

The `ifvtex` package

Heiko Oberdiek
<heiko.oberdiek at gmail.com>

2010/03/01 v1.5

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

1 Usage	1
2 Implementation	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>	4
2.5 <code>\ifvtex</code>	5
2.6 Mode and GeX switches	5
2.7 Protocol entry	5
3 Test	6
3.1 Catcode checks for loading	6
4 Installation	7
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
5 History	9
[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
6 Index	9

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 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 Implemenation

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 [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\ifvtexps
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     \textps true
171   \or % 3
172     \textps 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 \endpackage

```

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 CTAN¹:

[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](#)

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

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
```

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 \TeX distribution (`te \TeX` , `mik \TeX` , ...) relies on file name databases, you must refresh these. For example, `te \TeX` 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 pdfL^AT_EX:

```
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 iniT_EX.

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.

Symbols		
<code>\#</code>	213, 275	<code>\}</code> 212, 274
<code>\%</code>	278	<code>\]</code> 277
<code>\@</code>	214, 271	
<code>\@firstofone</code>	222, 225	<code>_</code> 279
<code>\@gobble</code>	219, 227	
<code>\@nil</code>	131, 133	A
<code>\@undefined</code>	52	<code>\advance</code> 252, 260
<code>\[</code>	276	<code>\aftergroup</code> 26
<code>\]</code>	272	B
<code>\{</code>	211, 273	<code>\body</code> 231, 235

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