

The grfext package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/09/30 v1.0

Abstract

This package provides macros for adding and reordering graphics extensions of package `graphics`.

Contents

1	Documentation	2
1.1	Introduction	2
1.2	User interface	2
1.3	Package loading	2
1.4	plain- <code>T_EX</code>	2
2	Implementation	3
2.1	Relead check and identification	3
2.2	Catcodes	4
2.3	plain- <code>T_EX</code>	4
2.4	Add	4
2.5	Check	5
2.6	Remove	6
2.7	Print	6
3	Test	7
3.1	Catcode checks for loading	7
3.2	Macro tests	8
3.2.1	Preamble	8
3.2.2	Detection of unwanted space	8
3.2.3	Generic test macros	8
3.2.4	Test for <code>\AppendGraphicsExtensions</code>	9
3.2.5	Test for <code>\PrependGraphicsExtensions</code>	9
3.2.6	Test for <code>\RemoveGraphicsExtensions</code>	10
4	Installation	11
4.1	Download	11
4.2	Bundle installation	11
4.3	Package installation	11
4.4	Refresh file name databases	11
4.5	Some details for the interested	12
5	References	12
6	History	12
	[2007/09/30 v1.0]	12
7	Index	12

1 Documentation

1.1 Introduction

If you are not familiar with L^AT_EX's graphics bundle, please read its documentation `grffile` [1]. The bundle contains two packages for graphics inclusion: `graphics` and `graphicx`. The first one is loaded by the second one that adds a key value interface.

Graphics files are included in both cases by macro `\includegraphics`. The file name extension can be omitted. Then the graphics package goes through a list of known extensions until it finds the graphics file. This extension list is set by `\DeclareGraphicsExtensions`. The previous contents of the list is overwritten.

1.2 User interface

This package `grfext` provides macros that adds entries to the list or remove them. The list may be empty or even undefined before. It is always defined afterwards, but can be empty (especially after removing entries).

```
\AppendGraphicsExtensions * {\langle ext-list \rangle}
\PrependGraphicsExtensions * {\langle ext-list \rangle}
```

The argument $\langle ext-list \rangle$ is a comma separated list whose entries are file name extensions including the dot. But first the entries are removed from `graphics`' extension list to avoid multiple occurrences of the same extension.

Then macro `\AppendGraphicsExtensions` adds the entries after the end of `graphics`' list, whereas macro `\PrependGraphicsExtensions` puts them in front of the list. The order matters if a graphics file is available in different acceptable formats. Then the first extension wins.

The star version of these commands only adds an extensions, if a specific graphics rule exists for that extension.

```
\RemoveGraphicsExtensions {\langle ext-list \rangle}
```

All occurrences of file extensions in $\langle ext-list \rangle$ are removed from `graphics`' extension list.

1.3 Package loading

The package does not define any options. It is loaded as usual in L^AT_EX, e.g.:

```
\usepackage{grfext}
```

```
\PrintGraphicsExtensions
```

Macro `\PrintGraphicsExtensions` writes the current graphics extensions list in the `.log` file. The macros described before do this automatically after their operation.

1.4 plain-T_EX

L^AT_EX's graphics packages can also be used with plain-T_EX. The necessary basic L^AT_EX macros are defined in `miniltx.tex`. This package `grfext` also relies on it. Example:

```
\input miniltx.tex\relax
\def\Gin@driver{pdftex.def}
\input graphicx.sty\relax
```

```

\input grfext.sty\relax
\resetatcatcode

```

2 Implementation

```
1 <*package>
```

2.1 Reload 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 \catcode123 1 % {
9 \catcode125 2 % }
10 \expandafter\let\expandafter\x\csname ver@grfext.sty\endcsname
11 \ifx\x\relax % plain-TeX, first loading
12 \else
13 \def\empty{}%
14 \ifx\x\empty % LaTeX, first loading,
15 % variable is initialized, but \ProvidesPackage not yet seen
16 \else
17 \catcode35 6 % #
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{grfext}{The package is already loaded}%
26 \aftergroup\endinput
27 \fi
28 \fi
29 \endgroup

```

Package identification:

```

30 \begingroup
31 \catcode35 6 % #
32 \catcode40 12 % (
33 \catcode41 12 % )
34 \catcode44 12 % ,
35 \catcode45 12 % -
36 \catcode46 12 % .
37 \catcode47 12 % /
38 \catcode58 12 % :
39 \catcode64 11 % @
40 \catcode91 12 % [
41 \catcode93 12 % ]
42 \catcode123 1 % {
43 \catcode125 2 % }
44 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
45 \def\x#1#2#3[#4]{\endgroup
46 \immediate\write-1{Package: #3 #4}%
47 \xdef#1{#4}%
48 }%
49 \else
50 \def\x#1#2[#3]{\endgroup
51 #2[#3]}%
52 \ifx#1\undefined

```

```

53      \xdef#1{#3}%
54      \fi
55      \ifx#1\relax
56      \xdef#1{#3}%
57      \fi
58  }%
59  \fi
60 \expandafter\x\csname ver@grfext.sty\endcsname
61 \ProvidesPackage{grfext}%
62 [2007/09/30 v1.0 Managing graphics extensions (H0)]

```

2.2 Catcodes

```

63 \begingroup
64 \catcode123 1 % {
65 \catcode125 2 % }
66 \def\x{\endgroup
67 \expandafter\edef\csname grfext@AtEnd\endcsname{%
68 \catcode35 \the\catcode35\relax
69 \catcode64 \the\catcode64\relax
70 \catcode123 \the\catcode123\relax
71 \catcode125 \the\catcode125\relax
72 }%
73 }%
74 \x
75 \catcode35 6 % #
76 \catcode64 11 % @
77 \catcode123 1 % {
78 \catcode125 2 % }
79 \def\TMP@EnsureCode#1#2{%
80 \edef\grfext@AtEnd{%
81 \grfext@AtEnd
82 \catcode#1 \the\catcode#1\relax
83 }%
84 \catcode#1 #2\relax
85 }
86 \TMP@EnsureCode{42}{12}% *
87 \TMP@EnsureCode{44}{12}% ,
88 \TMP@EnsureCode{47}{12}% /
89 \TMP@EnsureCode{58}{12}% :
90 \TMP@EnsureCode{60}{12}% <
91 \TMP@EnsureCode{61}{12}% =
92 \TMP@EnsureCode{62}{12}% >
93 \TMP@EnsureCode{91}{12}% [
94 \TMP@EnsureCode{93}{12}% ]

```

2.3 plain-TeX

\@expandtwoargs Requirement is miniltx.tex, but we need also L^AT_EX's \@expandtwoargs.

```

95 \@ifundefined{@expandtwoargs}{%
96 \def\@expandtwoargs#1#2#3{%
97 \edef\reserved@a{\noexpand#1{#2}{#3}}%
98 \reserved@a
99 }%
100 }{}

```

2.4 Add

\AppendGraphicsExtensions

```

101 \newcommand*\AppendGraphicsExtensions{%
102 \@ifundefined{Gin@extensions}{%
103 \let\Gin@extensions\@empty

```

```

104 }{}%
105 \@ifstar{\grfext@Append\grfext@Check}{\grfext@Append\grfext@@Add}%
106 }%

```

`\grfext@Append`

```

107 \def\grfext@Append#1#2{%
108   \let\grfext@Print\@gobble
109   \edef\grfext@next{%
110     \noexpand\grfext@Add\noexpand#1{%
111       \zap@space#2 \@empty
112     }{\noexpand\Gin@extensions,}{}%
113   }%
114   \grfext@next
115   \let\grfext@Print\grfext@@Print
116   \grfext@Print\AppendGraphicsExtensions
117 }

```

`\PrependGraphicsExtensions`

```

118 \newcommand*{\PrependGraphicsExtensions}{%
119   \@ifundefined{Gin@extensions}{%
120     \let\Gin@extensions\@empty
121   }{}%
122   \@ifstar{\grfext@Prepend\grfext@Check}{\grfext@Prepend\grfext@@Add}%
123 }%

```

`\grfext@Prepend`

```

124 \def\grfext@Prepend#1#2{%
125   \let\grfext@Print\@gobble
126   \edef\grfext@next{%
127     \noexpand\grfext@Add\noexpand#1{%
128       \zap@space#2 \@empty
129     }{}{\noexpand\Gin@extensions}%
130   }%
131   \grfext@next
132   \let\grfext@Print\grfext@@Print
133   \grfext@Print\PrependGraphicsExtensions
134 }

```

`\grfext@Add`

```

135 \def\grfext@Add#1#2{%
136   #1{#2}%
137 }

```

`\grfext@@Add`

```

138 \def\grfext@@Add#1#2#3{%
139   \RemoveGraphicsExtensions{#1}%
140   \ifx\Gin@extensions\@empty
141     \def\Gin@extensions{#1}%
142   \else
143     \edef\Gin@extensions{#2#1#3}%
144   \fi
145 }

```

2.5 Check

`\grfext@Check`

```

146 \def\grfext@Check#1{%
147   \let\grfext@tmp\@empty
148   \@for\grfext@ext:=#1\do{%
149     \@ifundefined{Gin@rule@\grfext@ext}{%
150       }{}%

```

```

151     \ifx\grfext@tmp\@empty
152     \let\grfext@tmp\grfext@ext
153     \else
154     \edef\grfext@tmp{\grfext@tmp,\grfext@ext}%
155     \fi
156   }%
157 }%
158 \ifx\grfext@tmp\@empty
159   \def\grfext@next##1##2{%
160 \else
161   \edef\grfext@next{%
162     \noexpand\grfext@@Add{\grfext@tmp}%
163   }%
164 \fi
165 \grfext@next
166 }

```

2.6 Remove

\RemoveGraphicsExtensions

```

167 \newcommand*{\RemoveGraphicsExtensions}[1]{%
168   \@ifundefined{Gin@extensions}{%
169     \def\Gin@extensions{%
170   }{%
171     \edef\grfext@tmp{\zap@space#1 \@empty}%
172     \@for\grfext@ext:=\grfext@tmp\do{%
173       \def\grfext@next{%
174         \let\grfext@tmp\Gin@extensions
175         \@expandtwoargs
176         \@removeelement\grfext@ext\Gin@extensions\Gin@extensions
177         \ifx\grfext@tmp\Gin@extensions
178           \let\grfext@next\relax
179         \fi
180         \grfext@next
181       }%
182       \grfext@next
183     }%
184   }%
185   \grfext@Print\RemoveGraphicsExtensions
186 }

```

2.7 Print

```
187 \RequirePackage{infwarerr}[2007/09/09]
```

\PrintGraphicsExtensions

```

188 \def\PrintGraphicsExtensions{%
189   \grfext@Print\PrintGraphicsExtensions
190 }

```

\grfext@Print

```

191 \def\grfext@Print#1{%
192   \@PackageInfo{grfext}{%
193     Graphics extension search list:\MessageBreak
194     \@ifundefined{Gin@extensions}{%
195       <unavailable>%
196     }{%
197       [\Gin@extensions]%
198     }\MessageBreak
199     \string#1%
200   }%
201 }

```

\grfext@@Print

202 \let\grfext@@Print\grfext@Print

203 \grfext@AtEnd

204 \end{package}

3 Test

Load miniltx for standard catcode test.

205 \test1

206 \input miniltx.tex\relax

207 \resetatcatcode

208 \test1

3.1 Catcode checks for loading

209 \test1

210 \catcode'\{=1 %

211 \catcode'\}=2 %

212 \catcode'\#=6 %

213 \catcode'\@=11 %

214 \expandafter\ifx\csname count@\endcsname\relax

215 \countdef\count@=255 %

216 \fi

217 \expandafter\ifx\csname @gobble\endcsname\relax

218 \long\def\@gobble#1{}\%

219 \fi

220 \expandafter\ifx\csname @firstofone\endcsname\relax

221 \long\def\@firstofone#1{#1}\%

222 \fi

223 \expandafter\ifx\csname loop\endcsname\relax

224 \expandafter\@firstofone

225 \else

226 \expandafter\@gobble

227 \fi

228 {%

229 \def\loop#1\repeat{%

230 \def\body{#1}\%

231 \iterate

232 }%

233 \def\iterate{%

234 \body

235 \let\next\iterate

236 \else

237 \let\next\relax

238 \fi

239 \next

240 }%

241 \let\repeat=\fi

242 }%

243 \def\RestoreCatcodes{}

244 \count@=0 %

245 \loop

246 \edef\RestoreCatcodes{%

247 \RestoreCatcodes

248 \catcode\the\count@=\the\catcode\count@\relax

249 }%

250 \ifnum\count@<255 %

251 \advance\count@ 1 %

252 \repeat

253

```

254 \def\RangeCatcodeInvalid#1#2{%
255   \count@=#1\relax
256   \loop
257     \catcode\count@=15 %
258   \ifnum\count@<#2\relax
259     \advance\count@ 1 %
260   \repeat
261 }
262 \expandafter\ifx\csname LoadCommand\endcsname\relax
263   \def\LoadCommand{\input grfext.sty\relax}%
264 \fi
265 \def\Test{%
266   \RangeCatcodeInvalid{0}{47}%
267   \RangeCatcodeInvalid{58}{64}%
268   \RangeCatcodeInvalid{91}{96}%
269   \RangeCatcodeInvalid{123}{255}%
270   \catcode'\@=12 %
271   \catcode'\=0 %
272   \catcode'\{=1 %
273   \catcode'\}=2 %
274   \catcode'\#=6 %
275   \catcode'\[=12 %
276   \catcode'\]=12 %
277   \catcode'\%=14 %
278   \catcode'\ =10 %
279   \catcode13=5 %
280   \LoadCommand
281   \RestoreCatcodes
282 }
283 \Test
284 \csname @@end\endcsname
285 \end
286 </test1>

```

3.2 Macro tests

3.2.1 Preamble

```

287 <*test2>
288 \NeedsTeXFormat{LaTeX2e}
289 \nofiles
290 \documentclass{article}
291 \usepackage[pdftex]{graphics}
292 \usepackage{grfext}[2007/09/30]
293 \usepackage{qstest}
294 \IncludeTests{*}
295 \LogTests{log}{*}{*}
296 \makeatletter

```

3.2.2 Detection of unwanted space

```

297 \let\orig@qstest\qstest
298 \let\orig@endqstest\endqstest
299 \def\qstest#1#2{%
300   \orig@qstest{#1}{#2}%
301   \setbox0\hbox\bgroup\begingroup\ignorespaces
302 }
303 \def\endqstest{%
304   \endgroup\egroup
305   \Expect*{\the\wd0}{0.0pt}%
306   \orig@endqstest
307 }

```

3.2.3 Generic test macros


```

308 \def\Test{\@ifnextchar\bgroup\TestInitList\TestInitCmd}%
309 \def\TestInitList#1{%
310   \def\Gin@extensions{#1}%
311   \@Test
312 }
313 \def\TestInitCmd#1{%
314   \let\Gin@extensions#1\relax
315   \@Test
316 }
317 \def\@Test#1#2{%
318   \TestCmd{#1}%
319   \expandafter\Expect\expandafter*{\Gin@extensions}{#2}%
320 }

```

3.2.4 Test for \AppendGraphicsExtensions

```

321 \begin{qstest}{AppendGraphicsExtensions}{AppendGraphicsExtensions}
322   \let\TestCmd\AppendGraphicsExtensions
323   \Test\@undefined{.test}{.test}%
324   \Test\relax{.test}{.test}%
325   \Test{}{.test}{.test}%
326   \Test{.foo}{.test}{.foo,.test}%
327   \Test{.foo,.bar}{.test}{.foo,.bar,.test}%
328   \Test{.test}{.test}{.test}%
329   \Test{.foo,.test}{.test}{.foo,.test}%
330   \Test{.test,.foo}{.test}{.foo,.test}%
331   \Test{.foo,.test,.bar}{.test}{.foo,.bar,.test}%
332   \Test{.foo}{.test,.bar}{.foo,.test,.bar}%
333   \Test{.foo}{.test,.foo}{.test,.foo}%
334   \Test{.foo}{.test,.bar,.x,.y}{.foo,.test,.bar,.x,.y}%
335   \def\TestCmd{\AppendGraphicsExtensions*}%
336   \Test\@undefined{.test}{}%
337   \Test\relax{.test}{}%
338   \Test{}{.test}{}%
339   \Test{.test}{.test}{.test}%
340   \Test{.foo}{.test}{.foo}%
341   \Test{.foo,.bar}{.test}{.foo,.bar}%
342   \Test{.test}{.test}{.test}%
343   \Test{.foo,.test}{.test}{.foo,.test}%
344   \Test{.test,.foo}{.test}{.test,.foo}%
345   \Test{.foo,.test,.bar}{.test}{.foo,.test,.bar}%
346   \Test{.foo}{.test,.bar}{.foo}%
347   \Test{.foo}{.test,.foo}{.foo}%
348   \DeclareGraphicsRule{.test}{test}{.test}{}%
349   \Test\@undefined{.test}{.test}%
350   \Test\relax{.test}{.test}%
351   \Test{}{.test}{.test}%
352   \Test{.foo}{.test}{.foo,.test}%
353   \Test{.foo,.bar}{.test}{.foo,.bar,.test}%
354   \Test{.test}{.test}{.test}%
355   \Test{.foo,.test}{.test}{.foo,.test}%
356   \Test{.test,.foo}{.test}{.foo,.test}%
357   \Test{.foo,.test,.bar}{.test}{.foo,.bar,.test}%
358   \Test{.foo}{.test,.bar}{.foo,.test}%
359   \Test{.foo}{.test,.foo}{.foo,.test}%
360   \DeclareGraphicsRule{.bar}{bar}{.bar}{}%
361   \Test{.foo}{.test,.bar}{.foo,.test,.bar}%
362   \Test{.bar}{.test,.bar}{.test,.bar}%
363   \Test{.bar}{.bar,.test}{.bar,.test}%
364 \end{qstest}

```

3.2.5 Test for \PrependGraphicsExtensions

```

365 \begin{qstest}{PrependGraphicsExtensions}{PrependGraphicsExtensions}
366   \let\TestCmd\PrependGraphicsExtensions

```

```

367 \Test\@undefined{.test}{.test}%
368 \Test\relax{.test}{.test}%
369 \Test{}{.test}{.test}%
370 \Test{.foo}{.test}{.test,.foo}%
371 \Test{.foo,.bar}{.test}{.test,.foo,.bar}%
372 \Test{.test}{.test}{.test}%
373 \Test{.foo,.test}{.test}{.test,.foo}%
374 \Test{.test,.foo}{.test}{.test,.foo}%
375 \Test{.foo,.test,.bar}{.test}{.test,.foo,.bar}%
376 \Test{.foo}{.test,.bar}{.test,.bar,.foo}%
377 \Test{.foo}{.test,.foo}{.test,.foo}%
378 \Test{.foo}{.test,.bar,.x,.y}{.test,.bar,.x,.y,.foo}%
379 \def\TestCmd{\PrependGraphicsExtensions*}%
380 \Test\@undefined{.test}{}%
381 \Test\relax{.test}{}%
382 \Test{}{.test}{}%
383 \Test{.test}{.test}{.test}%
384 \Test{.foo}{.test}{.foo}%
385 \Test{.foo,.bar}{.test}{.foo,.bar}%
386 \Test{.test}{.test}{.test}%
387 \Test{.foo,.test}{.test}{.foo,.test}%
388 \Test{.test,.foo}{.test}{.test,.foo}%
389 \Test{.foo,.test,.bar}{.test}{.foo,.test,.bar}%
390 \DeclareGraphicsRule{.test}{test}{.test}{}%
391 \Test\@undefined{.test}{.test}%
392 \Test\relax{.test}{.test}%
393 \Test{}{.test}{.test}%
394 \Test{.foo}{.test}{.test,.foo}%
395 \Test{.foo,.bar}{.test}{.test,.foo,.bar}%
396 \Test{.test}{.test}{.test}%
397 \Test{.foo,.test}{.test}{.test,.foo}%
398 \Test{.test,.foo}{.test}{.test,.foo}%
399 \Test{.foo,.test,.bar}{.test}{.test,.foo,.bar}%
400 \Test{.foo}{.test,.bar}{.test,.foo}%
401 \Test{.foo}{.test,.foo}{.test,.foo}%
402 \DeclareGraphicsRule{.bar}{bar}{.bar}{}%
403 \Test{.foo}{.test,.bar}{.test,.bar,.foo}%
404 \Test{.bar}{.test,.bar}{.test,.bar}%
405 \Test{.bar}{.bar,.test}{.bar,.test}%
406 \end{qstest}

```

3.2.6 Test for \RemoveGraphicsExtensions

```

407 \begin{qstest}{RemoveGraphicsExtensions}{RemoveGraphicsExtensions}
408 \let\TestCmd\RemoveGraphicsExtensions
409 \Test\@undefined{.test}{}%
410 \Test\relax{.test}{}%
411 \Test{}{.test}{}%
412 \Test{.foo}{.test}{.foo}%
413 \Test{.foo,.bar}{.test}{.foo,.bar}%
414 \Test{.test}{.test}{}%
415 \Test{.foo,.test}{.test}{.foo}%
416 \Test{.test,.foo}{.test}{.foo}%
417 \Test{.foo,.test,.bar}{.test}{.foo,.bar}%
418 \Test{.test,.test}{.test}{}%
419 \Test{.foo,.test,.bar,.test,.test,.bar}{.test}{.foo,.bar,.bar}%
420 \Test{.foo,.test,.bar,.foo,.test,.bar}{.test,.bar}{.foo,.foo}%
421 \Test{.foo,.test,.bar,.foo,.test,.bar}{.test,.bar}{.foo,.foo}%
422 \end{qstest}

423 \begin{document}
424 \end{document}
425 \end{test2}

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

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

[CTAN:macros/latex/contrib/oberdiek/grfext.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 T_EX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

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-T_EX:

```
tex grfext.dtx
```

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

<code>grfext.sty</code>	→ <code>tex/latex/oberdiek/grfext.sty</code>
<code>grfext.pdf</code>	→ <code>doc/latex/oberdiek/grfext.pdf</code>
<code>test/grfext-test1.tex</code>	→ <code>doc/latex/oberdiek/test/grfext-test1.tex</code>
<code>test/grfext-test2.tex</code>	→ <code>doc/latex/oberdiek/test/grfext-test2.tex</code>
<code>grfext.dtx</code>	→ <code>source/latex/oberdiek/grfext.dtx</code>

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 T_EX distribution (teT_EX, miK_T_EX, ...) relies on file name databases, you must refresh these. For example, teT_EX users run `texhash` or `mktextlsr`.

¹<http://ftp.ctan.org/tex-archive/>

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 grfext.pdf unpack_files output .
```

Unpacking with L^AT_EX. The `.dtx` chooses its action depending on the format:

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

L^AT_EX: Generate the documentation.

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

```
latex \let\install=y\input{grfext.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 grfext.dtx
makeindex -s gind.ist grfext.idx
pdflatex grfext.dtx
makeindex -s gind.ist grfext.idx
pdflatex grfext.dtx
```

5 References

- [1] David Carlisle, Sebastian Rahtz: *The graphics package*; 2006/02/20 v1.0o;
CTAN:macros/latex/required/graphics/graphics.dtx.

6 History

[2007/09/30 v1.0]

- First public version,

7 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		120, 128, 140, 147, 151, 158, 171
\#	212, 274	\@expandtwoargs 95, 175
\%	277	\@firstofone 221, 224
\@	213, 270	\@for 148, 172
\@PackageInfo	192	\@gobble 108, 125, 218, 226
\@Test	311, 315, 317	\@ifnextchar 308
\@empty	103, 111,	\@ifstar 105, 122

<code>\@ifundefined</code>	95, 102, 119, 149, 168, 194	<code>\grfext@Prepend</code>	122, <u>124</u>
<code>\@removeelement</code>	176	<code>\grfext@Print</code>	108, 115, 116, 125, 132, 133, 185, 189, <u>191</u> , 202
<code>\@undefined</code>	52, 323, 336, 349, 367, 380, 391, 409	<code>\grfext@tmp</code>	147, 151, 152, 154, 158, 162, 171, 172, 174, 177
<code>\[</code>	275	H	
<code>\]</code>	271	<code>\hbox</code>	301
<code>\{</code>	210, 272	I	
<code>\}</code>	211, 273	<code>\ifnum</code>	250, 258
<code>\]</code>	276	<code>\ifx</code>	11, 14, 18, 44, 52, 55, 140, 151, 158, 177, 214, 217, 220, 223, 262
<code>_</code>	278	<code>\ignorespaces</code>	301
A		<code>\immediate</code>	20, 46
<code>\advance</code>	251, 259	<code>\IncludeTests</code>	294
<code>\aftergroup</code>	26	<code>\input</code>	206, 263
<code>\AppendGraphicsExtensions</code>	2, <u>101</u> , 116, 322, 335	<code>\iterate</code>	231, 233, 235
B		L	
<code>\begin</code>	321, 365, 407, 423	<code>\LoadCommand</code>	263, 280
<code>\body</code>	230, 234	<code>\LogTests</code>	295
C		<code>\loop</code>	229, 245, 256
<code>\catcode</code>	3, 4, 5, 6, 7, 8, 9, 17, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 64, 65, 68, 69, 70, 71, 75, 76, 77, 78, 82, 84, 210, 211, 212, 213, 248, 257, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279	M	
<code>\count@</code>	215, 244, 248, 250, 251, 255, 257, 258, 259	<code>\makeatletter</code>	296
<code>\countdef</code>	215	<code>\MessageBreak</code>	193, 198
<code>\csname</code>	10, 18, 44, 60, 67, 214, 217, 220, 223, 262, 284	N	
D		<code>\NeedsTeXFormat</code>	288
<code>\DeclareGraphicsRule</code>	348, 360, 390, 402	<code>\newcommand</code>	101, 118, 167
<code>\do</code>	148, 172	<code>\next</code>	235, 237, 239
<code>\documentclass</code>	290	<code>\nofiles</code>	289
E		O	
<code>\empty</code>	13, 14	<code>\orig@endqstest</code>	298, 306
<code>\end</code>	285, 364, 406, 422, 424	<code>\orig@qstest</code>	297, 300
<code>\endcsname</code>	10, 18, 44, 60, 67, 214, 217, 220, 223, 262, 284	P	
<code>\endinput</code>	26	<code>\PackageInfo</code>	23
<code>\endqstest</code>	298, 303	<code>\PrependGraphicsExtensions</code>	118, 133, 366, 379
<code>\Expect</code>	305, 319	<code>\PrintGraphicsExtensions</code>	2, 188
G		<code>\ProvidesPackage</code>	15, 61
<code>\Gin@extensions</code>	103, 112, 120, 129, 140, 141, 143, 169, 174, 176, 177, 197, 310, 314, 319	Q	
<code>\grfext@@Add</code>	105, 122, <u>138</u> , 162	<code>\qstest</code>	297, 299
<code>\grfext@@Print</code>	115, 132, <u>202</u>	R	
<code>\grfext@Add</code>	110, 127, <u>135</u>	<code>\RangeCatcodeInvalid</code>	254, 266, 267, 268, 269
<code>\grfext@Append</code>	105, <u>107</u>	<code>\RemoveGraphicsExtensions</code>	2, 139, <u>167</u> , 408
<code>\grfext@AtEnd</code>	80, 81, 203	<code>\repeat</code>	229, 241, 252, 260
<code>\grfext@Check</code>	105, 122, <u>146</u>	<code>\RequirePackage</code>	187
<code>\grfext@ext</code>	148, 149, 152, 154, 172, 176	<code>\reserved@a</code>	97, 98
<code>\grfext@next</code>	109, 114, 126, 131, 159, 161, 165, 173, 178, 180, 182	<code>\resetatcatcode</code>	207
S		<code>\RestoreCatcodes</code>	243, 246, 247, 281
T		<code>\setbox</code>	301
T		<code>\Test</code>	265, 283, 308, 323, 324, 325, 326, 327, 328, 329, 330,

331, 332, 333, 334, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 361, 362, 363, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 403, 404, 405, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421	\the 68, 69, 70, 71, 82, 248, 305 \TMP@EnsureCode 79, 86, 87, 88, 89, 90, 91, 92, 93, 94
	U
	\usepackage 291, 292, 293
	W
	\wd 305
	\write 20, 46
	X
\TestCmd . . 318, 322, 335, 366, 379, 408 \TestInitCmd 308, 313 \TestInitList 308, 309	\x 10, 11, 14, 19, 23, 25, 45, 50, 60, 66, 74
	Z
	\zap@space 111, 128, 171