

# The `ifvtex` package

Heiko Oberdiek  
<heiko.oberdiek at gmail.com>

2008/11/04 v1.4

## Abstract

This package looks for  $\text{V}\text{T}_{\text{E}}\text{X}$ , implements and sets the switches `\ifvtex`, `\ifvtex<mode>`, `\ifvtexgex`. It works with plain or  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  formats.

## Contents

<b>1 Usage</b>	<b>1</b>
<b>2 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 Test</b>	<b>5</b>
3.1 Catcode checks for loading	5
<b>4 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 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]	9
[2008/11/04 v1.4]	9
<b>6 Index</b>	<b>9</b>

## 1 Usage

The package `ifvtex` can be used with both plain- $\text{T}_{\text{E}}\text{X}$  and  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ :

plain- $\text{T}_{\text{E}}\text{X}$ : `\input ifvtex.sty`

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ : `\usepackage{ifvtex}`

The package implements switches for  $\text{V}\text{T}_{\text{E}}\text{X}$  and its different modes and interprets  $\backslash\text{V}\text{T}_{\text{E}}\text{Xversion}$ ,  $\backslash\text{O}\text{p}\text{Mode}$ , and  $\backslash\text{gexmode}$ .

$\backslash\text{ifvtex}$

The package provides the switch  $\backslash\text{ifvtex}$ :

```
 $\backslash\text{ifvtex}$ 
... do things, if  $\text{V}\text{T}_{\text{E}}\text{X}$  is running ...
 $\backslash\text{else}$ 
... other  $\text{T}_{\text{E}}\text{X}$  compiler ...
 $\backslash\text{fi}$ 
```

Users of the package  $\text{ifthen}$  can use the switch as boolean:

```
 $\backslash\text{boolean}\{\text{ifvtex}\}$ 
```

$\backslash\text{ifvtexdvi}$   
 $\backslash\text{ifvtexpdf}$   
 $\backslash\text{ifvtexps}$   
 $\backslash\text{ifvtexhtml}$

$\text{V}\text{T}_{\text{E}}\text{X}$  knows different output modes that can be asked by these switches.

$\backslash\text{ifvtexgex}$

This switch shows, whether GeX is available.

## 2 Implementation

### 2.1 Reload check and package identification

```
1  $\langle$ *package $\rangle$ 
```

Reload check, especially if the package is not used with  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .

```
2  $\backslash\text{begingroup}$ 
3  $\backslash\text{catcode}44\ 12\ \%$  ,
4  $\backslash\text{catcode}45\ 12\ \%$  -
5  $\backslash\text{catcode}46\ 12\ \%$  .
6  $\backslash\text{catcode}58\ 12\ \%$  :
7  $\backslash\text{catcode}64\ 11\ \%$  @
8  $\backslash\text{catcode}123\ 1\ \%$  {
9  $\backslash\text{catcode}125\ 2\ \%$  }
10  $\backslash\text{expandafter}\backslash\text{let}\backslash\text{expandafter}\backslash\text{x}\backslash\text{csname}\ \text{ver@ifvtex.sty}\backslash\text{endcsname}$ 
11  $\backslash\text{ifx}\backslash\text{relax}\ \%$  plain-TeX, first loading
12  $\backslash\text{else}$ 
13  $\backslash\text{def}\backslash\text{empty}\{\}\%$ 
14  $\backslash\text{ifx}\backslash\text{empty}\ \%$  LaTeX, first loading,
15  $\%$  variable is initialized, but  $\backslash\text{ProvidesPackage}$  not yet seen
16  $\backslash\text{else}$ 
17  $\backslash\text{catcode}35\ 6\ \%$  #
18  $\backslash\text{expandafter}\backslash\text{ifx}\backslash\text{csname}\ \text{PackageInfo}\backslash\text{endcsname}\backslash\text{relax}$ 
19  $\backslash\text{def}\backslash\text{x}\#1\#2\{\%$ 
20  $\backslash\text{immediate}\backslash\text{write}-1\{\text{Package \#1 Info: \#2.}\}\%$ 
21  $\}$ 
22  $\backslash\text{else}$ 
23  $\backslash\text{def}\backslash\text{x}\#1\#2\{\backslash\text{PackageInfo}\{\#1\}\{\#2,\ \text{stopped}\}\}\%$ 
24  $\backslash\text{fi}$ 
25  $\backslash\text{x}\{\text{ifvtex}\}\{\text{The package is already loaded}\}\%$ 
26  $\backslash\text{aftergroup}\backslash\text{endinput}$ 
```

```

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 [2008/11/04 v1.4 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 \ifvtex

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

```

125 \newif\ifvtex
126 \begingroup\expandafter\expandafter\expandafter\endgroup
127 \expandafter\ifx\csname VTeXversion\endcsname\relax
128 \else
129 \begingroup\expandafter\expandafter\expandafter\endgroup
130 \expandafter\ifx\csname OpMode\endcsname\relax
131 \else
132 \vtexttrue
133 \fi
134 \fi

```

## 2.5 Mode and GeX switches

```

135 \newif\ifvtexdvi
136 \newif\ifvtexpdf
137 \newif\ifvtexps
138 \newif\ifvtexhtml

```

```

139 \newif\ifvtexgex
140 \ifvtex
141   \ifcase\OpMode\relax
142     \vtexdvitrue
143   \or % 1
144     \vtexpdftrue
145   \or % 2
146     \vtexpstrue
147   \or % 3
148     \vtexpstrue
149   \or\or\or\or\or\or\or % 10
150     \vtexhtmltrue
151   \fi
152 \begingroup\expandafter\expandafter\expandafter\endgroup
153 \expandafter\ifx\csname gexmode\endcsname\relax
154 \else
155   \ifnum\gexmode>0 %
156     \vtexgextrue
157   \fi
158 \fi
159 \fi

```

## 2.6 Protocol entry

Log comment:

```

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

```

## 3 Test

### 3.1 Catcode checks for loading

```

186 \test1
187 \catcode'\{=1 %
188 \catcode'\}=2 %
189 \catcode'\#=6 %
190 \catcode'\@=11 %

```

```

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

```

```

253 \catcode'\]=12 %
254 \catcode'\%=14 %
255 \catcode'\ =10 %
256 \catcode13=5 %
257 \LoadCommand
258 \RestoreCatcodes
259 }
260 \Test
261 \csname @@end\endcsname
262 \end
263 </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

```

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

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  $\TeX$  distribution (`teTeX`, `mikTeX`, ...) relies on file name databases, you must refresh these. For example, `teTeX` 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  $\LaTeX$ .** The `.dtx` chooses its action depending on the format:

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

**$\LaTeX$ :** Generate the documentation.

If you insist on using  $\LaTeX$  for `docstrip` (really, `docstrip` does not need  $\LaTeX$ ), 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 $\LaTeX$` :

```
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).

## 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	I
<code>\#</code> ..... 189, 251	<code>\i</code> ..... 101, 116, 117
<code>\%</code> ..... 254	<code>\ifcase</code> ..... 141
<code>\@</code> ..... 190, 247	<code>\ifnum</code> ..... 155, 227, 235
<code>\@firstofone</code> ..... 198, 201	<code>\ifvtx</code> ..... 2, <u>125</u> , 140, 171
<code>\@gobble</code> ..... 195, 203	<code>\ifvtx@AtEnd</code> ..... 80, 81, 121, 184
<code>\@undefined</code> ..... 52	<code>\ifvtxdvi</code> ..... 2, 135, 172
<code>\[</code> ..... 252	<code>\ifvtxgex</code> ..... 2, 139, 177
<code>\]</code> ..... 248	<code>\ifvtxhtml</code> ..... 2, 138, 175
<code>\{</code> ..... 187, 249	<code>\ifvtxpdf</code> ..... 2, 136, 173
<code>\}</code> ..... 188, 250	<code>\ifvtxps</code> ..... 2, 137, 174
<code>\]</code> ..... 253	<code>\ifx</code> ..... 11, 14, 18, 44, 52, 55, 99, 102, 127, 130, 153, 161, 191, 194, 197, 200, 239
<code>\_</code> ..... 255	<code>\immediate</code> ..... 20, 46, 163
<b>A</b>	<code>\input</code> ..... 240
<code>\advance</code> ..... 228, 236	<code>\iterate</code> ..... 208, 210, 212
<code>\aftergroup</code> ..... 26	<b>L</b>
<b>B</b>	<code>\LoadCommand</code> ..... 240, 257
<code>\body</code> ..... 207, 211	<code>\loop</code> ..... 206, 222, 233
<b>C</b>	<b>M</b>
<code>\catcode</code> 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, 187, 188, 189, 190, 225, 234, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256	<code>\MessageBreak</code> ..... 114
<code>\count@</code> ..... 192, 221, 225, 227, 228, 232, 234, 235, 236	<b>N</b>
<code>\countdef</code> ..... 192	<code>\newif</code> .... 125, 135, 136, 137, 138, 139
<code>\csname</code> ..... 10, 18, 44, 60, 67, 99, 101, 102, 127, 130, 153, 161, 167, 191, 194, 197, 200, 239, 261	<code>\newlinechar</code> ..... 109
<b>E</b>	<code>\next</code> ..... 212, 214, 216
<code>\empty</code> ..... 13, 14, 167	<b>O</b>
<code>\end</code> ..... 262	<code>\OpMode</code> ..... 141
<code>\endcsname</code> .... 10, 18, 44, 60, 67, 99, 101, 102, 127, 130, 153, 161, 167, 191, 194, 197, 200, 239, 261	<b>P</b>
<code>\endinput</code> ..... 26, 122	<code>\PackageError</code> ..... 112
<code>\errhelp</code> ..... 105	<code>\PackageInfo</code> ..... 23, 166
<code>\errmessage</code> ..... 106	<code>\ProvidesPackage</code> ..... 15, 61
<b>G</b>	<b>R</b>
<code>\gexmode</code> ..... 155	<code>\RangeCatcodeInvalid</code> ..... 231, 243, 244, 245, 246
	<code>\repeat</code> ..... 206, 218, 229, 237
	<code>\RestoreCatcodes</code> .. 220, 223, 224, 258
	<b>S</b>
	<code>\space</code> ..... 176, 177
	<b>T</b>
	<code>\Test</code> ..... 242, 260

<code>\the</code> .....	68, 69, 70, 71, 82, 225		
<code>\TMP@EnsureCode</code> .....	79, 86, 87,		
	88, 89, 90, 91, 92, 93, 94, 95, 96, 97		
			<b>W</b>
		<code>\write</code> .....	20, 46, 163
			<b>X</b>
		<code>\x</code> ..	10, 11, 14, 19, 23, 25, 45, 50, 60,
			66, 74, 103, 111, 116, 162, 166, 169
			<b>Y</b>
		<code>\y</code> .....	108, 114, 117
			<b>Z</b>
		<code>\z</code> .....	104, 105
	<b>V</b>		
<code>\vtexdvitrue</code> .....	142		
<code>\vtexgextrue</code> .....	156		
<code>\vtexhtmltrue</code> .....	150		
<code>\texpdftrue</code> .....	144		
<code>\texpstrue</code> .....	146, 148		
<code>\textrue</code> .....	132		