

The texdepends package*

Vincent DANJEAN

2008/01/28

Abstract

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

Contents

1	Introduction	2
2	Implementation of texdepends	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 graphics	11
2.4.2	Package figl _{at} ex	12
2.4.3	Package doc	14
2.4.4	Package index	15
2.4.5	Package makeidx	15
2.4.6	Package nomencl	15
2.4.7	Package bibtopic	15

*This file has version number Automatic, last revised 2008/01/28.

1 Introduction

This package automatically generates dependancies 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 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^AT_EX) file

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

`\TD@printFig` We need an XFig file

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

`\TD@printSubfig` Web need a subfig file (see figlatex.dtx)

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

```

\TD@printRequiredFile  A file is missing. texdepend will try to skip it this time, but it will be needed at
                        the next compilation
                        89 \def\TD@printRequiredFile#1{%
                        90   \TD@print{TD_\jobname\TD@extention _REQUIRED\space += #1}%
                        91 }%

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

\TD@printOut           We write in a file
                        93 \def\TD@printOut#1{%
                        94   \TD@print{TD_\jobname\TD@extention _OUTPUTS\space += #1}%
                        95 }%

\TD@printGraphicspath  We modify the graphicspath
                        96 \def\TD@printGraphicspath#1{%
                        97   \TD@print{TD_\jobname\TD@extention _GRAPHICSPATH\space := #1}%
                        98 }%

\TD@printBibfile       We use a bibfile (*.bib)
                        99 \def\TD@printBibfile#1{%
                        100   \TD@print{TD_\jobname\TD@extention _BIBFILES\space += #1.bib}%
                        101 }%

\TD@printBibstyle      We use a bibstyle file (*.bst)
                        102 \def\TD@printBibstyle#1{%
                        103   \TD@print{TD_\jobname\TD@extention _BIBSTYLES\space += #1.bst}%
                        104 }%

\TD@printBblfile       We use a bibliography file (*.bbl)
                        105 \def\TD@printBblfile#1{%
                        106   \TD@print{TD_\jobname\TD@extention _BBLFILES\space += #1.bbl}%
                        107 }%

\TD@printRebuildRule   We need a specific rule in the Makefile to check if a rebuild is needed
                        108 \def\TD@printRebuildRule#1{%
                        109   \TD@print{TD_\jobname\TD@extention _REBUILD_RULES\space += #1}%
                        110 }%

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

\TD@printIndexSource    The index file filled by LATEX
                        115 \def\TD@printIndexSource#1#2{%
                        116   \TD@print{TD_\jobname\TD@extention _INDEX_#1_SRC\space = #2}%
                        117 }%

```

`\TD@printIndexTarget` And the index file read by L^AT_EX after makeindex processing

```

118 \def\TD@printIndexTarget#1#2{%
119   \TD@print{TD_\jobname\TD@extention _INDEX_#1_TARGET\space = #2}%
120 }%

```

`\TD@printIndexStyle` The index style to be used with makeindex (never called for now)

```

121 \def\TD@printIndexStyle#1#2{%
122   \TD@print{TD_\jobname\TD@extention _INDEX_#1_STYLE\space = #2}%
123 }%

```

`\TD@printGlossary` And the same things for glossary. Its name

```

124 \def\TD@printGlossary#1{%
125   \TD@print{TD_\jobname\TD@extention _INDEXES += GLOSS}%
126   \TD@print{TD_\jobname\TD@extention _INDEXES_GLOSS\space += #1}%
127 }%

```

`\TD@printGlossarySource` The glossary file filled by L^AT_EX

```

128 \def\TD@printGlossarySource#1#2{%
129   \TD@print{TD_\jobname\TD@extention _GLOSS_#1_SRC\space = #2}%
130 }%

```

`\TD@printGlossaryTarget` And the glossary file read by L^AT_EX after makeindex processing

```

131 \def\TD@printGlossaryTarget#1#2{%
132   \TD@print{TD_\jobname\TD@extention _GLOSS_#1_TARGET\space = #2}%
133 }%

```

`\TD@printGlossaryStyle` The glossary style to be used with makeindex (never called for now)

```

134 \def\TD@printGlossaryStyle#1#2{%
135   \TD@print{TD_\jobname\TD@extention _GLOSS_#1_STYLE\space = #2}%
136 }%

```

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)

```

137 \def\TD@switch#1{%
138   \expandafter\let\expandafter\TD@local@old\csname#1\endcsname%
139   \expandafter\let\expandafter\TD@local@new\csname TD@#1\endcsname%
140   \ifx\TD@local@old\TD@local@new
141     \PackageWarning{texdepends}{#1 already redefined. Skipping it.}%
142   \else
143     \ifx\TD@local@old\relax
144       \TD@warning{Diverting empty macro '#1' \MessageBreak%
145         Probably a bug in texdepends.sty}%
146     \else
147       \TD@info{Diverting macro '#1'}%
148     \fi

```

```

149 \expandafter\let\csname TD@orig@#1\endcsname\TD@local@old%
150 \expandafter\let\csname#1\endcsname\TD@local@new%
151 \fi
152 }%

```

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

```

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

```

`\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 ;-)).

```

169 \def\TD@PackagePreDivert#1{%
170 \expandafter\def\csname TD@PackageStart@#1\endcsname ##1##2##3##4%
171 }%
172 \def\TD@PackagePostDivert#1{%
173 \expandafter\def\csname TD@PackageEnd@#1\endcsname ##1##2##3##4%
174 }%

```

2.2.4 Initialisation

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

```

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

```

And we write the first dependency (upon ourself)

```

182 \TD@printPackage{texdepends}

```

We write default graphicspath (is this really needed ?)

```
183 \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.

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

2.3 Diversion of L^AT_EX macros

2.3.1 Class and package loading

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

```
195 \def\TD@@fileswith@ptions#1[#2]#3[#4]{%
196   \ifx#1\@clsextension
197     \TD@printClass{#3}
198   \fi
199   \TD@orig@@fileswith@ptions{#1}#2{#3}#4}%
200 }%
201 \let\TD@@@fileswith@ptions\TD@@fileswith@ptions
202 \TD@switch{@fileswith@pti@ns}%
203 \TD@switch{@@fileswith@ptions}%
```

`\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

```
204 \def\TD@@onefilewithoptions#1[#2]#3#4{%
205   \TD@printPackage{#1}%
206   \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^AT_EX does not know it. So we try to honor the
debug option.

```
207   \TD@debug{Package '1' detected}
208   \@for\TD@package@option:=#2\do{%
209     \ifthenelse{\expandafter\isundefined\csname TD@option@TD@package@option\endcsname}{
```



```

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

```

2.3.2 Tracking input files

\TD@@iinput \@iinput and \@input@ are diverted to track loaded files

```

\TD@@input@ 245 \def\TD@@iinput#1{%
246   \TD@printInput{#1}%
247   \TD@orig@@iinput{#1}}%
248 \def\TD@input@#1{%
249   \IfFileExists{#1}{%
250     \TD@printInput{#1}%
251   }{%
252     \TD@orig@@input@{#1}}%
253 \TD@switch{@iinput}%
254 \TD@switch{@input@}%

```

```

255 \def\TD@verbatiminput#1{%
256   \TD@printInput{#1}%
257   \TD@orig@verbatiminput{#1}}%
258 \TD@PackagePostDivert{verbatim}{%
259   \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 260 \let\TD@orig@openout\openout
261 \def\TD@openout#1{%
262   \@ifnextchar={\TD@openout@eq#1}{\TD@openout@noeq#1}}
263 \def\TD@openout@eq#1=#2.#3{%
264   \TD@printOut{#2.#3}%
265   \TD@orig@openout#1=#2.#3%
266 }%
267 \def\TD@openout@noeq#1#2.#3{%
268   \TD@printOut{#2.#3}%
269   \TD@orig@openout#1#2.#3%
270 }%
271 %\TD@switch{openout}

```

2.3.4 Tracking bibliography files

```

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

```

```

296 }%
297 \TD@docswitch{bibliography}%
298 \TD@switch{bibliographystyle}%

```

2.3.5 Glossary and index files

```

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

```

2.4 Diversion of package's macros

2.4.1 Package graphics

```

311 \def\TD@Gin@getbase#1{%
312   \TD@orig@Gin@getbase{#1}%
313   \ifx\Gin@ext\relax%
314   \else%
315     \TD@printFig{\Gin@base\Gin@ext}%
316     \fi%
317 }%
318 \def\TD@Gin@include@eps#1{%
319   \IfFileExists{#1}{%
320     \TD@orig@Gin@include@eps{#1}%
321   }{%
322     \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
323       Skipping it this time (the last one however)\MessageBreak}%
324     \TD@missingDepends%
325     \TD@printRequiredFile{#1}%
326   }%
327 }%
328 \def\TD@Gin@include@pdf#1{%
329   \IfFileExists{#1}{%
330     \TD@orig@Gin@include@pdf{#1}%
331   }{%
332     \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
333       Skipping it this time (the last one however)\MessageBreak}%
334     \TD@missingDepends%
335     \TD@printRequiredFile{#1}%
336   }%
337 }%
338 \def\TD@graphicspath#1{%
339   \TD@printGraphicspath{#1}%

```

```

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

```

2.4.2 Package figlatex

```

376 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
377 % We do not want to fail when trying to load a missing subfig.
378 % \Gin@include@rawtex
379 \def\TD@FL@includerawtexgraphics[#1]#2{%
380 \IfFileExists{#2}{%
381   \TD@orig@FL@includerawtexgraphics[#1]{#2}%
382 }{%
383   \PackageWarning{texdepends}{Figure '#2' needed\MessageBreak
384     Skipping it this time (the last one however)\MessageBreak}%
385   \TD@missingDepends%
386   \TD@printRequiredFile{#2}%
387 }%

```

```

388 }%
389
390 \def\TD@Gininclude@rawtex#1{%
391   \IfFileExists{#1}{%
392     \TD@orig@Gininclude@rawtex{#1}%
393   }{%
394     \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
395       Skipping it this time (the last one however)\MessageBreak}%
396     \TD@missingDepends%
397     \TD@printRequiredFile{#1}%
398   }%
399 }%
400
401 % \@Gininclude@figtex
402 \def\TD@Gininclude@figtex#1{%
403   \IfFileExists{#1}{%
404     \% \PackageWarning{texdepends}{Gininclude File '#1' exists}%
405     \TD@orig@Gininclude@figtex{#1}%
406   }{%
407     \ifthenelse{\isundefined{\FL@subfig@mainfigname}}{%
408       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
409         Skipping it this time (the last one however)\MessageBreak}%
410     }{%
411       \PackageWarning{texdepends}{SubFigure '#1' from
412         \FL@subfig@mainfigname needed\MessageBreak
413         Skipping it this time (the last one however)\MessageBreak}%
414     }
415     \TD@missingDepends%
416     \TD@printRequiredFile{#1}%
417   }%
418 }%
419 % \@Gininclude@svgtext
420 \def\TD@Gininclude@svgtext#1{%
421   \IfFileExists{#1}{%
422     \% \PackageWarning{texdepends}{Gininclude File '#1' exists}%
423     \TD@orig@Gininclude@svgtext{#1}%
424   }{%
425     \ifthenelse{\isundefined{\FL@subfig@mainfigname}}{%
426       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
427         Skipping it this time (the last one however)\MessageBreak}%
428     }{%
429       \PackageWarning{texdepends}{SubFigure '#1' from
430         \FL@subfig@mainfigname needed\MessageBreak
431         Skipping it this time (the last one however)\MessageBreak}%
432     }
433     \TD@missingDepends%
434     \TD@printRequiredFile{#1}%
435   }%
436 }%
437

```

```

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

```

2.4.3 Package doc

```

483 \def\TD@PrintIndex{%
484   \TD@printIndexStyle{\jobname}{gind.ist}
485   \TD@printIndexTarget{\jobname}{\jobname.ind}

```

```

486 \TD@orig@PrintIndex%
487 }%
488 \def\TD@PrintChanges{%
489 \TD@printGlossaryStyle{\jobname}{gglo.ist}
490 \TD@printGlossaryTarget{\jobname}{\jobname.gls}
491 \TD@orig@PrintChanges%
492 }%
493 \TD@PackagePostDivert{doc}{%
494 \TD@docswitch{PrintIndex}%
495 \TD@docswitch{PrintChanges}%
496 }%

```

2.4.4 Package index

```

497 \def\TD@def@index#1#2#3#4{%
498 \TD@printIndex{#2}%
499 \TD@printIndexSource{#2}{\jobname.#3}%
500 \TD@printIndexTarget{#2}{\jobname.#4}
501 \TD@orig@def@index{#1}{#2}{#3}{#4}%
502 }%
503 \TD@PackagePostDivert{index}{%
504 \TD@switch{def@index}%
505 }%

```

2.4.5 Package makeidx

```

506 \def\TD@printindex{%
507 \TD@printIndexTarget{\jobname}{\jobname.ind}
508 \TD@orig@printindex%
509 }%
510 \TD@PackagePostDivert{makeidx}{%
511 \TD@docswitch{printindex}%
512 }%

```

2.4.6 Package nomencl

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

```

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

```

2.4.7 Package bibtopic

```

522 % Pour le package bibtopic
523 \def\TD@btSect{%

```

```

524 \@ifnextchar[{\TD@btSect@opt}{\TD@btSect@noopt}}
525 \def\TD@btSect@opt[#1]#2{%
526   \TD@bibli@files{#2}%
527   \TD@bibli@style{#1}%
528   \TD@orig@btSect[#1]{#2}}
529 \def\TD@btSect@noopt#1{%
530   \TD@bibli@files{#1}%
531   \TD@orig@btSect{#1}}
532
533 \def\TD@bt@write@auxfile{%
534   \TD@printBblfile{\thebt@auxfile}%
535   \TD@orig@bt@write@auxfile%
536 }%
537 \TD@PackagePostDivert{bibtopic}{%
538   \global\let\TD@bibli@bbl\relax%
539   \global\let\TD@biblio@rules@package\TD@biblio@rules@package@bibtopic%
540   \TD@switch{bt@write@auxfile}%
541   \TD@printOut{btbbl.aux}%
542   \TD@printRebuildRule{bibtopic}%
543   \TD@docswitch{btSect}%
544 }%
545 </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		C	
\@Gininclude@figtex	401	\CurrentOption	31, 32
\@Gininclude@rawtex	378	D	
\@Gininclude@svgtex	419	debug (option)	<u>3</u>
\@Gread@figtex	439	\documentclass	175, 177, 180
\@Gread@svgtex	454	E	
\@clsextension	196	extension (option)	<u>8</u>
\@for	164, 208, 278, 284	F	
\@ifnextchar	262, 524	\FL@subfig@check	468
\@tempboxa	349, 351, 352, 362, 364, 365, 447, 449, 450, 462, 464, 465	\FL@subfig@mainfigname 407, 412, 425, 430
\@twoclasseserror	175	G	
\@unknownoptionerror	19	\Gin@base	315
A		\Gin@defaultbp	351, 352, 364, 365, 449, 450, 464, 465
\AtBeginDocument	162	\Gin@ext	313, 315
\AtEndDocument	192		
B			
\boolean	56, 63		

\Gin@llx	350, 363, 448, 463	\TD@depends	184
\Gin@lly	350, 363, 448, 463	\TD@dependsWarning	184
\Gin@urx	351, 364, 449, 464	\TD@divert	225, 227, 231, 234, 236, 240
\Gin@ury	352, 365, 450, 465	\TD@docswitch	153, 297, 370–374, 475–481, 494, 495, 511, 518, 543
I		\TD@docswitchlist	153, 156, 157, 159, 164
\ifxetex	38	\TD@extention	10, 35, 40, 43, 46, 69, 70, 75, 78, 81, 84, 87, 90, 94, 97, 100, 103, 106, 109, 112, 113, 116, 119, 122, 125, 126, 129, 132, 135
\isundefined	35, 209, 407, 425	\TD@FL@includerawtexgraphics	379
N		\TD@FL@subfig@check	469
\newboolean	3	\TD@Gin@getbase	311
O		\TD@Gin@include@eps	318
options:		\TD@Gin@include@figtex	402
debug	3	\TD@Gin@include@pdf	328
extension	8	\TD@Gin@include@rawtex	390
\OR	214	\TD@Gin@include@svgtx	420
P		\TD@graphicspath	338
\PackageError	176	\TD@Gread@eps	342
\PackageInfo	59	\TD@Gread@figtex	440
R		\TD@Gread@pdf	355
\RequireFile	92	\TD@Gread@svgtx	455
S		\TD@info	55, 69, 147, 155, 163, 167, 230, 239
\setboolean	6	\TD@local@new	139, 140, 150
T		\TD@local@old	138, 140, 143, 149
\TD@@@fileswith@pti@ns	195	\TD@makeglossary	304
\TD@@bt@write@auxfile	533	\TD@makeindex	299
\TD@@fileswith@pti@ns	195	\TD@missingDepends	184, 324, 334, 385, 396, 415, 433
\TD@@iinput	245	\TD@openout	260
\TD@@input@	245	\TD@openout@eq	262, 263
\TD@@@onefilewithoptions	204	\TD@openout@noeq	262, 267
\TD@bibfile	278, 279	\TD@option@cmd	15, 18, 22, 25
\TD@bibli@bbl	272, 274, 277, 283, 538	\TD@option@debug	4
\TD@bibli@files	276, 290, 526, 530	\TD@option@extension	8
\TD@bibli@style	282, 294, 527	\TD@option@name	13, 14, 16
\TD@biblio@rules@package	539	\TD@option@split	12, 32
\TD@biblio@rules@package@bibtopic	539	\TD@option@value	24, 25
\TD@bibliography	289	\TD@orig@@bt@write@auxfile	535
\TD@bibliographystyle	293	\TD@orig@@fileswith@pti@ns	199
\TD@bibstyle	284, 285	\TD@orig@@iinput	247
\TD@btSect	523	\TD@orig@@input@	252
\TD@btSect@noopt	524, 529	\TD@orig@@@onefilewithoptions	233
\TD@btSect@opt	524, 525	\TD@orig@bibliography	291
\TD@cmd	164, 165	\TD@orig@bibliographystyle	295
\TD@debug	62, 207, 228, 237	\TD@orig@btSect	528, 531
\TD@def@index	497		

<code>\TD@orig@def@index</code>	501	<code>\TD@printGlossary</code>	<u>124</u> , 305
<code>\TD@orig@FL@includerawtexgraphics</code>	381	<code>\TD@printglossary</code>	513
<code>\TD@orig@Gin@getbase</code>	312	<code>\TD@printGlossarySource</code> ..	<u>128</u> , 306
<code>\TD@orig@Gin@include@eps</code>	320	<code>\TD@printGlossaryStyle</code>	<u>134</u> , 489, 520
<code>\TD@orig@Gin@include@figtex</code>	405	<code>\TD@printGlossaryTarget</code>	<u>131</u> , 490, 514
<code>\TD@orig@Gin@include@pdf</code>	330	<code>\TD@printGraphicspath</code> .	<u>96</u> , 183, 339
<code>\TD@orig@Gin@include@rawtex</code>	392	<code>\TD@PrintIndex</code>	483
<code>\TD@orig@Gin@include@svgtex</code>	423	<code>\TD@printIndex</code>	<u>111</u> , 300, 498
<code>\TD@orig@graphicspath</code>	340	<code>\TD@printindex</code>	506
<code>\TD@orig@Gread@eps</code>	345	<code>\TD@printIndexSource</code> ..	<u>115</u> , 301, 499
<code>\TD@orig@Gread@figtex</code>	443	<code>\TD@printIndexStyle</code>	<u>121</u> , 484
<code>\TD@orig@Gread@pdf</code>	358	<code>\TD@printIndexTarget</code>
<code>\TD@orig@Gread@svgtex</code>	458	<u>118</u> , 485, 500, 507
<code>\TD@orig@makeglossary</code>	307	<code>\TD@printInput</code>	<u>80</u> , 246, 250, 256
<code>\TD@orig@makeindex</code>	302	<code>\TD@printOut</code>	<u>93</u> , 264, 268, 541
<code>\TD@orig@openout</code>	260, 265, 269	<code>\TD@printPackage</code>	<u>77</u> , 182, 205
<code>\TD@orig@PrintChanges</code>	491	<code>\TD@printRebuildRule</code>	<u>108</u> , 542
<code>\TD@orig@printglossary</code>	515	<code>\TD@printRequiredFile</code> ...	<u>89</u> , 92, 325, 335, 386, 397, 416, 434, 470
<code>\TD@orig@PrintIndex</code>	486	<code>\TD@printSubfig</code>	<u>86</u> , 471
<code>\TD@orig@printindex</code>	508	<code>\TD@switch</code>	<u>137</u> , 165, 202, 203, 244, 253, 254, 259, 271, 298, 309, 310, 369, 504, 519, 540
<code>\TD@orig@verbatiminput</code>	257	<code>\TD@verbatiminput</code>	255
<code>\TD@package@option</code>	<code>\TD@warning</code>	<u>49</u> , 144
.....	208–210, 213, 214, 216, 218	<code>\TD@warningLine</code>	<u>52</u> , 210, 218
<code>\TD@PackagePostDivert</code> .	<u>169</u> , 258, 368, 474, 493, 503, 510, 517, 537	<code>\TD@write</code>	68, 70, 72
<code>\TD@PackagePreDivert</code>	<u>169</u>	<code>\thebtauxfile</code>	534
<code>\TD@poenout@eq</code>	<u>260</u>		
<code>\TD@poenout@noeq</code>	<u>260</u>		
<code>\TD@print</code>	<u>71</u> , 75, 78, 81, 84, 87, 90, 94, 97, 100, 103, 106, 109, 112, 113, 116, 119, 122, 125, 126, 129, 132, 135		
<code>\TD@printBblfile</code>	<u>105</u> , 273, 534		
<code>\TD@printBibfile</code>	<u>99</u> , 279		
<code>\TD@printBibstyle</code>	<u>102</u> , 285, 287		
<code>\TD@PrintChanges</code>	488		
<code>\TD@printClass</code>	<u>74</u> , 197		
<code>\TD@printFig</code>	<u>83</u> , 315		

U

<code>\usepackage</code>	178
--------------------------------	-----

Change History

v1.0.1	Rewrite of the package	1
General: Version 1.0.1 at last	...	1
v1.0.2	General: Better management of subfig files	1
General: Add support for package index.sty		1
v1.1.0	General: Management of svg files	1
General: Manage index/glossary	...	1