

# 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> . . . . .	3
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>6</b>
4.1	Download . . . . .	6
4.2	Bundle installation . . . . .	6
4.3	Package installation . . . . .	6
4.4	Refresh file name databases . . . . .	7
4.5	Some details for the interested . . . . .	7
<b>5</b>	<b>History</b>	<b>7</b>
[2001/09/26 v1.0]	. . . . .	7
[2006/02/20 v1.1]	. . . . .	7
[2007/01/10 v1.2]	. . . . .	8
[2007/09/09 v1.3]	. . . . .	8
<b>6</b>	<b>Index</b>	<b>8</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 % LaTeX
13      \else
14        1%
15      \fi
16    \fi
17  \else
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  \endgroup
```

```

27 \expandafter\endinput
28 \fi
29 \endgroup
Package identification:
30 \begingroup
31 \catcode40 12 % (
32 \catcode41 12 % )
33 \catcode44 12 % ,
34 \catcode45 12 % -
35 \catcode46 12 % .
36 \catcode47 12 % /
37 \catcode58 12 % :
38 \catcode64 11 % @
39 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
40 \def\x#1#2#3[#4]{\endgroup
41 \immediate\write-1{Package: #3 #4}%
42 \xdef#1{#4}%
43 }%
44 \else
45 \def\x#1#2[#3]{\endgroup
46 #2[#{#3}]]%
47 \ifx#1\relax
48 \xdef#1{#3}%
49 \fi
50 }%
51 \fi
52 \expandafter\x\csname ver@ifvtex.sty\endcsname
53 \ProvidesPackage{ifvtex}%
54 [2007/09/09 v1.3 Switches for detecting VTeX and its modes (HO)]

```

## 2.2 Catcodes

```

55 \expandafter\edef\csname ifvtex@AtEnd\endcsname{%
56 \catcode64 \the\catcode64\relax
57 }
58 \catcode64 11 % @
59 \def\TMP@EnsureCode#1#2{%
60 \edef\ifvtex@AtEnd{%
61 \ifvtex@AtEnd
62 \catcode#1 \the\catcode#1\relax
63 }%
64 \catcode#1 #2\relax
65 }
66 \TMP@EnsureCode{10}{12}% ^^J
67 \TMP@EnsureCode{39}{12}% '
68 \TMP@EnsureCode{44}{12}% ,
69 \TMP@EnsureCode{45}{12}% -
70 \TMP@EnsureCode{46}{12}% .
71 \TMP@EnsureCode{47}{12}% /
72 \TMP@EnsureCode{58}{12}% :
73 \TMP@EnsureCode{60}{12}% <
74 \TMP@EnsureCode{61}{12}% =
75 \TMP@EnsureCode{62}{12}% >
76 \TMP@EnsureCode{94}{7}% ^
77 \TMP@EnsureCode{96}{12}% `

```

## 2.3 Check for previously defined \ifvtex

```

78 \begingroup
79 \expandafter\ifx\csname ifvtex\endcsname\relax
80 \else
81 \edef\i/{\expandafter\string\csname ifvtex\endcsname}%
82 \expandafter\ifx\csname PackageError\endcsname\relax
83 \def\x#1#2{%

```

```

84      \edef\z{#2}%
85      \expandafter\errhelp\expandafter{\z}%
86      \errmessage{Package ifvtex Error: #1}%
87    }%
88    \def\y{^^J}%
89    \newlinechar=10 %
90  \else
91    \def\x#1#2{%
92      \PackageError{ifvtex}{#1}{#2}%
93    }%
94    \def\y{\MessageBreak}%
95  \fi
96  \x{Name clash, \i/ is already defined}{%
97    Incompatible versions of \i/ can cause problems,\y
98    therefore package loading is aborted.%
99  }%
100 \endgroup
101 \ifvtex@AtEnd
102 \expandafter\endinput
103 \fi
104 \endgroup

```

## 2.4 \ifvtex

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

```

105 \newif\ifvtex
106 \begingroup\expandafter\expandafter\expandafter\endgroup
107 \expandafter\ifx\csname VTeXversion\endcsname\relax
108 \else
109   \begingroup\expandafter\expandafter\expandafter\endgroup
110   \expandafter\ifx\csname OpMode\endcsname\relax
111   \else
112     \vtexttrue
113   \fi
114 \fi

```

## 2.5 Mode and GeX switches

```

115 \newif\ifvtexdvi
116 \newif\ifvtexpdf
117 \newif\ifvtexps
118 \newif\ifvtexhtml
119 \newif\ifvtexgex
120 \ifvtex
121   \ifcase\OpMode\relax
122     \vtextdvi true
123   \or % 1
124     \vtextpdf true
125   \or % 2
126     \vtextps true
127   \or % 3
128     \vtextps true
129   \or\or\or\or\or\or\or % 10
130     \vtexthtml true
131   \fi
132 \begingroup\expandafter\expandafter\expandafter\endgroup
133 \expandafter\ifx\csname gexmode\endcsname\relax
134 \else
135   \ifnum\gexmode>0 %
136     \vtextgex true
137   \fi
138 \fi

```

139 \fi

## 2.6 Protocol entry

Log comment:

```
140 \begingroup
141   \expandafter\ifx\csname PackageInfo\endcsname\relax
142     \def\x#1#2{%
143       \immediate\write-1{Package #1 Info: #2.}%
144     }%
145   \else
146     \let\x\PackageInfo
147     \expandafter\let\csname on@line\endcsname\empty
148   \fi
149   \x{ifvtex}{%
150     VTeX %
151     \ifvtex
152       in \ifvtexdvi DVI\fi
153       \ifvtexpdf PDF\fi
154       \ifvtexps PS\fi
155       \ifvtexhtml HTML\fi
156       \space mode %
157       with\ifvtexgex\else out\fi\space GeX %
158     \else
159       not %
160     \fi
161     detected%
162   }%
163 \endgroup
164 \ifvtex@AtEnd
165 \</package>
```

## 3 Test

### 3.1 Catcode checks for loading

```
166 <*test1>
167 \catcode'\@=11 %
168 \def\RestoreCatcodes{}
169 \count@=0 %
170 \loop
171   \edef\RestoreCatcodes{%
172     \RestoreCatcodes
173     \catcode\the\count@=\the\catcode\count@\relax
174   }%
175   \ifnum\count@<255 %
176     \advance\count@\@ne
177   \repeat
178
179 \def\RangeCatcodeInvalid#1#2{%
180   \count@=#1\relax
181   \loop
182     \catcode\count@=15 %
183     \ifnum\count@<#2\relax
184       \advance\count@\@ne
185     \repeat
186 }
187 \def\Test{%
188   \RangeCatcodeInvalid{0}{47}%
189   \RangeCatcodeInvalid{58}{64}%
190   \RangeCatcodeInvalid{91}{96}%
```

```

191 \RangeCatcodeInvalid{123}{255}%
192 \catcode'\@=12 %
193 \catcode'\=0 %
194 \catcode'\{=1 %
195 \catcode'\}=2 %
196 \catcode'\#=6 %
197 \catcode'\[=12 %
198 \catcode'\]=12 %
199 \catcode'\%=14 %
200 \catcode'\_ =10 %
201 \catcode13=5 %
202 \input ifvtex.sty\relax
203 \RestoreCatcodes
204 }
205 \Test
206 \csname @@end\endcsname
207 \end
208 </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:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

*TDS* refers to the standard “A Directory Structure for  $\TeX$  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- $\TeX$ :

```
tex ifvtex.dtx
```

---

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

**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  $\text{\TeX}$  distribution (`te $\text{\TeX}$` , `mik $\text{\TeX}$` , ...) relies on file name databases, you must refresh these. For example, `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> .....	196
<code>\%</code> .....	199
<code>\@</code> .....	167, 192
<code>\@ne</code> .....	176, 184
<code>\[</code> .....	197
<code>\]</code> .....	193
<code>\{</code> .....	194
<code>\}</code> .....	195
<code>\]</code> .....	198
<code>\_</code> .....	200
A	
<code>\advance</code> .....	176, 184
C	
<code>\catcode</code> .....	3, 4, 5, 6, 7, 31, 32, 33, 34, 35, 36, 37, 38, 56, 58, 62, 64, 167, 173, 182, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201
<code>\count@</code> .....	169, 173, 175, 176, 180, 182, 183, 184
<code>\csname</code> .....	8, 18, 39, 52, 55, 79, 81, 82, 107, 110, 133, 141, 147, 206
E	
<code>\empty</code> .....	12, 147
<code>\end</code> .....	207
<code>\endcsname</code> .....	8, 18, 39, 52, 55, 79, 81, 82, 107, 110, 133, 141, 147, 206
<code>\endinput</code> .....	27, 102
<code>\errhelp</code> .....	85
<code>\errmessage</code> .....	86
G	
<code>\gexmode</code> .....	135
I	
<code>\i</code> .....	81, 96, 97
<code>\ifcase</code> .....	9, 121
<code>\ifnum</code> .....	135, 175, 183
<code>\ifvtex</code> .....	2, <u>105</u> , 120, 151
<code>\ifvtex@AtEnd</code> .....	60, 61, 101, 164
<code>\ifvtexdvi</code> .....	2, 115, 152
<code>\ifvtexgex</code> .....	2, 119, 157
<code>\ifvtexhtml</code> .....	2, 118, 155
<code>\ifvtexpdf</code> .....	2, 116, 153
<code>\ifvtexps</code> .....	2, 117, 154
<code>\ifx</code> .....	10, 12, 18, 39, 47, 79, 82, 107, 110, 133, 141
<code>\immediate</code> .....	20, 41, 143
<code>\input</code> .....	202
L	
<code>\loop</code> .....	170, 181
M	
<code>\MessageBreak</code> .....	94
N	
<code>\newif</code> .....	105, 115, 116, 117, 118, 119
<code>\newlinechar</code> .....	89
O	
<code>\OpMode</code> .....	121
P	
<code>\PackageError</code> .....	92
<code>\PackageInfo</code> .....	23, 146
<code>\ProvidesPackage</code> .....	53
R	
<code>\RangeCatcodeInvalid</code> .....	179, 188, 189, 190, 191
<code>\repeat</code> .....	177, 185
<code>\RestoreCatcodes</code> ..	168, 171, 172, 203
S	
<code>\space</code> .....	156, 157
T	
<code>\Test</code> .....	187, 205
<code>\the</code> .....	56, 62, 173
<code>\TMP@EnsureCode</code> .....	59, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77
V	
<code>\vtexdvitrue</code> .....	122
<code>\vtexgextrue</code> .....	136
<code>\vtexhtmltrue</code> .....	130
<code>\vtexpdftrue</code> .....	124
<code>\vtexpstrue</code> .....	126, 128
<code>\vtexttrue</code> .....	112
W	
<code>\write</code> .....	20, 41, 143



	<b>X</b>	<b>Y</b>
		<b>\y</b> .....
		88, 94, 97
<b>\x</b> .....	8, 10, 12, 19, 23, 25,	<b>Z</b>
	40, 45, 52, 83, 91, 96, 142, 146, 149	<b>\z</b> .....
		84, 85