

The `ifvtex` package

Heiko Oberdiek*

<heiko.oberdiek at gmail.com>

2016/05/16 v1.6

Abstract

This package looks for \TeX , implements and sets the switches `\ifvtex`, `\ifvtex<mode>`, `\ifvtexgex`. It works with plain or \LaTeX formats.

Contents

1 Usage	2
2 Implemenation	2
2.1 Reload check and package identification	2
2.2 Catcodes	3
2.3 Check for previously defined <code>\ifvtex</code>	4
2.4 Provide <code>\newif</code>	5
2.5 <code>\ifvtex</code>	5
2.6 Mode and \TeX switches	5
2.7 Protocol entry	6
3 Test	6
3.1 Catcode checks for loading	6
4 Installation	8
4.1 Download	8
4.2 Bundle installation	8
4.3 Package installation	9
4.4 Refresh file name databases	9
4.5 Some details for the interested	9
5 Catalogue	9
6 History	10
[2001/09/26 v1.0]	10
[2006/02/20 v1.1]	10
[2007/01/10 v1.2]	10
[2007/09/09 v1.3]	10
[2008/11/04 v1.4]	10
[2010/03/01 v1.5]	10
[2016/05/16 v1.6]	10
7 Index	11

*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

1 Usage

The package `ifvtex` can be used with both plain `TEX` and `LATEX`:

plain T_EX: `\input ifvtex.sty`

L^AT_EX 2_ε: `\usepackage{ifvtex}`

The package implements switches for `VTEX` and its different modes and interprets `\VTeXversion`, `\OpMode`, and `\gexmode`.

`\ifvtex`

The package provides the switch `\ifvtex`:

```
\ifvtex
... do things, if VTEX is running ...
\else
... other TEX compiler ...
\fi
```

Users of the package `ifthen` can use the switch as boolean:

```
\boolean{ifvtex}
```

```
\ifvtexdvi
\ifvtexpdf
\ifvtexps
\ifvtexhtml
```

`VTEX` knows different output modes that can be asked by these switches.

`\ifvtexgex`

This switch shows, whether `GeX` is available.

2 Implemenation

2.1 Reload check and package identification

```
1 (*package)
```

Reload check, especially if the package is not used with `LATEX`.

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3 \catcode13=5 % ^^M
4 \endlinechar=13 %
5 \catcode35=6 % #
6 \catcode39=12 % '
7 \catcode44=12 % ,
8 \catcode45=12 % -
9 \catcode46=12 % .
10 \catcode58=12 % :
11 \catcode64=11 % @
12 \catcode123=1 % {
13 \catcode125=2 % }
14 \expandafter\let\expandafter\x\csname ver@ifvtex.sty\endcsname
15 \ifx\x\relax % plain-TeX, first loading
16 \else
17 \def\empty{}%
```

```

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

```

Package identification:

```

33 \begingroup\catcode61\catcode48\catcode32=10\relax%
34 \catcode13=5 % ^^M
35 \endlinechar=13 %
36 \catcode35=6 % #
37 \catcode39=12 % '
38 \catcode40=12 % (
39 \catcode41=12 % )
40 \catcode44=12 % ,
41 \catcode45=12 % -
42 \catcode46=12 % .
43 \catcode47=12 % /
44 \catcode58=12 % :
45 \catcode64=11 % @
46 \catcode91=12 % [
47 \catcode93=12 % ]
48 \catcode123=1 % {
49 \catcode125=2 % }
50 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
51 \def\x#1#2#3[#4]{\endgroup
52 \immediate\write-1{Package: #3 #4}%
53 \xdef#1{#4}%
54 }%
55 \else
56 \def\x#1#2[#3]{\endgroup
57 #2[#{#3}]%
58 \ifx#1\@undefined
59 \xdef#1{#3}%
60 \fi
61 \ifx#1\relax
62 \xdef#1{#3}%
63 \fi
64 }%
65 \fi
66 \expandafter\x\csname ver@ifvtex.sty\endcsname
67 \ProvidesPackage{ifvtex}%
68 [2016/05/16 v1.6 Detect VTeX and its facilities (HO)]%

```

2.2 Catcodes

```

69 \begingroup\catcode61\catcode48\catcode32=10\relax%
70 \catcode13=5 % ^^M
71 \endlinechar=13 %
72 \catcode123=1 % {
73 \catcode125=2 % }
74 \catcode64=11 % @
75 \def\x{\endgroup
76 \expandafter\edef\csname ifvtex@AtEnd\endcsname{%

```

```

77 \endlinechar=\the\endlinechar\relax
78 \catcode13=\the\catcode13\relax
79 \catcode32=\the\catcode32\relax
80 \catcode35=\the\catcode35\relax
81 \catcode61=\the\catcode61\relax
82 \catcode64=\the\catcode64\relax
83 \catcode123=\the\catcode123\relax
84 \catcode125=\the\catcode125\relax
85 }%
86 }%
87 \x\catcode61\catcode48\catcode32=10\relax%
88 \catcode13=5 % ^^M
89 \endlinechar=13 %
90 \catcode35=6 % #
91 \catcode64=11 % @
92 \catcode123=1 % {
93 \catcode125=2 % }
94 \def\TMP@EnsureCode#1#2{%
95 \edef\ifvtex@AtEnd{%
96 \ifvtex@AtEnd
97 \catcode#1=\the\catcode#1\relax
98 }%
99 \catcode#1=#2\relax
100 }
101 \TMP@EnsureCode{10}{12}% ^^J
102 \TMP@EnsureCode{39}{12}% '
103 \TMP@EnsureCode{44}{12}% ,
104 \TMP@EnsureCode{45}{12}% -
105 \TMP@EnsureCode{46}{12}% .
106 \TMP@EnsureCode{47}{12}% /
107 \TMP@EnsureCode{58}{12}% :
108 \TMP@EnsureCode{60}{12}% <
109 \TMP@EnsureCode{62}{12}% >
110 \TMP@EnsureCode{94}{7}% ^
111 \TMP@EnsureCode{96}{12}% `
112 \edef\ifvtex@AtEnd{\ifvtex@AtEnd\noexpand\endinput}

```

2.3 Check for previously defined \ifvtex

```

113 \begingroup
114 \expandafter\ifx\csname ifvtex\endcsname\relax
115 \else
116 \edef\i/{\expandafter\string\csname ifvtex\endcsname}%
117 \expandafter\ifx\csname PackageError\endcsname\relax
118 \def\x#1#2{%
119 \edef\z{#2}%
120 \expandafter\errhelp\expandafter{\z}%
121 \errmessage{Package ifvtex Error: #1}%
122 }%
123 \def\y{^^J}%
124 \newlinechar=10 %
125 \else
126 \def\x#1#2{%
127 \PackageError{ifvtex}{#1}{#2}%
128 }%
129 \def\y{\MessageBreak}%
130 \fi
131 \x{Name clash, \i/ is already defined}{%
132 Incompatible versions of \i/ can cause problems,\y
133 therefore package loading is aborted.%
134 }%
135 \endgroup
136 \expandafter\ifvtex@AtEnd

```

```

137 \fi%
138 \endgroup

```

2.4 Provide \newif

```

139 \begingroup\expandafter\expandafter\expandafter\endgroup
140 \expandafter\ifx\csname newif\endcsname\relax

\ifvtex@newif

141 \def\ifvtex@newif#1{%
142   \begingroup
143   \escapechar=-1 %
144   \expandafter\endgroup
145   \expandafter\ifvtex@@newif\string#1\@nil
146 }%

\ifvtex@@newif

147 \def\ifvtex@@newif#1#2#3\@nil{%
148   \expandafter\edef\csname#3true\endcsname{%
149     \let
150     \expandafter\noexpand\csname if#3\endcsname
151     \expandafter\noexpand\csname iftrue\endcsname
152   }%
153   \expandafter\edef\csname#3false\endcsname{%
154     \let
155     \expandafter\noexpand\csname if#3\endcsname
156     \expandafter\noexpand\csname iffalse\endcsname
157   }%
158   \csname#3false\endcsname
159 }%

160 \else

\ifvtex@newif

161 \expandafter\let\expandafter\ifvtex@newif\csname newif\endcsname
162 \fi

```

2.5 \ifvtex

\ifvtex Create and set the switch. \newif initializes the switch with \iffalse.

```

163 \ifvtex@newif\ifvtex

164 \begingroup\expandafter\expandafter\expandafter\endgroup
165 \expandafter\ifx\csname VTeXversion\endcsname\relax
166 \else
167   \begingroup\expandafter\expandafter\expandafter\endgroup
168   \expandafter\ifx\csname OpMode\endcsname\relax
169   \else
170     \vtexttrue
171   \fi
172 \fi

```

2.6 Mode and GeX switches

```

173 \ifvtex@newif\ifvtexdvi
174 \ifvtex@newif\ifvtexpdf
175 \ifvtex@newif\ifvtexps
176 \ifvtex@newif\ifvtexhtml
177 \ifvtex@newif\ifvtexgex
178 \ifvtex
179   \ifcase\OpMode\relax
180     \vtextdvitrue
181   \or % 1

```

```

182 \vtxpdftrue
183 \or % 2
184 \vtxpstrue
185 \or % 3
186 \vtxpstrue
187 \or\or\or\or\or\or\or % 10
188 \vtxhtmltrue
189 \fi
190 \begingroup\expandafter\expandafter\expandafter\endgroup
191 \expandafter\ifx\csname gexmode\endcsname\relax
192 \else
193 \ifnum\gexmode>0 %
194 \vtxgextrue
195 \fi
196 \fi
197 \fi

```

2.7 Protocol entry

Log comment:

```

198 \begingroup
199 \expandafter\ifx\csname PackageInfo\endcsname\relax
200 \def\x#1#2{%
201 \immediate\write-1{Package #1 Info: #2.}%
202 }%
203 \else
204 \let\x\PackageInfo
205 \expandafter\let\csname on@line\endcsname\empty
206 \fi
207 \x{ifvtex}{%
208 VTeX %
209 \ifvtex
210 in \ifvtexdvi DVI\fi
211 \ifvtexpdf PDF\fi
212 \ifvtexps PS\fi
213 \ifvtexhtml HTML\fi
214 \space mode %
215 with\ifvtexgex\else out\fi\space GeX %
216 \else
217 not %
218 \fi
219 detected%
220 }%
221 \endgroup
222 \ifvtex@AtEnd%
223 \endpackage

```

3 Test

3.1 Catcode checks for loading

```

224 \test1\
225 \catcode`\{=1 %
226 \catcode`\}=2 %
227 \catcode`\#=6 %
228 \catcode`\@=11 %
229 \expandafter\ifx\csname count@\endcsname\relax
230 \countdef\count@=255 %
231 \fi
232 \expandafter\ifx\csname @gobble\endcsname\relax
233 \long\def\@gobble#1{%

```

```

234 \fi
235 \expandafter\ifx\csname @firstofone\endcsname\relax
236 \long\def\@firstofone#1{#1}%
237 \fi
238 \expandafter\ifx\csname loop\endcsname\relax
239 \expandafter\@firstofone
240 \else
241 \expandafter\@gobble
242 \fi
243 {%
244 \def\loop#1\repeat{%
245 \def\body{#1}%
246 \iterate
247 }%
248 \def\iterate{%
249 \body
250 \let\next\iterate
251 \else
252 \let\next\relax
253 \fi
254 \next
255 }%
256 \let\repeat=\fi
257 }%
258 \def\RestoreCatcodes{}
259 \count@=0 %
260 \loop
261 \edef\RestoreCatcodes{%
262 \RestoreCatcodes
263 \catcode\the\count@=\the\catcode\count@\relax
264 }%
265 \ifnum\count@<255 %
266 \advance\count@ 1 %
267 \repeat
268
269 \def\RangeCatcodeInvalid#1#2{%
270 \count@=#1\relax
271 \loop
272 \catcode\count@=15 %
273 \ifnum\count@<#2\relax
274 \advance\count@ 1 %
275 \repeat
276 }
277 \def\RangeCatcodeCheck#1#2#3{%
278 \count@=#1\relax
279 \loop
280 \ifnum#3=\catcode\count@
281 \else
282 \errmessage{%
283 Character \the\count@\space
284 with wrong catcode \the\catcode\count@\space
285 instead of \number#3%
286 }%
287 \fi
288 \ifnum\count@<#2\relax
289 \advance\count@ 1 %
290 \repeat
291 }
292 \def\space{ }
293 \expandafter\ifx\csname LoadCommand\endcsname\relax
294 \def\LoadCommand{\input ifvtex.sty\relax}%
295 \fi

```

```

296 \def\Test{%
297   \RangeCatcodeInvalid{0}{47}%
298   \RangeCatcodeInvalid{58}{64}%
299   \RangeCatcodeInvalid{91}{96}%
300   \RangeCatcodeInvalid{123}{255}%
301   \catcode`\@=12 %
302   \catcode`\|=0 %
303   \catcode`\%=14 %
304   \LoadCommand
305   \RangeCatcodeCheck{0}{36}{15}%
306   \RangeCatcodeCheck{37}{37}{14}%
307   \RangeCatcodeCheck{38}{47}{15}%
308   \RangeCatcodeCheck{48}{57}{12}%
309   \RangeCatcodeCheck{58}{63}{15}%
310   \RangeCatcodeCheck{64}{64}{12}%
311   \RangeCatcodeCheck{65}{90}{11}%
312   \RangeCatcodeCheck{91}{91}{15}%
313   \RangeCatcodeCheck{92}{92}{0}%
314   \RangeCatcodeCheck{93}{96}{15}%
315   \RangeCatcodeCheck{97}{122}{11}%
316   \RangeCatcodeCheck{123}{255}{15}%
317   \RestoreCatcodes
318 }
319 \Test
320 \csname @@end\endcsname
321 \end
322 </test1>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/iftex.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/iftex.pdf](#) Documentation.

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for T_EX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```

chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/

```

¹<http://ctan.org/pkg/iftex>

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain $\text{T}_{\text{E}}\text{X}$:

```
tex ifvtex.dtx
```

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

```
ifvtex.sty      → tex/generic/oberdiek/ifvtex.sty
ifvtex.pdf      → doc/latex/oberdiek/ifvtex.pdf
test/ifvtex-test1.tex → doc/latex/oberdiek/test/ifvtex-test1.tex
ifvtex.dtx      → source/latex/oberdiek/ifvtex.dtx
```

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

4.4 Refresh file name databases

If your $\text{T}_{\text{E}}\text{X}$ distribution (te $\text{T}_{\text{E}}\text{X}$, mik $\text{T}_{\text{E}}\text{X}$, ...) relies on file name databases, you must refresh these. For example, te $\text{T}_{\text{E}}\text{X}$ users run `texhash` or `mktextlsr`.

4.5 Some details for the interested

Unpacking with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. The `.dtx` chooses its action depending on the format:

plain $\text{T}_{\text{E}}\text{X}$: Run docstrip and extract the files.

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$: Generate the documentation.

If you insist on using $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ for docstrip (really, docstrip does not need $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{ifvtex.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdf $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$:

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

5 Catalogue

The following XML file can be used as source for the [\$\text{T}_{\text{E}}\text{X}\$ Catalogue](#). The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `ifvtex.xml`.

```
323 (*catalogue)
324 <?xml version='1.0' encoding='us-ascii'?>
325 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
```

```

326 <entry datestamp='$Date$' modifier='$Author$' id='ifvtex'>
327   <name>ifvtex</name>
328   <caption>Detects use of VTeX and its facilities.</caption>
329   <authorref id='auth:oberdiek' />
330   <copyright owner='Heiko Oberdiek' year='2001,2006-2008,2010' />
331   <license type='lppl1.3' />
332   <version number='1.6' />
333   <description>
334     The package looks for VTeX and sets the switch <tt>\ifvtex</tt>.
335     In the presence of VTeX, the mode switches <tt>\ifvtexdvi</tt>,
336     <tt>\ifvtexpdf</tt> and <tt>\ifvtexps</tt> are set;
337     <tt>\ifvtexgex</tt> tells you whether GeX is operating.
338   <p/>
339   The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
340 </description>
341   <documentation details='Package documentation'
342     href='ctan:/macros/latex/contrib/oberdiek/ifvtex.pdf' />
343   <ctan file='true' path='/macros/latex/contrib/oberdiek/ifvtex.dtx' />
344   <miktex location='oberdiek' />
345   <texlive location='oberdiek' />
346   <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip' />
347 </entry>
348 </catalogue>

```

6 History

[2001/09/26 v1.0]

- First public version.

[2006/02/20 v1.1]

- DTX framework.
- Undefined tests changed.

[2007/01/10 v1.2]

- Fix of the `\ProvidesPackage` description.

[2007/09/09 v1.3]

- Catcode section added.

[2008/11/04 v1.4]

- Bug fix: Misspelled `\OpMode` (found by Hideo Umeki).

[2010/03/01 v1.5]

- Compatibility with `iniTEX`.

[2016/05/16 v1.6]

- Documentation updates.

7 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

Symbols	
\#	227
\%	303
\@	228, 301
\@firstofone	236, 239
\@gobble	233, 241
\@nil	145, 147
\@undefined	58
\\	302
\{	225
\}	226
A	
\advance	266, 274, 289
\aftergroup	29
B	
\body	245, 249
C	
\catcode	2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 69, 70, 72, 73, 74, 78, 79, 80, 81, 82, 83, 84, 87, 88, 90, 91, 92, 93, 97, 99, 225, 226, 227, 228, 263, 272, 280, 284, 301, 302, 303
\count@	230, 259, 263, 265, 266, 270, 272, 273, 274, 278, 280, 283, 284, 288, 289
\countdef	230
\csname	14, 21, 50, 66, 76, 114, 116, 117, 140, 148, 150, 151, 153, 155, 156, 158, 161, 165, 168, 191, 199, 205, 229, 232, 235, 238, 293, 320
E	
\empty	17, 18, 205
\end	321
\endcsname	14, 21, 50, 66, 76, 114, 116, 117, 140, 148, 150, 151, 153, 155, 156, 158, 161, 165, 168, 191, 199, 205, 229, 232, 235, 238, 293, 320
\endinput	29, 112
\endlinechar	4, 35, 71, 77, 89
\errhelp	120
\errmessage	121, 282
\escapechar	143
G	
\gexmode	193
I	
\i	116, 131, 132
\ifcase	179
\ifnum	193, 265, 273, 280, 288
\ifvtx	2, 163, 178, 209, 334
\ifvtx@@newif	145, 147
\ifvtx@AtEnd	95, 96, 112, 136, 222
\ifvtx@newif	141, 161, 163, 173, 174, 175, 176, 177
\ifvtxdvi	2, 173, 210, 335
\ifvtxgex	2, 177, 215, 337
\ifvtxhtml	2, 176, 213
\ifvtxpdf	2, 174, 211, 336
\ifvtxps	2, 175, 212, 336
\ifx	15, 18, 21, 50, 58, 61, 114, 117, 140, 165, 168, 191, 199, 229, 232, 235, 238, 293
\immediate	23, 52, 201
\input	294
\iterate	246, 248, 250
L	
\LoadCommand	294, 304
\loop	244, 260, 271, 279
M	
\MessageBreak	129
N	
\newlinechar	124
\next	250, 252, 254
\number	285
O	
\OpMode	179
P	
\PackageError	127
\PackageInfo	26, 204
\ProvidesPackage	19, 67
R	
\RangeCatcodeCheck	277, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316
\RangeCatcodeInvalid	269, 297, 298, 299, 300
\repeat	244, 256, 267, 275, 290
\RestoreCatcodes	258, 261, 262, 317
S	
\space	214, 215, 283, 284, 292
T	
\Test	296, 319
\the	77, 78, 79, 80, 81, 82, 83, 84, 97, 263, 283, 284
\TMP@EnsureCode	94, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111

V		X	
<code>\texdvi</code>	180	<code>\x</code>	14, 15, 18, 22, 26, 28, 51, 56, 66,
<code>\texge</code>	194		75, 87, 118, 126, 131, 200, 204, 207
<code>\texhtml</code>	188		
<code>\texpdf</code>	182		
<code>\texps</code>	184, 186		
<code>\textrue</code>	170	<code>\y</code>	123, 129, 132
W		Z	
<code>\write</code>	23, 52, 201	<code>\z</code>	119, 120