

# The ifvtex package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2007/09/09 v1.3

## 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</b>	<b>Usage</b>	<b>1</b>
<b>2</b>	<b>Implementation</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	<code>\ifvtex</code> . . . . .	4
2.5	Mode and GeX switches . . . . .	4
2.6	Protocol entry . . . . .	5
<b>3</b>	<b>Test</b>	<b>5</b>
3.1	Catcode checks for loading . . . . .	5
<b>4</b>	<b>Installation</b>	<b>7</b>
4.1	Download . . . . .	7
4.2	Bundle installation . . . . .	7
4.3	Package installation . . . . .	7
4.4	Refresh file name databases . . . . .	8
4.5	Some details for the interested . . . . .	8
<b>5</b>	<b>History</b>	<b>8</b>
	[2001/09/26 v1.0] . . . . .	8
	[2006/02/20 v1.1] . . . . .	8
	[2007/01/10 v1.2] . . . . .	8
	[2007/09/09 v1.3] . . . . .	8
<b>6</b>	<b>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  $\text{\TeX}$  and its different modes and interprets `\TeXversion`, `\OpMode`, and `\gexmode`.

`\ifvtex`

The package provides the switch `\ifvtex`:

```
\ifvtex
... do things, if  $\text{\TeX}$  is running ...
\else
... other  $\text{\TeX}$  compiler ...
\fi
```

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

```
\boolean{ifvtex}
```

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

$\text{\TeX}$  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  $\text{\LaTeX}$ .

```
2 \begingroup
3   \catcode44 12 % ,
4   \catcode45 12 % -
5   \catcode46 12 % .
6   \catcode58 12 % :
7   \catcode64 11 % @
8   \expandafter\let\expandafter\x\csname ver@ifvtex.sty\endcsname
9   \ifcase 0%
10    \ifx\x\relax % plain
11    \else
12      \ifx\x\empty %  $\text{\LaTeX}$ 
13      \else
14        1%
15      \fi
16    \fi
17  \else
18    \catcode35 6 % #
19    \catcode123 1 % {
20    \catcode125 2 % }
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   \endgroup
30   \expandafter\endinput
31   \fi
32 \endgroup
Package identification:
33 \begingroup
34   \catcode35 6 % #
35   \catcode40 12 % (
36   \catcode41 12 % )
37   \catcode44 12 % ,
38   \catcode45 12 % -
39   \catcode46 12 % .
40   \catcode47 12 % /
41   \catcode58 12 % :
42   \catcode64 11 % @
43   \catcode123 1 % {
44   \catcode125 2 % }
45   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
46     \def\x#1#2#3[#4]{\endgroup
47       \immediate\write-1{Package: #3 #4}%
48       \xdef#1{#4}%
49     }%
50   \else
51     \def\x#1#2[#3]{\endgroup
52       #2[{#3}]%
53       \ifx#1\relax
54         \xdef#1{#3}%
55       \fi
56     }%
57   \fi
58 \expandafter\x\csname ver@ifvtex.sty\endcsname
59 \ProvidesPackage{ifvtex}%
60 [2007/09/09 v1.3 Switches for detecting VTeX and its modes (HO)]

```

## 2.2 Catcodes

```

61 \begingroup
62   \catcode123 1 % {
63   \catcode125 2 % }
64   \def\x{\endgroup
65     \expandafter\edef\csname ifvtex@AtEnd\endcsname{%
66       \catcode35 \the\catcode35\relax
67       \catcode64 \the\catcode64\relax
68       \catcode123 \the\catcode123\relax
69       \catcode125 \the\catcode125\relax
70     }%
71   }%
72 \x
73 \catcode35 6 % #
74 \catcode64 11 % @
75 \catcode123 1 % {
76 \catcode125 2 % }
77 \def\TMP@EnsureCode#1#2{%
78   \edef\ifvtex@AtEnd{%
79     \ifvtex@AtEnd
80     \catcode#1 \the\catcode#1\relax
81   }%
82   \catcode#1 #2\relax
83 }
84 \TMP@EnsureCode{10}{12}% ^^J
85 \TMP@EnsureCode{39}{12}% '

```



```

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

```

## 2.3 Check for previously defined \ifvtex

```

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

```

## 2.4 \ifvtex

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

```

123 \newif\ifvtex
124 \begingroup\expandafter\expandafter\expandafter\endgroup
125 \expandafter\ifx\csname VTeXversion\endcsname\relax
126 \else
127 \begingroup\expandafter\expandafter\expandafter\endgroup
128 \expandafter\ifx\csname OpMode\endcsname\relax
129 \else
130 \texttrue
131 \fi
132 \fi

```

## 2.5 Mode and GeX switches

```

133 \newif\ifvtexdvi
134 \newif\ifvtexpdf
135 \newif\ifvtexps
136 \newif\ifvtexhtml
137 \newif\ifvtexgex
138 \ifvtex

```



```

139 \ifcase\OpMode\relax
140   \texdvi true
141   \or % 1
142     \texpdf true
143     \or % 2
144       \texps true
145       \or % 3
146         \texps true
147         \or\or\or\or\or\or\or % 10
148         \texhtml true
149   \fi
150 \begingroup\expandafter\expandafter\expandafter\endgroup
151 \expandafter\ifx\csname gexmode\endcsname\relax
152 \else
153   \ifnum\gexmode>0 %
154     \texgex true
155   \fi
156 \fi
157 \fi

```

## 2.6 Protocol entry

Log comment:

```

158 \begingroup
159 \expandafter\ifx\csname PackageInfo\endcsname\relax
160   \def\x#1#2{%
161     \immediate\write-1{Package #1 Info: #2.}%
162   }%
163 \else
164   \let\x\PackageInfo
165   \expandafter\let\csname on@line\endcsname\empty
166 \fi
167 \x{ifvtex}{%
168   VTeX %
169   \ifvtex
170     in \ifvtexdvi DVI\fi
171     \ifvtexpdf PDF\fi
172     \ifvtexps PS\fi
173     \ifvtexhtml HTML\fi
174     \space mode %
175     with\ifvtexgex\else out\fi\space GeX %
176   \else
177     not %
178   \fi
179   detected%
180 }%
181 \endgroup
182 \ifvtex@AtEnd
183 \</package>

```

## 3 Test

### 3.1 Catcode checks for loading

```

184 \test1
185 \catcode'\{=1 %
186 \catcode'\}=2 %
187 \catcode'\#=6 %
188 \catcode'\@=11 %
189 \expandafter\ifx\csname count@\endcsname\relax
190   \countdef\count@=255 %

```



```

191 \fi
192 \expandafter\ifx\csname @gobble\endcsname\relax
193   \long\def\@gobble#1{%
194 \fi
195 \expandafter\ifx\csname @firstofone\endcsname\relax
196   \long\def\@firstofone#1{#1}%
197 \fi
198 \expandafter\ifx\csname loop\endcsname\relax
199   \expandafter\@firstofone
200 \else
201   \expandafter\@gobble
202 \fi
203 {%
204   \def\loop#1\repeat{%
205     \def\body{#1}%
206     \iterate
207   }%
208   \def\iterate{%
209     \body
210     \let\next\iterate
211   \else
212     \let\next\relax
213   \fi
214   \next
215 }%
216 \let\repeat=\fi
217 }%
218 \def\RestoreCatcodes{}
219 \count@=0 %
220 \loop
221   \edef\RestoreCatcodes{%
222     \RestoreCatcodes
223     \catcode\the\count@=\the\catcode\count@\relax
224   }%
225 \ifnum\count@<255 %
226   \advance\count@ 1 %
227 \repeat
228
229 \def\RangeCatcodeInvalid#1#2{%
230   \count@=#1\relax
231   \loop
232     \catcode\count@=15 %
233     \ifnum\count@<#2\relax
234       \advance\count@ 1 %
235     \repeat
236 }
237 \expandafter\ifx\csname LoadCommand\endcsname\relax
238   \def\LoadCommand{\input ifvtex.sty\relax}%
239 \fi
240 \def\Test{%
241   \RangeCatcodeInvalid{0}{47}%
242   \RangeCatcodeInvalid{58}{64}%
243   \RangeCatcodeInvalid{91}{96}%
244   \RangeCatcodeInvalid{123}{255}%
245   \catcode'\@=12 %
246   \catcode'\=0 %
247   \catcode'\{=1 %
248   \catcode'\}=2 %
249   \catcode'\#=6 %
250   \catcode'\[=12 %
251   \catcode'\]=12 %
252   \catcode'\%=14 %

```



```

253 \catcode'\ =10 %
254 \catcode13=5 %
255 \LoadCommand
256 \RestoreCatcodes
257 }
258 \Test
259 \csname @@end\endcsname
260 \end
261 </test1>

```

## 4 Installation

### 4.1 Download

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

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

[CTAN: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](#)

TDS refers to the standard “A Directory Structure for T<sub>E</sub>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-T<sub>E</sub>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.

---

<sup>1</sup><http://ftp.ctan.org/tex-archive/>



## 4.4 Refresh file name databases

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

## 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  $\text{\LaTeX}$ .** The `.dtx` chooses its action depending on the format:

**plain- $\text{\TeX}$ :** Run `docstrip` and extract the files.

**$\text{\LaTeX}$ :** Generate the documentation.

If you insist on using  $\text{\LaTeX}$  for `docstrip` (really, `docstrip` does not need  $\text{\LaTeX}$ ), 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.



## 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; numbers in roman refer to the code lines where the entry is used.

Symbols	
<code>\#</code> .....	187, 249
<code>\%</code> .....	252
<code>\@</code> .....	188, 245
<code>\@firstofone</code> .....	196, 199
<code>\@gobble</code> .....	193, 201
<code>\[</code> .....	250
<code>\]</code> .....	246
<code>\{</code> .....	185, 247
<code>\}</code> .....	186, 248
<code>\]</code> .....	251
<code>\_</code> .....	253
A	
<code>\advance</code> .....	226, 234
B	
<code>\body</code> .....	205, 209
C	
<code>\catcode</code> ...	3, 4, 5, 6, 7, 18, 19, 20, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 62, 63, 66, 67, 68, 69, 73, 74, 75, 76, 80, 82, 185, 186, 187, 188, 223, 232, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254
<code>\count@</code> .....	190, 219, 223, 225, 226, 230, 232, 233, 234
<code>\countdef</code> .....	190
<code>\csname</code> .....	8, 21, 45, 58, 65, 97, 99, 100, 125, 128, 151, 159, 165, 189, 192, 195, 198, 237, 259
E	
<code>\empty</code> .....	12, 165
<code>\end</code> .....	260
<code>\endcsname</code> .....	8, 21, 45, 58, 65, 97, 99, 100, 125, 128, 151, 159, 165, 189, 192, 195, 198, 237, 259
<code>\endinput</code> .....	30, 120
<code>\errhelp</code> .....	103
<code>\errmessage</code> .....	104
G	
<code>\gexmode</code> .....	153
I	
<code>\i</code> .....	99, 114, 115
<code>\ifcase</code> .....	9, 139
<code>\ifnum</code> .....	153, 225, 233
<code>\ifvtx</code> .....	2, <u>123</u> , 138, 169
<code>\ifvtx@AtEnd</code> .....	78, 79, 119, 182
<code>\ifvtxdvi</code> .....	2, 133, 170
<code>\ifvtxgex</code> .....	2, 137, 175
<code>\ifvtxhtml</code> .....	2, 136, 173
<code>\ifvtxpdf</code> .....	2, 134, 171
<code>\ifvtxps</code> .....	2, 135, 172
<code>\ifx</code> .....	10, 12, 21, 45, 53, 97, 100, 125, 128, 151, 159, 189, 192, 195, 198, 237
 	9
<code>\immediate</code> .....	23, 47, 161
<code>\input</code> .....	238
<code>\iterate</code> .....	206, 208, 210
L	
<code>\LoadCommand</code> .....	238, 255
<code>\loop</code> .....	204, 220, 231
M	
<code>\MessageBreak</code> .....	112
N	
<code>\newif</code> ....	123, 133, 134, 135, 136, 137
<code>\newlinechar</code> .....	107
<code>\next</code> .....	210, 212, 214
O	
<code>\OpMode</code> .....	139
P	
<code>\PackageError</code> .....	110
<code>\PackageInfo</code> .....	26, 164
<code>\ProvidesPackage</code> .....	59
R	
<code>\RangeCatcodeInvalid</code> .....	229, 241, 242, 243, 244
<code>\repeat</code> .....	204, 216, 227, 235
<code>\RestoreCatcodes</code> ..	218, 221, 222, 256
S	
<code>\space</code> .....	174, 175
T	
<code>\Test</code> .....	240, 258
<code>\the</code> .....	66, 67, 68, 69, 80, 223
<code>\TMP@EnsureCode</code> .....	77, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95
V	
<code>\vtxdvitrue</code> .....	140
<code>\vtxgextrue</code> .....	154
<code>\vtxhtmltrue</code> .....	148
<code>\vtxpdftrue</code> .....	142
<code>\vtxpsttrue</code> .....	144, 146
<code>\vtxtrue</code> .....	130
W	
<code>\write</code> .....	23, 47, 161
X	
<code>\x</code> ..	8, 10, 12, 22, 26, 28, 46, 51, 58, 64, 72, 101, 109, 114, 160, 164, 167
Y	
<code>\y</code> .....	106, 112, 115
Z	
<code>\z</code> .....	102, 103