

# The grffile package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2008/08/11 v1.8

## Abstract

The package extends the file name processing of package `graphics` to support a larger range of file names. For example, the file name may contain several dots. Or in case of pdfTeX in PDF mode the file name may contain spaces.

## Contents

<b>1</b>	<b>Usage</b>	<b>2</b>
1.1	Option <code>multidot</code>	2
1.2	Option <code>babel</code>	2
1.3	Option <code>extendedchars</code>	2
1.4	Option <code>encoding</code>	2
1.4.1	Option <code>inputencoding</code>	3
1.4.2	Option <code>filenameencoding</code>	3
1.4.3	Example	3
1.5	Option <code>space</code>	3
1.6	General use	4
1.7	Default settings	4
<b>2</b>	<b>Implementation</b>	<b>4</b>
2.1	Identification	4
2.2	Catcode stuff	4
2.3	Options	4
<b>3</b>	<b>Installation</b>	<b>8</b>
3.1	Download	8
3.2	Bundle installation	8
3.3	Package installation	9
3.4	Refresh file name databases	9
3.5	Some details for the interested	9
<b>4</b>	<b>References</b>	<b>10</b>
<b>5</b>	<b>History</b>	<b>10</b>
	[2004/07/18 v0.5]	10
	[2006/08/15 v1.0]	10
	[2006/08/17 v1.1]	10
	[2006/11/30 v1.2]	10
	[2007/04/11 v1.3]	10
	[2007/06/13 v1.4]	10
	[2007/08/16 v1.5]	10
	[2007/11/11 v1.6]	10
	[2007/11/24 v1.7]	10
	[2008/08/11 v1.8]	10

## 1 Usage

### 1.1 Option `multidot`

The file name parsing of package `graphics` is changed, in order to detect known extensions. This allows both the use of dots inside the base file name and extensions with several dots.

Assume there are two files in the current directory: `Hello.World.eps` and `Hello.World.pdf`. `\includegraphics{Hello.World}` will find `Hello.World.pdf` with driver `pdftex` or `Hello.World.eps` with driver `dvips`.

**Limitations:** Problem could occur on systems, which don't use the dot as extension delimiter. These systems needs an own `texsys.cfg` containing definitions for `\filename@parse`. The author could not test that, due to a missing example.

### 1.2 Option `babel`

This option allows the use of shorthand characters of package `babel` inside the `graphics` file name. Additionally the tilde '~' is supported. The option is turned on as default. (In version v1.1 or below of this package, the features of this option were part of option `extendedchars`.)

Example:

```
\usepackage[frenchb]{babel}
\usepackage{grffile}
Image: \includegraphics{C:/path/image}
```

### 1.3 Option `extendedchars`

If the input encoding is the same encoding as the encoding that is used for file names and the driver allows non-ascii characters. Without option `extendedchars` the 8-bit characters are expanded, if they are active characters. For example, see the  $\text{\LaTeX}$  package `inputenc`. However a file name is not input for  $\text{\LaTeX}$ . Therefore this option `extendedchars` removes the active status and the 8-bit characters are not expandable any more.

Example:

```
\usepackage[latin1]{inputenc}
\usepackage[extendedchars]{grffile}
\includegraphics{Bäckerstraße}
```

If the `draft` option of the `graphics` package is enabled, the file name is printed with the current font encoding for `\ttfamily`. Thus it is possible, that such characters are omitted or the wrong characters are displayed, if the font encoding is not the same as the file name encoding.

### 1.4 Option `encoding`

Consider the following scenario. Your file system is using UTF-8 as encoding for file names. But you use `latin1` as input encoding for your  $\text{\TeX}$  files, because some packages are not ready for multi-byte encodings (`listings`, ...).

Then this option `encoding` loads support for converting encodings by loading package `stringenc`. The option is not defined after the preamble, because  $\text{\LaTeX}$  limits package loading to the preamble.

File names are converted, if package `stringenc` is loaded and the encodings are known, see options `inputencoding` and `filenameencoding`.

### 1.4.1 Option inputencoding

Option `inputencoding` specifies the encoding of the file name in your  $\text{\TeX}$  input file.

Package `inputenx` and package `inputenc` since version 2006/02/22 v1.1a remember the name of the input encoding that is looked up by this package. Therefore option `inputencoding` is usually not mandatory.

### 1.4.2 Option filenameencoding

This is the encoding of the filename of your file system. This option is mandatory, file names are not converted without this option. The option is disabled, if the value is empty.

### 1.4.3 Example

Back to the scenario where the file system uses UTF-8 and the  $\text{\LaTeX}$  input files are encoded in `latin1`.

```
\usepackage[latin1]{inputenc}[2006/02/22]
% \usepackage[latin1]{inputenx}
\usepackage{graphicx}
\usepackage[encoding,filenameencoding=utf8]{grffile}
```

For older versions of package `inputenc` option `inputencoding` provides the necessary informations.

```
\usepackage[latin1]{inputenc}
\usepackage{graphicx}
\usepackage{grffile}
\grffilessetup{
  encoding,
  inputencoding=latin1,
  filenameencoding=utf8,
}
```

## 1.5 Option space

This option allows graphics file names that contain spaces if possible.

In general it is not possible to use space inside file names, because  $\text{\TeX}$  considers the space character as termination in its syntax for commands that expect a file name.

Regarding graphics inclusion with the package `graphics` file names are used in two or three contexts:

1. The basic `\special` statement or primitive command for graphics inclusion. The `\special` statements for drivers `dvips` or `dvipdfm` do not allow spaces. However  $\text{pdf}\text{\TeX}$ 's primitive `\pdfximage` uses curly braces to delimit the file name and allows spaces.
2. `\includegraphics` checks the existence of the file. Also it looks for the right extension if the extension is not given. If  $\text{pdf}\text{\TeX}$  1.30 is given, the file existence test can be rewritten using a new primitive that allows spaces. This works in both modes DVI and PDF.
3. Sometimes files are read as  $\text{\TeX}$  input files. For example, `.bb` files or MPS files.

If  $\text{pdf}\text{\TeX}$  1.30 or greater is used in PDF mode then the graphics file names may contain spaces except for MPS files. Therefore option `space` is only enabled by default, if the supported  $\text{pdf}\text{\TeX}$  in PDF mode is detected. You can enable the option manually, if you know, your DVI driver supports spaces in its `\special` syntax and if there is no need to read the image file as  $\text{\TeX}$  input file (third context).

## 1.6 General use

The options can be given at many places:

1. As package options:  
`\usepackage[<options>]{grffile}`
2. Setup command of package grffile:  
`\grffilesetup{<options>}`
3. The options are also available as options for package `graphicx`:  
`\setkeys{Gin}{<options>}`
4. If package `graphicx` is loaded the options can also be applied for a single image:  
`\includegraphics[<options>]{...}`

## 1.7 Default settings

<code>multidot</code>	<code>true</code>	
<code>babel</code>	<code>true</code>	
<code>extendedchars</code>	<code>false</code>	
<code>space</code>	<code>true</code>	if pdf $\TeX$ 1.30 or greater is used in PDF mode
	<code>false</code>	otherwise

## 2 Implementation

### 2.1 Identification

```
1 (*package)
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{grffile}%
4 [2008/08/11 v1.8 Extended file name support for graphics (HO)]%
```

### 2.2 Catcode stuff

```
5 \edef\grffile@RestoreCatcodes{%
6   \catcode'\noexpand\=\the\catcode'\=\relax
7   \catcode'\noexpand\:\the\catcode'\:\relax
8   \catcode'\noexpand\.\the\catcode'\.\relax
9   \catcode'\noexpand\''\the\catcode'\'\relax
10  \catcode'\noexpand\<\the\catcode'\<\relax
11  \catcode'\noexpand\>\the\catcode'\>\relax
12  \catcode'\noexpand*\the\catcode'\*\relax
13  \catcode'\noexpand^\the\catcode'\^\relax
14  \catcode'\noexpand~\the\catcode'\~\relax
15 }
16 \@makeother\=
17 \@makeother\:
18 \@makeother\.
19 \@makeother\'
20 \@makeother\<
21 \@makeother\>
22 \@makeother\*
23 \catcode'\^=7 %
24 \catcode'\~=active
```

### 2.3 Options

```
25 \RequirePackage{ifpdf}
26 \RequirePackage{kvoptions}[2006/08/17]
27 \SetupKeyvalOptions{%
28   family=Gin,%
29   prefix=grffile@%
```

```

30 }
31 \DeclareBoolOption[true]{multidot}
32 \DeclareBoolOption[true]{babel}
33 \DeclareBoolOption[false]{extendedchars}
34 \DeclareBoolOption{space}
35 \DeclareVoidOption{encoding}{%
36   \RequirePackage{stringenc}\relax
37 }
38 \DeclareStringOption{inputencoding}
39 \DeclareStringOption{filenameencoding}
40 \DeclareDefaultOption{%
41   \PassOptionsToPackage{CurrentOption}{graphics}%
42 }

```

Default setting for option space.

```

43 \RequirePackage{pdftexcmds}[2007/11/11]
44 \begingroup\expandafter\expandafter\expandafter\endgroup
45 \expandafter\ifx\csname pdf@filesize\endcsname\relax
46   \grffile@spacefalse
47   \let\grffile@space@disabled\@empty
48   \def\grffile@spacetrue{%
49     \PackageWarning{grffile}{%
50       Option 'space' is not available,\MessageBreak
51       because it needs pdfTeX >= 1.30%
52     }%
53   }%
54 \else
55   \ifpdf
56     \grffile@spacetrue
57   \else
58     \grffile@spacefalse
59   \fi
60 \fi
61 \ProcessKeyvalOptions*
62 \AtBeginDocument{%
63   \DisableKeyvalOption[package=grffile]{Gin}{encoding}%
64 }
65 \RequirePackage{graphics}

```

`\grffilesetup`

```

66 \newcommand*{\grffilesetup}{%
67   \setkeys{Gin}%
68 }

69 \let\grffile@org@Gininclude@graphics\Gininclude@graphics
70 \renewcommand*{\Gininclude@graphics}{%
71   \ifx\grffile@filenameencoding\@empty
72   \else
73     \ifx\grffile@inputencoding\@empty
74       \expandafter\ifx\csname inputencodingname\endcsname\relax
75         \expandafter\ifx\csname
76           CurrentInputEncodingOption\endcsname\relax
77         \else
78           \let\grffile@inputencoding\CurrentInputEncodingOption
79         \fi
80       \else
81         \let\grffile@inputencoding\inputencodingname
82       \fi
83     \fi
84     \ifx\grffile@inputencoding\@empty
85     \else
86       \grffile@extendedcharstrue
87     \fi

```

```

88 \fi
89 \ifnum0\ifgrffile@babel 1\fi\ifgrffile@extendedchars 1\fi>\z@
90 \begingroup

```

Support of babel's shorthand characters.

```

91 \ifgrffile@babel
92 \csname @safe@activetrue\endcsname

```

Support of active tilde.

```

93 \edef~{\string~}%

```

Support of characters controlled by package inputenc.

```

94 \fi
95 \ifgrffile@extendedchars
96 \grffile@inputenc@loop\^^A\^^H%
97 \grffile@inputenc@loop\^^K\^^K%
98 \grffile@inputenc@loop\^^N\^^_%
99 \grffile@inputenc@loop\^^?\^^ff%
100 \fi
101 \expandafter\grffile@extchar@Ginclude@graphics
102 \else
103 \expandafter\grffile@Ginclude@graphics
104 \fi
105 }
106 \def\grffile@extchar@Ginclude@graphics#1{%
107 \def\grffile@filename{#1}%
108 \ifx\grffile@inputencoding\@empty
109 \else
110 \ifx\grffile@filenameencoding\@empty
111 \else
112 \ifx\grffile@inputencoding\grffile@filenameencoding
113 \else
114 \expandafter\ifx\csname StringEncodingConvert\endcsname\relax
115 \PackageError{grffile}{%
116 Package 'stringenc' is not loaded,\MessageBreak
117 omitting file name conversion%
118 }\@ehc
119 \else
120 \StringEncodingConvert\grffile@temp\grffile@filename
121 \grffile@inputencoding\grffile@filenameencoding
122 \StringEncodingSuccessFailure{%
123 \let\grffile@filename\grffile@temp
124 }{%
125 \PackageError{grffile}{%
126 Filename conversion failed%
127 }\@ehc
128 }%
129 \fi
130 \fi
131 \fi
132 \fi
133 \edef\x{\endgroup
134 \noexpand\grffile@Ginclude@graphics{\grffile@filename}%
135 }%
136 \x
137 }
138 \def\grffile@inputenc@loop#1#2{%
139 \count@=#1\relax
140 \loop
141 \begingroup
142 \uccode'\~= \count@
143 \uppercase{%
144 \endgroup
145 \edef~{\string~}%

```

```

146     }%
147     \ifnum\count@<‘#2\relax
148         \advance\count@\@ne
149     \repeat
150 }

Support for option space
151 \def\grffile@space@getbase#1{%
152     \edef\grffile@tempa{%
153         \def\noexpand\@tempa####1#1\noexpand\@nil{%
154             \def\noexpand\Gin@base{####1}%
155         }%
156     }%
157     \grffile@ifFileExists{\filename@area\filename@base#1}{%
158         \grffile@tempa
159         \expandafter\@tempa\grffile@file@found\@nil
160         \edef\Gin@ext{#1}%
161     }{%
162     }%
163 }
164 \def\grffile@ifFileExists#1{%
165     \expandafter\expandafter\expandafter
166     \ifx\expandafter\expandafter\expandafter\\\pdf@filesize{#1}\\\%
167         \let\reserved@a\@secondoftwo
168         \ifx\input@path\@undefined
169             \else
170                 \expandafter\@tfor\expandafter\reserved@b\expandafter
171                 : \expandafter=\input@path\do{%
172                     \expandafter\expandafter\expandafter
173                     \ifx\expandafter\expandafter\expandafter
174                         \\\pdf@filesize{\reserved@b#1}\\\%
175                     \else
176                         \edef\grffile@file@found{\reserved@b#1}%
177                         \let\reserved@a\@firstoftwo
178                         \break@tfor
179                     \fi
180                 }%
181             \fi
182             \expandafter\reserved@a
183         \else
184             \edef\grffile@file@found{#1}%
185             \expandafter\@firstoftwo
186         \fi
187     }
188
189 \def\grffile@Gininclude@graphics#1{%
190     \begingroup
191         \ifgrffile@space
192             \let\Gin@getbase\grffile@space@getbase
193         \fi
194         \ifgrffile@multidot
195             \let\filename@base\@empty
196             \let\filename@simple\grffile@filename@simple
197         \fi
198         \grffile@org@Gininclude@graphics{#1}%
199     \endgroup
200 }%
201
202 \def\grffile@filename@simple#1.#2\\\{%
203     \ifx\\#2\\\%
204         \let\filename@ext\relax
205     \else
206         \expandafter\ifx\csname

```

```

207      Gin@rule@.\filename@dot #2\\endcsname\relax
208      \edef\filename@base{\filename@base #1.}%
209      \grffile@ReturnAfterFiFiBase{\grffile@filename@simple #2\\}%
210      \else
211      \edef\filename@ext{\filename@dot #2\\}%
212      \fi
213  \fi
214  \edef\filename@base{\filename@base #1}%
215 }
216 \def\grffile@ReturnAfterFiFiBase#1#2\filename@base#3{\fi\fi#1}
      Print current option setting
217 \def\grffile@option@status#1{%
218   \begingroup
219   \let\on@line\@empty
220   \PackageInfo{grffile}{%
221     Option ‘#1’ is %
222     \expandafter\ifx\csname ifgrffile@#1\expandafter\endcsname
223       \csname iftrue\endcsname
224     set to ‘true’%
225   \else
226     \expandafter\ifx\csname grffile@#1@disabled\endcsname\@empty
227     not available%
228   \else
229     set to ‘false’%
230   \fi
231   \fi
232   }%
233 \endgroup
234 }
235 \grffile@option@status{multidot}
236 \grffile@option@status{extendedchars}
237 \grffile@option@status{space}
238 \grffile@RestoreCatcodes
239 \end{package}

```

## 3 Installation

### 3.1 Download

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

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

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

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

### 3.2 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
```

---

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



**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/
```

### 3.3 Package installation

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

```
tex grffile.dtx
```

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

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

### 3.4 Refresh file name databases

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

### 3.5 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 grffile.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{grffile.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 pdf $\LaTeX$ :

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

## 4 References

- [1] David Carlisle, Sebastian Rahtz: *The graphics package*; 2006/02/20 v1.0o; [CTAN:macros/latex/required/graphics/graphics.dtx](#).
- [2] Sebastian Rahtz, Heiko Oberdiek: *The graphicx package*; 1999/02/16 v1.0f; [CTAN:macros/latex/required/graphics/graphicx.dtx](#).

## 5 History

### [2004/07/18 v0.5]

- First version, published in newsgroup [de.comp.text.tex](#):  
“[Re: Dateinamenproblem](#)”<sup>2</sup>

### [2006/08/15 v1.0]

- File existence check by new primitives of pdfTeX 1.30.
- Implementation partly rewritten.
- New DTX framework.

### [2006/08/17 v1.1]

- Adaptation to version 2.3 of package kvoptions.

### [2006/11/30 v1.2]

- New option babel. Before this feature was part of option extendedchars.

### [2007/04/11 v1.3]

- Line ends sanitized.

### [2007/06/13 v1.4]

- Encoding support added with options encoding, inputencoding, and filenameencoding.

### [2007/08/16 v1.5]

- Bug fix in encoding support.

### [2007/11/11 v1.6]

- Use of package pdftexcmds for L<sup>A</sup>T<sub>E</sub>X support.

### [2007/11/24 v1.7]

- Bug fix of broken previous version.

### [2008/08/11 v1.8]

- Code is not changed.
- URLs updated.

---

<sup>2</sup>Url: <http://groups.google.com/group/de.comp.text.tex/msg/b85984095d1a3c95>

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.

11

<b>O</b>		<b>S</b>	
\on@line .....	219	\setkeys .....	67
<b>P</b>		\SetupKeyvalOptions .....	27
\PackageError .....	115, 125	\StringEncodingConvert .....	120
\PackageInfo .....	220	\StringEncodingSuccessFailure ..	122
\PackageWarning .....	49	<b>T</b>	
\PassOptionsToPackage .....	41	\the .....	6, 7, 8, 9, 10, 11, 12, 13, 14
\pdf@filesize .....	166, 174	<b>U</b>	
\ProcessKeyvalOptions .....	61	\uccode .....	142
\ProvidesPackage .....	3	\uppercase .....	143
<b>R</b>		<b>X</b>	
\renewcommand .....	70	\x .....	133, 136
\repeat .....	149	<b>Z</b>	
\RequirePackage ....	25, 26, 36, 43, 65	\z@ .....	89
\reserved@a .....	167, 177, 182		
\reserved@b .....	170, 174, 176		