

# The **epstopdf** package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2006/08/26 v1.3

## 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

<b>1 Usage</b>	<b>1</b>
<b>2 Implementation</b>	<b>2</b>
2.1 Relead check and identification . . . . .	2
2.2 Help macros for miniltx mode . . . . .	2
2.3 Checks . . . . .	3
2.4 Adding conversion support . . . . .	3
<b>3 Installation</b>	<b>4</b>
3.1 Some details for the interested . . . . .	4
<b>4 History</b>	<b>5</b>
[2001/01/06 v1.0] . . . . .	5
[2001/02/04 v1.1] . . . . .	5
[2006/02/20 v1.2] . . . . .	5
[2006/08/26 v1.3] . . . . .	5
<b>5 Index</b>	<b>5</b>

## 1 Usage

Required:   \* The program ‘epstopdf’.  
          \* The feature ‘\write18’ has to be enabled to get  
            the conversion via the program epstopdf work:  
            \* command line option: -shell-escape  
              example: pdflatex -shell-escape test.tex  
            \* configuraton file ‘texmf.cnf’: shell\_escape = 1

Use:       The package is loaded after `graphic{s,x}`, eg:  
            `\usepackage[pdftex]{graphicx}`  
            `\usepackage{epstopdf}`  
Images with extension ‘.eps’ are now detected  
and supported:  
\* Implicitly: `\includegraphics{bild}`  
  If ‘bild.eps’ can only be found,  
  then it is converted to the file ‘bild.pdf’,  
  that will be used by pdfTeX.  
  On the next ocurrences or on the next pdfTeX 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 Implementation

```
1 (*package)
```

### 2.1 Reload check and identification

Reload check, especially if the package is not used with L<sup>A</sup>T<sub>E</sub>X.

```

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

```

Package identification:

```

25 \begingroup
26 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
27 \def\x#1#2#3[#4]{\endgroup
28 \immediate\write-1{Package: #3 #4}%
29 \xdef#1{#4}%
30 }%
31 \else
32 \def\x#1#2[#3]{\endgroup
33 #2[#3]}%
34 \ifx#1\relax
35 \xdef#1{#3}%
36 \fi
37 }%
38 \fi
39 \expandafter\x\csname ver@epstopdf.sty\endcsname
40 \ProvidesPackage{epstopdf}%
41 [2006/08/26 v1.3 Conversion with epstopdf on the fly (H0)]

```

### 2.2 Help macros for miniltx mode

```

42 \@ifundefined{PackageWarning}{%
43 \def\PackageWarning#1#2{%
44 \begingroup
45 \newlinechar=10 %
46 \def\MessageBreak{%
47 ^^J(#1)\@spaces\@spaces\@spaces\@spaces

```

```

48     }%
49     \immediate\write16{^^JPackage #1 Warning: #2\on@line.^^J}%
50   \endgroup
51 }%
52 }{}
53 \@ifundefined{PackageWarningNoLine}{%
54   \def\PackageWarningNoLine#1#2{%
55     \PackageWarning{#1}{#2\@gobble}%
56   }%
57 }{}
58 \@ifundefined{on@line}{%
59   \def\on@line{ on input line \the\inputlineno}%
60 }{}
61 \@ifundefined{@spaces}{%
62   \def@spaces{\space\space\space\space}%
63 }{}

```

## 2.3 Checks

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

```

64 \begingroup\expandafter\expandafter\expandafter\endgroup
65 \expandafter\ifx\csname Gin@setfile\endcsname\relax
66   \PackageWarningNoLine{epstopdf}{%
67     No graphics package \string'graphic{s,x}\string' found%
68   }%
69   \expandafter\endinput
70 \fi

```

Check, whether pdfTeX.def is loaded. `\ver@pdfTeX.def` is not available with minilTeX, thus we test for `\Gin@driver`.

```

71 \begingroup
72   \def\x{pdfTeX.def}%
73   \ifx\Gin@driver\x
74   \else
75     \PackageWarningNoLine{epstopdf}{%
76       Graphics driver file \string'pdfTeX.def\string' not found%
77     }
78   \expandafter\endgroup\expandafter\endinput
79 \fi
80 \endgroup

```

Check, whether the shell escape feature is enabled.

```

81 \begingroup
82   \expandafter\ifx\csname pdfshellescape\endcsname\relax
83   \else
84     \ifnum\pdfshellescape>0 %
85     \else
86       \PackageWarningNoLine{epstopdf}{%
87         Shell escape feature is not enabled%
88       }%
89     \fi
90   \fi
91 \endgroup

```

## 2.4 Adding conversion support

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

```

92 \let\ETEorg@Gin@setfile\Gin@setfile
93 \def\Gin@setfile#1#2#3{%
94   \if'\@car #3\relax\@nil
95     \immediate\write18{\@cdr #3\@empty\@nil}%
96     \ETEorg@Gin@setfile{#1}{#2}{\Gin@base #2}%

```

```

97 \else
98   \ETEorg@Gin@setfile{#1}{#2}{#3}%
99 \fi
100 }

```

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

```

101 \ifundefined{Gin@extensions}{%
102   \def\Gin@extensions{.eps}%
103 }{%
104   \expandafter\ifx\expandafter\indent\Gin@extensions\indent
105   \def\Gin@extensions{.eps}%
106 \else
107   \expandafter\def\expandafter\Gin@extensions\expandafter{%
108     \Gin@extensions,.eps%
109   }%
110 \fi
111 }

\DeclareGraphicsRule for .eps
112 \@namedef{Gin@rule@.eps}#1{{pdf}}{.pdf}{'epstopdf #1}}
113 \end{package}

```

## 3 Installation

**CTAN.** This package is available on CTAN<sup>1</sup>:

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

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

**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.

**Refresh file databases.** If your TeX distribution (TeX, MiKTeX, ...) rely on file databases, you must refresh these. For example, TeX users run `texhash` or `mktexlsr`.

### 3.1 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 .
```

---

<sup>1</sup><http://ftp.ctan.org/tex-archive/>

**Unpacking with L<sup>A</sup>T<sub>E</sub>X.** The .dtx chooses its action depending on the format:

**plain-T<sub>E</sub>X:** Run docstrip and extract the files.

**L<sup>A</sup>T<sub>E</sub>X:** Generate the documentation.

If you insist on using L<sup>A</sup>T<sub>E</sub>X for docstrip (really, docstrip does not need L<sup>A</sup>T<sub>E</sub>X), then inform the autodetect routine about your intension:

```
latex \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<sup>A</sup>T<sub>E</sub>X:

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

## 4 History

[2001/01/06 v1.0]

- First public version, published in the pdfT<sub>E</sub>X 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.

## 5 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		
	\@gobble	55
\@car	\@ifundefined	42, 53, 58, 61, 101
\@cdr	\@namedef	112
\@empty	\@nil	94, 95

\@spaces .....	47, 62	<b>M</b>	
		\MessageBreak .....	46
<b>C</b>		<b>N</b>	
\csname .....	3, 13, 26, 39, 65, 82	\newlinechar .....	45
<b>E</b>		<b>O</b>	
\empty .....	7	\on@line .....	49, 59
\endcsname .....	3, 13, 26, 39, 65, 82	<b>P</b>	
\endinput .....	22, 69, 78	\PackageInfo .....	18
\ETEorg@Gin@setfile .....	92, 96, 98	\PackageWarning .....	43, 55
<b>G</b>		\PackageWarningNoLine ..	54, 66, 75, 86
\Gin@base .....	96	\pdfshellescape .....	84
\Gin@driver .....	73	\ProvidesPackage .....	40
\Gin@extensions .....	102, 104, 105, 107, 108	<b>S</b>	
\Gin@setfile .....	92, 93	\space .....	62
<b>I</b>		<b>T</b>	
\if .....	94	\the .....	59
\ifcase .....	4	<b>W</b>	
\ifnum .....	84	\write .....	15, 28, 49, 95
\ifx ....	5, 7, 13, 26, 34, 65, 73, 82, 104	<b>X</b>	
\immediate .....	15, 28, 49, 95	\x ..	3, 5, 7, 14, 18, 20, 27, 32, 39, 72, 73
\indent .....	104		
\inputlineno .....	59		