

# The texdepends package\*

Vincent DANJEAN

2018/09/04

## Abstract

This package allows L<sup>A</sup>T<sub>E</sub>X to automatically generate dependencies while compiling documents.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Implementation of texdepends</b>	<b>2</b>
2.1	Options . . . . .	2
2.2	Tools . . . . .	3
2.2.1	Debug/info wrapper . . . . .	3
2.2.2	Writing dependencies . . . . .	4
2.2.3	Diverting a macro . . . . .	6
2.2.4	Initialisation . . . . .	7
2.2.5	Warning upon missing figures . . . . .	8
2.3	Diversion of L <sup>A</sup> T <sub>E</sub> X macros . . . . .	8
2.3.1	Class and package loading . . . . .	8
2.3.2	Tracking input files . . . . .	9
2.3.3	Tracking output files . . . . .	10
2.3.4	Tracking bibliography files . . . . .	10
2.3.5	Glossary and index files . . . . .	11
2.4	Diversion of package's macros . . . . .	11
2.4.1	Package graphics . . . . .	11
2.4.2	Package figlatex . . . . .	12
2.4.3	Package doc . . . . .	15
2.4.4	Package index . . . . .	15
2.4.5	Package makeidx . . . . .	15
2.4.6	Package nomencl . . . . .	15
2.4.7	Package bibtopic . . . . .	16
2.4.8	Package comment . . . . .	16

---

\*This file has version number v2.3.0, last revised 2018/09/04.

# 1 Introduction

This package automatically generates dependancies on L<sup>A</sup>T<sub>E</sub>X files and packages, figures, bibliography files, glossary and index files. The generated file is then used by a Makefile to handle dependancies.

## 2 Implementation of texdepends

### 2.1 Options

```
1 <*package>
```

texdepends is easier to write with the ifthen package.

```
2 \RequirePackage{ifthen}
```

**debug** The debug option (enable debug messages, useful for debugging)

```
3 \newboolean{TD@debug}
4 \newcommand{\TD@option@debug}[1][true]{%
5   \%PackageWarningNoLine{texdepends}{Setting debug to #1}
6   \setboolean{TD@debug}{#1}%
7 }%
```

**extension** The extension option. We can use it to let texdepends know whether we compile a dvi or a pdf file.

```
8 \def\TD@option@extension[#1]{%
9   \%PackageWarningNoLine{texdepends}{Using extention #1}
10  \def\TD@extention{#1}%
11 }%
```

Managing options with specific code. Using the keyval package could be a solution, but this would add one more dependency.

```
12 \def\TD@option@split#1=#2=#3\relax{%
13   \def\TD@option@name{#1}%
14   \ifx\TD@option@name\@empty\else
15     \expandafter\let\expandafter\TD@option@cmd
16     \csname TD@option@\TD@option@name\endcsname
17     \%PackageWarningNoLine{texdepends}{1: '#1', 2: '#2', 3: '#3'}
18     \ifx\TD@option@cmd\relax
19       \@unknownoptionerror%
20     \else
21       \ifx\@empty#3\@empty
22         \TD@option@cmd\relax%
23       \else
24         \def\TD@option@value{#2}%
25         \TD@option@cmd[\TD@option@value]%
26       \fi
27     \fi
28   \fi
29 }%
```

All option are managed by the same code.

```
30 \DeclareOption*{%
31   %\PackageWarningNoLine{texdepends}{Option \CurrentOption}%
32   \expandafter\TD@option@split\CurrentOption==\relax%
33 }%
```

So lets process options

```
34 \ProcessOptions
```

If the `extension` option is not used, we try to find ourselves the good extension based on TeX engine or mode.

```
35 \ifthenelse{\isundefined{\TD@extention}}{%
36   \PackageWarningNoLine{texdepends}{No extension given: detecting it}%
37   \RequirePackage{ifxetex,ifpdf}%
38   \ifxetex%
39     \PackageWarningNoLine{texdepends}{Detected XeTeX: using .pdf}%
40     \def\TD@extention{.pdf}%
41   \else\ifpdf%
42     \PackageWarningNoLine{texdepends}{Detected PDF mode: using .pdf}%
43     \def\TD@extention{.pdf}%
44   \else%
45     \PackageWarningNoLine{texdepends}{Detected DVI mode. Using .dvi}%
46     \def\TD@extention{.dvi}%
47   \fi\fi%
48 }{}%
```

## 2.2 Tools

Some macros to help development

### 2.2.1 Debug/info wrapper

```
\TD@warning To write a warning
49 \newcommand{\TD@warning}[1]{%
50   \PackageWarningNoLine{texdepends}{#1}%
51 }%

\TD@warningLine To write a warning with line info
52 \newcommand{\TD@warningLine}[1]{%
53   \PackageWarning{texdepends}{#1}%
54 }%

\TD@info To write an info (as a warning if the debug option is set
55 \newcommand{\TD@info}[1]{%
56   \ifthenelse{\boolean{TD@debug}}{%
57     \PackageWarningNoLine{texdepends}{#1}%
58   }{%
59     \PackageInfo{texdepends}{#1}%
60   }%
61 }%
```

`\TD@debug` To write a message if the `debug` option is set

```
62 \newcommand{\TD@debug}[1]{%
63   \ifthenelse{\boolean{TD@debug}}{%
64     \PackageWarningNoLine{texdepends}{#1}%
65   }{%
66   }%
67 }%
```

### 2.2.2 Writing dependencies

We open the file where we will write dependencies and other infos.

```
68 \newwrite\TD@write
69 \TD@info{Writing info in '\jobname\TD@extention.mk'}
70 \immediate\openout\TD@write\jobname\TD@extention.mk%
```

And we define a few macro to write in this file

`\TD@print` We write something in the file

```
71 \def\TD@print#1{%
72   \immediate\write\TD@write{#1}%
73 }%
```

`\TD@printClass` We need a class

```
74 \def\TD@printClass#1{%
75   \TD@print{TD_\jobname\TD@extention _INPUTS\space += #1.cls}%
76 }%
```

`\TD@printPackage` We need a package

```
77 \def\TD@printPackage#1{%
78   \TD@print{TD_\jobname\TD@extention _INPUTS\space += #1.sty}%
79 }%
```

`\TD@printInput` We need a (L<sup>A</sup>T<sub>E</sub>X) file

```
80 \def\TD@printInput#1{%
81   \TD@print{TD_\jobname\TD@extention _INPUTS\space += #1}%
82 }%
```

`\TD@printFig` We want to ignore a dependency

```
83 \def\TD@printNoDependsFile#1{%
84   \TD@print{TD_\jobname\TD@extention _DEPENDS_EXCLUDE\space += #1}%
85 }%
```

`\TD@printFig` We need an XFig file

```
86 \def\TD@printFig#1{%
87   \TD@print{TD_\jobname\TD@extention _FIGURES\space += #1}%
88 }%
```

```

\TD@printSubfig Web need a subfig file (see figlatex.dtx)
89 \def\TD@printSubfig#1{%
90 \TD@print{TD_\jobname\TD@extention _SUBFIGS\space += #1}%
91 }%

\TD@printRequiredFile A file is missing. texdepend will try to skip it this time, but it will be needed at
the next compilation
92 \def\TD@printRequiredFile#1{%
93 \TD@print{TD_\jobname\TD@extention _REQUIRED\space += #1}%
94 }%

\RequireFile LaTeX users can directly declare a dependencie
95 \let\RequireFile\TD@printRequiredFile

\TD@printOut We write in a file
96 \def\TD@printOut#1{%
97 \TD@print{TD_\jobname\TD@extention _OUTPUTS\space += #1}%
98 }%

\TD@printGraphicspath We modify the graphicspath
99 \def\TD@printGraphicspath#1{%
100 \TD@print{TD_\jobname\TD@extention _GRAPHICSPATH\space := #1}%
101 }%

\TD@printBibfile We use a bibfile (*.bib)
102 \def\TD@printBibfile#1{%
103 \TD@print{TD_\jobname\TD@extention _BIBFILES\space += #1.bib}%
104 }%

\TD@printBibstyle We use a bibstyle file (*.bst)
105 \def\TD@printBibstyle#1{%
106 \TD@print{TD_\jobname\TD@extention _BIBSTYLES\space += #1.bst}%
107 }%

\TD@printBblfile We use a bibliography file (*.bbl)
108 \def\TD@printBblfile#1{%
109 \TD@print{TD_\jobname\TD@extention _BBLFILES\space += #1.bbl}%
110 }%

\TD@printRebuildRule We need a specific rule in the Makefile to check if a rebuild is needed
111 \def\TD@printRebuildRule#1{%
112 \TD@print{TD_\jobname\TD@extention _REBUILD_RULES\space += #1}%
113 }%

\TD@printIndex We use a new index
114 \def\TD@printIndex#1{%
115 \TD@print{TD_\jobname\TD@extention _INDEXES\space += INDEX}%
116 \TD@print{TD_\jobname\TD@extention _INDEXES_INDEX\space += #1}%
117 }%

```

```

\TD@printIndexSource The index file filled by LATEX
118 \def\TD@printIndexSource#1#2{%
119 \TD@print{TD_\jobname\TD@extention _INDEX_#1_SRC\space = #2}%
120 }%

\TD@printIndexTarget And the index file read by LATEX after makeindex processing
121 \def\TD@printIndexTarget#1#2{%
122 \TD@print{TD_\jobname\TD@extention _INDEX_#1_TARGET\space = #2}%
123 }%

\TD@printIndexStyle The index style to be used with makeindex (never called for now)
124 \def\TD@printIndexStyle#1#2{%
125 \TD@print{TD_\jobname\TD@extention _INDEX_#1_STYLE\space = #2}%
126 }%

\TD@printGlossary And the same things for glossary. Its name
127 \def\TD@printGlossary#1{%
128 \TD@print{TD_\jobname\TD@extention _INDEXES += GLOSS}%
129 \TD@print{TD_\jobname\TD@extention _INDEXES_GLOSS\space += #1}%
130 }%

\TD@printGlossarySource The glossary file filled by LATEX
131 \def\TD@printGlossarySource#1#2{%
132 \TD@print{TD_\jobname\TD@extention _GLOSS_#1_SRC\space = #2}%
133 }%

\TD@printGlossaryTarget And the glossary file read by LATEX after makeindex processing
134 \def\TD@printGlossaryTarget#1#2{%
135 \TD@print{TD_\jobname\TD@extention _GLOSS_#1_TARGET\space = #2}%
136 }%

\TD@printGlossaryStyle The glossary style to be used with makeindex (never called for now)
137 \def\TD@printGlossaryStyle#1#2{%
138 \TD@print{TD_\jobname\TD@extention _GLOSS_#1_STYLE\space = #2}%
139 }%

```

### 2.2.3 Diverting a macro

```

\TD@switch \TD@switch replace the macro \#1 by the macro \TD@#1. The original macro is
stored in the \TD@orig@#1 macro. Nothing is done if \#1 is already \TD@#1 (that
is if \TD@switch is called several times with the same argument)
140 \def\TD@switch#1{%
141 \expandafter\let\expandafter\TD@local@old\csname#1\endcsname%
142 \expandafter\let\expandafter\TD@local@new\csname TD@#1\endcsname%
143 \ifx\TD@local@old\TD@local@new
144 \PackageWarning{texdepends}{#1 already redefined. Skipping it.}%
145 \else
146 \ifx\TD@local@old\relax

```

```

147     \TD@warning{Diverting empty macro '#1' \MessageBreak%
148         Probably a bug in texdepends.sty}%
149     \else
150     \TD@info{Diverting macro '#1'}%
151     \fi
152     \expandafter\let\csname TD@orig@#1\endcsname\TD@local@old%
153     \expandafter\let\csname#1\endcsname\TD@local@new%
154     \fi
155 }%

```

\TD@docswitch Sometimes, we want the diversion to be delayed until the start of the document.

```

156 \def\TD@docswitchlist{}%
157 \def\TD@docswitch#1{%
158     \TD@info{Registering macro '#1' for diversion}%
159     \ifx\TD@docswitchlist\@empty%
160         \xdef\TD@docswitchlist{#1}%
161     \else%
162         \xdef\TD@docswitchlist{\TD@docswitchlist,#1}%
163     \fi%
164 }%
165 \AtBeginDocument{%
166     \TD@info{Diverting registered macros...}%
167     \@for\TD@cmd:=\TD@docswitchlist\do{%
168         \TD@switch{\TD@cmd}%
169     }%
170     \TD@info{Divertions done}%
171 }%

```

\TD@PackagePreDivert And then, two macros to insert command just before or after loading a package.

\TD@PackagePostDivert The first argument is the name of the package we want to add an hook before or after, and then there is the definition of the hook that will be called. Four parameters (#1, ..., #4) are available (the same as the @onefilewithoptions L<sup>A</sup>T<sub>E</sub>X macro, ie package name, options, and the two remaining ;-)).

```

172 \def\TD@PackagePreDivert#1{%
173     \expandafter\def\csname TD@PackageStart@#1\endcsname ##1##2##3##4%
174 }%
175 \def\TD@PackagePostDivert#1{%
176     \expandafter\def\csname TD@PackageEnd@#1\endcsname ##1##2##3##4%
177 }%

```

#### 2.2.4 Initialisation

We need to be loaded *before* \documentclass

```

178 \ifx\documentclass\@twoclasseserror
179     \PackageError{texdepends}{texdepends needs to be loaded BEFORE
180 \string\documentclass \MessageBreak
181 Use \string\RequirePackage\space instead of \string\usepackage
182 \MessageBreak as the later is not allowed before
183 \string\documentclass}

```

```

184 \fi
    And we write the first dependency (upon ourself)
185 \TD@printPackage{texdepends}
    We write default graphicspath (is this really needed ?)
186 \TD@printGraphicspath{{.}}

```

### 2.2.5 Warning upon missing figures

`\TD@depends` If some figures are missing, `\TD@missingDepends` will be called so that a warning  
`\TD@dependsWarning` will be displayed at the end of the compilation. Figures should be generated before  
`\TD@missingDepends` the next run.

```

187 \def\TD@depends{ }%
188 \def\TD@dependsWarning{%
189   \PackageWarningNoLine{texdepends}{Figure(s) missing. Check
190     dependencies again}%
191 }%
192 \def\TD@missingDepends{%
193   \global\let\TD@depends\TD@dependsWarning%
194 }%
195 \AtEndDocument{%
196   \TD@depends%
197 }%

```

## 2.3 Diversion of L<sup>A</sup>T<sub>E</sub>X macros

### 2.3.1 Class and package loading

`\TD@@fileswith@pti@ns` Diverting `\@fileswith@pti@ns` and `@@fileswith@pti@ns` so that we can track  
`\TD@@@fileswith@pti@ns` class loading. Package are tracked with an other macro (see just after).

```

198 \def\TD@@fileswith@pti@ns#1[#2]#3[#4]{%
199   \ifx#1\@clsextension
200     \TD@printClass{#3}
201   \fi
202   \TD@orig@@fileswith@pti@ns{#1}#2{#3}#4}%
203 }%
204 \let\TD@@@fileswith@pti@ns\TD@@fileswith@pti@ns
205 \TD@switch{@fileswith@pti@ns}%
206 \TD@switch{@@fileswith@pti@ns}%

```

`\TD@@onefilewithoptions` By diverting `\@onefilewithoptions`, we can track all loaded packages. We write  
the dependency, and we look at the loaded package to see if we need some special  
requirements

```

207 \def\TD@@onefilewithoptions#1[#2][#3]#4{%
208   \TD@printPackage{#1}%
209   \ifthenelse{equal{#1}{texdepends}}{ }%

```



The `texdepends` package is special : we know it has already been loaded (we are running code in it) but perhaps L<sup>A</sup>T<sub>E</sub>X does not know it. So we try to honor the `debug` option.

```

210 \TD@debug{Package ' #1 ' detected}
211 \@for\TD@package@option:=#2\do{%
212 \ifthenelse{\expandafter\isundefined\csname TD@option@\TD@package@option\endcsname}{
213 \TD@warningLine{Ignoring unknown option '\TD@package@option'}
214 }{
215 \ifthenelse{%
216 \equal{\TD@package@option}{debug}%
217 %\OR\equal{\TD@package@option}{debug}%
218 }{
219 \csname TD@option@\TD@package@option\endcsname%
220 }{
221 \TD@warningLine{Ignoring option '\TD@package@option'
222 \MessageBreak #1 already loaded
223 }
224 }
225 }
226 }
227 }{%
228 \expandafter\let\expandafter\TD@divert
229 \csname TD@PackageStart@#1\endcsname
230 \ifx\TD@divert\relax
231 \TD@debug{No preamble for package #1}
232 \else
233 \TD@info{Preamble for package #1}
234 \TD@divert{#1}{#2}{#3}{#4}%
235 \fi
236 \TD@orig@@onefilewithoptions{#1}[#2][#3]{#4}%
237 \expandafter\let\expandafter\TD@divert
238 \csname TD@PackageEnd@#1\endcsname
239 \ifx\TD@divert\relax
240 \TD@debug{No postamble for package #1}
241 \else
242 \TD@info{Postamble for package #1}
243 \TD@divert{#1}{#2}{#3}{#4}%
244 \fi
245 }%
246 }%
247 \TD@switch{@onefilewithoptions}%

```

### 2.3.2 Tracking input files

```

\TD@@input \@input and \input@ are diverted to track loaded files
\TD@@input@
248 \def\TD@@input#1{%
249 \TD@printInput{#1}%
250 \TD@orig@@input{#1}}%
251 \def\TD@@input@#1{%

```

```

252 \IfFileExists{#1}{%
253   \TD@printInput{#1}%
254 }{%
255   \TD@orig@input@{#1}}%
256 \TD@switch{@input}%
257 \TD@switch{@input@}%
258 \def\TD@verbatiminput#1{%
259   \TD@printInput{#1}%
260   \TD@orig@verbatiminput{#1}}%
261 \TD@PackagePostDivert{verbatim}{%
262   \TD@switch{verbatiminput}}%

```

### 2.3.3 Tracking output files

```

\TD@openout Note: that does not work. Currently, we parse the log file instead of using this. If
\TD@poenout@eq someone has ideas...
\TD@poenout@noeq 263 \let\TD@orig@openout\openout
264 \def\TD@openout#1{%
265   \ifnextchar={\TD@openout@eq#1}{\TD@openout@noeq#1}}
266 \def\TD@openout@eq#1=#2.#3{%
267   \TD@printOut{#2.#3}%
268   \TD@orig@openout#1=#2.#3%
269 }%
270 \def\TD@openout@noeq#1#2.#3{%
271   \TD@printOut{#2.#3}%
272   \TD@orig@openout#1#2.#3%
273 }%
274 %\TD@switch{openout}

```

### 2.3.4 Tracking bibliography files

```

275 \def\TD@bibli@bbl{%
276   \TD@printBblfile{\jobname}%
277   \global\let\TD@bibli@bbl\relax%
278 }%
279 \def\TD@bibli@files#1{%
280   \TD@bibli@bbl%
281   \@for\TD@bibfile:=#1\do{%
282     \TD@printBibfile{\TD@bibfile}%
283   }%
284 }
285 \def\TD@bibli@style#1{%
286   \TD@bibli@bbl%
287   \@for\TD@bibstyle:=#1\do{%
288     % \TD@printBibstyle{\TD@bibstyle}%
289   }%
290   \TD@printBibstyle{#1}%
291 }
292 \def\TD@bibliography#1{%

```

```

293 \TD@bibli@files{#1}%
294 \TD@orig@bibliography{#1}%
295 }%
296 \def\TD@bibliographystyle#1{%
297 \TD@bibli@style{#1}%
298 \TD@orig@bibliographystyle{#1}%
299 }%
300 \TD@docswitch{bibliography}%
301 \TD@switch{bibliographystyle}%

```

### 2.3.5 Glossary and index files

```

302 \def\TD@makeindex{%
303 \TD@printIndex{\jobname}%
304 \TD@printIndexSource{\jobname}{\jobname.idx}%
305 \TD@orig@makeindex%
306 }%
307 \def\TD@makeglossary{%
308 \TD@printGlossary{\jobname}%
309 \TD@printGlossarySource{\jobname}{\jobname.glo}%
310 \TD@orig@makeglossary%
311 }%
312 \TD@switch{makeindex}%
313 \TD@switch{makeglossary}%

```

## 2.4 Diversion of package's macros

### 2.4.1 Package graphics

```

314 \def\TD@Gin@getbase#1{%
315 \TD@orig@Gin@getbase{#1}%
316 \ifx\Gin@ext\relax%
317 \else%
318 \TD@printFig{\Gin@base\Gin@ext}%
319 \fi%
320 }%
321 \def\TD@Gin@include@eps#1{%
322 \IfFileExists{#1}{%
323 \TD@orig@Gin@include@eps{#1}%
324 }{%
325 \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
326 Skipping it this time (the last one however)\MessageBreak}%
327 \TD@missingDepends%
328 \TD@printRequiredFile{#1}%
329 }%
330 }%
331 \def\TD@Gin@include@pdf#1{%
332 \IfFileExists{#1}{%
333 \TD@orig@Gin@include@pdf{#1}%
334 }{%
335 \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
336 Skipping it this time (the last one however)\MessageBreak}%

```

```

337 \TD@missingDepends%
338 \TD@printRequiredFile{#1}%
339 }%
340 }%
341 \def\TD@graphicspath#1{%
342 \TD@printGraphicspath{#1}%
343 \TD@orig@graphicspath{#1}%
344 }%
345 \def\TD@Gread@eps#1{%
346 \IfFileExists{#1}{%
347 \PackageWarning{texdepends}{Gread File '1' exists}%
348 \TD@orig@Gread@eps{#1}%
349 }{%
350 \PackageWarning{texdepends}{No '1' file \MessageBreak
351 using 1 for graphic dimensions}%
352 \setbox\@tempboxa\hbox{\relax}%
353 \def\Gin@llx{1}\let\Gin@lly\Gin@llx
354 \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
355 \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
356 }%
357 }%
358 \def\TD@Gread@pdf#1{%
359 \IfFileExists{#1}{%
360 \PackageWarning{texdepends}{Gread File '1' exists}%
361 \TD@orig@Gread@pdf{#1}%
362 }{%
363 \PackageWarning{texdepends}{No '1' file \MessageBreak
364 using 1 for graphic dimensions}%
365 \setbox\@tempboxa\hbox{\relax}%
366 \def\Gin@llx{1}\let\Gin@lly\Gin@llx
367 \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
368 \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
369 }%
370 }%
371 \TD@PackagePostDivert{graphics}{%
372 \TD@switch{graphicspath}%
373 \TD@docswitch{Gin@getbase}%
374 \TD@docswitch{Gin@include@eps}%
375 \TD@docswitch{Gin@include@pdf}%
376 \TD@docswitch{Gread@eps}%
377 \TD@docswitch{Gread@pdf}%
378 }%

```

## 2.4.2 Package figlatex

```

379 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
380 % We do not want to fail when trying to load a missing subfig.
381 % \@@Gin@include@rawtex
382 \def\TD@FL@includerawtexgraphics[#1]#2{%
383 \IfFileExists{#2}{%
384 \TD@orig@FL@includerawtexgraphics[#1]{#2}%

```

```

385 }{%
386   \PackageWarning{texdepends}{Figure '#2' needed\MessageBreak
387     Skipping it this time (the last one however)\MessageBreak}%
388   \TD@missingDepends%
389   \TD@printRequiredFile{#2}%
390 }%
391 }%
392
393 \def\TD@Ginclude@rawtex#1{%
394   \IfFileExists{#1}{%
395     \TD@orig@Ginclude@rawtex{#1}%
396   }{%
397     \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
398       Skipping it this time (the last one however)\MessageBreak}%
399     \TD@missingDepends%
400     \TD@printRequiredFile{#1}%
401   }%
402 }%
403
404 % \@Ginclude@figtex
405 \def\TD@Ginclude@figtex#1{%
406   \IfFileExists{#1}{%
407     \PackageWarning{texdepends}{Ginclude File '#1' exists}%
408     \TD@orig@Ginclude@figtex{#1}%
409   }{%
410     \ifthenelse{\isundefined{\FL@subfig@mainfigname}}{%
411       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
412         Skipping it this time (the last one however)\MessageBreak}%
413     }{%
414       \PackageWarning{texdepends}{SubFigure '#1' from
415         \FL@subfig@mainfigname needed\MessageBreak
416         Skipping it this time (the last one however)\MessageBreak}%
417     }
418     \TD@missingDepends%
419     \TD@printRequiredFile{#1}%
420   }%
421 }%
422 % \@Ginclude@svgtex
423 \def\TD@Ginclude@svgtex#1{%
424   \IfFileExists{#1}{%
425     \PackageWarning{texdepends}{Ginclude File '#1' exists}%
426     \TD@orig@Ginclude@svgtex{#1}%
427   }{%
428     \ifthenelse{\isundefined{\FL@subfig@mainfigname}}{%
429       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
430         Skipping it this time (the last one however)\MessageBreak}%
431     }{%
432       \PackageWarning{texdepends}{SubFigure '#1' from
433         \FL@subfig@mainfigname needed\MessageBreak
434         Skipping it this time (the last one however)\MessageBreak}%

```

```

435     }
436     \TD@missingDepends%
437     \TD@printRequiredFile{#1}%
438 }%
439 }%
440
441
442 % \@Gread@figtex
443 \def\TD@Gread@figtex#1{%
444   \IfFileExists{#1}{%
445     \%PackageWarning{texdepends}{Gread File '1' exists}%
446     \TD@orig@Gread@figtex{#1}%
447   }{%
448     \PackageWarning{texdepends}{No '1' file \MessageBreak
449       using 1 for graphic dimensions}%
450     \setbox\@tempboxa\hbox{\relax}%
451     \def\Gin@llx{1}\let\Gin@lly\Gin@llx
452     \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
453     \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
454   }%
455 }%
456
457 % \@Gread@svgtx
458 \def\TD@Gread@svgtx#1{%
459   \IfFileExists{#1}{%
460     \%PackageWarning{texdepends}{Gread File '1' exists}%
461     \TD@orig@Gread@svgtx{#1}%
462   }{%
463     \PackageWarning{texdepends}{No '1' file \MessageBreak
464       using 1 for graphic dimensions}%
465     \setbox\@tempboxa\hbox{\relax}%
466     \def\Gin@llx{1}\let\Gin@lly\Gin@llx
467     \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
468     \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
469   }%
470 }%
471 % \FL@subfig@check
472 \def\TD@FL@subfig@check#1{%
473   \TD@printRequiredFile{#1}%
474   \TD@printSubfig{#1}%
475 }%
476 %
477 \TD@PackagePostDivert{figlax}{%
478   \TD@docswitch{FL@includerawtexgraphics}%
479   \TD@docswitch{Ginclude@rawtex}%
480   \TD@docswitch{Ginclude@figtex}%
481   \TD@docswitch{Ginclude@svgtx}%
482   \TD@docswitch{Gread@figtex}%
483   \TD@docswitch{Gread@svgtx}%
484   \TD@docswitch{FL@subfig@check}%

```

485 }%

### 2.4.3 Package doc

```
486 \def\TD@PrintIndex{%
487   \TD@printIndexStyle{\jobname}{gind.ist}
488   \TD@printIndexTarget{\jobname}{\jobname.ind}
489   \TD@orig@PrintIndex%
490 }%
491 \def\TD@PrintChanges{%
492   \TD@printGlossaryStyle{\jobname}{gglo.ist}
493   \TD@printGlossaryTarget{\jobname}{\jobname.gls}
494   \TD@orig@PrintChanges%
495 }%
496 \TD@PackagePostDivert{doc}{%
497   \TD@docswitch{PrintIndex}%
498   \TD@docswitch{PrintChanges}%
499 }%
```

### 2.4.4 Package index

```
500 \def\TD@def@index#1#2#3#4{%
501   \TD@printIndex{#2}%
502   \TD@printIndexSource{#2}{\jobname.#3}%
503   \TD@printIndexTarget{#2}{\jobname.#4}
504   \TD@orig@def@index{#1}{#2}{#3}{#4}%
505 }%
506 \TD@PackagePostDivert{index}{%
507   \TD@switch{def@index}%
508 }%
```

### 2.4.5 Package makeidx

```
509 \def\TD@printindex{%
510   \TD@printIndexTarget{\jobname}{\jobname.ind}
511   \TD@orig@printindex%
512 }%
513 \TD@PackagePostDivert{makeidx}{%
514   \TD@docswitch{printindex}%
515 }%
```

### 2.4.6 Package nomencl

We divert makeglossary again after loading this package as its redefines this macro (loosing our diversion). We also setup the glossary style for this package.

```
516 \def\TD@printglossary{%
517   \TD@printGlossaryTarget{\jobname}{\jobname.gls}
518   \TD@orig@printglossary%
519 }%
520 \TD@PackagePostDivert{nomencl}{%
521   \TD@docswitch{printglossary}%
522   \TD@switch{makeglossary}%
523   \TD@printGlossaryStyle{\jobname}{nomencl.ist}
```

524 }%

## 2.4.7 Package bibtopic

```

525 % For the bibtopic package
526 \def\TD@btSect{%
527   \@ifnextchar[{\TD@btSect@opt}{\TD@btSect@noopt}}
528 \def\TD@btSect@opt[#1]#2{%
529   \TD@bibli@files{#2}%
530   \TD@bibli@style{#1}%
531   \TD@orig@btSect[#1]{#2}}
532 \def\TD@btSect@noopt#1{%
533   \TD@bibli@files{#1}%
534   \TD@orig@btSect{#1}}
535
536 \def\TD@@bt@write@auxfile{%
537   \TD@printBblfile{\thebtauxfile}%
538   \TD@orig@@bt@write@auxfile%
539 }%
540 \TD@PackagePostDivert{bibtopic}{%
541   \global\let\TD@bibli@bbl\relax%
542   \global\let\TD@biblio@rules@package\TD@biblio@rules@package@bibtopic%
543   \TD@switch{@bt@write@auxfile}%
544   \TD@printOut{btbbl.aux}%
545   \TD@printRebuildRule{bibtopic}%
546   \TD@docswitch{btSect}%
547 }%
```

## 2.4.8 Package comment

```

548 % For the comment package, the |comment.cut| file must not be used as
549 % a dependency: it is read and written several times during a compilation.
550 \TD@PackagePostDivert{comment}{%
551   \TD@printNoDependsFile{comment.cut}%
552 }%
553 </package>
```

# 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		
\@Ginclude@figtex	404	\@clsextension 199
\@Ginclude@rawtex	381	\@for 167, 211, 281, 287
\@Ginclude@svgtex	422	\@ifnextchar 265, 527
\@Gread@figtex	442	\@tempboxa 352, 354, 355, 365, 367,
\@Gread@svgtex	457	368, 450, 452, 453, 465, 467, 468
		\@twoclasseserror 178



\@unknownoptionerror .....	19		
<b>A</b>			<b>R</b>
\AtBeginDocument .....	165	\RequireFile .....	<u>95</u>
\AtEndDocument .....	195	<b>S</b>	
<b>B</b>		\setboolean .....	6
\boolean .....	56, 63	<b>T</b>	
<b>C</b>		\TD@@@fileswith@pti@ns .....	<u>198</u>
\CurrentOption .....	31, 32	\TD@@bt@write@auxfile .....	<u>536</u>
<b>D</b>		\TD@@fileswith@pti@ns .....	<u>198</u>
debug (option) .....	<u>3</u>	\TD@@iinput .....	<u>248</u>
\documentclass .....	178, 180, 183	\TD@@input@ .....	<u>248</u>
<b>E</b>		\TD@@onefilewithoptions .....	<u>207</u>
extension (option) .....	<u>8</u>	\TD@bibfile .....	281, 282
<b>F</b>		\TD@bibli@bbl ..	275, 277, 280, 286, 541
\FL@subfig@check .....	471	\TD@bibli@files ..	279, 293, 529, 533
\FL@subfig@mainfigname .....		\TD@bibli@style .....	285, 297, 530
.....	410, 415, 428, 433	\TD@bibli@rules@package .....	542
<b>G</b>		\TD@bibli@rules@package@bibtopic .....	542
\Gin@base .....	318	\TD@bibliography .....	292
\Gin@defaultbp .....	354,	\TD@bibliographystyle .....	296
355, 367, 368, 452, 453, 467, 468		\TD@bibstyle .....	287, 288
\Gin@ext .....	316, 318	\TD@btSect .....	526
\Gin@llx .....	353, 366, 451, 466	\TD@btSect@noopt .....	527, 532
\Gin@lly .....	353, 366, 451, 466	\TD@btSect@opt .....	527, 528
\Gin@urx .....	354, 367, 452, 467	\TD@cmd .....	167, 168
\Gin@ury .....	355, 368, 453, 468	\TD@debug .....	<u>62</u> , 210, 231, 240
<b>I</b>		\TD@def@index .....	500
\ifxetex .....	38	\TD@depends .....	<u>187</u>
\isundefined .....	35, 212, 410, 428	\TD@dependsWarning .....	<u>187</u>
<b>N</b>		\TD@divert ..	228, 230, 234, 237, 239, 243
\newboolean .....	3	\TD@docswitch ..	<u>156</u> , 300, 373–377,
<b>O</b>		478–484, 497, 498, 514, 521, 546	
options:		\TD@docswitchlist .....	
debug .....	<u>3</u>	.....	156, 159, 160, 162, 167
extension .....	<u>8</u>	\TD@extention .....	10,
\OR .....	217	35, 40, 43, 46, 69, 70, 75, 78,	
<b>P</b>		81, 84, 87, 90, 93, 97, 100, 103,	
\PackageError .....	179	106, 109, 112, 115, 116, 119,	
\PackageInfo .....	59	122, 125, 128, 129, 132, 135, 138	
		\TD@FL@includerawtexgraphics ..	382
		\TD@FL@subfig@check .....	472
		\TD@Gin@getbase .....	314
		\TD@Gin@include@eps .....	321
		\TD@Gin@include@figtex .....	405
		\TD@Gin@include@pdf .....	331
		\TD@Gin@include@rawtex .....	393
		\TD@Gin@include@svgtex .....	423
		\TD@graphicspath .....	341

\TD@Gread@eps	345	\TD@orig@verbatiminput	260
\TD@Gread@figtex	443	\TD@package@option	
\TD@Gread@pdf	358	.....	211–213, 216, 217, 219, 221
\TD@Gread@svgtx	458	\TD@PackagePostDivert	
\TD@info	55,	.....	172, 261, 371,
69, 150, 158, 166, 170, 233, 242		477, 496, 506, 513, 520, 540, 550	
\TD@local@new	142, 143, 153	\TD@PackagePreDivert	172
\TD@local@old	141, 143, 146, 152	\TD@poenout@eq	263
\TD@makeglossary	307	\TD@poenout@noeq	263
\TD@makeindex	302	\TD@print	71, 75, 78,
\TD@missingDepends		81, 84, 87, 90, 93, 97, 100, 103,	
187, 327, 337, 388, 399, 418, 436		106, 109, 112, 115, 116, 119,	
\TD@openout	263	122, 125, 128, 129, 132, 135, 138	
\TD@openout@eq	265, 266	\TD@printBblfile	108, 276, 537
\TD@openout@noeq	265, 270	\TD@printBibfile	102, 282
\TD@option@cmd	15, 18, 22, 25	\TD@printBibstyle	105, 288, 290
\TD@option@debug	4	\TD@PrintChanges	491
\TD@option@extension	8	\TD@printClass	74, 200
\TD@option@name	13, 14, 16	\TD@printFig	83, 86, 318
\TD@option@split	12, 32	\TD@printGlossary	127, 308
\TD@option@value	24, 25	\TD@printglossary	516
\TD@orig@bt@write@auxfile	538	\TD@printGlossarySource	131, 309
\TD@orig@@fileswith@ptions	202	\TD@printGlossaryStyle	137, 492, 523
\TD@orig@Ginput	250	\TD@printGlossaryTarget	134, 493, 517
\TD@orig@Ginput@	255	\TD@printGraphicspath	99, 186, 342
\TD@orig@Gonefilewithoptions	236	\TD@PrintIndex	486
\TD@orig@bibliography	294	\TD@printIndex	114, 303, 501
\TD@orig@bibliographystyle	298	\TD@printindex	509
\TD@orig@btSect	531, 534	\TD@printIndexSource	118, 304, 502
\TD@orig@def@index	504	\TD@printIndexStyle	124, 487
\TD@orig@FL@includerawtexgraphics		\TD@printIndexTarget	
.....	384	.....	121, 488, 503, 510
\TD@orig@Gin@getbase	315	\TD@printInput	80, 249, 253, 259
\TD@orig@Gin@Ginclude@eps	323	\TD@printNoDependsFile	83, 551
\TD@orig@Gin@Ginclude@figtex	408	\TD@printOut	96, 267, 271, 544
\TD@orig@Gin@Ginclude@pdf	333	\TD@printPackage	77, 185, 208
\TD@orig@Gin@Ginclude@rawtex	395	\TD@printRebuildRule	111, 545
\TD@orig@Gin@Ginclude@svgtx	426	\TD@printRequiredFile	92, 95,
\TD@orig@Ggraphicspath	343	328, 338, 389, 400, 419, 437, 473	
\TD@orig@Gread@eps	348	\TD@printSubfig	89, 474
\TD@orig@Gread@figtex	446	\TD@switch	140, 168, 205,
\TD@orig@Gread@pdf	361	206, 247, 256, 257, 262, 274,	
\TD@orig@Gread@svgtx	461	301, 312, 313, 372, 507, 522, 543	
\TD@orig@makeglossary	310	\TD@verbatiminput	258
\TD@orig@makeindex	305	\TD@warning	49, 147
\TD@orig@openout	263, 268, 272	\TD@warningLine	52, 213, 221
\TD@orig@PrintChanges	494	\TD@write	68, 70, 72
\TD@orig@printglossary	518	\thebtauxfile	537
\TD@orig@PrintIndex	489		
\TD@orig@printindex	511		

## U

`\usepackage` . . . . . 181

## Change History

v1.0.1	subfig files . . . . . 1
General: Version 1.0.1 at last . . . 1	
v1.0.2	v1.3.0
General: Add support for package	General: Management of svg files 1
index.sty . . . . . 1	v2.2.5
v1.1.0	General: No changes in
General: Manage index/glossary . 1	texdepends.dtx . . . . . 1
Rewrite of the package . . . . . 1	v2.3.0
v1.2.0	General: ignore comment.cut file
General: Better management of	from the comment package . . 1