<code>\Gin@driver</code>	83
<code>\catcode</code> 3, 4, 5, 6, 7, 31, 32, 33, 34, 35		<code>\Gin@extensions</code>	112, 114, 115, 117, 118
		<code>\Gin@setfile</code>	102, 103

I		P	
\if	104	\PackageInfo	23
\ifcase	9	\PackageWarning	53, 65
\ifnum	94	\PackageWarningNoLine	64, 76, 85, 96
\ifx	10, 12, 18, 36, 44, 75, 83, 92, 114	\pdfshellescape	94
\immediate	20, 38, 59, 105	\ProvidesPackage	50
\indent	114		
\inputlineno	69	S	
		\space	72
M		T	
\MessageBreak	56	\the	69
N		W	
\newlinechar	55	\write	20, 38, 59, 105
O		X	
\on@line	59, 69	\x	8, 10, 12, 19, 23, 25, 37, 42, 49, 82, 83