The \texttt{ltxcmds} package

Heiko Oberdiek*

2023-12-04 v1.26

Abstract

The package \texttt{ltxcmds} exports some utility macros from the L\LaTeX\ kernel into a separate namespace and also provides them for other formats such as plain-L\LaTeX.  

Contents

\section{Documentation}
\begin{enumerate}
\setcounter{enumi}{0}
\item \texttt{Introduction} \hfill 3 \item \texttt{Numbers} \hfill 3 \item \texttt{Scratch registers} \hfill 3 \item \texttt{Argument killers} \hfill 3 \item \texttt{Argument grabbers} \hfill 4 \item \texttt{List helpers} \hfill 4 \item \texttt{Tail recursion} \hfill 5 \item \texttt{Empty macro} \hfill 5 \item \texttt{Characters} \hfill 5 \item \texttt{Boolean switch} \hfill 5 \item \texttt{Command definitions} \hfill 5 \item \texttt{Stripping} \hfill 6 \item \texttt{File management} \hfill 6 \item \texttt{File extensions} \hfill 6 \item \texttt{Load check} \hfill 6 \item \texttt{Version date check} \hfill 7 \item \texttt{Macro additions} \hfill 7 \item \texttt{Next character detection} \hfill 7 \item \texttt{Expandable test for emptiness} \hfill 8 \item \texttt{Stripping spaces} \hfill 8 \item \texttt{Check for emptiness of boxes} \hfill 9 \end{enumerate}

\section{Implementation}
\begin{enumerate}
\setcounter{enumi}{0}
\item \texttt{Identification} \hfill 9 \item \texttt{Numbers} \hfill 11 \item \texttt{Scratch registers} \hfill 11 \item \texttt{Argument killers} \hfill 13 \item \texttt{Argument grabbers} \hfill 13 \item \texttt{List helpers} \hfill 14 \item \texttt{Tail recursion} \hfill 16 \item \texttt{Empty macro} \hfill 16 \item \texttt{Characters} \hfill 16 \item \texttt{Boolean switch} \hfill 16 \item \texttt{Command definitions} \hfill 17 \end{enumerate}

\footnote{Please report any issues at \url{https://github.com/ho-tex/ltxcmds/issues}}
2.12 Stripping .................................................. 18
2.13 File management ........................................... 18
  2.13.1 File extensions ...................................... 18
  2.13.2 Load check .......................................... 19
  2.13.3 Version date check ................................. 19
2.14 Macro additions ......................................... 19
2.15 Next character detection .............................. 20
2.16 $\langle$\texttt{\textbackslash lmbox}$\rangle$ .......................... 21
2.17 Help macros .............................................. 22
2.18 Expandable test for emptiness ......................... 22
  2.18.1 Vanilla \LaTeX ........................................ 22
  2.18.2 With \texttt{\textbackslash detokenize} .................. 23
  2.18.3 \texttt{\textbackslash ifblank} ......................... 24
2.19 \texttt{\textbackslash zap\textbackslash space} ..................... 24
2.20 \texttt{\textbackslash IfBoxEmpty} ....................... 24

3 Installation .................................................. 25
  3.1 Download ................................................. 25
  3.2 Package installation .................................... 25
  3.3 Refresh file name databases ......................... 26
  3.4 Some details for the interested .................... 26

4 References .................................................. 26

5 History ....................................................... 27
  [2009/08/05 v1.0] ........................................... 27
  [2009/12/12 v1.1] .......................................... 27
  [2010/01/28 v1.2] .......................................... 27
  [2010/03/01 v1.3] .......................................... 27
  [2010/03/09 v1.4] .......................................... 27
  [2010/04/08 v1.5] .......................................... 28
  [2010/04/16 v1.6] .......................................... 28
  [2010/04/26 v1.7] .......................................... 28
  [2010/09/11 v1.8] .......................................... 28
  [2010/10/25 v1.9] .......................................... 28
  [2010/10/31 v1.10] ........................................ 28
  [2010/11/12 v1.11] ......................................... 28
  [2010/12/02 v1.12] ......................................... 28
  [2010/12/04 v1.13] ......................................... 28
  [2010/12/07 v1.14] ......................................... 28
  [2010/12/12 v1.15] ......................................... 28
  [2011/02/04 v1.16] ......................................... 28
  [2011/02/05 v1.17] ......................................... 29
  [2011/03/16 v1.18] ......................................... 29
  [2011/04/14 v1.19] ......................................... 29
  [2011/04/18 v1.20] ......................................... 29
  [2011/08/22 v1.21] ......................................... 29
  [2011/11/09 v1.22] ......................................... 29
  [2016/05/16 v1.23] ......................................... 29
  [2019/12/15 v1.24] ......................................... 29
  [2020-05-10 v1.25] ....................................... 29
  [2023-12-04 v1.26] ....................................... 29

6 Index .......................................................... 29
1 Documentation

1.1 Introduction

Many of my packages also support other formats such as plain-\TeX. Because I am rather familiar with the utility macros from \LaTeX’s kernel (e.g. \texttt{\@gobble}, \texttt{\@firstoftwo}), I found myself rewriting them again and again, because they are lacking in plain-\TeX.

Therefore this package provides often used macros and similar ones with the name prefix \texttt{\ltx@}. This avoids also faulty redefinitions. I remember an example where a package redefined \texttt{\@firstoftwo} with forgetting \texttt{\long}.

1.2 Numbers

\begin{center}
\begin{tabular}{ll}
\texttt{\ltx@zero} & \rightarrow 0 \\
\texttt{\ltx@one} & \rightarrow 1 \\
\texttt{\ltx@two} & \rightarrow 2 \\
\texttt{\ltx@cclv} & \rightarrow 255 \\
\texttt{\ltx@minusone} & \rightarrow -1
\end{tabular}
\end{center}

These commands are numbers 0, 1, 2, 255 and -1. They are not digits and a space is not gobbled afterwards. Macro \texttt{\ltx@minusone} is available since version 2010/12/12 v1.15.

1.3 Scratch registers

Following the conventions of plain \TeX and \LaTeX the first ten registers are free to use. Even numbered registers are for local, odd numbered for global use.

\begin{center}
\texttt{\ltx@(Loc,Glob)(Toks,Dimen,Skip)(A,B,C,D,E)}
\end{center}

The name consists of the prefix \texttt{\ltx@}, then Loc or Glob for local or global usage follows. The register type is given by Toks for token register, Dimen for dimen register and Skip for skip register. As last part the registers are numbered from A to E. Example: \texttt{\ltx@LocToksA}.

Since 2011/04/14 v1.19.

1.4 Argument killers

\begin{center}
\begin{tabular}{ll}
\texttt{\ltx@gobble \{\{I\}\}} & \rightarrow \\
\texttt{\ltx@gobbletwo \{\{I\}\} \{\{2\}\}} & \rightarrow \\
\texttt{\ltx@gobblethree \{\{I\}\} \{\{2\}\} \{\{3\}\}} & \rightarrow \\
\texttt{\ltx@gobblefour \{\{I\}\} \{\{2\}\} \{\{3\}\} \{\{4\}\}} & \rightarrow
\end{tabular}
\end{center}

\begin{center}
\texttt{\ltx@GobbleNum \{(num)\} \{\{I\}\} \{\{2\}\} \ldots \{\{(num)\}\}} \rightarrow
\end{center}

The first argument \texttt{(num)} of macro \texttt{\ltx@GobbleNum} specifies, how many following arguments are eaten. Macro \texttt{\ltx@GobbleNum} is expandable in exact two expansion steps.
1.5 Argument grabbers

Macros \texttt{\textbackslash firstofone}, \texttt{\textbackslash firstoftwo} and \texttt{\textbackslash firstofthree} were added in version 2010/11/12 v1.11. Macros \texttt{\textbackslash firstoffour}, \ldots, \texttt{\textbackslash fourthoffour} were added in version 2011/02/04 v1.16.

1.6 List helpers

Macros with uppercase letters are expandable in two expansion steps. Changes in version 2023-12-04 v1.26:
• Macros \ltex@carsecond, \ltex@carthird, \ltex@carfourth, \ltex@CarNUMth added.

• Macros \ltex@cdr, \ltex@cdrtwo, \ltex@cdrthree, \ltex@cdrfour, \ltex@CdrNUM are expandable in two expansion steps and retain spaces and braces after the first gobbled arguments.

1.7 Tail recursion

\ltex@ReturnAfterFi \{1\} \fi → \fi \{1\}
\ltex@ReturnAfterElseFi \{1\} \else \{2\} \fi → \fi \{1\}

1.8 Empty macro

\ltex@empty →

1.9 Characters

\ltex@space → \ 
\ltex@percentchar → %
\ltex@backslashchar → \ 
\ltex@hashchar → # (since v1.7)
\ltex@leftbracechar → \{ (since v1.8)
\ltex@rightbracechar → \} (since v1.8)

1.10 Boolean switch

\ltex@newif \{cmd\}
\ltex@newif defines a new boolean switch \langle cmd \rangle like \newif. Unlike plain \TeX’s \newif, \ltex@newif is not \outer. The command \langle cmd \rangle must start with the two characters if.

\ltex@newglobalif \{cmd\}
\ltex@newglobalif defines a new boolean \langle cmd \rangle like \ltex@newif. However the switch setting commands, \langle cmd \rangle without the prefix if and followed by true or false are acting globally.

1.11 Command definitions

\ltex@ifundefined \{cmd\} \{yes\} \{no\}
If \e-\TeX is available, \ifcsname is used that does not have the side effect of defining undefined commands with meaning of \relax. This command is always expandable. Change in version 1.1: Also the meaning \relax is always considered “undefined”.

5
\ltx@ifundefined \{⟨cmd⟩\} \{⟨yes⟩\} \{⟨no⟩\}

If \TeX{} is available, \texttt{\ifcsname} is used that does not have the side effect of defining undefined commands with meaning of \texttt{\relax}. Also it always checks for the meaning of \texttt{\relax} and considers this as undefined. This macro is not expandable without \TeX{}.

\ltx@localexpandafter

It expands the token after the next token but in a local context. That is the difference to \texttt{\expandafter}. The local context discards the side effect of \texttt{\csname} and let the command undefined after the expansion step.

1.12 Stripping

\ltx@removeprefix \ltx@stripprefix

All tokens up to and including the next available character ‘>’ are thrown away. Usually it is used to strip the first part of the output of the commands \texttt{\meaning} or \texttt{\pdflastmatch}. Macro \texttt{\ltx@removeprefix} has the same meaning as \LaTeX{}’s \texttt{\strip@prefix}, whereas macro \texttt{\ltx@stripprefix} expands the next token once before stripping the prefix.

\ltx@onelvel@sanitize \{⟨macro⟩\}

Macro \texttt{\ltx@onelvel@sanitize} provides \LaTeX{}’s \texttt{\@onelvel@sanitize}. The macro is expanded once and the contents is converted to characters with catcode 12 (other) and space tokens with catcode 10 (space). Then then sanitized contents is stored into the macro again. Since version 1.12.

1.13 File management

All macros in this section are expandable like the counterparts of the \LaTeX{} kernel. Also they can be used after the preamble.

1.13.1 File extensions

\ltx@clsextension \ltx@pkgextension

Macros \texttt{\ltx@clsextension} and \texttt{\ltx@pkgextension} stores the strings \texttt{cls} and \texttt{sty}. In opposite to \LaTeX{}’s \texttt{\@clsextension} and \texttt{\@styextension} they can also be used after \texttt{\begin{document}}.

1.13.2 Load check

\ltx@ifclassloaded \{⟨class⟩\} \{⟨yes⟩\} \{⟨no⟩\} \ltx@ifpackageloaded \{⟨package⟩\} \{⟨yes⟩\} \{⟨no⟩\}

Macros \texttt{\ltx@ifclassloaded} and \texttt{\ltx@ifpackageloaded} execute \texttt{⟨yes⟩}, if the \texttt{⟨class⟩} or \texttt{⟨package⟩} is loaded, otherwise \texttt{⟨no⟩} is called. Both \texttt{⟨class⟩} and \texttt{⟨package⟩} are specified without extension. The macros can also be used after \texttt{\begin{document}}.
\ltx@iffileloaded {(file)} {(yes)} {(no)}

If \LaTeX{}'s \texttt{ProvidesFile} macro was called before using \texttt{(file)} as argument, then \ltx@iffileloaded calls \texttt{(yes)}, otherwise \texttt{(no)}. Therefore it is possible that the \texttt{(file)} is loaded, but \texttt{(no)} is executed because of a missing \texttt{ProvidesFile}. The \LaTeX{} kernel does not have a counterpart of \ltx@iffileloaded.

Note that the file name used in \texttt{ProvidesFile} and \ltx@iffileloaded must match. For example, if \TeX{}'s default extension \texttt{.tex} was given in the first command, then it must also specified in the latter command and vice versa.

1.13.3 Version date check

\ltx@ifclasslater {(class)} {(date)} {(yes)} {(no)}
\ltx@ifpackagelater {(package)} {(date)} {(yes)} {(no)}
\ltx@iffilelater {(file)} {(date)} {(yes)} {(no)}

If a \texttt{ProvidesClass}/\texttt{ProvidesPackage}/\texttt{ProvidesFile} command with exact the same class/package/file was executed before with an optional argument that starts with a \LaTeX{} version date, then this version date is compared with the argument \texttt{(date)}. If they are equal or if the version date is the later date, then \texttt{(yes)} is called. In all other cases \texttt{(no)} is executed.

A \LaTeX{} date has the format YY/MM/DD with YYYY as year with four digits, MM as month with two digits and DD as day with two digits. If pdf\TeX{}'s \texttt{pdfmatch} is available, then it is used to detect the version date, to reject invalid date formats and to reject some invalid dates. Dates before 1994/01/01 are always invalid, because version dates are introduced with \LaTeX{} 2ε in 1994.

1.14 Macro additions

\ltx@GlobalAppendToMacro {(cmd)} {(addition)}
\ltx@LocalAppendToMacro {(cmd)} {(addition)}

The \texttt{(addition)} is appended to the parameterless macro \texttt{(cmd)}. If \texttt{(cmd)} is undefined or has the meaning \texttt{\relax}, then it will be initialized as empty macro beforehand. Due to a bug \texttt{(addition)} must not contain \texttt{\par} before version 2010/10/25 v1.9.

\ltx@GlobalPrependToMacro {(cmd)} {(addition)}
\ltx@LocalPrependToMacro {(cmd)} {(addition)}

The \texttt{(addition)} is prepended to the parameterless macro \texttt{(cmd)}. If \texttt{(cmd)} is undefined or has the meaning \texttt{\relax}, then it will be initialized as empty macro beforehand. The macros were added in version 2011/08/22 v1.21.

1.15 Next character detection

\ltx@ifnextchar {(char)} {(yes)} {(no)}

If next character is \texttt{(char)} then \texttt{(yes)} is called, otherwise \texttt{(no)}. The character is not removed. Spaces are silently removed when looking for \texttt{(char)} as \LaTeX{}'s version \texttt{\kernel@ifnextchar} does. But there are also small differences:

- The space can be used as \texttt{(char)}. In this case optional spaces before \texttt{(char)} are not supported of course.
• If the optional space is a command that is a character (defined by \let or \futurelet), then \kernel@ifnextchar breaks with a TeX error. \ltx@ifnextchar silently removes this token as optional space.

Since 2010/03/01 v1.3.

\ltx@ifnextchar@nospace \{⟨char⟩\} \{⟨yes⟩\} \{⟨no⟩\}

Macro \ltx@ifnextchar@nospace behaves like macro \ltx@ifnextchar with the exception that optional spaces are not supported before ⟨char⟩. Since 2011/04/14 v1.19.

1.16 \ltx@leavevmode, \ltx@mbox

\ltx@leavevmode

Macro \ltx@leavevmode calls pdfTeX’s \quitvmode. Otherwise \leavevmode is used and defined if it is necessary.

\ltx@mbox

Macro \ltx@mbox reimplements \mbox with two changes. Instead of \leavevmode it uses \ltx@leavevmode and stops right after \hbox. Especially it does not grab the argument and allows the extended syntax of \hbox.

1.17 Expandable test for emptiness

\ltx@ifempty \{⟨stuff⟩\} \{⟨yes⟩\} \{⟨no⟩\}

Macro \ltx@ifempty checks in exact two expansion steps whether ⟨stuff⟩ is empty or contains token. Depending on the result ⟨yes⟩ or ⟨no⟩ is executed. The token in ⟨stuff⟩ may contain \par and unmatched conditionals (\if, \else, \fi, ...). Since version 2010/11/12 v1.11.

\ltx@ifblank \{⟨stuff⟩\} \{⟨yes⟩\} \{⟨no⟩\}

Macro \ltx@ifblank tests in exact two expansion steps if ⟨stuff⟩ is empty or contain only blank spaces. In this case argument ⟨yes⟩ is called. If ⟨stuff⟩ contains other tokens than spaces then ⟨no⟩ is executed. Since version 2010/12/04 v1.13.

1.18 Stripping spaces

\ltx@zapspace \{⟨stuff⟩\}

Macro \ltx@zapspace strips spaces from ⟨stuff⟩ that are not hidden inside curly braces. Like LATEX’s \zap@space it is expandable. Differences:

• Syntax: \zap@space also expects a space token and \empty after ⟨stuff⟩.
• Macro \ltx@zapspace is expandable in exact two expansion steps.
• Macro \ltx@zapspace always retains curly braces.
• Macro \zap@space has a bug. It stops stripping spaces after a token group in curly braces if the first two tokens inside the group are equal.
• Macro \ltx@zapspace also works with \par and conditionals (\if, \else, \fi, ...).

Macro \ltx@zapspace is available since version 2010/12/07 v1.14.

1.19 Check for emptiness of boxes

\ltx@ifBoxEmpty{(\texttt{box register number})} {\langle \texttt{yes} \rangle} {\langle \texttt{no} \rangle}

Macro \ltx@ifBoxEmpty calls \langle \texttt{yes} \rangle if the box exists (\ifvoid returns false) and the box does not contain any content. Otherwise if the box is void or contains something, then \langle \texttt{no} \rangle is executed. Thus being empty means that the box exists and is either an \texttt{hbox} or a \texttt{vbox} and may even have dimensions other than 0.0 pt, but the box does not contain anything. Macro \ltx@ifBoxEmpty is available since 2010/02/04 v1.16.

\ltx@ifBoxVoidOrEmpty{(\texttt{box register number})} {\langle \texttt{yes} \rangle} {\langle \texttt{no} \rangle}

Macro \ltx@ifBoxVoidOrEmpty calls \langle \texttt{yes} \rangle if the box is either void or does not contain any content. Otherwise \langle \texttt{no} \rangle is executed. Macro \ltx@ifBoxVoidOrEmpty is available since 2010/02/04 v1.16.

2 Implementation

2.1 Identification

\begin{verbatim}
\texttt{\langle \ast \texttt{package} \rangle}
\end{verbatim}

Reload check, especially if the package is not used with \TeX.  

\begin{verbatim}
\texttt{\langle \begin{verbatim}
\begin{verbatim}
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode39=12 % '
\catcode44=12 % ,
\catcode45=12 % -
\catcode46=12 % .
\catcode58=12 % :
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\expandafter\let\expandafter\x\csname ver@ltxcmds.sty\endcsname
\ifx\x\relax % plain-\TeX, first loading,
\else
\def\empty{}
\ifx\x\empty % LaTeX, first loading, % variable is initialized, but \ProvidesPackage not yet seen
\else
\expandafter\if\csname PackageInfo\endcsname\relax
\def\x#1#2{%
\immediate\write-1{Package \#1 Info: #2.}%
}\else
\PackageInfo{\x}{#2, stopped}%
\fi
\expandafter\let\expandafter\x\csname ver@ltxcmds.sty\endcsname\endcsname
\fi
\if\x\texttt{\langle \texttt{relax} \rangle, first loading,}
\else
\def\empty{\texttt{\langle \texttt{empty} \rangle}}
\if\x\texttt{\langle \texttt{empty} \rangle \ LaTeX, first loading,}
\else
\PackageInfo{\x}{\x\#1\#2\texttt{\langle \texttt{PackageInfo\##1\#2, stopped} \rangle}}
\fi
\fi
\end{verbatim}
\end{verbatim}
\end{verbatim}
\endinput
\endgroup%
\end{verbatim}
\endinput
\endgroup%
\end{verbatim}
\endinput
\endgroup%
\end{verbatim}
\endinput
\endgroup%
\end{verbatim}
\endinput
\endgroup%
Package identification:
\begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode39=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
  \fi
\expandafter\x\csname ver@ltxcmds.sty\endcsname
\ProvidesPackage{ltxcmds}[
  [2023-12-04 v1.26 LaTeX kernel commands for general use (HO)]%
\begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode39=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
  \fi
\expandafter\x\csname ver@ltxcmds.sty\endcsname
\ProvidesPackage{ltxcmds}%
[2023-12-04 v1.26 LaTeX kernel commands for general use (HO)]%
\begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode39=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
  \fi
\expandafter\x\csname ver@ltxcmds.sty\endcsname
\ProvidesPackage{ltxcmds}%
[2023-12-04 v1.26 LaTeX kernel commands for general use (HO)]%
2.2 Numbers

\ltx@zero
\chardef\ltx@zero=0 %
\ltx@one
\chardef\ltx@one=1 %
\ltx@two
\chardef\ltx@two=2 %
\ltx@active
\chardef\ltx@active=13 %
\ltx@cclv
\chardef\ltx@cclv=255 %
\ltx@minusone
\def\ltx@minusone{\ltx@one}

2.3 Scratch registers

\ltx@LocToksA
\toksdef\ltx@LocToksA=0 %
\ltx@LocToksB
\toksdef\ltx@LocToksB=2 %
\ltx@LocToksC
\toksdef\ltx@LocToksC=4 %
\ltx@LocToksD
\toksdef\ltx@LocToksD=6 %
\ltx@LocToksE
\toksdef\ltx@LocToksE=8 %
\ltx@GlobToksA  
129 \toksdef\ltx@GlobToksA=1 %
\ltx@GlobToksB  
130 \toksdef\ltx@GlobToksB=3 %
\ltx@GlobToksC  
131 \toksdef\ltx@GlobToksC=5 %
\ltx@GlobToksD  
132 \toksdef\ltx@GlobToksD=7 %
\ltx@GlobToksE  
133 \toksdef\ltx@GlobToksE=9 %
\ltx@LocDimenA  
134 \dimendef\ltx@LocDimenA=0 %
\ltx@LocDimenB  
135 \dimendef\ltx@LocDimenB=2 %
\ltx@LocDimenC  
136 \dimendef\ltx@LocDimenC=4 %
\ltx@LocDimenD  
137 \dimendef\ltx@LocDimenD=6 %
\ltx@LocDimenE  
138 \dimendef\ltx@LocDimenE=8 %
\ltx@GlobDimenA  
139 \dimendef\ltx@GlobDimenA=1 %
\ltx@GlobDimenB  
140 \dimendef\ltx@GlobDimenB=3 %
\ltx@GlobDimenC  
141 \dimendef\ltx@GlobDimenC=5 %
\ltx@GlobDimenD  
142 \dimendef\ltx@GlobDimenD=7 %
\ltx@GlobDimenE  
143 \dimendef\ltx@GlobDimenE=9 %
\ltx@LocSkipA  
144 \skipdef\ltx@LocSkipA=0 %
\ltx@LocSkipB  
145 \skipdef\ltx@LocSkipB=2 %
\ltx@LocSkipC  
146 \skipdef\ltx@LocSkipC=4 %
\ltx@LocSkipD  
147 \skipdef\ltx@LocSkipD=6 %
\ltx@LocSkipE  
148 \skipdef\ltx@LocSkipE=8 %
2.4 Argument killers

\ltx@gobble
\long\def\ltx@gobble#1{}

\ltx@gobbletwo
\long\def\ltx@gobbletwo#1#2{}

\ltx@gobblethree
\long\def\ltx@gobblethree#1#2#3{}

\ltx@gobblefour
\long\def\ltx@gobblefour#1#2#3#4{}

\ltx@GobbleNum
\def\ltx@GobbleNum#1{%
  \romannumeral
  \csname ltx@zero%
  \expandafter\LTXcmds@GobbleNum
  \romannumeral\LTXcmds@num{#1}000{m\endcsname}%
}

\LTXcmds@GobbleNum
\def\LTXcmds@GobbleNum#1{%
  \csname LTXcmds@G#1\LTXcmds@GobbleNum
}

\LTXcmds@Gm
\long\def\LTXcmds@Gm#1{%
  \endcsname
}

2.5 Argument grabbers

\ltx@firstofone
\long\def\ltx@firstofone#1{(#1}

\ltx@firstoftwo
\long\def\ltx@firstoftwo#1#2{(#1}

\ltx@secondoftwo
\long\def\ltx@secondoftwo#1#2{(#2}

\ltx@firstofthree
\long\def\ltx@firstofthree#1#2#3{(#1}
2.6 List helpers

\ltx@secondofthree
\long\def\ltx@secondofthree#1#2#3{#2}

\ltx@thirdofthree
\long\def\ltx@thirdofthree#1#2#3{#3}%

\ltx@firstoffour
\long\def\ltx@firstoffour#1#2#3#4{#1}

\ltx@secondoffour
\long\def\ltx@secondoffour#1#2#3#4{#2}

\ltx@thirdoffour
\long\def\ltx@thirdoffour#1#2#3#4{#3}%

\ltx@fourthoffour
\long\def\ltx@fourthoffour#1#2#3#4{#4}%

\ltx@carzero
\long\def\ltx@carzero#1\@nil{}%

\ltx@cdrzero
\def\ltx@cdrzero{\romannumeral\LTXcmds@cdrzero\ltx@zero}

\ltx@car
\long\def\ltx@car#1#2\@nil{#1}

\ltx@cdr
\long\def\ltx@cdr#1{\romannumeral\LTXcmds@cdrzero\ltx@zero}

\ltx@cartwo
\long\def\ltx@cartwo#1#2#3\@nil{#1#2}

\ltx@carsecond
\long\def\ltx@carsecond#1#2#3\@nil{#2}

\ltx@cdrtwo
\long\def\ltx@cdrtwo#1#2{\romannumeral\LTXcmds@cdrzero\ltx@zero}

\ltx@carthree
\long\def\ltx@carthree#1#2#3#4\@nil{#1#2#3}

\ltx@carthird
\long\def\ltx@carthird#1#2#3#4\@nil{#3}

\ltx@cdrthree
\long\def\ltx@cdrthree#1#2#3#4\@nil{#4}

\ltx@cdrthird
\long\def\ltx@cdrthird#1#2#3#4\@nil{#4}
2.7 Tail recursion

\def\ltx@ReturnAfterFi#1\fi{\fi#1}

\def\ltx@ReturnAfterElseFi#1\else#2\fi{\fi#1}

2.8 Empty macro

\def\ltx@empty{}

2.9 Characters

\def\ltx@space{ }

\def\ltx@percentchar{
  \begingroup
  \lccode'0='\%
  \lowercase{\endgroup}
  \def\ltx@percentchar{0}\%
}

\def\ltx@backslashchar{
  \begingroup
  \lccode'0='\\
  \lowercase{\endgroup}
  \def\ltx@backslashchar{0}\%
}

\def\ltx@hashchar{
  \begingroup
  \lccode'0='\#
  \lowercase{\endgroup}
  \def\ltx@hashchar{0}\%
}

\def\ltx@leftbracechar{
  \begingroup
  \lccode'0='\{
  \lowercase{\endgroup}
  \def\ltx@leftbracechar{0}\%
}

\def\ltx@rightbracechar{
  \begingroup
  \lccode'0='\}
  \lowercase{\endgroup}
  \def\ltx@rightbracechar{0}\%
}

2.10 Boolean switch

\def\ltx@newif#1{\begingroup
  \escapechar=-1
  \expandafter\endgroup
  \expandafter\LTXcmds@newif\string#1\@nil
}

16
2.11 Command definitions

\ltx@ifundefined
\def\ltx@ifundefined#1{\expandafter\ifx\csname #1\endcsname\relax...
\ltxIfUndefined
\def\ltxIfUndefined#1{%
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname #1\endcsname\relax
\expandafter\firstoftwo
\else
\expandafter\secondoftwo
\fi
}\%
\expandafter\gobble
\else
\expandafter\firstofone
\fi
{%}
\ltxIfUndefined
\ltxIfUndefined\ltxIfUndefined\ltxIfUndefined
}

2.12 Stripping
\ltxRemovePrefix
\def\ltxRemovePrefix#1>{}
\ltxStripPrefix
\def\ltxStripPrefix{%
\expandafter\ltxRemovePrefix\meaning#1%
\}
\ltxOnelevelSanitize
\def\ltxOnelevelSanitize#1{%
\edef#1{%
\expandafter\ltxRemovePrefix\meaning#1%
\}
\}

2.13 File management
2.13.1 File extensions
\ltxCloseExtension
\def\ltxCloseExtension{cls}
2.13.2 Load check

\texttt{\ltx@iffileloaded}\ltx@iffileloaded\#1{%\ltx@ifundefined{ver@#1}\ltx@secondoftwo\ltx@firstoftwo\}}

\texttt{\ltx@ifclassloaded}\ltx@ifclassloaded\#1{%\ltx@iffileloaded{#1.\ltx@clsextension}{\}}

\texttt{\ltx@ifpackageloaded}\ltx@ifpackageloaded\#1{%\ltx@iffileloaded{#1.\ltx@pkgextension}{\}}

2.13.3 Version date check

changed 2020-05-10 to adapt to dates with dashes (ISO) The core of the commands are copies from the latex commands.

\texttt{\ltx@ifl@ter}\ltx@ifl@ter\#1\#2{%\ltx@ifl@t@r\csname ver@#2.#1\endcsname}

\texttt{\ltx@ifl@ter}\ltx@ifl@ter\#1\#2{%\ltx@ifl@t@r\csname ver@#2.#1\endcsname}

\texttt{\ltx@ifl@ter}\ltx@ifl@ter\#1\#2{%\ltx@ifl@t@r\csname ver@#2.#1\endcsname}

2.14 Macro additions

\texttt{\ltx@GlobalAppendToMacro}\long\ltx@GlobalAppendToMacro\#1\#2{%\ltx@undefined\ltx@empty\else\relax\relax\#1\fi\#2\#3\#4\#5\}
2.15 Next character detection
\ltx@ifnextchar\#1\#2\#3{\%
\begingroup
\let\LTXcmds@CharToken= \relax
\ltx@LocToksA{\endgroup\#2}\%
\ltx@LocToksB{\endgroup\#3}\%
\futurelet\LTXcmds@LetToken\LTXcmds@ifnextchar\}

\LTXcmds@ifnextchar
\def\LTXcmds@ifnextchar{\ifx\LTXcmds@LetToken\LTXcmds@CharToken
\the\expandafter\ltx@LocToksA
\else
\expandafter\ifx\csname LTXcmds@LetToken\endcsname\LTXcmds@SpaceToken
\expandafter\expandafter\expandafter\LTXcmds@@ifnextchar
\else
\the\expandafter\expandafter\expandafter\ltx@LocToksB
\fi
\fi
}

\LTXcmds@@ifnextchar
\futurelet\LTXcmds@LetToken\LTXcmds@ifnextchar
\def\LTXcmds@@ifnextchar{\expandafter\futurelet\expandafter\LTXcmds@LetToken\expandafter\LTXcmds@ifnextchar
\romannumeral-\%.}

\LTXcmds@SpaceToken
\ltx@firstofone{\let\LTXcmds@SpaceToken= } %

\ltx@ifnextchar@nospace
\long\def\ltx@ifnextchar@nospace\#1\#2\#3{\%
\begingroup
\let\LTXcmds@CharToken= \relax
\ltx@LocToksA{\endgroup\#2}\%
\ltx@LocToksB{\endgroup\#3}\%
\futurelet\LTXcmds@LetToken\LTXcmds@ifnextchar@nospace\}

\LTXcmds@ifnextchar@nospace
\def\LTXcmds@ifnextchar@nospace{\ifx\LTXcmds@LetToken\LTXcmds@CharToken
\expandafter\ltx@LocToksA
\else
\ltx@LocToksB
\fi
}

\LTXcmds@leavevmode, \LTXcmds@mbox
\long\def\ltx@leavevmode{\ifx\LTXcmds@LetToken\LTXcmds@CharToken
\expandafter\ltx@LocToksA
\else
\expandafter\ltx@LocToksB
\fi
}

2.16 \LTXcmds@leavevmode, \LTXcmds@mbox

\long\def\ltx@IfUndefined{quitmode}{\%
\ltx@IfUndefined{leavingvmode}{\%
\ltx@IfUndefined{voidb@x}{\%

21
2.17 Help macros

2.18 Expandable test for emptiness

2.18.1 Vanilla \TeX

\texttt{\the\numexpr\number1\relax} The macro is based on \texttt{\ifempty} of Robert R. Schneck \cite{1} and \texttt{\ifnull} of Ulrich Diez \cite{2}. There are three cases to consider:

1. \#1 is empty,
2. #1 is not empty and the first token is not a begingroup character,
3. #1 starts with a begingroup character (catcode 1).

2.18.2 With \detokenize

Ahmed Musa provided `\ifstrempty` using `\detokenize` and `\pdfstrcmp` [3]. Ulrich Diez, GL, Heiko Oberdiek improved it further by removing `\pdfstrcmp` and taking three arguments [4, 5, 6, 7, 8].
In case of $\varepsilon$-TEX the test for an empty box is done via `\lastnodetype` as suggested by David Kastrup [9].

```latex
\ltx@IfBoxEmpty
\def\ltx@IfBoxEmpty#1{\ifvoid#1\relax\else
& \begingroup
& \setbox\ltx@zero=\ifhbox#1\hbox\else\vbox\fi{\ifhmode\unhcopy\else\unvcopy\fi#1\relax
& \expandafter
& }\expandafter\endgroup\ifnum\lastnodetype<\ltx@zero
& \expandafter\expandafter\expandafter\ltx@firstoftwo
& \else
& \expandafter\expandafter\expandafter\ltx@secondoftwo
& \fi
```

Implementation using $\varepsilon$-TEX’s `\lastnodetype`.
Implementation without $\varepsilon$-TEX using a signature at the beginning of the test box.

\begin{verbatim}
\begingroup
\setbox\ltx@zero=\ifhbox#1\hbox\else\vbox\fi{% 
\penalty\ltx@one
\ifhmode\unhcopy\else\unvcopy\fi#1\relax 
\expandafter}
\ifnum\lastpenalty=\ltx@one
Box 0 has been changed and is restored by closing the group.
\endgroup
\begingroup
\setbox\ltx@zero=\ifhbox#1\hbox\else\vbox\fi{% 
\penalty\ltx@two
\ifhmode\unhcopy\else\unvcopy\fi#1\relax 
\expandafter}
\ifnum\lastpenalty=\ltx@two
\def\next{\endgroup\expandafter\ltx@firstoftwo}\
\else
\def\next{\endgroup\expandafter\ltx@secondoftwo}\
\fi
\else
\def\next{\endgroup\expandafter\ltx@secondoftwo}\
\fi

\ltx@IfBoxVoidOrEmpty
\def\ltx@IfBoxVoidOrEmpty#1{% 
\ifvoid#1\relax 
\expandafter\ltx@thirdoffour
\fi
\ltx@IfBoxEmpty{#1}%
\}
\endgroup
\def\ltx@IfBoxVoidOrEmpty#1{% 
\ifvoid#1\relax 
\expandafter\ltx@thirdoffour
\fi
\ltx@IfBoxEmpty{#1}%
\}
\LTXcmds@AtEnd%
\endgroup
\end{verbatim}

\ltx@IfBoxVoidOrEmpty
\def\ltx@IfBoxVoidOrEmpty#1{% 
\ifvoid#1\relax 
\expandafter\ltx@thirdoffour
\fi
\ltx@IfBoxEmpty{#1}%
\}
\LTXcmds@AtEnd%

3 Installation

3.1 Download

Package. This package is available on CTAN\textsuperscript{1}:

\begin{itemize}
\item \url{CTAN:macros/latex/contrib/ltxcmds/ltxcmds.dtx} The source file.
\item \url{CTAN:macros/latex/contrib/ltxcmds/ltxcmds.pdf} Documentation.
\end{itemize}

3.2 Package installation

The package is at best installed with the package manager of the \TeX{} system. Manual installation is possible too:

Unpacking. The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain \TeX{}:

```
tex ltxcmds.dtx
```

\textsuperscript{1}CTAN:pkg/ltxcmds
TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
ltxcmds.sty → tex/generic/ltxcmds/ltxcmds.sty
ltxcmds.pdf → doc/latex/ltxcmds/ltxcmds.pdf
ltxcmds.dtx → source/latex/ltxcmds/ltxcmds.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`.

### 3.3 Refresh file name databases

If your TeX distribution (TeX Live, MiKTeX, ...) relies on file name databases, you must refresh these. For example, TeX Live users run `texhash` or `mktexlsr`.

### 3.4 Some details for the interested

**Unpacking with \TeX{}**. 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{ltxcmds.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 ltxcmds.dtx
makeindex -s gind.ist ltxcmds.idx
pdflatex ltxcmds.dtx
makeindex -s gind.ist ltxcmds.idx
pdflatex ltxcmds.dtx
```

### 4 References

[1] Robert R. Schneck: *Re: \texttt{\textbackslash ifempty} solution (was Macro puzzle: maximally general \texttt{\textbackslash ifempty})*; newsgroup `comp.text.tex`, news:SeeFlada_6@corp.newsgroups.com, 2003-06-17.  
https://groups.google.com/group/comp.text.tex/msg/be03a159ec374895

[2] Ulrich Diez: *Re: TeX refuses to strip outer braces in argument*; newsgroup  
`comp.text.tex`, news:ihk3t8$ee7$i@news.albasani.net, 2010-11-12.  
https://groups.google.com/group/comp.text.tex/msg/803bd57221a04996

`comp.text.tex`, news:f5496afe-40ed-42bd-b629-a2419ecf7c0d0  
o14g2000prn.googlegroups.com, 2010-12-03.  
https://groups.google.com/group/comp.text.tex/msg/fbf7d61a0c3a807d
5 History

[2009/08/05 v1.0]
- First version.

[2009/12/12 v1.1]
- Short title shortened.
  - \ltx@ifUndefined added.

[2010/01/28 v1.2]
- \ltx@RemovePrefix and \ltx@StripPrefix added.
- \ltx@ifclassloaded, \ltx@ifpackageloaded, \ltx@iffileloaded, \ltx@ifclasslater, \ltx@ifpackagelater, \ltx@iffilelater, \ltx@clsextension, \ltx@pkgextension added.
- \ltx@GlobalAppendToMacro, \ltx@LocalAppendToMacro added.

[2010/03/01 v1.3]
- \ltx@newif added.
- \ltx@ifnextchar added.
- Numbers \ltx@zero, \ltx@one, \ltx@two, \ltx@cclv added.

[2010/03/09 v1.4]
- \ltx@pkgextension and \ltx@clsextension are hardcoded to avoid trouble with \onlypreamble.
[2010/04/08 v1.5]
- \texttt{\LaTeX}{\backslash cartwo}, \texttt{\LaTeX}{\backslash cdrtwo}, \texttt{\LaTeX}{\backslash carthree}, \texttt{\LaTeX}{\backslash cdrthree}, \texttt{\LaTeX}{\backslash carfour}, \texttt{\LaTeX}{\backslash cdrfour} added.
- \texttt{\LaTeX}{\backslash ReturnAfterFi} and \texttt{\LaTeX}{\backslash ReturnAfterElseFi} fixed.

[2010/04/16 v1.6]
- \texttt{\LaTeX}{\backslash leavevmode}, \texttt{\LaTeX}{\backslash mbox} added.

[2010/04/26 v1.7]
- \texttt{\LaTeX}{\backslash GobbleNum}, \texttt{\LaTeX}{\backslash CdrNum}, \texttt{\LaTeX}{\backslash CarNum} added.
- \texttt{\LaTeX}{\backslash carzero}, \texttt{\LaTeX}{\backslash cdrzero} added.
- \texttt{\LaTeX}{\backslash hashchar} added.

[2010/09/11 v1.8]
- \texttt{\LaTeX}{\backslash leftbracechar}, \texttt{\LaTeX}{\backslash rightbracechar} added.

[2010/10/25 v1.9]
- \texttt{\LaTeX}{\backslash LocalAppendToMacro} and \texttt{\LaTeX}{\backslash GlobalAppendToMacro} are now \texttt{\LaTeX}{\backslash long}.

[2010/10/31 v1.10]
- \texttt{\LaTeX}{\backslash newglobalif} added.

[2010/11/12 v1.11]
- \texttt{\LaTeX}{\backslash ifempty} added.
- \texttt{\LaTeX}{\backslash firstofthree}, \texttt{\LaTeX}{\backslash secondofthree}, \texttt{\LaTeX}{\backslash thirdofthree} added.

[2010/12/02 v1.12]
- \texttt{\LaTeX}{\backslash onelevel\@sanitize} added.
- \texttt{\LaTeX}{\backslash \ATeXcmds\@num} fixed for the case with \texttt{\LaTeX}{\backslash numexpr} (bug found by GL).

[2010/12/04 v1.13]
- \texttt{\LaTeX}{\backslash ifblank} added.
- Optimization for \texttt{\LaTeX}{\backslash ifempty}.

[2010/12/07 v1.14]
- \texttt{\LaTeX}{\backslash zap\@space} added.

[2010/12/12 v1.15]
- \texttt{\LaTeX}{\backslash minus\@one} added.

[2011/02/04 v1.16]
- \texttt{\LaTeX}{\backslash IfBoxEmpty} and \texttt{\LaTeX}{\backslash IfBoxVoidOrEmpty} added.
- \texttt{\LaTeX}{\backslash firstoffour}, ..., \texttt{\LaTeX}{\backslash fourthoffour} added.
• \texttt{\LaTeX@IfBoxEmpty}: an empty box may have non-zero dimensions.

• \texttt{\LaTeX@ifclasslater} fixed.

• \texttt{\LaTeX@ifnextchar}: detection of optional spaces modified.
  • \texttt{\LaTeX(Loc,Glob)(Toks,Dimen,Skip)(A,B,C,D,E)} added.

• \texttt{\LaTeX@ifnextchar} with conditional support (thanks GL for bug report).

• \texttt{\LaTeX@GlobalPrependToMacro, \LaTeX@LocalPrependToMacro} added (feature request of Martin M"unch).

• \texttt{\LaTeX@carsecond, \LaTeX@carthird, \LaTeX@carfourth, \LaTeX@CarNumth} added.
  • \texttt{\LaTeX@cdrzero, \LaTeX@cdr, \LaTeX@cdrtwo, \texttt{\LaTeX@cdrthree, \LaTeX@cdrfour, \LaTeX@CdrNum} modified to retain braces and spaces. They are expandable in two expansion steps.

• Documentation updates.

• Documentation updates.

• Changed the definitions of \texttt{\LaTeX@iffilelater, \LaTeX@ifpackagelater} and \texttt{\LaTeX@ifclasslater} to support dates in ISO format in same way as the \LaTeX kernel does it since 2017. The commands now use the same test as the \LaTeX kernel. \texttt{\pdfmatch} is no longer used with pdftex, and the tests for dates before 1994 have been removed.

• corrected \texttt{\LaTeX@ifl@t@r} for plain.

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

\texttt\$ \hspace{2cm} 610, 613\
\texttt\# \hspace{2cm} 254
<table>
<thead>
<tr>
<th>Command</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>\skipdef</td>
<td>144, 145, 146, 147, 148, 149, 150, 151, 152, 153</td>
</tr>
<tr>
<td>\unhbox</td>
<td>503, 515, 518</td>
</tr>
<tr>
<td>\unhcopy</td>
<td>622, 634, 642</td>
</tr>
<tr>
<td>\unvcopy</td>
<td>622, 634, 642</td>
</tr>
<tr>
<td>\the</td>
<td>77, 78, 79, 80, 81, 82, 83, 84, 97, 404, 418, 433, 448, 461, 467, 486, 539</td>
</tr>
<tr>
<td>\vbox</td>
<td>621, 632, 640</td>
</tr>
<tr>
<td>\voidb@x</td>
<td>518</td>
</tr>
<tr>
<td>\TMP@EnsureCode</td>
<td>94, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114</td>
</tr>
<tr>
<td>\write</td>
<td>23, 52</td>
</tr>
<tr>
<td>\toksdef</td>
<td>124, 125, 126, 127, 128, 129, 130, 131, 132, 133</td>
</tr>
<tr>
<td>\x</td>
<td>14, 15, 18, 22, 26, 28, 51, 56, 66, 75, 87</td>
</tr>
</tbody>
</table>