The texdepends $package^*$

Vincent Danjean

2021/10/26

Abstract

This package allows $\ensuremath{\mbox{IAT}_{\mbox{E}}}\mbox{X}$ to automatically generate dependencies while compiling documents.

Contents

1 Introduction					
2 Implementation of texdepends 2					
2.1^{-}	Option	ns	2		
2.2	_		3		
			3		
			4		
	2.2.3		6		
	2.2.4		7		
	2.2.5		8		
			8		
	2.3.1		8		
	2.3.2		9		
	2.3.3		10		
	2.3.4		10		
	2.3.5		11		
2.4	Divers		11		
	2.4.1		11		
			12		
			15		
			15		
			15		
			15		
		· · · · · · · · · · · · · · · · · · ·	16		
			16		
	Imp 2.1 2.2 2.3	Implementa 2.1 Option 2.2 Tools 2.2.1 2.2.2 2.2.3 2.2.4 2.2.5 2.3 Divers 2.3.1 2.3.2 2.3.3 2.3.4 2.3.5 2.4 Divers	Implementation of texdepends 2.1 Options 2.2 Tools 2.2.1 Debug/info wrapper 2.2.2 Writing dependencies 2.2.3 Diverting a macro 2.2.4 Initialisation 2.2.5 Warning upon missing figures 2.3 Diversion of LATEX macros 2.3.1 Class and package loading 2.3.2 Tracking input files 2.3.3 Tracking output files 2.3.4 Tracking bibliography files 2.3.5 Glossary and index files 2.4 Diversion of package's macros 2.4.1 Package graphics 2.4.2 Package figlatex 2.4.3 Package doc 2.4.4 Package index 2.4.5 Package makeidx 2.4.6 Package nomencl 2.4.7 Package bibtopic		

^{*}This file has version number v2.4.3, last revised 2021/10/26.

1 Introduction

This package automatically generates dependancies on LATEX 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 debuging)
3 \newboolean{TD@debug}
4 \newcommand{\TD@option@debug}[1][true]{%
5 %\PackageWarningNoLine{texdepends}{Setting debug to #1}
6 \setboolean{TD@debug}{#1}%
7 }%
The extension option. We can use it to let texdepends know whether we
```

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

```
All option are managed by the same code.
                 30 \DeclareOption*{%
                     %\PackageWarningNoLine{texdepends}{Option \CurrentOption}%
                     \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}}{\%
                     \PackageWarningNoLine{texdepends}{No extension given: detecting it}%
                 36
                     \RequirePackage{ifxetex,ifpdf}%
                 37
                     \ifxetex%
                 38
                        \PackageWarningNoLine{textdepends}{Detected XeTeX: using .pdf}%
                 39
                        \def\TD@extention{.pdf}%
                 40
                     \else\ifpdf%
                 41
                        \PackageWarningNoLine{texdepends}{Detected PDF mode: using .pdf}%
                 42
                       \def\TD@extention{.pdf}%
                 43
                 44
                     \else%
                        \PackageWarningNoLine{texdepends}{Detected DVI mode. Using .dvi}%
                 45
                        \def\TD@extention{.dvi}%
                     \fi\fi%
                 47
                 48 }{}%
                 2.2
                       Tools
                 Some macros to help development
                 2.2.1 Debug/info wrapper
                To write a warning
   \TD@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]{%
                     \ifthenelse{\boolean{TD@debug}}{%
```

\PackageWarningNoLine{texdepends}{#1}%

\PackageInfo{texdepends}{#1}%

58

59

60

}{%

```
61 }%
                  62 }%
       \TD@debug To write a message if the debug option is set
                  63 \newcommand{\TD@debug}[1]{%
                      \ifthenelse{\boolean{TD@debug}}{%
                        \PackageWarningNoLine{texdepends}{#1}%
                  65
                  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
                     {%
                        \% TODO: fix pb when #1 contains '#'
                  74
                  75
                        %\noexpandarg%
                  76
                        \StrSubstitute{#1}{//}{/}[\TD@write@text]%
                        \immediate\write\TD@write{\TD@write@text}%
                        %\immediate\write\TD@write{#1}%
                  78
                  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 (LATEX) 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 Web 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{%
                        \label{top:linear} $$115 $$ $$ TD@print{TD_{jobname}TD@extention _BIBSTYLES\space += $$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{%
                             \TD@print{TD_\jobname\TD@extention _INDEXES\space += INDEX}%
                              \TD@print{TD_\jobname\TD@extention _INDEXES_INDEX\space += #1}%
                        125
                        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 IATEX after makeindex processing
                        130 \def\TD@printIndexTarget#1#2{%
                              \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{%
                              \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{%
                             \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 \TDCorigC#1 macro. Nothing is done if \#1 is already \TDC#1 (that
                         is if \TD@switch is called severall times with the same argument)
                        149 \def\TD@switch#1{%
```

```
\expandafter\let\expandafter\TD@local@old\csname#1\endcsname%
              150
                    \expandafter\let\expandafter\TD@local@new\csname TD@#1\endcsname%
              151
                    \ifx\TD@local@old\TD@local@new
              152
                      \PackageWarning{texdepends}{#1 already redefined. Skipping it.}%
              153
              154
                    \else
                      \ifx\TD@local@old\relax
              155
              156
                        \TD@warning{Diverting empty macro '#1' \MessageBreak%
                          Probably a bug in texdepends.sty}%
              157
              158
                      \else
                        \TD@info{Diverting macro '#1'}%
              159
                      \fi
              160
                      \expandafter\let\csname TD@orig@#1\endcsname\TD@local@old%
               161
                      \expandafter\let\csname#1\endcsname\TD@local@new%
              163
              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{%
                    \TD@info{Registering macro '#1' for diversion}%
              167
                    \ifx\TD@docswitchlist\@empty%
              168
                      \xdef\TD@docswitchlist{#1}%
              169
                    \else%
              170
              171
                      \xdef\TD@docswitchlist{\TD@docswitchlist,#1}%
              172
                    \fi%
              173 }%
              174 \AtBeginDocument{%
                    \TD@info{Diverting registered macros...}%
                    \@for\TD@cmd:=\TD@docswitchlist\do{%
              176
                      \TD@switch{\TD@cmd}%
              177
              178
                    \TD@info{Divertions done}%
              179
              180 }%
```

\TD@PackagePreDivert \TD@PackagePostDivert And then, two macros to insert command just before or after loading a package. 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 <code>@onefilewithoptions</code> LATEX 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 \TD@missingDepends If some figures are missing, \TD@missingDepends will be called so that a warning will be displayed at the end of the compilation. Figures should be generated before 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 \TD@@@fileswith@pti@ns Diverting \@fileswith@pti@ns and @@fileswith@pti@ns so that we can track 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\{\%
     \TD@printPackage{#1}%
     218
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}
       \@for\TD@package@option:=#2\do{%
220
         \ifthenelse{\expandafter\isundefined\csname TD@option@\TD@package@option\endcsname}{
221
           \TD@warningLine{Ignoring unknown option '\TD@package@option'}
222
223
         }{
224
           \ifthenelse{%
             \equal{\TD@package@option}{debug}%
225
             %\OR\equal{\TD@package@option}{debug}%
226
           }{
227
             \csname TD@option@\TD@package@option\endcsname%
228
           }{
229
             \TD@warningLine{Ignoring option '\TD@package@option'
230
               \MessageBreak #1 already loaded
231
232
             }
           }
233
         }
234
235
       }
236
     }{%
237
       \expandafter\let\expandafter\TD@divert
       \csname TD@PackageStart@#1\endcsname
238
       \ifx\TD@divert\relax
239
         \TD@debug{No preamble for package #1}
240
241
         \TD@info{Preamble for package #1}
242
243
         \TD@divert{#1}{#2}{#3}{#4}%
244
       \TD@orig@@onefilewithoptions{#1}[#2][#3]{#4}%
245
       \expandafter\let\expandafter\TD@divert
246
       \csname TD@PackageEnd@#1\endcsname
247
       \ifx\TD@divert\relax
248
249
         \TD@debug{No postamble for package #1}
250
         \TD@info{Postamble for package #1}
251
         \TD@divert{#1}{#2}{#3}{#4}%
252
       \fi
253
     }%
254
255 }%
256 \TD@switch{@onefilewithoptions}%
```

2.3.2 Tracking input files

 $\label{localine} $$\TD@0iinput \ \end \end{subarray}$ are diverted to track loaded files $$\TD@0iinput@257 \end{$

```
\TD@printInput{#1}%
258
     \TD@orig@@iinput{#1}}%
259
260 \def\TD@@input@#1{%
     \IfFileExists{#1}{%
261
       \TD@printInput{#1}%
262
263
     }{}%
264
     \TD@orig@@input@{#1}}%
265 \TD@switch{@iinput}%
266 \TD@switch{@input@}%
267 \def\TD@verbatiminput#1{%
     \TD@printInput{#1}%
268
     \TD@orig@verbatiminput{#1}}%
270 \TD@PackagePostDivert{verbatim}{%
     \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
                   someone has ideas...
 \TD@poenout@eq
\verb|\TD@poenout@noeq||_{272} \verb|\letTD@orig@openout@noeq||_{272} \\
                  273 \def\TD@openout#1{%}
                        \@ifnextchar={\TD@openout@eq#1}{\TD@openout@noeq#1}}
                  275 \def\TD@openout@eq#1=#2.#3{%
                        \TD@printOut{#2.#3}%
                        \TD@orig@openout#1=#2.#3%
                  277
                  278 }%
                  279 \def\TD@openout@noeq#1#2.#3{%
                        \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{%
       \TD@printBblfile{\jobname}%
285
        \global\let\TD@bibli@bbl\relax%
286
287 }%
288 \def\TD@bibli@files#1{%
     \TD@bibli@bbl%
289
     \c \c \TD@bibfile:=#1\do{\%}
290
        \TD@printBibfile{\TD@bibfile}%
291
292
293 }
294 \def\TD@bibli@style#1{%
     \TD@bibli@bbl%
295
     %\@for\TD@bibstyle:=#1\do{%
296
        \TD@printBibstyle{\TD@bibstyle}%
297
    %}
298
```

```
300 }
301 \def\TD@bibliography#1{%
     \TD@bibli@files{#1}%
     \TD@orig@bibliography{#1}%
303
304 }%
305 \def\TD@bibliographystyle#1{%
     \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{%
     \TD@printIndex{\jobname}%
     \TD@printIndexSource{\jobname}{\jobname.idx}%
     \TD@orig@makeindex%
314
315 }%
316 \def\TD@makeglossary{%
     \TD@printGlossary{\jobname}%
317
     \TD@printGlossarySource{\jobname}{\jobname.glo}%
     \TD@orig@makeglossary%
320 }%
321 \TD@switch{makeindex}%
```

\TD@printBibstyle{#1}%

299

2.4 Diversion of package's macros

2.4.1 Package graphics

322 \TD@switch{makeglossary}%

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

```
}{%
343
       \PackageWarning{texdepends}{Figure '#1' needed\MessageBreak
344
         Skipping it this time (the last one however)\MessageBreak}%
345
       \TD@missingDepends%
346
       \TD@printRequiredFile{#1}{pdf}%
347
348
    }%
349 }%
   \def\TD@graphicspath#1{%
350
     \TD@printGraphicspath{#1}%
351
     \TD@orig@graphicspath{#1}%
352
353 }%
354 \def\TD@Gread@eps#1{%
     \IfFileExists{#1}{%
355
       %\PackageWarning{texdepends}{Gread File '#1' exists}%
356
       \TD@orig@Gread@eps{#1}%
357
     }{%
358
       \PackageWarning{texdepends}{No '#1' file \MessageBreak
359
         using 1 for graphic dimensions}%
360
361
       \setbox\@tempboxa\hbox{\relax}%
362
       \def\Gin@llx{1}\let\Gin@lly\Gin@llx
       \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
363
364
       \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
    }%
365
366 }%
367 \def\TD@Gread@pdf#1{%
     \IfFileExists{#1}{%
       %\PackageWarning{texdepends}{Gread File '#1' exists}%
369
       \TD@orig@Gread@pdf{#1}%
370
     }{%
371
       \PackageWarning{texdepends}{No '#1' file \MessageBreak
372
         using 1 for graphic dimensions}%
373
374
       \setbox\@tempboxa\hbox{\relax}%
375
       \def\Gin@llx{1}\let\Gin@lly\Gin@llx
       \Gin@defaultbp\Gin@urx{\wd\@tempboxa}%
376
377
       \Gin@defaultbp\Gin@ury{\ht\@tempboxa}%
378
    }%
379 }%
380 \TD@PackagePostDivert{graphics}{%
     \TD@switch{graphicspath}%
381
     \TD@docswitch{Gin@getbase}%
382
383
     \TD@docswitch{Ginclude@eps}%
     \TD@docswitch{Ginclude@pdf}%
384
     \TD@docswitch{Gread@eps}%
385
     \TD@docswitch{Gread@pdf}%
386
387 }%
2.4.2 Package figlatex
389 % We do not want to fail when trying to load a missing subfig.
```

390 % \@Ginclude@rawtex

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

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

```
\TD@docswitch{Gread@figtex}%
491
     \TD@docswitch{Gread@svgtex}%
492
     \TD@docswitch{FL@subfig@check}%
494 }%
 2.4.3 Package doc
495 \def\TD@PrintIndex{%
     \TD@printIndexStyle{\jobname}{gind.ist}
     \TD@printIndexTarget{\jobname}{\jobname.ind}
     \TD@orig@PrintIndex%
498
499 }%
500 \def\TD@PrintChanges{%
     \TD@printGlossaryStyle{\jobname}{gglo.ist}
501
     \verb|\TD@printGlossaryTarget{\jobname}{\jobname.gls}| \\
502
     \TD@orig@PrintChanges%
503
504 }%
505 \TD@PackagePostDivert{doc}{%
     \TD@docswitch{PrintIndex}%
     \TD@docswitch{PrintChanges}%
507
508 }%
 2.4.4 Package index
509 \def\TD@def@index#1#2#3#4{%
     \TD@printIndex{#2}%
     \TD@printIndexSource{#2}{\jobname.#3}%
     \TD@printIndexTarget{#2}{\jobname.#4}
     \TD@orig@def@index{#1}{#2}{#3}{#4}%
513
514 }%
515 \TD@PackagePostDivert{index}{%
     \TD@switch{def@index}%
517 }%
2.4.5 Package makeidx
518 \def\TD@printindex{%
     \TD@printIndexTarget{\jobname}{\jobname.ind}
520
     \TD@orig@printindex%
521 }%
522 \TD@PackagePostDivert{makeidx}{%
     \TD@docswitch{printindex}%
524 }%
 2.4.6 Package nomencl
 We divert makeglossary again after loading this package as its redefines this
 macro (loosing our divertion). We also setup the glossary style for this package.
525 \def\TD@printglossary{%
     \TD@printGlossaryTarget{\jobname}{\jobname.gls}
     \TD@orig@printglossary%
527
```

528 }%

529 \TD@PackagePostDivert{nomencl}{%

```
\TD@docswitch{printglossary}%
530
     \TD@switch{makeglossary}%
531
    \TD@printGlossaryStyle{\jobname}{nomencl.ist}
532
533 }%
2.4.7 Package bibtopic
534 % For the bibtopic package
535 \def\TD@btSect{%
    \@ifnextchar[{\TD@btSect@opt}{\TD@btSect@noopt}}
537 \def\TD@btSect@opt[#1]#2{%
     \TD@bibli@files{#2}%
538
539
     \TD@bibli@style{#1}%
     \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}%
     \TD@orig@@bt@write@auxfile%
547
548 }%
549 \TD@PackagePostDivert{bibtopic}{%
     \global\let\TD@bibli@bbl\relax%
550
     \global\let\TD@biblio@rules@package\TD@biblio@rules@package@bibtopic%
551
     \TD@switch{@bt@write@auxfile}%
552
553
     \TD@printOut{btbbl.aux}%
     \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}{%
   \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	$\verb \Ginclude@svgtex 431$
\@Ginclude@figtex 413	$\verb \@Gread@figtex 451 $
\@Ginclude@rawtex 390	\@Gread@svgtex 466

\@clsextension	extension $\underline{8}$ \OR
\@tempboxa 361, 363, 364, 374, 376,	P 100
377, 459, 461, 462, 474, 476, 477	\PackageError 188
\@twoclasseserror 187	\PackageInfo 60
\@unknownoptionerror 19	D
A	R
A	\RequireFile <u>102</u>
\AtBeginDocument	\mathbf{S}
\AtEndDocument 204	
В	
	\StrSubstitute 76
\boolean 57, 64	Т
\mathbf{C}	-
\CurrentOption 31, 32	\TDQQQfileswithQptiQns 207 \TDQQbtQwriteQauxfile 545
(ouriendpoion or, or	
D	\TD@@fileswith@pti@ns 207
debug (option) <u>3</u>	\TD@@iinput
\documentclass 187, 189, 192	\TD@@input@
21, 22,	\TD@@onefilewithoptions 216
${f E}$	\TD@bibfile
extension (option) $\underline{8}$	\TD@bibli@files 288, 302, 538, 542
_	
${f F}$	\TD@bibli@style 294, 306, 539
\FL@subfig@check 480	\TD@biblio@rules@package 551
\FL@subfig@mainfigname	\TD@biblio@rules@package@bibtopic
$\dots \dots $	\TD@\:\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
	\TD@bibliography
\mathbf{G}	\TD@bibliographystyle 305 \TD@bibstyle 296, 297
\Gin@base 327	\TD@btSect
\Gin@defaultbp 363,	
364, 376, 377, 461, 462, 476, 477	\TD@btSect@noopt 536, 541 \TD@btSect@opt 536, 537
\Gin@ext 325, 327	
\Gin@llx 362, 375, 460, 475	\TD@cmd
\Gin@lly $362, 375, 460, 475$	\TD@debug <u>63</u> , 219, 240, 249 \TD@def@index 509
\Gin@urx 363, 376, 461, 476	
\Gin@ury 364, 377, 462, 477	\TD@depends
_	\TD@dependsWarning 196
I	\TD0divert 237, 239, 243, 246, 248, 252
\ifxetex 38	\TD@docswitch . <u>165</u> , 309, 382–386,
\isundefined 35, 221, 419, 437	487–493, 506, 507, 523, 530, 555
N	\TD@docswitchlist
N	165, 168, 169, 171, 176
\newboolean	\TD@extention 10, 35,
\noexpandarg 75	40, 43, 46, 70, 71, 82, 85, 88, 91, 94, 97, 100, 106, 109, 112,
0	
options:	115, 118, 121, 124, 125, 128, 131, 134, 137, 138, 141, 144, 147
	\TD@FL@includerawtexgraphics 391
debug \ldots 3	/inatraincinderameaxRiahnics 931

	\
\TD@FL@subfig@check 481	\TD@orig@Gread@svgtex 470
\TD@Gin@getbase 323	\TD@orig@makeglossary 319
\TD@Ginclude@eps 330	\TD@orig@makeindex 314
\TD@Ginclude@figtex 414	\TD@orig@openout 272, 277, 281
\TD@Ginclude@pdf	$\verb \TD@orig@PrintChanges 503 $
\TD@Ginclude@rawtex 402	\TD@orig@printglossary 527
\TD@Ginclude@svgtex 432	\TD@orig@PrintIndex 498
\TD@graphicspath 350	\TD@orig@printindex 520
\TD@Gread@eps 354	\TD@orig@verbatiminput 269
\TD@Gread@figtex	\TD@package@option
\TD@Gread@pdf 367	220–222, 225, 226, 228, 230
\TD@Gread@svgtex	\TD@PackagePostDivert
\TD@info <u>56,</u>	181, 270, 380,
70, 159, 167, 175, 179, 242, 251	486, 505, 515, 522, 529, 549, 559
\TD@local@new 151, 152, 162	\TD@PackagePreDivert 181
\TD@local@old 150, 152, 155, 161	\TD@poenout@eq 272
\TD@makeglossary 316	\TD@poenout@noeq 272
\TD@makeindex	\TD@print 72, 82, 85, 88,
\TD@missingDepends	91, 94, 97, 100, 106, 109, 112,
196, 336, 346, 397, 408, 427, 445	115, 118, 121, 124, 125, 128,
\TD@openout	131, 134, 137, 138, 141, 144, 147
\TD@openout@eq 274, 275	\TD@printBblfile 117, 285, 546
\TD@openout@noeq 274, 279	\TD@printBibfile 111, 291
\TD@option@cmd 15, 18, 22, 25	\TD@printBibstyle <u>114</u> , 297, 299
\TDCoptionCdebug 4	
\TD@option@extension 8	\TD@PrintChanges 500
\TD@option@name 13, 14, 16	\TD@printClass <u>81,</u> 209
\TD@option@split 12, 32	\TD@printFig 90, 93, 327
\TD@option@value 24, 25	\TD@printGlossary 136, 317
\TD@orig@@bt@write@auxfile 547	\TD@printglossary 525
\TD@orig@@fileswith@pti@ns 211	\TD@printGlossarySource $\underline{140}$, 318
\TD@orig@@iinput	\TD@printGlossaryStyle $\underline{146}$, 501 , 532
\TD@orig@@input@	\TD@printGlossaryTarget $\underline{143}, 502, 526$
\TD@orig@@onefilewithoptions 245	\TD@printGraphicspath . $\underline{108}$, 195 , 351
\TD@orig@bibliography 303	\TD@PrintIndex 495
\TD@orig@bibliographystyle 307	\TD@printIndex $\dots 123, 312, 510$
\TD@orig@btSect 540, 543	\TD@printindex 518
\TD@orig@def@index 513	\TD@printIndexSource <u>127</u> , 313, 511
\TD@orig@FL@includerawtexgraphics	\TD@printIndexStyle 133, 496
\TD@orig@Gin@getbase 324	\TD@printIndexTarget
	<u>130,</u> 497, 512, 519
\TD@orig@Ginclude@eps 332 \TD@orig@Ginclude@figtex 417	\TD@printInput 87, 258, 262, 268
\TD@orig@Ginclude@rigtex 417 \TD@orig@Ginclude@pdf 342	\TD@printNoDependsFile 90, 560
	\TD@printOut <u>105,</u> 276, 280, 553
\TD@orig@Ginclude@rawtex 404 \TD@orig@Ginclude@svgtex 435	\TD@printPackage 84, 194, 217
\TD@orig@graphicspath 352	\TD@printRebuildRule 120, 554
\TD@orig@Gread@eps	\TD@printRequiredFile 99, 103,
\TD@orig@Gread@figtex	337, 347, 398, 409, 428, 446, 482
\TD@orig@Gread@pdf	\TD@printSubfig 96, 483
/inaoiikagieadahdi	(10ehr 11109ff) 11 (10ehr 11109ff) 1409

\TD@switch 149 , 177, 214,	${f U}$	
215, 256, 265, 266, 271, 283, 310, 321, 322, 381, 516, 531, 552 \TD@verbatiminput	\usepackage 190	
Change History		
v1.0.1	v2.2.5	
General: Version 1.0.1 at last \dots 1	General: No changes in	
v1.0.2	texdepends.dtx $\dots \dots 1$ v2.3.0	
General: Add support for package index.sty	General: ignore comment.cut file	
v1.1.0	from the comment package 1	
General: Manage index/glossary . 1	General: No changes in	
Rewrite of the package 1	$texdepends.dtx \dots \dots 1$	
v1.2.0	v2.4.2	
General: Better management of	General: Replace // by / in paths 1	
subfig files	v2.4.3	
. = . 9 . 9	General: No changes in	
General: Management of svg files 1	$texdepends.dtx \dots 1$	