

The `texdepends` package*

Vincent DANJEAN

2021/01/03

Abstract

This package allows L^AT_EX to automatically generate dependencies while compiling documents.

Contents

1	Introduction	2
2	Implementation of <code>texdepends</code>	2
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 ^A T _E 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 <code>graphics</code>	11
2.4.2	Package <code>figlatex</code>	12
2.4.3	Package <code>doc</code>	15
2.4.4	Package <code>index</code>	15
2.4.5	Package <code>makeidx</code>	15
2.4.6	Package <code>nomencl</code>	15
2.4.7	Package <code>bibtopic</code>	16
2.4.8	Package <code>comment</code>	16

*This file has version number v2.4.2, last revised 2021/01/03.

1 Introduction

This package automatically generates dependencies on L^AT_EX files and packages, figures, bibliography files, glossary and index files. The generated file is then used by a Makefile to handle dependencies.

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 extension #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 options are managed by the same code.

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

So let's 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 \RequirePackage{xstring}%
50 \newcommand{\TD@warning}[1]{%
51   \PackageWarningNoLine{texdepends}{#1}%
52 }%
```

`\TD@warningLine` To write a warning with line info

```
53 \newcommand{\TD@warningLine}[1]{%
54   \PackageWarning{texdepends}{#1}%
55 }%
```

`\TD@info` To write an info (as a warning if the `debug` option is set)

```
56 \newcommand{\TD@info}[1]{%
57   \ifthenelse{\boolean{TD@debug}}{%
58     \PackageWarningNoLine{texdepends}{#1}%
59   }{%
60     \PackageInfo{texdepends}{#1}%
61   }%
62 }%
```

```
61 }%
62 }%
```

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

```
63 \newcommand{\TD@debug}[1]{%
64   \ifthenelse{\boolean{TD@debug}}{%
65     \PackageWarningNoLine{texdepends}{#1}%
66   }{%
67   }%
68 }%
```

2.2.2 Writing dependencies

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

```
69 \newwrite\TD@write
70 \TD@info{Writing info in '\jobname\TD@extention.mk'}
71 \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

```
72 \def\TD@print#1{%
73   {%
74     % TODO: fix pb when #1 contains '#'
75     \%noexpandarg%
76     \StrSubstitute{#1}{/}{/}[\TD@write@text]%
77     \immediate\write\TD@write{\TD@write@text}%
78     \%immediate\write\TD@write{#1}%
79   }
80 }%
```

`\TD@printClass` We need a class

```
81 \def\TD@printClass#1{%
82   \TD@print{TD_\jobname\TD@extention _INPUTS\space += #1.cls}%
83 }%
```

`\TD@printPackage` We need a package

```
84 \def\TD@printPackage#1{%
85   \TD@print{TD_\jobname\TD@extention _INPUTS\space += #1.sty}%
86 }%
```

`\TD@printInput` We need a (L^AT_EX) file

```
87 \def\TD@printInput#1{%
88   \TD@print{TD_\jobname\TD@extention _INPUTS\space += #1}%
89 }%
```

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

```
90 \def\TD@printNoDependsFile#1{%
91   \TD@print{TD_\jobname\TD@extention _DEPENDS_EXCLUDE\space += #1}%
92 }%
```

```

\TD@printFig We need an XFig file
93 \def\TD@printFig#1{%
94 \TD@print{TD_\jobname\TD@extention _FIGURES\space += #1}%
95 }%

\TD@printSubfig We need a subfig file (see figlatex.dtx)
96 \def\TD@printSubfig#1{%
97 \TD@print{TD_\jobname\TD@extention _SUBFIGS\space += #1}%
98 }%

\TD@printRequiredFile A file is missing. texdepend will try to skip it this time, but it will be needed at
the next compilation
99 \def\TD@printRequiredFile#1#2{%
100 \TD@print{TD_\jobname\TD@extention _REQUIRED\space += #1}%
101 }%

\RequireFile LaTeX users can directly declare a dependencie
102 \def\RequireFile#1{%
103 \TD@printRequiredFile{#1}{User}%
104 }%

\TD@printOut We write in a file
105 \def\TD@printOut#1{%
106 \TD@print{TD_\jobname\TD@extention _OUTPUTS\space += #1}%
107 }%

\TD@printGraphicspath We modify the graphicspath
108 \def\TD@printGraphicspath#1{%
109 \TD@print{TD_\jobname\TD@extention _GRAPHICSPATH\space := #1}%
110 }%

\TD@printBibfile We use a bibfile (*.bib)
111 \def\TD@printBibfile#1{%
112 \TD@print{TD_\jobname\TD@extention _BIBFILES\space += #1.bib}%
113 }%

\TD@printBibstyle We use a bibstyle file (*.bst)
114 \def\TD@printBibstyle#1{%
115 \TD@print{TD_\jobname\TD@extention _BIBSTYLES\space += #1.bst}%
116 }%

\TD@printBblfile We use a bibliography file (*.bbl)
117 \def\TD@printBblfile#1{%
118 \TD@print{TD_\jobname\TD@extention _BBLFILES\space += #1.bbl}%
119 }%

\TD@printRebuildRule We need a specific rule in the Makefile to check if a rebuild is needed
120 \def\TD@printRebuildRule#1{%
121 \TD@print{TD_\jobname\TD@extention _REBUILD_RULES\space += #1}%
122 }%

```

```

\TD@printIndex We use a new index
123 \def\TD@printIndex#1{%
124 \TD@print{TD_\jobname\TD@extention _INDEXES\space += INDEX}%
125 \TD@print{TD_\jobname\TD@extention _INDEXES_INDEX\space += #1}%
126 }%

```

```

\TD@printIndexSource The index file filled by LATEX
127 \def\TD@printIndexSource#1#2{%
128 \TD@print{TD_\jobname\TD@extention _INDEX_#1_SRC\space = #2}%
129 }%

```

```

\TD@printIndexTarget And the index file read by LATEX after makeindex processing
130 \def\TD@printIndexTarget#1#2{%
131 \TD@print{TD_\jobname\TD@extention _INDEX_#1_TARGET\space = #2}%
132 }%

```

```

\TD@printIndexStyle The index style to be used with makeindex (never called for now)
133 \def\TD@printIndexStyle#1#2{%
134 \TD@print{TD_\jobname\TD@extention _INDEX_#1_STYLE\space = #2}%
135 }%

```

```

\TD@printGlossary And the same things for glossary. Its name
136 \def\TD@printGlossary#1{%
137 \TD@print{TD_\jobname\TD@extention _INDEXES += GLOSS}%
138 \TD@print{TD_\jobname\TD@extention _INDEXES_GLOSS\space += #1}%
139 }%

```

```

\TD@printGlossarySource The glossary file filled by LATEX
140 \def\TD@printGlossarySource#1#2{%
141 \TD@print{TD_\jobname\TD@extention _GLOSS_#1_SRC\space = #2}%
142 }%

```

```

\TD@printGlossaryTarget And the glossary file read by LATEX after makeindex processing
143 \def\TD@printGlossaryTarget#1#2{%
144 \TD@print{TD_\jobname\TD@extention _GLOSS_#1_TARGET\space = #2}%
145 }%

```

```

\TD@printGlossaryStyle The glossary style to be used with makeindex (never called for now)
146 \def\TD@printGlossaryStyle#1#2{%
147 \TD@print{TD_\jobname\TD@extention _GLOSS_#1_STYLE\space = #2}%
148 }%

```

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)
149 \def\TD@switch#1{%

```

```

150 \expandafter\let\expandafter\TD@local@old\csname#1\endcsname%
151 \expandafter\let\expandafter\TD@local@new\csname TD@#1\endcsname%
152 \ifx\TD@local@old\TD@local@new
153   \PackageWarning{texdepends}{#1 already redefined. Skipping it.}%
154 \else
155   \ifx\TD@local@old\relax
156     \TD@warning{Diverting empty macro '#1' \MessageBreak%
157       Probably a bug in texdepends.sty}%
158   \else
159     \TD@info{Diverting macro '#1'}%
160   \fi
161   \expandafter\let\csname TD@orig@#1\endcsname\TD@local@old%
162   \expandafter\let\csname#1\endcsname\TD@local@new%
163 \fi
164 }%

```

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

```

165 \def\TD@docswitchlist{}%
166 \def\TD@docswitch#1{%
167   \TD@info{Registering macro '#1' for diversion}%
168   \ifx\TD@docswitchlist\@empty%
169     \xdef\TD@docswitchlist{#1}%
170   \else%
171     \xdef\TD@docswitchlist{\TD@docswitchlist,#1}%
172   \fi%
173 }%
174 \AtBeginDocument{%
175   \TD@info{Diverting registered macros...}%
176   \@for\TD@cmd:=\TD@docswitchlist\do{%
177     \TD@switch{\TD@cmd}%
178   }%
179   \TD@info{Divertions done}%
180 }%

```

`\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^AT_EX macro, ie package name, options, and the two remaining ;-)).

```

181 \def\TD@PackagePreDivert#1{%
182   \expandafter\def\csname TD@PackageStart@#1\endcsname ##1##2##3##4%
183 }%
184 \def\TD@PackagePostDivert#1{%
185   \expandafter\def\csname TD@PackageEnd@#1\endcsname ##1##2##3##4%
186 }%

```

2.2.4 Initialisation

We need to be loaded *before* `\documentclass`

```

187 \ifx\documentclass\@twoclasseserror
188 \PackageError{texdepends}{texdepends needs to be loaded BEFORE
189 \string\documentclass \MessageBreak
190 Use \string\RequirePackage\space instead of \string\usepackage
191 \MessageBreak as the later is not allowed before
192 \string\documentclass}
193 \fi
    And we write the first dependency (upon ourself)
194 \TD@printPackage{texdepends}
    We write default graphicspath (is this really needed ?)
195 \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.

```

196 \def\TD@depends{ }%
197 \def\TD@dependsWarning{%
198 \PackageWarningNoLine{texdepends}{Figure(s) missing. Check
199 dependencies again}%
200 }%
201 \def\TD@missingDepends{%
202 \global\let\TD@depends\TD@dependsWarning%
203 }%
204 \AtEndDocument{%
205 \TD@depends%
206 }%

```

2.3 Diversion of \LaTeX 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).

```

207 \def\TD@@fileswith@pti@ns#1[#2]#3[#4]{%
208 \ifx#1\@clsextension
209 \TD@printClass{#3}
210 \fi
211 \TD@orig@@fileswith@pti@ns{#1}[#2]{#3}[#4]%
212 }%
213 \let\TD@@@fileswith@pti@ns\TD@@fileswith@pti@ns
214 \TD@switch{@fileswith@pti@ns}%
215 \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


```

216 \def\TD@onefilewithoptions#1[#2][#3]#4{%
217 \TD@printPackage{#1}%
218 \ifthenelse{\equal{#1}{texdepends}}{%
The texdepends package is special : we know it has already been loaded (we are
running code in it) but perhaps LATEX does not know it. So we try to honor the
debug option.
219 \TD@debug{Package ' #1 ' detected}
220 \@for\TD@package@option:=#2\do{%
221 \ifthenelse{\expandafter\isundefined\csname TD@option@\TD@package@option\endcsname}{
222 \TD@warningLine{Ignoring unknown option '\TD@package@option'}
223 }{
224 \ifthenelse{%
225 \equal{\TD@package@option}{debug}%
226 %\OR\equal{\TD@package@option}{debug}%
227 }{
228 \csname TD@option@\TD@package@option\endcsname%
229 }{
230 \TD@warningLine{Ignoring option '\TD@package@option'
231 \MessageBreak #1 already loaded
232 }
233 }
234 }
235 }
236 }{%
237 \expandafter\let\expandafter\TD@divert
238 \csname TD@PackageStart@#1\endcsname
239 \ifx\TD@divert\relax
240 \TD@debug{No preamble for package #1}
241 \else
242 \TD@info{Preamble for package #1}
243 \TD@divert{#1}{#2}{#3}{#4}%
244 \fi
245 \TD@orig@@onefilewithoptions{#1}[#2][#3]{#4}%
246 \expandafter\let\expandafter\TD@divert
247 \csname TD@PackageEnd@#1\endcsname
248 \ifx\TD@divert\relax
249 \TD@debug{No postamble for package #1}
250 \else
251 \TD@info{Postamble for package #1}
252 \TD@divert{#1}{#2}{#3}{#4}%
253 \fi
254 }%
255 }%
256 \TD@switch{@onefilewithoptions}%

```

2.3.2 Tracking input files

```

\TD@@input \@input and \@input@ are diverted to track loaded files
\TD@@input@ 257 \def\TD@@input#1{%

```

```

258 \TD@printInput{#1}%
259 \TD@orig@input{#1}}%
260 \def\TD@input@#1{%
261 \IfFileExists{#1}{%
262 \TD@printInput{#1}%
263 }{%
264 \TD@orig@input@{#1}}%
265 \TD@switch{@input}%
266 \TD@switch{@input@}%
267 \def\TD@verbatiminput#1{%
268 \TD@printInput{#1}%
269 \TD@orig@verbatiminput{#1}}%
270 \TD@PackagePostDivert{verbatim}{%
271 \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 272 \let\TD@orig@openout\openout
273 \def\TD@openout#1{%
274 \@ifnextchar={\TD@openout@eq#1}{\TD@openout@noeq#1}}
275 \def\TD@openout@eq#1=#2.#3{%
276 \TD@printOut{#2.#3}%
277 \TD@orig@openout#1=#2.#3%
278 }%
279 \def\TD@openout@noeq#1#2.#3{%
280 \TD@printOut{#2.#3}%
281 \TD@orig@openout#1#2.#3%
282 }%
283 %\TD@switch{openout}

```

2.3.4 Tracking bibliography files

```

284 \def\TD@bibli@bbl{%
285 \TD@printBblfile{\jobname}%
286 \global\let\TD@bibli@bbl\relax%
287 }%
288 \def\TD@bibli@files#1{%
289 \TD@bibli@bbl%
290 \@for\TD@bibfile:=#1\do{%
291 \TD@printBibfile{\TD@bibfile}%
292 }%
293 }
294 \def\TD@bibli@style#1{%
295 \TD@bibli@bbl%
296 %\@for\TD@bibstyle:=#1\do{%
297 % \TD@printBibstyle{\TD@bibstyle}%
298 %}

```

```

299 \TD@printBibstyle{#1}%
300 }
301 \def\TD@bibliography#1{%
302 \TD@bibli@files{#1}%
303 \TD@orig@bibliography{#1}%
304 }%
305 \def\TD@bibliographystyle#1{%
306 \TD@bibli@style{#1}%
307 \TD@orig@bibliographystyle{#1}%
308 }%
309 \TD@docswitch{bibliography}%
310 \TD@switch{bibliographystyle}%

```

2.3.5 Glossary and index files

```

311 \def\TD@makeindex{%
312 \TD@printIndex{\jobname}%
313 \TD@printIndexSource{\jobname}{\jobname.idx}%
314 \TD@orig@makeindex%
315 }%
316 \def\TD@makeglossary{%
317 \TD@printGlossary{\jobname}%
318 \TD@printGlossarySource{\jobname}{\jobname.glo}%
319 \TD@orig@makeglossary%
320 }%
321 \TD@switch{makeindex}%
322 \TD@switch{makeglossary}%

```

2.4 Diversion of package's macros

2.4.1 Package graphics

```

323 \def\TD@Gin@getbase#1{%
324 \TD@orig@Gin@getbase{#1}%
325 \ifx\Gin@ext\relax%
326 \else%
327 \TD@printFig{\Gin@base\Gin@ext}%
328 \fi%
329 }%
330 \def\TD@Gin@include@eps#1{%
331 \IfFileExists{#1}{%
332 \TD@orig@Gin@include@eps{#1}%
333 }{%
334 \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
335 Skipping it this time (the last one however)\MessageBreak}%
336 \TD@missingDepends%
337 \TD@printRequiredFile{#1}{eps}%
338 }%
339 }%
340 \def\TD@Gin@include@pdf#1{%
341 \IfFileExists{#1}{%
342 \TD@orig@Gin@include@pdf{#1}%

```

```

343 }{%
344   \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
345     Skipping it this time (the last one however)\MessageBreak}%
346   \TD@missingDepends%
347   \TD@printRequiredFile{#1}{pdf}%
348 }%
349 }%
350 \def\TD@graphicspath#1{%
351   \TD@printGraphicspath{#1}%
352   \TD@orig@graphicspath{#1}%
353 }%
354 \def\TD@Gread@eps#1{%
355   \IfFileExists{#1}{%
356     \%PackageWarning{texdepends}{Gread File '#1' exists}%
357     \TD@orig@Gread@eps{#1}%
358   }{%
359     \PackageWarning{texdepends}{No '#1' file \MessageBreak
360       using 1 for graphic dimensions}%
361     \setbox\@tempboxa\hbox{\relax}%
362     \def\Gin@llx{1}\let\Gin@lly\Gin@llx
363     \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
364     \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
365   }%
366 }%
367 \def\TD@Gread@pdf#1{%
368   \IfFileExists{#1}{%
369     \%PackageWarning{texdepends}{Gread File '#1' exists}%
370     \TD@orig@Gread@pdf{#1}%
371   }{%
372     \PackageWarning{texdepends}{No '#1' file \MessageBreak
373       using 1 for graphic dimensions}%
374     \setbox\@tempboxa\hbox{\relax}%
375     \def\Gin@llx{1}\let\Gin@lly\Gin@llx
376     \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
377     \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
378   }%
379 }%
380 \TD@PackagePostDivert{graphics}{%
381   \TD@switch{graphicspath}%
382   \TD@docswitch{Gin@getbase}%
383   \TD@docswitch{Gin@include@eps}%
384   \TD@docswitch{Gin@include@pdf}%
385   \TD@docswitch{Gread@eps}%
386   \TD@docswitch{Gread@pdf}%
387 }%

```

2.4.2 Package figlatex

```

388 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
389 % We do not want to fail when trying to load a missing subfig.
390 % \@Gin@rawtex

```

```

391 \def\TD@FL@includerawtexgraphics[#1]#2{%
392   \IfFileExists{#2}{%
393     \TD@orig@FL@includerawtexgraphics[#1]{#2}%
394   }{%
395     \PackageWarning{texdepends}{Figure '#2' needed\MessageBreak
396       Skipping it this time (the last one however)\MessageBreak}%
397     \TD@missingDepends%
398     \TD@printRequiredFile{#2}{rawtexgraphics}%
399   }%
400 }%
401
402 \def\TD@Gininclude@rawtex#1{%
403   \IfFileExists{#1}{%
404     \TD@orig@Gininclude@rawtex{#1}%
405   }{%
406     \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
407       Skipping it this time (the last one however)\MessageBreak}%
408     \TD@missingDepends%
409     \TD@printRequiredFile{#1}{rawtex}%
410   }%
411 }%
412
413 % \@Gininclude@figtex
414 \def\TD@Gininclude@figtex#1{%
415   \IfFileExists{#1}{%
416     %\PackageWarning{texdepends}{Gininclude File '#1' exists}%
417     \TD@orig@Gininclude@figtex{#1}%
418   }{%
419     \ifthenelse{\isundefined{\FL@subfig@mainfigname}}{%
420       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
421         Skipping it this time (the last one however)\MessageBreak}%
422     }{%
423       \PackageWarning{texdepends}{SubFigure '#1' from
424         \FL@subfig@mainfigname needed\MessageBreak
425         Skipping it this time (the last one however)\MessageBreak}%
426     }
427     \TD@missingDepends%
428     \TD@printRequiredFile{#1}{figtex}%
429   }%
430 }%
431 % \@Gininclude@svgtex
432 \def\TD@Gininclude@svgtex#1{%
433   \IfFileExists{#1}{%
434     %\PackageWarning{texdepends}{Gininclude File '#1' exists}%
435     \TD@orig@Gininclude@svgtex{#1}%
436   }{%
437     \ifthenelse{\isundefined{\FL@subfig@mainfigname}}{%
438       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
439         Skipping it this time (the last one however)\MessageBreak}%
440     }{%

```

```

441     \PackageWarning{texdepends}{SubFigure '#1' from
442     \FL@subfig@mainfigname needed\MessageBreak
443     Skipping it this time (the last one however)\MessageBreak}%
444   }
445   \TD@missingDepends%
446   \TD@printRequiredFile{#1}{svgtex}%
447 }%
448 }%
449
450
451 % \@Gread@figtex
452 \def\TD@Gread@figtex#1{%
453   \IfFileExists{#1}{%
454     \PackageWarning{texdepends}{Gread File '#1' exists}%
455     \TD@orig@Gread@figtex{#1}%
456   }{%
457     \PackageWarning{texdepends}{No '#1' file \MessageBreak
458     using 1 for graphic dimensions}%
459     \setbox\@tempboxa\hbox{\relax}%
460     \def\Gin@llx{1}\let\Gin@lly\Gin@llx
461     \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
462     \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
463   }%
464 }%
465
466 % \@Gread@svgtex
467 \def\TD@Gread@svgtex#1{%
468   \IfFileExists{#1}{%
469     \PackageWarning{texdepends}{Gread File '#1' exists}%
470     \TD@orig@Gread@svgtex{#1}%
471   }{%
472     \PackageWarning{texdepends}{No '#1' file \MessageBreak
473     using 1 for graphic dimensions}%
474     \setbox\@tempboxa\hbox{\relax}%
475     \def\Gin@llx{1}\let\Gin@lly\Gin@llx
476     \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
477     \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
478   }%
479 }%
480 % \FL@subfig@check
481 \def\TD@FL@subfig@check#1{%
482   \TD@printRequiredFile{#1}{subfigcheck}%
483   \TD@printSubfig{#1}%
484 }%
485 %
486 \TD@PackagePostDivert{figlatex}{%
487   \TD@docswitch{FL@includerawtexgraphics}%
488   \TD@docswitch{Ginclude@rawtex}%
489   \TD@docswitch{Ginclude@figtex}%
490   \TD@docswitch{Ginclude@svgtex}%

```

```

491 \TD@docswitch{Gread@figtex}%
492 \TD@docswitch{Gread@svgtex}%
493 \TD@docswitch{FL@subfig@check}%
494 }%

```

2.4.3 Package doc

```

495 \def\TD@PrintIndex{%
496 \TD@printIndexStyle{\jobname}{gind.ist}
497 \TD@printIndexTarget{\jobname}{\jobname.ind}
498 \TD@orig@PrintIndex%
499 }%
500 \def\TD@PrintChanges{%
501 \TD@printGlossaryStyle{\jobname}{gglo.ist}
502 \TD@printGlossaryTarget{\jobname}{\jobname.gls}
503 \TD@orig@PrintChanges%
504 }%
505 \TD@PackagePostDivert{doc}{%
506 \TD@docswitch{PrintIndex}%
507 \TD@docswitch{PrintChanges}%
508 }%

```

2.4.4 Package index

```

509 \def\TD@def@index#1#2#3#4{%
510 \TD@printIndex{#2}%
511 \TD@printIndexSource{#2}{\jobname.#3}%
512 \TD@printIndexTarget{#2}{\jobname.#4}
513 \TD@orig@def@index{#1}{#2}{#3}{#4}%
514 }%
515 \TD@PackagePostDivert{index}{%
516 \TD@switch{def@index}%
517 }%

```

2.4.5 Package makeidx

```

518 \def\TD@printindex{%
519 \TD@printIndexTarget{\jobname}{\jobname.ind}
520 \TD@orig@printindex%
521 }%
522 \TD@PackagePostDivert{makeidx}{%
523 \TD@docswitch{printindex}%
524 }%

```

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.

```

525 \def\TD@printglossary{%
526 \TD@printGlossaryTarget{\jobname}{\jobname.gls}
527 \TD@orig@printglossary%
528 }%
529 \TD@PackagePostDivert{nomencl}{%

```

```

530 \TD@docswitch{printglossary}%
531 \TD@switch{makeglossary}%
532 \TD@printGlossaryStyle{\jobname}{nomencl.ist}
533 }%

```

2.4.7 Package bibtopic

```

534 % For the bibtopic package
535 \def\TD@btSect{%
536 \@ifnextchar[{\TD@btSect@opt}{\TD@btSect@noopt}}
537 \def\TD@btSect@opt[#1]#2{%
538 \TD@bibli@files{#2}%
539 \TD@bibli@style{#1}%
540 \TD@orig@btSect{#1}{#2}}
541 \def\TD@btSect@noopt#1{%
542 \TD@bibli@files{#1}%
543 \TD@orig@btSect{#1}}
544
545 \def\TD@@bt@write@auxfile{%
546 \TD@printBblfile{\thebtauxfile}%
547 \TD@orig@@bt@write@auxfile%
548 }%
549 \TD@PackagePostDivert{bibtopic}{%
550 \global\let\TD@bibli@bbl\relax%
551 \global\let\TD@biblio@rules@package\TD@biblio@rules@package@bibtopic%
552 \TD@switch{@bt@write@auxfile}%
553 \TD@printOut{btbbl.aux}%
554 \TD@printRebuildRule{bibtopic}%
555 \TD@docswitch{btSect}%
556 }%

```

2.4.8 Package comment

```

557 % For the comment package, the |comment.cut| file must not be used as
558 % a dependency: it is read and written several times during a compilation.
559 \TD@PackagePostDivert{comment}{%
560 \TD@printNoDependsFile{comment.cut}%
561 }%
562 </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	<code>\@Gininclude@svgtex</code>	431
<code>\@Gininclude@figtex</code>		<code>\@Gread@figtex</code>	451
<code>\@Gininclude@crawtex</code>	390	<code>\@Gread@svgtex</code>	466

<code>\@clsextension</code>	208	<code>extension</code>	<u>8</u>
<code>\@for</code>	176, 220, 290, 296	<code>\OR</code>	226
<code>\@ifnextchar</code>	274, 536		
<code>\@tempboxa</code> 361, 363, 364, 374, 376, 377, 459, 461, 462, 474, 476, 477			
<code>\@twoerrors</code>	187		
<code>\@unknownoptionerror</code>	19		
		P	
		<code>\PackageError</code>	188
		<code>\PackageInfo</code>	60
		R	
		<code>\RequireFile</code>	<u>102</u>
		S	
		<code>\setboolean</code>	6
		<code>\StrSubstitute</code>	76
		T	
		<code>\TD@@@fileswith@ptions</code>	<u>207</u>
		<code>\TD@bt@write@auxfile</code>	545
		<code>\TD@fileswith@ptions</code>	<u>207</u>
		<code>\TD@input</code>	<u>257</u>
		<code>\TD@input@</code>	<u>257</u>
		<code>\TD@onefilewithoptions</code>	<u>216</u>
		<code>\TD@bibfile</code>	290, 291
		<code>\TD@bibli@bbl</code> . 284, 286, 289, 295, 550	
		<code>\TD@bibli@files</code> .. 288, 302, 538, 542	
		<code>\TD@bibli@style</code>	294, 306, 539
		<code>\TD@biblio@rules@package</code>	551
		<code>\TD@biblio@rules@package@bibtopic</code>	551
		<code>\TD@bibliography</code>	301
		<code>\TD@bibliographystyle</code>	305
		<code>\TD@bibstyle</code>	296, 297
		<code>\TD@btSect</code>	535
		<code>\TD@btSect@noopt</code>	536, 541
		<code>\TD@btSect@opt</code>	536, 537
		<code>\TD@cmd</code>	176, 177
		<code>\TD@debug</code>	<u>63</u> , 219, 240, 249
		<code>\TD@def@index</code>	509
		<code>\TD@depends</code>	<u>196</u>
		<code>\TD@dependsWarning</code>	<u>196</u>
		<code>\TD@divert</code> 237, 239, 243, 246, 248, 252	
		<code>\TD@docswitch</code> . <u>165</u> , 309, 382–386, 487–493, 506, 507, 523, 530, 555	
		<code>\TD@docswitchlist</code>	
		165, 168, 169, 171, 176
		<code>\TD@extension</code>	10, 35, 40, 43, 46, 70, 71, 82, 85, 88, 91, 94, 97, 100, 106, 109, 112, 115, 118, 121, 124, 125, 128, 131, 134, 137, 138, 141, 144, 147
		<code>\TD@FL@includerawtexgraphics</code> ..	391
<code>\@clsextension</code>	208		
<code>\@for</code>	176, 220, 290, 296		
<code>\@ifnextchar</code>	274, 536		
<code>\@tempboxa</code> 361, 363, 364, 374, 376, 377, 459, 461, 462, 474, 476, 477			
<code>\@twoerrors</code>	187		
<code>\@unknownoptionerror</code>	19		
A			
<code>\AtBeginDocument</code>	174		
<code>\AtEndDocument</code>	204		
B			
<code>\boolean</code>	57, 64		
C			
<code>\CurrentOption</code>	31, 32		
D			
<code>debug (option)</code>	<u>3</u>		
<code>\documentclass</code>	187, 189, 192		
E			
<code>extension (option)</code>	<u>8</u>		
F			
<code>\FL@subfig@check</code>	480		
<code>\FL@subfig@mainfigname</code>			
.....	419, 424, 437, 442		
G			
<code>\Gin@base</code>	327		
<code>\Gin@defaultbp</code>	363, 364, 376, 377, 461, 462, 476, 477		
<code>\Gin@ext</code>	325, 327		
<code>\Gin@llx</code>	362, 375, 460, 475		
<code>\Gin@lly</code>	362, 375, 460, 475		
<code>\Gin@urx</code>	363, 376, 461, 476		
<code>\Gin@ury</code>	364, 377, 462, 477		
I			
<code>\ifxetex</code>	38		
<code>\isundefined</code>	35, 221, 419, 437		
N			
<code>\newboolean</code>	3		
<code>\noexpandarg</code>	75		
O			
options:			
<code>debug</code>	<u>3</u>		

<code>\TD@FL@subfig@check</code>	481	<code>\TD@orig@Gread@svgtex</code>	470
<code>\TD@Gin@getbase</code>	323	<code>\TD@orig@makeglossary</code>	319
<code>\TD@Gin@include@eps</code>	330	<code>\TD@orig@makeindex</code>	314
<code>\TD@Gin@include@figtex</code>	414	<code>\TD@orig@openout</code>	272, 277, 281
<code>\TD@Gin@include@pdf</code>	340	<code>\TD@orig@PrintChanges</code>	503
<code>\TD@Gin@include@rawtex</code>	402	<code>\TD@orig@printglossary</code>	527
<code>\TD@Gin@include@svgtex</code>	432	<code>\TD@orig@PrintIndex</code>	498
<code>\TD@graphicspath</code>	350	<code>\TD@orig@printindex</code>	520
<code>\TD@Gread@eps</code>	354	<code>\TD@orig@verbatiminput</code>	269
<code>\TD@Gread@figtex</code>	452	<code>\TD@package@option</code>	
<code>\TD@Gread@pdf</code>	367	220–222, 225, 226, 228, 230
<code>\TD@Gread@svgtex</code>	467	<code>\TD@PackagePostDivert</code>	
<code>\TD@info</code>	56,	181, 270, 380,
70, 159, 167, 175, 179, 242, 251		486, 505, 515, 522, 529, 549, 559	
<code>\TD@local@new</code>	151, 152, 162	<code>\TD@PackagePreDivert</code>	181
<code>\TD@local@old</code>	150, 152, 155, 161	<code>\TD@poenout@eq</code>	272
<code>\TD@makeglossary</code>	316	<code>\TD@poenout@noeq</code>	272
<code>\TD@makeindex</code>	311	<code>\TD@print</code>	72, 82, 85, 88,
<code>\TD@missingDepends</code>		91, 94, 97, 100, 106, 109, 112,	
196, 336, 346, 397, 408, 427, 445		115, 118, 121, 124, 125, 128,	
<code>\TD@openout</code>	272	131, 134, 137, 138, 141, 144, 147	
<code>\TD@openout@eq</code>	274, 275	<code>\TD@printBblfile</code>	117, 285, 546
<code>\TD@openout@noeq</code>	274, 279	<code>\TD@printBibfile</code>	111, 291
<code>\TD@option@cmd</code>	15, 18, 22, 25	<code>\TD@printBibstyle</code>	114, 297, 299
<code>\TD@option@debug</code>	4	<code>\TD@PrintChanges</code>	500
<code>\TD@option@extension</code>	8	<code>\TD@printClass</code>	81, 209
<code>\TD@option@name</code>	13, 14, 16	<code>\TD@printFig</code>	90, 93, 327
<code>\TD@option@split</code>	12, 32	<code>\TD@printGlossary</code>	136, 317
<code>\TD@option@value</code>	24, 25	<code>\TD@printglossary</code>	525
<code>\TD@orig@bt@write@auxfile</code>	547	<code>\TD@printGlossarySource</code>	140, 318
<code>\TD@orig@@fileswith@ptions</code>	211	<code>\TD@printGlossaryStyle</code>	146, 501, 532
<code>\TD@orig@@iinput</code>	259	<code>\TD@printGlossaryTarget</code>	143, 502, 526
<code>\TD@orig@@iinput@</code>	264	<code>\TD@printGraphicspath</code>	108, 195, 351
<code>\TD@orig@@onefilewithoptions</code>	245	<code>\TD@PrintIndex</code>	495
<code>\TD@orig@bibliography</code>	303	<code>\TD@printIndex</code>	123, 312, 510
<code>\TD@orig@bibliographystyle</code>	307	<code>\TD@printindex</code>	518
<code>\TD@orig@btSect</code>	540, 543	<code>\TD@printIndexSource</code>	127, 313, 511
<code>\TD@orig@def@index</code>	513	<code>\TD@printIndexStyle</code>	133, 496
<code>\TD@orig@FL@includerawtexgraphics</code>		<code>\TD@printIndexTarget</code>	
.	393	130, 497, 512, 519
<code>\TD@orig@Gin@getbase</code>	324	<code>\TD@printInput</code>	87, 258, 262, 268
<code>\TD@orig@Gin@include@eps</code>	332	<code>\TD@printNoDependsFile</code>	90, 560
<code>\TD@orig@Gin@include@figtex</code>	417	<code>\TD@printOut</code>	105, 276, 280, 553
<code>\TD@orig@Gin@include@pdf</code>	342	<code>\TD@printPackage</code>	84, 194, 217
<code>\TD@orig@Gin@include@rawtex</code>	404	<code>\TD@printRebuildRule</code>	120, 554
<code>\TD@orig@Gin@include@svgtex</code>	435	<code>\TD@printRequiredFile</code>	99, 103,
<code>\TD@orig@graphicspath</code>	352	337, 347, 398, 409, 428, 446, 482	
<code>\TD@orig@Gread@eps</code>	357	<code>\TD@printSubfig</code>	96, 483
<code>\TD@orig@Gread@figtex</code>	455		
<code>\TD@orig@Gread@pdf</code>	370		

`\TD@switch` 149, 177, 214,
 215, 256, 265, 266, 271, 283,
 310, 321, 322, 381, 516, 531, 552
`\TD@verbatiminput` 267
`\TD@warning` 49, 156
`\TD@warningLine` 53, 222, 230
`\TD@write` 69, 71, 77, 78
`\TD@write@text` 76, 77
`\thebtauxfile` 546

U

`\usepackage` 190

Change History

<p> v1.0.1 General: Version 1.0.1 at last . . . 1 v1.0.2 General: Add support for package index.sty 1 v1.1.0 General: Manage index/glossary . 1 Rewrite of the package 1 v1.2.0 General: Better management of subfig files 1 v1.3.0 General: Management of svg files 1 </p>	<p> v2.2.5 General: No changes in texdepends.dtx 1 v2.3.0 General: ignore comment.cut file from the comment package . . 1 v2.4.1 General: No changes in texdepends.dtx 1 v2.4.2 General: Replace // by / in paths 1 </p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------