The ifvtex package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2008/11/04 v1.4

Abstract

This package looks for VTEX, implements and sets the switches `\ifvtex`, `\ifvtex⟨mode⟩`, `\ifvtexgex`. It works with plain or LaTeX formats.

Contents

1 Usage **1**

2 Implemenation **2**
2.1 Reload check and package identification . . . 2
2.2 Catcodes . . . 3
2.3 Check for previously defined `\ifvtex` . . . 4
2.4 `\ifvtex` . . . 4
2.5 Mode and GeX switches . . . 4
2.6 Protocol entry . . . 5

3 Test **5**
3.1 Catcode checks for loading . . . 5

4 Installation **7**
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

5 History **8**
[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

6 Index **9**

1 Usage

The package ifvtex can be used with both plain-TEX and LaTeX:

plain-TEX: `\input ifvtex.sty`

LaTeX 2ε: `\usepackage{ifvtex}`

The package implements switches for VTEX 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`

VTEX 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 LATEX.

```
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
```

```
    \fi
  \fi
\endgroup
```

Package identification:

```
\begingroup
  \catcode35 6 % #
  \catcode40 12 % (
  \catcode41 12 % )
  \catcode44 12 % ,
  \catcode45 12 % -
  \catcode46 12 % .
  \catcode47 12 % /
  \catcode58 12 % :
  \catcode64 11 % @
  \catcode91 12 % [
  \catcode93 12 % ]
  \catcode123 1 % {
  \catcode125 2 % }
  \expandafter\ifx\csname ProvidesPackage\endcsname\relax
    \def\x#1#2#3[#4]{\endgroup
      \immediate\write-1{Package: #3 #4}%
      \xdef#1{#4}%
    }%
  \else
    \def\x#1#2[#3]{\endgroup
      #2[{#3}]%
      \ifx#1\@undefined
        \xdef#1{#3}%
      \fi
      \ifx#1\relax
        \xdef#1{#3}%
      \fi
    }%
  \fi
\expandafter\x\csname ver@ifvtex.sty\endcsname
\ProvidesPackage{ifvtex}%
  [2008/11/04 v1.4 Switches for detecting VTeX and its modes (HO)]
```

2.2 Catcodes

```
\begingroup
  \catcode123 1 % {
  \catcode125 2 % }
  \def\x{\endgroup
    \expandafter\edef\csname ifvtex@AtEnd\endcsname{%
      \catcode35 \the\catcode35\relax
      \catcode64 \the\catcode64\relax
      \catcode123 \the\catcode123\relax
      \catcode125 \the\catcode125\relax
    }%
  }%
\x
\catcode35 6 % #
\catcode64 11 % @
\catcode123 1 % {
\catcode125 2 % }
\def\TMP@EnsureCode#1#2{%
  \edef\ifvtex@AtEnd{%
    \ifvtex@AtEnd
    \catcode#1 \the\catcode#1\relax
  }%
  \catcode#1 #2\relax
}
```

```
\TMP@EnsureCode{10}{12}% ^^J
\TMP@EnsureCode{39}{12}% '
\TMP@EnsureCode{44}{12}% ,
\TMP@EnsureCode{45}{12}% -
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{58}{12}% :
\TMP@EnsureCode{60}{12}% <
\TMP@EnsureCode{61}{12}% =
\TMP@EnsureCode{62}{12}% >
\TMP@EnsureCode{94}{7}% ^
\TMP@EnsureCode{96}{12}% `
```

2.3 Check for previously defined \ifvtex

```
\begingroup
  \expandafter\ifx\csname ifvtex\endcsname\relax
  \else
    \edef\i/{\expandafter\string\csname ifvtex\endcsname}%
    \expandafter\ifx\csname PackageError\endcsname\relax
      \def\x#1#2{%
        \edef\z{#2}%
        \expandafter\errhelp\expandafter{\z}%
        \errmessage{Package ifvtex Error: #1}%
      }%
      \def\y{^^J}%
      \newlinechar=10 %
    \else
      \def\x#1#2{%
        \PackageError{ifvtex}{#1}{#2}%
      }%
      \def\y{\MessageBreak}%
    \fi
    \x{Name clash, \i/ is already defined}{%
      Incompatible versions of \i/ can cause problems,\y
      therefore package loading is aborted.%
    }%
    \endgroup
    \ifvtex@AtEnd
    \expandafter\endinput
  \fi
\endgroup
```

2.4 \ifvtex

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

```
\newif\ifvtex
```

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname VTeXversion\endcsname\relax
\else
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname OpMode\endcsname\relax
  \else
    \vtextrue
  \fi
\fi
```

2.5 Mode and GeX switches

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

```
\newif\ifvtexgex
\ifvtex
  \ifcase\OpMode\relax
    \vtexdvitrue
  \or % 1
    \vtexpdftrue
  \or % 2
    \vtexpstrue
  \or % 3
    \vtexpstrue
  \or\or\or\or\or\or\or % 10
    \vtexhtmltrue
  \fi
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname gexmode\endcsname\relax
  \else
    \ifnum\gexmode>0 %
      \vtexgextrue
    \fi
  \fi
\fi
```

2.6 Protocol entry

Log comment:

```
\begingroup
  \expandafter\ifx\csname PackageInfo\endcsname\relax
    \def\x#1#2{%
      \immediate\write-1{Package #1 Info: #2.}%
    }%
  \else
    \let\x\PackageInfo
    \expandafter\let\csname on@line\endcsname\empty
  \fi
  \x{ifvtex}{%
    VTeX %
    \ifvtex
      in \ifvtexdvi DVI\fi
         \ifvtexpdf PDF\fi
         \ifvtexps PS\fi
         \ifvtexhtml HTML\fi
      \space mode %
      with\ifvtexgex\else out\fi\space GeX %
    \else
      not %
    \fi
    detected%
  }%
\endgroup
```

```
\ifvtex@AtEnd
```

⟨/package⟩

3 Test

3.1 Catcode checks for loading

⟨∗test1⟩

```
\catcode`\{=1 %
\catcode`\}=2 %
\catcode`\#=6 %
\catcode`\@=11 %
```

```
\expandafter\ifx\csname count@\endcsname\relax
  \countdef\count@=255 %
\fi
\expandafter\ifx\csname @gobble\endcsname\relax
  \long\def\@gobble#1{}%
\fi
\expandafter\ifx\csname @firstofone\endcsname\relax
  \long\def\@firstofone#1{#1}%
\fi
\expandafter\ifx\csname loop\endcsname\relax
  \expandafter\@firstofone
\else
  \expandafter\@gobble
\fi
{%
  \def\loop#1\repeat{%
    \def\body{#1}%
    \iterate
  }%
  \def\iterate{%
    \body
      \let\next\iterate
    \else
      \let\next\relax
    \fi
    \next
  }%
  \let\repeat=\fi
}%
\def\RestoreCatcodes{}
\count@=0 %
\loop
  \edef\RestoreCatcodes{%
    \RestoreCatcodes
    \catcode\the\count@=\the\catcode\count@\relax
  }%
\ifnum\count@<255 %
  \advance\count@ 1 %
\repeat

\def\RangeCatcodeInvalid#1#2{%
  \count@=#1\relax
  \loop
    \catcode\count@=15 %
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\expandafter\ifx\csname LoadCommand\endcsname\relax
  \def\LoadCommand{\input ifvtex.sty\relax}%
\fi
\def\Test{%
  \RangeCatcodeInvalid{0}{47}%
  \RangeCatcodeInvalid{58}{64}%
  \RangeCatcodeInvalid{91}{96}%
  \RangeCatcodeInvalid{123}{255}%
  \catcode`\@=12 %
  \catcode`\\=0 %
  \catcode`\{=1 %
  \catcode`\}=2 %
  \catcode`\#=6 %
  \catcode`\[=12 %
```

```
  \catcode`\]=12 %
  \catcode`\%=14 %
  \catcode`\ =10 %
  \catcode13=5 %
  \LoadCommand
  \RestoreCatcodes
}
\Test
\csname @@end\endcsname
\end
```

⟨/test1⟩

4 Installation

4.1 Download

Package. This package is available on CTAN[1]:

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 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
```

[1] ftp://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 pdfLATEX:

```
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: Mispelled `\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

\# 189, 251
\% 254
\@ 190, 247
\@firstofone 198, 201
\@gobble 195, 203
\@undefined 52
\[. 252
\\ 248
\{ 187, 249
\} 188, 250
\] 253

\␣ 255

A

\advance 228, 236
\aftergroup 26

B

\body 207, 211

C

\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, 187, 188, 189, 190, 225, 234, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256
\count@ 192, 221, 225, 227, 228, 232, 234, 235, 236
\countdef 192
\csname 10, 18, 44, 60, 67, 99, 101, 102, 127, 130, 153, 161, 167, 191, 194, 197, 200, 239, 261

E

\empty 13, 14, 167
\end 262
\endcsname 10, 18, 44, 60, 67, 99, 101, 102, 127, 130, 153, 161, 167, 191, 194, 197, 200, 239, 261
\endinput 26, 122
\errhelp 105
\errmessage 106

G

\gexmode 155

I

\i 101, 116, 117
\ifcase 141
\ifnum 155, 227, 235
\ifvtex *2*, 125, 140, 171
\ifvtex@AtEnd 80, 81, 121, 184
\ifvtexdvi *2*, 135, 172
\ifvtexgex *2*, 139, 177
\ifvtexhtml *2*, 138, 175
\ifvtexpdf *2*, 136, 173
\ifvtexps *2*, 137, 174
\ifx 11, 14, 18, 44, 52, 55, 99, 102, 127, 130, 153, 161, 191, 194, 197, 200, 239
\immediate 20, 46, 163
\input 240
\iterate 208, 210, 212

L

\LoadCommand 240, 257
\loop 206, 222, 233

M

\MessageBreak 114

N

\newif 125, 135, 136, 137, 138, 139
\newlinechar 109
\next 212, 214, 216

O

\OpMode 141

P

\PackageError 112
\PackageInfo 23, 166
\ProvidesPackage 15, 61

R

\RangeCatcodeInvalid 231, 243, 244, 245, 246
\repeat 206, 218, 229, 237
\RestoreCatcodes . . 220, 223, 224, 258

S

\space 176, 177

T

\Test 242, 260

\the 68, 69, 70, 71, 82, 225
\TMP@EnsureCode 79, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97

V

\vtexdvitrue 142
\vtexgextrue 156
\vtexhtmltrue 150
\vtexpdftrue 144
\vtexpstrue 146, 148
\vtextrue 132

W

\write 20, 46, 163

X

\x . 10, 11, 14, 19, 23, 25, 45, 50, 60, 66, 74, 103, 111, 116, 162, 166, 169

Y

\y 108, 114, 117

Z

\z 104, 105