

# The `ifvtex` package

Heiko Oberdiek  
<heiko.oberdiek at gmail.com>

2010/03/01 v1.5

## Abstract

This package looks for VT<sub>E</sub>X, implements and sets the switches `\ifvtex`, `\ifvtex<mode>`, `\ifvtexgex`. It works with plain or L<sup>A</sup>T<sub>E</sub>X formats.

## Contents

<b>1 Usage</b>	<b>1</b>
<b>2 Implemenation</b>	<b>2</b>
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> . . . . .	4
2.5 <code>\ifvtex</code> . . . . .	5
2.6 Mode and GeX switches . . . . .	5
2.7 Protocol entry . . . . .	5
<b>3 Test</b>	<b>6</b>
3.1 Catcode checks for loading . . . . .	6
<b>4 Installation</b>	<b>7</b>
4.1 Download . . . . .	7
4.2 Bundle installation . . . . .	8
4.3 Package installation . . . . .	8
4.4 Refresh file name databases . . . . .	8
4.5 Some details for the interested . . . . .	8
<b>5 History</b>	<b>9</b>
[2001/09/26 v1.0] . . . . .	9
[2006/02/20 v1.1] . . . . .	9
[2007/01/10 v1.2] . . . . .	9
[2007/09/09 v1.3] . . . . .	9
[2008/11/04 v1.4] . . . . .	9
[2010/03/01 v1.5] . . . . .	9
<b>6 Index</b>	<b>9</b>

## 1 Usage

The package `ifvtex` can be used with both plain T<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X:

plain T<sub>E</sub>X: `\input ifvtex.sty`

L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>: `\usepackage{ifvtex}`

The package implements switches for  $\text{V}\text{T}\text{E}\text{X}$  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
```

$\text{V}\text{T}\text{E}\text{X}$  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  $\text{L}\text{A}\text{T}\text{E}\text{X}$ .

```
2 \begingroup
3  \catcode44 12 % ,
4  \catcode45 12 % -
5  \catcode46 12 % .
6  \catcode58 12 % :
7  \catcode64 11 % @
8  \catcode123 1 % {
9  \catcode125 2 % }
10 \expandafter\let\expandafter\x\csname ver@ifvtex.sty\endcsname
11 \ifx\x\relax % plain-Tex, first loading
12 \else
13   \def\empty{}%
14   \ifx\x\empty % LaTeX, first loading,
15     % variable is initialized, but \ProvidesPackage not yet seen
16   \else
17     \catcode35 6 % #
18     \expandafter\ifx\csname PackageInfo\endcsname\relax
19     \def\x#1#2{%
20       \immediate\write-1{Package #1 Info: #2.}%
21     }%
22   \else
23     \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
24   \fi
25   \x{ifvtex}{The package is already loaded}%
26 \aftergroup\endinput
```

```

27   \fi
28   \fi
29 \endgroup
Package identification:
30 \begingroup
31   \catcode35 6 % #
32   \catcode40 12 % (
33   \catcode41 12 % )
34   \catcode44 12 % ,
35   \catcode45 12 % -
36   \catcode46 12 % .
37   \catcode47 12 % /
38   \catcode58 12 % :
39   \catcode64 11 % @
40   \catcode91 12 % [
41   \catcode93 12 % ]
42   \catcode123 1 % {
43   \catcode125 2 % }
44   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
45     \def\x#1#2#3[#4]{\endgroup
46       \immediate\write-1{Package: #3 #4}%
47       \xdef#1{#4}%
48     }%
49   \else
50     \def\x#1#2[#3]{\endgroup
51       #2[#3]}%
52     \ifx#1@undefined
53       \xdef#1{#3}%
54     \fi
55     \ifx#1\relax
56       \xdef#1{#3}%
57     \fi
58   }%
59   \fi
60 \expandafter\x\csname ver@ifvtex.sty\endcsname
61 \ProvidesPackage{ifvtex}%
62 [2010/03/01 v1.5 Switches for detecting VTex and its modes (HO)]

```

## 2.2 Catcodes

```

63 \begingroup
64   \catcode123 1 % {
65   \catcode125 2 % }
66   \def\x{\endgroup
67     \expandafter\edef\csname ifvtex@AtEnd\endcsname{%
68       \catcode35 \the\catcode35\relax
69       \catcode64 \the\catcode64\relax
70       \catcode123 \the\catcode123\relax
71       \catcode125 \the\catcode125\relax
72     }%
73   }%
74 \x
75 \catcode35 6 % #
76 \catcode64 11 % @
77 \catcode123 1 % {
78 \catcode125 2 % }
79 \def\TMP@EnsureCode#1#2{%
80   \edef@ifvtex@AtEnd{%
81     \ifvtex@AtEnd
82     \catcode#1 \the\catcode#1\relax
83   }%
84   \catcode#1 #2\relax
85 }

```

```

86 \TMP@EnsureCode{10}{12}% ^^J
87 \TMP@EnsureCode{39}{12}% '
88 \TMP@EnsureCode{44}{12}% ,
89 \TMP@EnsureCode{45}{12}% -
90 \TMP@EnsureCode{46}{12}% .
91 \TMP@EnsureCode{47}{12}% /
92 \TMP@EnsureCode{58}{12}% :
93 \TMP@EnsureCode{60}{12}% <
94 \TMP@EnsureCode{61}{12}% =
95 \TMP@EnsureCode{62}{12}% >
96 \TMP@EnsureCode{94}{7}% ^
97 \TMP@EnsureCode{96}{12}% ‘

```

## 2.3 Check for previously defined `\ifvtex`

```

98 \begingroup
99 \expandafter\ifx\csname ifvtex\endcsname\relax
100 \else
101 \edef\i/{\expandafter\string\csname ifvtex\endcsname}%
102 \expandafter\ifx\csname PackageError\endcsname\relax
103 \def\x#1#2{%
104 \edef\z{#2}%
105 \expandafter\errhelp\expandafter{\z}%
106 \errmessage{Package ifvtex Error: #1}%
107 }%
108 \def\y{^^J}%
109 \newlinechar=10 %
110 \else
111 \def\x#1#2{%
112 \PackageError{ifvtex}{#1}{#2}%
113 }%
114 \def\y{\MessageBreak}%
115 \fi
116 \x{Name clash, \i/ is already defined}{%
117 Incompatible versions of \i/ can cause problems,\y
118 therefore package loading is aborted.%
119 }%
120 \endgroup
121 \ifvtex@AtEnd
122 \expandafter\endinput
123 \fi
124 \endgroup

```

## 2.4 Provide `\newif`

```

125 \begingroup\expandafter\expandafter\expandafter\endgroup
126 \expandafter\ifx\csname newif\endcsname\relax

\ifvtex@newif

127 \def\ifvtex@newif#1{%
128 \begingroup
129 \escapechar=-1 %
130 \expandafter\endgroup
131 \expandafter\ifvtex@@newif\string#1\@nil
132 }%

\ifvtex@@newif

133 \def\ifvtex@@newif#1#2#3\@nil{%
134 \expandafter\edef\csname#3true\endcsname{%
135 \let
136 \expandafter\noexpand\csname if#3\endcsname
137 \expandafter\noexpand\csname iftrue\endcsname
138 }%
139 \expandafter\edef\csname#3false\endcsname{%

```

```

140      \let
141      \expandafter\noexpand\csname if#3\endcsname
142      \expandafter\noexpand\csname iffalse\endcsname
143      }%
144      \csname#3false\endcsname
145      }%

146 \else

```

`\ifvtex@newif`

```

147 \expandafter\let\expandafter\ifvtex@newif\csname newif\endcsname
148 \fi

```

## 2.5 `\ifvtex`

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

```

149 \ifvtex@newif\ifvtex

150 \begingroup\expandafter\expandafter\expandafter\endgroup
151 \expandafter\ifx\csname VTexVersion\endcsname\relax
152 \else
153 \begingroup\expandafter\expandafter\expandafter\endgroup
154 \expandafter\ifx\csname OpMode\endcsname\relax
155 \else
156 \texttrue
157 \fi
158 \fi

```

## 2.6 Mode and GeX switches

```

159 \ifvtex@newif\ifvtexdvi
160 \ifvtex@newif\ifvtexpdf
161 \ifvtex@newif\ifvtexpst
162 \ifvtex@newif\ifvtexhtml
163 \ifvtex@newif\ifvtexgex
164 \ifvtex
165 \ifcase\OpMode\relax
166 \textdvi true
167 \or % 1
168 \textpdf true
169 \or % 2
170 \textpst true
171 \or % 3
172 \textpst true
173 \or\or\or\or\or\or % 10
174 \texthtml true
175 \fi
176 \begingroup\expandafter\expandafter\expandafter\endgroup
177 \expandafter\ifx\csname gexmode\endcsname\relax
178 \else
179 \ifnum\gexmode>0 %
180 \textgex true
181 \fi
182 \fi
183 \fi

```

## 2.7 Protocol entry

Log comment:

```

184 \begingroup
185 \expandafter\ifx\csname PackageInfo\endcsname\relax
186 \def\x#1#2{%
187 \immediate\write-1{Package #1 Info: #2.}%

```

```

188    }%
189 \else
190   \let\x\PackageInfo
191   \expandafter\let\csname on@line\endcsname\empty
192 \fi
193 \x{ifvtex}{%
194   VTeX %
195   \ifvtex
196     in \ifvtexdvi DVI\fi
197     \ifvtexpdf PDF\fi
198     \ifvtexps PS\fi
199     \ifvtexhtml HTML\fi
200     \space mode %
201     with\ifvtexgex\else out\fi\space GeX %
202   \else
203     not %
204   \fi
205   detected%
206 }%
207 \endgroup
208 \ifvtex@AtEnd
209 

```

### 3 Test

#### 3.1 Catcode checks for loading

```

210 (*test1)
211 \catcode'\{=1 %
212 \catcode'\}=2 %
213 \catcode'\#=6 %
214 \catcode'\@=11 %
215 \expandafter\ifx\csname count@\endcsname\relax
216   \countdef\count@=255 %
217 \fi
218 \expandafter\ifx\csname @gobble\endcsname\relax
219   \long\def\@gobble#1{}%
220 \fi
221 \expandafter\ifx\csname @firstofone\endcsname\relax
222   \long\def\@firstofone#1{#1}%
223 \fi
224 \expandafter\ifx\csname loop\endcsname\relax
225   \expandafter\@firstofone
226 \else
227   \expandafter\@gobble
228 \fi
229 {%
230   \def\loop#1\repeat{%
231     \def\body{#1}%
232     \iterate
233   }%
234   \def\iterate{%
235     \body
236     \let\next\iterate
237   \else
238     \let\next\relax
239   \fi
240   \next
241 }%
242 \let\repeat=\fi
243 }%

```

```
244 \def\RestoreCatcodes{}
245 \count@=0 %
246 \loop
247 \edef\RestoreCatcodes{%
248   \RestoreCatcodes
249   \catcode\the\count@=\the\catcode\count@\relax
250 }%
251 \ifnum\count@<255 %
252 \advance\count@ 1 %
253 \repeat
254
255 \def\RangeCatcodeInvalid#1#2{%
256 \count@=#1\relax
257 \loop
258 \catcode\count@=15 %
259 \ifnum\count@<#2\relax
260 \advance\count@ 1 %
261 \repeat
262 }
263 \expandafter\ifx\csname LoadCommand\endcsname\relax
264 \def\LoadCommand{\input ifvtex.sty\relax}%
265 \fi
266 \def\Test{%
267 \RangeCatcodeInvalid{0}{47}%
268 \RangeCatcodeInvalid{58}{64}%
269 \RangeCatcodeInvalid{91}{96}%
270 \RangeCatcodeInvalid{123}{255}%
271 \catcode'\@=12 %
272 \catcode'\=0 %
273 \catcode'\{=1 %
274 \catcode'\}=2 %
275 \catcode'\#=6 %
276 \catcode'\ [=12 %
277 \catcode'\]=12 %
278 \catcode'\%=14 %
279 \catcode'\ =10 %
280 \catcode13=5 %
281 \LoadCommand
282 \RestoreCatcodes
283 }
284 \Test
285 \csname @@end\endcsname
286 \end
287 </test1>
```

## 4 Installation

### 4.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/ifvtex.dtx](http://ctan.org/tex-archive/macros/latex/contrib/oberdiek/ifvtex.dtx) The source file.

[CTAN:macros/latex/contrib/oberdiek/ifvtex.pdf](http://ctan.org/tex-archive/macros/latex/contrib/oberdiek/ifvtex.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](http://ctan.org/tex-archive/install/macros/latex/contrib/oberdiek.tds.zip)

<sup>1</sup>[ftp://ftp.ctan.org/tex-archive/](http://ftp://ftp.ctan.org/tex-archive/)

*TDS* refers to the standard “A Directory Structure for  $\text{T}_\text{E}\text{X}$  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/
```

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

## 4.5 Some details for the interested

**Attached source.** The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk ifvtex.pdf unpack_files output .
```

**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 $\text{\LaTeX}$ :

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

## 5 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 ini $\text{\TeX}$ .

## 6 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.

	<b>Symbols</b>	
<code>\#</code> .....	<i>213, 275</i>	<code>\}</code> .....
<code>\%</code> .....	<i>278</i>	<code>\]</code> .....
<code>\@</code> .....	<i>214, 271</i>	
<code>\@firstofone</code> .....	<i>222, 225</i>	<code>\_</code> .....
<code>\@gobble</code> .....	<i>219, 227</i>	
<code>\@nil</code> .....	<i>131, 133</i>	<b>A</b>
<code>\@undefined</code> .....	<i>52</i>	<code>\advance</code> .....
<code>\[</code> .....	<i>276</i>	<code>\aftergroup</code> .....
<code>\</code> .....	<i>272</i>	
<code>\f</code> .....	<i>211, 273</i>	<b>B</b>
		<code>\body</code> .....
		<i>231, 235</i>

<b>C</b>		<b>L</b>	
\catcode	3, 4, 5, 6, 7, 8, 9, 17, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 64, 65, 68, 69, 70, 71, 75, 76, 77, 78, 82, 84, 211, 212, 213, 214, 249, 258, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280	\LoadCommand	264, 281
\count@	216, 245, 249, 251, 252, 256, 258, 259, 260	\loop	230, 246, 257
\countdef	216	<b>M</b>	
\csname	10, 18, 44, 60, 67, 99, 101, 102, 126, 134, 136, 137, 139, 141, 142, 144, 147, 151, 154, 177, 185, 191, 215, 218, 221, 224, 263, 285	\MessageBreak	114
<b>E</b>		<b>N</b>	
\empty	13, 14, 191	\newlinechar	109
\end	286	\next	236, 238, 240
\endcsname	10, 18, 44, 60, 67, 99, 101, 102, 126, 134, 136, 137, 139, 141, 142, 144, 147, 151, 154, 177, 185, 191, 215, 218, 221, 224, 263, 285	<b>O</b>	
\endinput	26, 122	\OpMode	165
\errhelp	105	<b>P</b>	
\errmsgmessage	106	\PackageError	112
\escapechar	129	\PackageInfo	23, 190
<b>G</b>		\ProvidesPackage	15, 61
\gexmode	179	<b>R</b>	
<b>I</b>		\RangeCatcodeInvalid	255, 267, 268, 269, 270
\i	101, 116, 117	\repeat	230, 242, 253, 261
\ifcase	165	\RestoreCatcodes	244, 247, 248, 282
\ifnum	179, 251, 259	<b>S</b>	
\ifvtex	2, 149, 164, 195	\space	200, 201
\ifvtex@newif	131, 133	<b>T</b>	
\ifvtex@AtEnd	80, 81, 121, 208	\Test	266, 284
\ifvtex@newif	127, 147, 149, 159, 160, 161, 162, 163	\the	68, 69, 70, 71, 82, 249
\ifvtexdvi	2, 159, 196	\TMP@EnsureCode	79, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97
\ifvtexgex	2, 163, 201	<b>V</b>	
\ifvtexhtml	2, 162, 199	\vtexdvitrue	166
\ifvtexpdf	2, 160, 197	\vtexgextrue	180
\ifvtexps	2, 161, 198	\vtexhtmltrue	174
\ifx	11, 14, 18, 44, 52, 55, 99, 102, 126, 151, 154, 177, 185, 215, 218, 221, 224, 263	\vtexpdftrue	168
\immediate	20, 46, 187	\vtexpstrue	170, 172
\input	264	\vtextrue	156
\iterate	232, 234, 236	<b>W</b>	
		\write	20, 46, 187
		<b>X</b>	
		\x	10, 11, 14, 19, 23, 25, 45, 50, 60, 66, 74, 103, 111, 116, 186, 190, 193
		<b>Y</b>	
		\y	108, 114, 117
		<b>Z</b>	
		\z	104, 105