

# The grfext package

Heiko Oberdiek

<heiko.oberdiek at gmail.com>

2010/08/19 v1.1

## Abstract

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

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	Introduction . . . . .	2
1.2	User interface . . . . .	2
1.3	Package loading . . . . .	2
1.4	Option support for package <code>graphicx</code> . . . . .	3
1.5	plain <code>T<sub>E</sub>X</code> . . . . .	3
<b>2</b>	<b>Implementation</b>	<b>3</b>
2.1	Relead check and identification . . . . .	3
2.2	Catcodes . . . . .	5
2.3	plain <code>T<sub>E</sub>X</code> . . . . .	5
2.4	Add . . . . .	5
2.5	Check . . . . .	6
2.6	Remove . . . . .	7
2.7	Print . . . . .	7
2.8	Defining options for package <code>graphicx</code> . . . . .	8
<b>3</b>	<b>Test</b>	<b>8</b>
3.1	Catcode checks for loading . . . . .	8
3.2	Macro tests . . . . .	10
3.2.1	Preamble . . . . .	10
3.2.2	Detection of unwanted space . . . . .	10
3.2.3	Generic test macros . . . . .	11
3.2.4	Test for <code>\AppendGraphicsExtensions</code> . . . . .	11
3.2.5	Test for <code>\PrependGraphicsExtensions</code> . . . . .	12
3.2.6	Test for <code>\RemoveGraphicsExtensions</code> . . . . .	12
<b>4</b>	<b>Installation</b>	<b>13</b>
4.1	Download . . . . .	13
4.2	Bundle installation . . . . .	13
4.3	Package installation . . . . .	13
4.4	Refresh file name databases . . . . .	14
4.5	Some details for the interested . . . . .	14
<b>5</b>	<b>Catalogue</b>	<b>15</b>

<b>6</b>	<b>References</b>	<b>15</b>
<b>7</b>	<b>History</b>	<b>15</b>
	[2007/09/30 v1.0] . . . . .	15
	[2010/08/19 v1.1] . . . . .	15
<b>8</b>	<b>Index</b>	<b>15</b>

# 1 Documentation

## 1.1 Introduction

If you are not familiar with L<sup>A</sup>T<sub>E</sub>X'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).

<code>\AppendGraphicsExtensions * {⟨ext-list⟩}</code> <code>\PrependGraphicsExtensions * {⟨ext-list⟩}</code>
---

The argument `⟨ext-list⟩` 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.

<code>\RemoveGraphicsExtensions {⟨ext-list⟩}</code>
---

All occurrences of file extensions in `⟨ext-list⟩` are removed from `graphics`' extension list.

## 1.3 Package loading

The package does not define any options. It is loaded as usual in L<sup>A</sup>T<sub>E</sub>X, e.g.:

```
\usepackage{grfext}
```

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

## 1.4 Option support for package `graphicx`

Package `graphicx` uses the interface of package `keyval` in order to specify options for `\includegraphics`. The options can also be set using

```
\setkeys{Gin}{\langle options \rangle}
```

The four user macros with the two star forms are available as options in family `Gin` as well:

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

This makes it easier to locally change the extension list for an included graphics, e.g.:

```
\includegraphics[RemoveGraphicsExtensions={.pdf,PDF}]{image}
```

## 1.5 plain $\text{\TeX}$

$\text{\LaTeX}$ 's graphics packages can also be used with plain  $\text{\TeX}$ . The necessary basic  $\text{\LaTeX}$  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 \langle *package \rangle
```

### 2.1 Reload check and identification

Reload check, especially if the package is not used with  $\text{\LaTeX}$ .

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%  
3 \catcode13=5 % ^~M  
4 \endlinechar=13 %  
5 \catcode35=6 % #  
6 \catcode39=12 % '  
7 \catcode44=12 % ,  
8 \catcode45=12 % -  
9 \catcode46=12 % .  
10 \catcode58=12 % :  
11 \catcode64=11 % @  
12 \catcode123=1 % {
```

```

13 \catcode125=2 % }
14 \expandafter\let\expandafter\x\csname ver@grfext.sty\endcsname
15 \ifx\x\relax % plain-TeX, first loading
16 \else
17 \def\empty{}%
18 \ifx\x\empty % LaTeX, first loading,
19 % variable is initialized, but \ProvidesPackage not yet seen
20 \else
21 \expandafter\ifx\csname PackageInfo\endcsname\relax
22 \def\x#1#2{%
23 \immediate\write-1{Package #1 Info: #2.}%
24 }%
25 \else
26 \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
27 \fi
28 \x{grfext}{The package is already loaded}%
29 \aftergroup\endinput
30 \fi
31 \fi
32 \endgroup%

```

Package identification:

```

33 \begingroup\catcode61\catcode48\catcode32=10\relax%
34 \catcode13=5 % ^M
35 \endlinechar=13 %
36 \catcode35=6 % #
37 \catcode39=12 % '
38 \catcode40=12 % (
39 \catcode41=12 % )
40 \catcode44=12 % ,
41 \catcode45=12 % -
42 \catcode46=12 % .
43 \catcode47=12 % /
44 \catcode58=12 % :
45 \catcode64=11 % @
46 \catcode91=12 % [
47 \catcode93=12 % ]
48 \catcode123=1 % {
49 \catcode125=2 % }
50 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
51 \def\x#1#2#3[#4]{\endgroup
52 \immediate\write-1{Package: #3 #4}%
53 \xdef#1{#4}%
54 }%
55 \else
56 \def\x#1#2[#3]{\endgroup
57 #2[#3]}%
58 \ifx#1\@undefined
59 \xdef#1{#3}%
60 \fi
61 \ifx#1\relax
62 \xdef#1{#3}%
63 \fi
64 }%
65 \fi
66 \expandafter\x\csname ver@grfext.sty\endcsname
67 \ProvidesPackage{grfext}%
68 [2010/08/19 v1.1 Manage graphics extensions (HO)]%

```

## 2.2 Catcodes

```

69 \begingroup\catcode61\catcode48\catcode32=10\relax%
70 \catcode13=5 % ^^M
71 \endlinechar=13 %
72 \catcode123=1 % {
73 \catcode125=2 % }
74 \catcode64=11 % @
75 \def\x{\endgroup
76   \expandafter\edef\csname grfext@AtEnd\endcsname{%
77     \endlinechar=\the\endlinechar\relax
78     \catcode13=\the\catcode13\relax
79     \catcode32=\the\catcode32\relax
80     \catcode35=\the\catcode35\relax
81     \catcode61=\the\catcode61\relax
82     \catcode64=\the\catcode64\relax
83     \catcode123=\the\catcode123\relax
84     \catcode125=\the\catcode125\relax
85   }%
86 }%
87 \x\catcode61\catcode48\catcode32=10\relax%
88 \catcode13=5 % ^^M
89 \endlinechar=13 %
90 \catcode35=6 % #
91 \catcode64=11 % @
92 \catcode123=1 % {
93 \catcode125=2 % }
94 \def\TMP@EnsureCode#1#2{%
95   \edef\grfext@AtEnd{%
96     \grfext@AtEnd
97     \catcode#1=\the\catcode#1\relax
98   }%
99   \catcode#1=#2\relax
100 }
101 \TMP@EnsureCode{42}{12}% *
102 \TMP@EnsureCode{44}{12}% ,
103 \TMP@EnsureCode{47}{12}% /
104 \TMP@EnsureCode{58}{12}% :
105 \TMP@EnsureCode{60}{12}% <
106 \TMP@EnsureCode{62}{12}% >
107 \TMP@EnsureCode{91}{12}% [
108 \TMP@EnsureCode{93}{12}% ]
109 \edef\grfext@AtEnd{\grfext@AtEnd\noexpand\endinput}

```

## 2.3 plain T<sub>E</sub>X

\@expandtwoargs Requirement is miniltx.tex, but we need also L<sup>A</sup>T<sub>E</sub>X's \@expandtwoargs.

```

110 \@ifundefined{@expandtwoargs}{%
111   \def\@expandtwoargs#1#2#3{%
112     \edef\reserved@a{\noexpand#1{#2}{#3}}%
113     \reserved@a
114   }%
115 }{}

```

## 2.4 Add

\AppendGraphicsExtensions

```

116 \newcommand*{\AppendGraphicsExtensions}{%
117   \@ifundefined{Gin@extensions}{%

```

```

118 \let\Gin@extensions\@empty
119 }{}%
120 \@ifstar{\grfext@Append\grfext@Check}{\grfext@Append\grfext@@Add}%
121 }%

```

\grfext@Append

```

122 \def\grfext@Append#1#2{%
123 \let\grfext@Print\@gobble
124 \edef\grfext@next{%
125 \noexpand\grfext@Add\noexpand#1{%
126 \zap@space#2 \@empty
127 }{\noexpand\Gin@extensions,}{}%
128 }%
129 \grfext@next
130 \let\grfext@Print\grfext@@Print
131 \grfext@Print\AppendGraphicsExtensions
132 }

```

\PrependGraphicsExtensions

```

133 \newcommand*{\PrependGraphicsExtensions}{%
134 \@ifundefined{Gin@extensions}{%
135 \let\Gin@extensions\@empty
136 }{}%
137 \@ifstar{\grfext@Prepend\grfext@Check}{\grfext@Prepend\grfext@@Add}%
138 }%

```

\grfext@Prepend

```

139 \def\grfext@Prepend#1#2{%
140 \let\grfext@Print\@gobble
141 \edef\grfext@next{%
142 \noexpand\grfext@Add\noexpand#1{%
143 \zap@space#2 \@empty
144 }{}{\noexpand\Gin@extensions}%
145 }%
146 \grfext@next
147 \let\grfext@Print\grfext@@Print
148 \grfext@Print\PrependGraphicsExtensions
149 }

```

\grfext@Add

```

150 \def\grfext@Add#1#2{%
151 #1{#2}%
152 }

```

\grfext@@Add

```

153 \def\grfext@@Add#1#2#3{%
154 \RemoveGraphicsExtensions{#1}%
155 \ifx\Gin@extensions\@empty
156 \def\Gin@extensions{#1}%
157 \else
158 \edef\Gin@extensions{#2#1#3}%
159 \fi
160 }

```

## 2.5 Check

\grfext@Check

```

161 \def\grfext@Check#1{%
162   \let\grfext@tmp\@empty
163   \@for\grfext@ext:=#1\do{%
164     \@ifundefined{Gin@rule@\grfext@ext}{%
165       }{%
166         \ifx\grfext@tmp\@empty
167           \let\grfext@tmp\grfext@ext
168         \else
169           \edef\grfext@tmp{\grfext@tmp,\grfext@ext}%
170         \fi
171       }%
172     }%
173   \ifx\grfext@tmp\@empty
174     \def\grfext@next##1##2{%
175   \else
176     \edef\grfext@next{%
177       \noexpand\grfext@@Add{\grfext@tmp}%
178     }%
179   \fi
180   \grfext@next
181 }

```

## 2.6 Remove

\RemoveGraphicsExtensions

```

182 \newcommand*\RemoveGraphicsExtensions}[1]{%
183   \@ifundefined{Gin@extensions}{%
184     \def\Gin@extensions{%
185   }{%
186     \edef\grfext@tmp{\zap@space#1 \@empty}%
187     \@for\grfext@ext:=\grfext@tmp\do{%
188       \def\grfext@next{%
189         \let\grfext@tmp\Gin@extensions
190         \@expandtwoargs
191         \@removeelement\grfext@ext\Gin@extensions\Gin@extensions
192         \ifx\grfext@tmp\Gin@extensions
193           \let\grfext@next\relax
194         \fi
195         \grfext@next
196       }%
197       \grfext@next
198     }%
199   }%
200   \grfext@Print\RemoveGraphicsExtensions
201 }

```

## 2.7 Print

202 \RequirePackage{infwarerr}[2007/09/09]

\PrintGraphicsExtensions

```

203 \def\PrintGraphicsExtensions{%
204   \grfext@Print\PrintGraphicsExtensions
205 }

```

\grfext@Print

```

206 \def\grfext@Print#1{%
207   \@PackageInfo{grfext}{%

```

```

208 Graphics extension search list:\MessageBreak
209 \@ifundefined{Gin@extensions}{%
210   <unavailable>%
211 }{%
212   [\Gin@extensions]%
213 }\MessageBreak
214 \string#1%
215 }%
216 }

```

\grfext@@Print

```
217 \let\grfext@@Print\grfext@Print
```

## 2.8 Defining options for package **graphicx**

```

218 \RequirePackage{kvdefinekeys}[2010/03/01]
219 \kv@define@key{Gin}{AppendGraphicsExtensions}{%
220   \AppendGraphicsExtensions{#1}%
221 }
222 \kv@define@key{Gin}{AppendGraphicsExtensions*}{%
223   \AppendGraphicsExtensions*{#1}%
224 }
225 \kv@define@key{Gin}{PrependGraphicsExtensions}{%
226   \PrependGraphicsExtensions{#1}%
227 }
228 \kv@define@key{Gin}{PrependGraphicsExtensions*}{%
229   \PrependGraphicsExtensions*{#1}%
230 }
231 \kv@define@key{Gin}{RemoveGraphicsExtensions}{%
232   \RemoveGraphicsExtensions{#1}%
233 }
234 \kv@define@key{Gin}{PrintGraphicsExtensions}[]{%
235   \PrintGraphicsExtensions
236 }
237 \grfext@AtEnd%
238 </package>

```

## 3 Test

Load miniltx for standard catcode test.

```

239 < *test1>
240 \input miniltx.tex\relax
241 \resetatcatcode
242 </test1>

```

### 3.1 Catcode checks for loading

```

243 < *test1>
244 \catcode'\{=1 %
245 \catcode'\}=2 %
246 \catcode'\#=6 %
247 \catcode'\@=11 %
248 \expandafter\ifx\csname count@\endcsname\relax
249   \countdef\count@=255 %
250 \fi
251 \expandafter\ifx\csname @gobble\endcsname\relax
252   \long\def\@gobble#1{%

```



```

253 \fi
254 \expandafter\ifx\csname @firstofone\endcsname\relax
255 \long\def\@firstofone#1{#1}%
256 \fi
257 \expandafter\ifx\csname loop\endcsname\relax
258 \expandafter\@firstofone
259 \else
260 \expandafter\@gobble
261 \fi
262 {%
263 \def\loop#1\repeat{%
264 \def\body{#1}%
265 \iterate
266 }%
267 \def\iterate{%
268 \body
269 \let\next\iterate
270 \else
271 \let\next\relax
272 \fi
273 \next
274 }%
275 \let\repeat=\fi
276 }%
277 \def\RestoreCatcodes{}
278 \count@=0 %
279 \loop
280 \edef\RestoreCatcodes{%
281 \RestoreCatcodes
282 \catcode\the\count@=\the\catcode\count@\relax
283 }%
284 \ifnum\count@<255 %
285 \advance\count@ 1 %
286 \repeat
287
288 \def\RangeCatcodeInvalid#1#2{%
289 \count@=#1\relax
290 \loop
291 \catcode\count@=15 %
292 \ifnum\count@<#2\relax
293 \advance\count@ 1 %
294 \repeat
295 }
296 \def\RangeCatcodeCheck#1#2#3{%
297 \count@=#1\relax
298 \loop
299 \ifnum#3=\catcode\count@
300 \else
301 \errmessage{%
302 Character \the\count@\space
303 with wrong catcode \the\catcode\count@\space
304 instead of \number#3%
305 }%
306 \fi
307 \ifnum\count@<#2\relax
308 \advance\count@ 1 %
309 \repeat
310 }

```

```

311 \def\space{ }
312 \expandafter\ifx\csname LoadCommand\endcsname\relax
313 \def\LoadCommand{\input grfext.sty\relax}%
314 \fi
315 \def\Test{%
316 \RangeCatcodeInvalid{0}{47}%
317 \RangeCatcodeInvalid{58}{64}%
318 \RangeCatcodeInvalid{91}{96}%
319 \RangeCatcodeInvalid{123}{255}%
320 \catcode'\@=12 %
321 \catcode'\=0 %
322 \catcode'\%=14 %
323 \LoadCommand
324 \RangeCatcodeCheck{0}{36}{15}%
325 \RangeCatcodeCheck{37}{37}{14}%
326 \RangeCatcodeCheck{38}{47}{15}%
327 \RangeCatcodeCheck{48}{57}{12}%
328 \RangeCatcodeCheck{58}{63}{15}%
329 \RangeCatcodeCheck{64}{64}{12}%
330 \RangeCatcodeCheck{65}{90}{11}%
331 \RangeCatcodeCheck{91}{91}{15}%
332 \RangeCatcodeCheck{92}{92}{0}%
333 \RangeCatcodeCheck{93}{96}{15}%
334 \RangeCatcodeCheck{97}{122}{11}%
335 \RangeCatcodeCheck{123}{255}{15}%
336 \RestoreCatcodes
337 }
338 \Test
339 \csname @@end\endcsname
340 \end
341 </test1>

```

## 3.2 Macro tests

### 3.2.1 Preamble

```

342 <*test2>
343 \NeedsTeXFormat{LaTeX2e}
344 \nofiles
345 \documentclass{article}
346 \usepackage[pdftex]{graphics}
347 \usepackage{grfext}[2010/08/19]
348 \usepackage{qstest}
349 \IncludeTests{*}
350 \LogTests{log}{*}{*}
351 \makeatletter

```

### 3.2.2 Detection of unwanted space

```

352 \let\orig@qstest\qstest
353 \let\orig@endqstest\endqstest
354 \def\qstest#1#2{%
355 \orig@qstest{#1}{#2}%
356 \setbox0\hbox\bgroup\beginingroup\ignorespaces
357 }
358 \def\endqstest{%
359 \endgroup\egroup
360 \Expect*{\the\wd0}{0.0pt}%
361 \orig@endqstest
362 }

```

### 3.2.3 Generic test macros

```
363 \def\Test{\@ifnextchar\bgroup\TestInitList\TestInitCmd}%
364 \def\TestInitList#1{%
365   \def\Gin@extensions{#1}%
366   \@Test
367 }
368 \def\TestInitCmd#1{%
369   \let\Gin@extensions#1\relax
370   \@Test
371 }
372 \def\@Test#1#2{%
373   \TestCmd{#1}%
374   \expandafter\Expect\expandafter*{\Gin@extensions}{#2}%
375 }
```

### 3.2.4 Test for \AppendGraphicsExtensions

```
376 \begin{qstest}{\AppendGraphicsExtensions}{\AppendGraphicsExtensions}
377   \let\TestCmd\AppendGraphicsExtensions
378   \Test\@undefined{.test}{.test}%
379   \Test\relax{.test}{.test}%
380   \Test{}{.test}{.test}%
381   \Test{.foo}{.test}{.foo,.test}%
382   \Test{.foo,.bar}{.test}{.foo,.bar,.test}%
383   \Test{.test}{.test}{.test}%
384   \Test{.foo,.test}{.test}{.foo,.test}%
385   \Test{.test,.foo}{.test}{.foo,.test}%
386   \Test{.foo,.test,.bar}{.test}{.foo,.bar,.test}%
387   \Test{.foo}{.test,.bar}{.foo,.test,.bar}%
388   \Test{.foo}{.test,.foo}{.test,.foo}%
389   \Test{.foo}{.test,.bar,.x,.y}{.foo,.test,.bar,.x,.y}%
390   \def\TestCmd{\AppendGraphicsExtensions*}%
391   \Test\@undefined{.test}{}%
392   \Test\relax{.test}{}%
393   \Test{}{.test}{}%
394   \Test{.test}{.test}{.test}%
395   \Test{.foo}{.test}{.foo}%
396   \Test{.foo,.bar}{.test}{.foo,.bar}%
397   \Test{.test}{.test}{.test}%
398   \Test{.foo,.test}{.test}{.foo,.test}%
399   \Test{.test,.foo}{.test}{.test,.foo}%
400   \Test{.foo,.test,.bar}{.test}{.foo,.test,.bar}%
401   \Test{.foo}{.test,.bar}{.foo}%
402   \Test{.foo}{.test,.foo}{.foo}%
403   \DeclareGraphicsRule{.test}{test}{.test}{}%
404   \Test\@undefined{.test}{.test}%
405   \Test\relax{.test}{.test}%
406   \Test{}{.test}{.test}%
407   \Test{.foo}{.test}{.foo,.test}%
408   \Test{.foo,.bar}{.test}{.foo,.bar,.test}%
409   \Test{.test}{.test}{.test}%
410   \Test{.foo,.test}{.test}{.foo,.test}%
411   \Test{.test,.foo}{.test}{.foo,.test}%
412   \Test{.foo,.test,.bar}{.test}{.foo,.bar,.test}%
413   \Test{.foo}{.test,.bar}{.foo,.test}%
414   \Test{.foo}{.test,.foo}{.foo,.test}%
415   \DeclareGraphicsRule{.bar}{bar}{.bar}{}%
416   \Test{.foo}{.test,.bar}{.foo,.test,.bar}%
417   \Test{.bar}{.test,.bar}{.test,.bar}%

```

```

418 \Test{.bar}{.bar,.test}{.bar,.test}%
419 \end{qstest}

```

### 3.2.5 Test for \PrependGraphicsExtensions

```

420 \begin{qstest}{PrependGraphicsExtensions}{PrependGraphicsExtensions}
421 \let\TestCmd\PrependGraphicsExtensions
422 \Test\@undefined{.test}{.test}%
423 \Test\relax{.test}{.test}%
424 \Test{}{.test}{.test}%
425 \Test{.foo}{.test}{.test,.foo}%
426 \Test{.foo,.bar}{.test}{.test,.foo,.bar}%
427 \Test{.test}{.test}{.test}%
428 \Test{.foo,.test}{.test}{.test,.foo}%
429 \Test{.test,.foo}{.test}{.test,.foo}%
430 \Test{.foo,.test,.bar}{.test}{.test,.foo,.bar}%
431 \Test{.foo}{.test,.bar}{.test,.bar,.foo}%
432 \Test{.foo}{.test,.foo}{.test,.foo}%
433 \Test{.foo}{.test,.bar,.x,.y}{.test,.bar,.x,.y,.foo}%
434 \def\TestCmd{\PrependGraphicsExtensions*}%
435 \Test\@undefined{.test}{}%
436 \Test\relax{.test}{}%
437 \Test{}{.test}{}%
438 \Test{.test}{.test}{.test}%
439 \Test{.foo}{.test}{.foo}%
440 \Test{.foo,.bar}{.test}{.foo,.bar}%
441 \Test{.test}{.test}{.test}%
442 \Test{.foo,.test}{.test}{.foo,.test}%
443 \Test{.test,.foo}{.test}{.test,.foo}%
444 \Test{.foo,.test,.bar}{.test}{.foo,.test,.bar}%
445 \DeclareGraphicsRule{.test}{test}{.test}{}%
446 \Test\@undefined{.test}{.test}%
447 \Test\relax{.test}{.test}%
448 \Test{}{.test}{.test}%
449 \Test{.foo}{.test}{.test,.foo}%
450 \Test{.foo,.bar}{.test}{.test,.foo,.bar}%
451 \Test{.test}{.test}{.test}%
452 \Test{.foo,.test}{.test}{.test,.foo}%
453 \Test{.test,.foo}{.test}{.test,.foo}%
454 \Test{.foo,.test,.bar}{.test}{.test,.foo,.bar}%
455 \Test{.foo}{.test,.bar}{.test,.foo}%
456 \Test{.foo}{.test,.foo}{.test,.foo}%
457 \DeclareGraphicsRule{.bar}{bar}{.bar}{}%
458 \Test{.foo}{.test,.bar}{.test,.bar,.foo}%
459 \Test{.bar}{.test,.bar}{.test,.bar}%
460 \Test{.bar}{.bar,.test}{.bar,.test}%
461 \end{qstest}

```

### 3.2.6 Test for \RemoveGraphicsExtensions

```

462 \begin{qstest}{RemoveGraphicsExtensions}{RemoveGraphicsExtensions}
463 \let\TestCmd\RemoveGraphicsExtensions
464 \Test\@undefined{.test}{}%
465 \Test\relax{.test}{}%
466 \Test{}{.test}{}%
467 \Test{.foo}{.test}{.foo}%
468 \Test{.foo,.bar}{.test}{.foo,.bar}%
469 \Test{.test}{.test}{}%
470 \Test{.foo,.test}{.test}{.foo}%
471 \Test{.test,.foo}{.test}{.foo}%

```

```

472 \Test{.foo,.test,.bar}{.test}{.foo,.bar}%
473 \Test{.test,.test}{.test}{}%
474 \Test{.foo,.test,.bar,.test,.test,.bar}{.test}{.foo,.bar,.bar}%
475 \Test{.foo,.test,.bar,.foo,.test,.bar}{.test,.bar}{.foo,.foo}%
476 \Test{.foo,.test,.bar,.foo,.test,.bar}{.test,.bar}{.foo,.foo}%
477 \end{qstest}

478 \begin{document}
479 \end{document}
480 \end{test2}

```

## 4 Installation

### 4.1 Download

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

[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<sub>E</sub>X 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<sub>E</sub>X:

```
tex grfext.dtx
```

---

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

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

```
grfext.sty          → tex/latex/oberdiek/grfext.sty
grfext.pdf          → doc/latex/oberdiek/grfext.pdf
test/grfext-test1.tex → doc/latex/oberdiek/test/grfext-test1.tex
test/grfext-test2.tex → doc/latex/oberdiek/test/grfext-test2.tex
grfext.dtx          → source/latex/oberdiek/grfext.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  $\text{\TeX}$  distribution (`te $\text{\TeX}$` , `mik $\text{\TeX}$` , ...) relies on file name databases, you must refresh these. For example, `te $\text{\TeX}$`  users run `texhash` or `mktextlsr`.

## 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  $\text{\LaTeX}$ .** The `.dtx` chooses its action depending on the format:

**plain  $\text{\TeX}$ :** Run `docstrip` and extract the files.

**$\text{\LaTeX}$ :** Generate the documentation.

If you insist on using  $\text{\LaTeX}$  for `docstrip` (really, `docstrip` does not need  $\text{\LaTeX}$ ), 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 `pdf $\text{\LaTeX}$` :

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

## 5 Catalogue

The following XML file can be used as source for the [T<sub>E</sub>X Catalogue](#). The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `grfext.xml`.

```
481 (*catalogue)
482 <?xml version='1.0' encoding='us-ascii'?>
483 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
484 <entry datestamp='$Date$' modifier='$Author$' id='grfext'>
485   <name>grfext</name>
486   <caption>Manipulate the graphics package's list of extensions.</caption>
487   <authorref id='auth:oberdiek'>
488     <copyright owner='Heiko Oberdiek' year='2007,2010'>/>
489     <license type='lppl1.3'>/>
490     <version number='1.1'>/>
491     <description>
492       This package provides macros for adding to, and reordering the list of
493       graphics file extensions recognised by package
494       <xref refid='graphics'>graphics</xref>.
495     <p/>
496     The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
497   </description>
498   <documentation details='Package documentation'
499     href='ctan:/macros/latex/contrib/oberdiek/grfext.pdf'>/>
500   <ctan file='true' path='/macros/latex/contrib/oberdiek/grfext.dtx'>/>
501   <miktex location='oberdiek'>/>
502   <texlive location='oberdiek'>/>
503   <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'>/>
504 </entry>
505 </catalogue>
```

## 6 References

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

## 7 History

[2007/09/30 v1.0]

- First public version.

[2010/08/19 v1.1]

- User macros are also made available as keyval options for package `graphicx`.

## 8 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; plain numbers refer to the code lines where the entry is used.

<b>Symbols</b>	<b>\%</b> . . . . .	<b>322</b>
<b>\#</b> . . . . .	<b>246</b>	<b>\@</b> . . . . . 247, 320

\@PackageInfo	207		
\@Test	366, 370, 372		
\@empty	118, 126, 135, 143, 155, 162, 166, 173, 186		
\@expandtwoargs	110, 190		
\@firstofone	255, 258		
\@for	163, 187		
\@gobble	123, 140, 252, 260		
\@ifnextchar	363		
\@ifstar	120, 137		
\@ifundefined	110, 117, 134, 164, 183, 209		
\@removeelement	191		
\@undefined	58, 378, 391, 404, 422, 435, 446, 464		
\\	321		
\{	244		
\}	245		
<b>A</b>		<b>G</b>	
\advance	285, 293, 308	\Gin@extensions	118, 127, 135, 144, 155, 156, 158, 184, 189, 191, 192, 212, 365, 369, 374
\aftergroup	29	\grfext@@Add	120, 137, 153, 177
\AppendGraphicsExtensions	2, 116, 131, 220, 223, 377, 390	\grfext@@Print	130, 147, 217
<b>B</b>		\grfext@Add	125, 142, 150
\begin	376, 420, 462, 478	\grfext@Append	120, 122
\body	264, 268	\grfext@AtEnd	95, 96, 109, 237
<b>C</b>		\grfext@Check	120, 137, 161
\catcode	2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 69, 70, 72, 73, 74, 78, 79, 80, 81, 82, 83, 84, 87, 88, 90, 91, 92, 93, 97, 99, 244, 245, 246, 247, 282, 291, 299, 303, 320, 321, 322	\grfext@ext	163, 164, 167, 169, 187, 191
\count@	249, 278, 282, 284, 285, 289, 291, 292, 293, 297, 299, 302, 303, 307, 308	\grfext@next	124, 129, 141, 146, 174, 176, 180, 188, 193, 195, 197
\countdef	249	\grfext@Prepend	137, 139
\csname	14, 21, 50, 66, 76, 248, 251, 254, 257, 312, 339	\grfext@Print	123, 130, 131, 140, 147, 148, 200, 204, 206, 217
<b>D</b>		\grfext@tmp	162, 166, 167, 169, 173, 177, 186, 187, 189, 192
\DeclareGraphicsRule	403, 415, 445, 457	<b>H</b>	
\do	163, 187	\hbox	356
\documentclass	345	<b>I</b>	
<b>E</b>		\ifnum	284, 292, 299, 307
\empty	17, 18	\ifx	15, 18, 21, 50, 58, 61, 155, 166, 173, 192, 248, 251, 254, 257, 312
\end	340, 419, 461, 477, 479	\ignorespaces	356
\endcsname	14, 21, 50, 66, 76, 248, 251, 254, 257, 312, 339	\immediate	23, 52
\endinput	29, 109	\IncludeTests	349
\endlinechar	4, 35, 71, 77, 89	\input	240, 313
\endqstest	353, 358	\iterate	265, 267, 269
\errmessage	301	<b>K</b>	
\Expect	360, 374	\kv@define@key	219, 222, 225, 228, 231, 234
<b>F</b>		<b>L</b>	
<b>G</b>		\LoadCommand	313, 323
<b>H</b>		\LogTests	350
<b>I</b>		\loop	263, 279, 290, 298
<b>J</b>		<b>M</b>	
<b>K</b>		\makeatletter	351
<b>L</b>		\MessageBreak	208, 213
<b>M</b>		<b>N</b>	
<b>N</b>		\NeedsTeXFormat	343
<b>O</b>		\newcommand	116, 133, 182
<b>P</b>		\next	269, 271, 273
<b>Q</b>		\nofiles	344
<b>R</b>		\number	304
<b>S</b>		<b>O</b>	
<b>T</b>		\orig@endqstest	353, 361
<b>U</b>		\orig@qstest	352, 355
<b>V</b>		<b>P</b>	
<b>W</b>		\PackageInfo	26
<b>X</b>		\PrependGraphicsExtensions	133, 148, 226, 229, 421, 434
<b>Y</b>		\PrintGraphicsExtensions	3, 203, 235



<code>\ProvidesPackage</code> . . . . .	19, 67	412, 413, 414, 416, 417, 418, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 458, 459, 460, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476
<b>Q</b>		
<code>\qstest</code> . . . . .	352, 354	
<b>R</b>		
<code>\RangeCatcodeCheck</code> . . . . .		
. . . . .	296, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335	
<code>\RangeCatcodeInvalid</code> . . . . .		
. . . . .	288, 316, 317, 318, 319	
<code>\RemoveGraphicsExtensions</code> . . . . .		
. . . . .	2, 154, 182, 232, 463	
<code>\repeat</code> . . . . .	263, 275, 286, 294, 309	
<code>\RequirePackage</code> . . . . .	202, 218	
<code>\reserved@a</code> . . . . .	112, 113	
<code>\resetatcatcode</code> . . . . .	241	
<code>\RestoreCatcodes</code> . . . . .	277, 280, 281, 336	
<b>S</b>		
<code>\setbox</code> . . . . .	356	
<code>\space</code> . . . . .	302, 303, 311	
<b>T</b>		
<code>\Test</code> . . . . .	315, 338, 363, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 404, 405, 406, 407, 408, 409, 410, 411,	
<code>\TestCmd</code> . . . . .	373, 377, 390, 421, 434, 463	
<code>\TestInitCmd</code> . . . . .	363, 368	
<code>\TestInitList</code> . . . . .	363, 364	
<code>\the</code> . . . . .	77, 78, 79, 80, 81, 82, 83, 84, 97, 282, 302, 303, 360	
<code>\TMP@EnsureCode</code> . . . . .	94, 101, 102, 103, 104, 105, 106, 107, 108	
<b>U</b>		
<code>\usepackage</code> . . . . .	346, 347, 348	
<b>W</b>		
<code>\wd</code> . . . . .	360	
<code>\write</code> . . . . .	23, 52	
<b>X</b>		
<code>\x</code> . . . . .	14, 15, 18, 22, 26, 28, 51, 56, 66, 75, 87	
<b>Z</b>		
<code>\zap@space</code> . . . . .	126, 143, 186	