Contents

Introduction 1

Providing \texttt{xparse} in the format 1

Other changes to the \LaTeX\ kernel 2

\texttt{symbol} in math mode for large Unicode values 2
Correct Unicode value of \texttt{\textasciitilde y} (\textbar{} y) 2
Add support for Unicode soft hyphens 2
Fix capital accents in Unicode engines 2
Support \texttt{calc} in various kernel commands 2
Spaces in filenames of included files 2
Set a non-zero \texttt{\baselineskip} in text scripts 2
Spacing issues when using \texttt{\linethickness} 2
Better support for the legacy series default interface 2
Support for uncommon font series defaults 2
Checking the current font series context 2
Avoid spurious package option warning 3
Adjusting \texttt{fleqn} 3
Provide \texttt{\clap} 3
Fix to legacy math alphabet interface 3
Added tests for format, package and class dates 3
Avoid a problem with \texttt{\verb} 3
Record the counter name stepped by \texttt{\refstepcounter} 3
Add support for Unicode soft hyphens 3
Native \LaTeX\ behaviour for \texttt{-} 4
Allow \texttt{\par} commands inside \texttt{\typeout} 4

Changes to packages in the \texttt{graphics} category 4

Generate a warning if existing color definition is changed 4

Changes to packages in the \texttt{tools} category 4

\texttt{array}: Support stretchable glue in \texttt{w}-columns 4
\texttt{array}: Use math mode for \texttt{w} and \texttt{W}-cells in \texttt{array} 4
\texttt{xr}: Support for spaces in filenames 4

Changes to packages in the \texttt{amsmath} category 4

Placement corrections for two accent commands 4
Fixes to \texttt{aligned} and \texttt{gathered} 4

Changes to the \texttt{babel} package 4

Introduction

This document is under construction ...

Providing \texttt{xparse} in the format

The official interface in the \LaTeX\ 2ε kernel for creating document-level commands has always been \texttt{\newcommand}. This was a big step forward from \LaTeX\ 2.09. However, it was still very limited in the types of command it can create: those taking at most one optional argument in square brackets, then zero or more mandatory arguments. Richer syntaxes required use of the \TeX\ \texttt{\def} primitive along with appropriate low-level macro programming.

The \LaTeX\ team started work on a comprehensive document-command parser, \texttt{xparse}, in the late 1990s. In the past decade, the experimental ideas it provides have been carefully worked through and moved to a stable footing. As such, \texttt{xparse} is now used to define a very large number of document and package commands. It does this by providing a rich and self-consistent syntax to describe a wide range interfaces seen in \LaTeX\ packages.

The ideas developed in \texttt{xparse} are now sufficiently well-tested that the majority can be transferred into the \LaTeX\ kernel. Thus the following commands have been added:

- \texttt{\NewDocumentEnvironment}, \texttt{\RenewDocumentEnvironment}, \texttt{\ProvideDocumentEnvironment}, \texttt{\DeclareDocumentEnvironment}
- \texttt{\BooleanTrue}, \texttt{\BooleanFalse}
- \texttt{\IfBooleanTF}, \texttt{\IfBooleanT}, \texttt{\IfBooleanF}
- \texttt{\IfNoValueTF}, \texttt{\IfNoValueT}, \texttt{\IfNoValueF}
- \texttt{\IfValueTF}, \texttt{\IfValueT}, \texttt{\IfValueF}
- \texttt{\SplitArgument}, \texttt{\SplitList}, \texttt{\TrimSpaces}, \texttt{\ProcessList}, \texttt{\ReverseBoolean}
- \texttt{\GetDocumentCommandArgSpec}, \texttt{\GetDocumentEnvironmentArgSpec}

Most, but not all, of the argument types defined by \texttt{xparse} are now supported at the kernel level. In
particular, the types \textbf{g/G}, \textit{l} and \textit{u} are not provided by the kernel code; these are deprecated but still available by explicitly loading \texttt{xpase}. All other argument types are now available directly within the \TeX\ kernel.

**Other changes to the \TeX\ kernel**

\texttt{symbol} in math mode for large Unicode values

The \TeX\ kernel defines the command \texttt{symbol}, which allows characters to be typeset by entering their 'slot number'. With the \texttt{LuaTeX} and \texttt{XeTeX} engines, these slot numbers can extend to very large values to accommodate Unicode characters in the upper Unicode planes (e.g., bold mathematical capital \textit{A} is slot number '1D400 in hex or 119808 in decimal). The \texttt{XeTeX} engine did not allow \texttt{symbol} in math mode for values above \texttt{2^16}, and this limitation has now been lifted.  

\texttt{github issue 152}

Set a non-zero \texttt{baselineskip} in text scripts

As \texttt{textsuperscript} and \texttt{textsubscript} usually contain only a few characters on a single line the \texttt{baselineskip} was set to zero. However, \texttt{hyperref} uses that value to determine the height of a link box which consequently came out far too small. This has been adjusted. 

\texttt{github issue 249}

Spacing issues when using \texttt{linethickness}

In some circumstances the use of \texttt{linethickness} introduced a spurious space that shifted objects in a picture environments to the right. This has been corrected. 

\texttt{github issue 274}

Better support for the legacy series default interface

In the initial implementation of \TeX's font selection scheme (NFSS) changes to any default where always carried out by redefining some commands, e.g., \texttt{seriesdefault}. In 2019 we introduced various extensions and with it new methods of customising certain parts of NFSS, e.g., the recommended way for changing the series default(s) is now through \texttt{DeclareFontSeriesDefault} [1]. In this release we improved the support for legacy documents using the old method was improved to cover additional edge cases. 

\texttt{github issues 306,315}

Support for uncommon font series defaults

If a font family was set up with fairly unusual font series defaults, e.g.,

\begin{verbatim}
\renewcommand{\ttfamily}{\bf}\sf
\end{verbatim}

then a switch between the main document families, e.g., \texttt{\ttfamily...\rmfamily} did not always correctly continued typesetting in medium or bold series if that involved adjusting the values used by \texttt{mdseries} or \texttt{bfsseries}. This has now been corrected. 

\texttt{github issue 291}

Checking the current font series context

Sometimes it is necessary to define commands that act differently when used in bold context (e.g., inside \texttt{\textbf{bf}}. Now that it is possible in \TeX\ to specify different "bf" defaults based for each of the three meta families (\texttt{rm}, \texttt{sf} and \texttt{tt}) via \texttt{\DeclareFontSeriesDefault}, it is not any longer easy to answer the question “am I typesetting in a bold context?”. To help with this problem a new command was provided:

\begin{verbatim}
\includeonly, which is a comma-separated list of files to process, can also contain spaces with any leading and trailing spaces stripped from the individual filenames while the spaces in the file names will remain intact. 

\texttt{github issue 217}

\texttt{Fix capital accents in Unicode engines}

In Unicode engines the capital accents such as \texttt{\textsc{capitalcedilla}}, etc. have been implemented as trivial short hands for the normal accents (because other than Computer Modern virtually no fonts support them), but that failed when \texttt{hyperref} got loaded. This has been corrected.  

\texttt{github issue 326}

\texttt{Support calc in various kernel commands}

The \texttt{\hspace}, \texttt{\hspace}, \texttt{\addspace}, \texttt{\ } and other commands simply passed their argument to a \TeX\ primitive to produce the necessary space. As a result it was impossible to specify anything other than a simple dimension value in such arguments. This has been changed, so that now \texttt{calc} syntax is also supported with these commands. 

\texttt{github issue 332}

\texttt{Spaces in filenames of included files}

File names containing spaces lead to unexpected results when used in the commands \texttt{\include} and \texttt{\includeonly}. This has now been fixed and the argument to \texttt{\include} can contain file name containing spaces. Leading or trailing spaces will be stripped off but spaces within the file name are kept. The argument to
Avoid spurious package option warning

When a package is loaded with a number of options, say \( X \), \( Y \) and \( Z \), and then later another loading attempt was made with a subset of the options or no options, it was possible that you got an error message that option \( X \) is not known to the package. This obviously incorrect error was due to some timing issue where the list of available options got lost prematurely. This has now been fixed. (github issue 22)

Adjusting fleqn

In amsmath the \texttt{mathindent} parameter used with the fleqn design is a rubber length parameter allowing for setting it to a value such as \texttt{1em minus 1em}, i.e., so the the normal indentation can be reduced in case of very wide math displays. This is now also supported by the \LaTeX{} standard classes.

In addition a compressible space between formula and equation number in the equation environment got added when the fleqn option is used so that a very wide formula doesn’t bump into the equation number. (github issue 252)

Provide \texttt{\clap}

\LaTeX{} has inherited \texttt{\clap} and \texttt{\rlap} from plain \TeX{} (zero-sized boxes whose content sticks out to the left or right, respectively) but there isn’t a corresponding \texttt{\clap} command that centers the material. This missing command was added by several addon packages, e.g., mathtools, and has now been added to the kernel.

Fix to legacy math alphabet interface

When using the \LaTeX{} 2.09 legacy math alphabet interface, e.g., \texttt{\$\text{sf} \, -1\$} instead of \texttt{\$\text{mathsf}(-1)\$}, an extra math Ord atom was added to the formula in case the math alphabet was used for the first time. In some cases this math atom would change the spacing, e.g., change the unary minus sign into a binary minus in the above example. This has finally been fixed. (gnats issue latex/3357)

\texttt{\IfFontSeriesContextTF{(context)}}
\texttt{\{(true code)\} \{(false code)\}}

The \texttt{(context)} can be either \texttt{bf} (bold) or \texttt{md} (medium) and depending on whether or not the current font is recognized as being selected through \texttt{\bfseries} or \texttt{\mdseries} the \texttt{(true code)} or \texttt{(false code)} is executed. As an example

\texttt{\usepackage{bm} \% (bold math)  
\newcommand{\vbeta}{\IfFontSeriesContextTF{\bfseries}{\{(true code)\}}{\{(false code)\}}}{\ensuremath{\bm{\beta}}}%  
{\ensuremath{\bm{\beta}}}%}

This way you can write \texttt{\vbeta-isotopes} and if used in a heading it comes out in a bolder version. (github issue 336)

\texttt{\IfFormatAtLeastTF{(date)}}
\texttt{\{(true code)\} \{(false code)\}}

and for package and class tests

\texttt{\IfPackageAtLeastTF{(package name)}{(date)}}
\texttt{\{(true code)\} \{(false code)\}}

For compatibility reasons the legacy commands remain available, but we suggest to replace them over time and use the new interfaces in new code. (github issue 186)

Avoid a problem with \texttt{\verb}

If a user typed \texttt{\verb\textasciitilde!\verb!foo} instead of \texttt{\verb\textasciitilde!\verb!foo} by mistake, then surprisingly the result was “\textasciitilde\textasciitilde\verb!foo” without any warning or error. What happened was that the \textasciitilde became the argument delimiter due to the rather complex processing done by \texttt{\verb} to render verbatim. This now got fixed and spaces directly following the \texttt{\verb} or \texttt{\verb*} are ignored as elsewhere. (github issue 327)

Record the counter name stepped by \texttt{\refstepcounter}

\texttt{\refstepcounter} now stores the name of counter in \texttt{\@currentcounter}. This allows packages like \texttt{zref} and hyperref to store the name without having to patch \texttt{\refstepcounter}. (github issue 300)

Add support for Unicode soft hyphens

For a long time, the UTF-8 option for inputenc made the Unicode soft hyphen character \texttt{(U+00AD)} an alias for the \LaTeX{} soft hyphen \texttt{\textasciitilde}. The Unicode engines X\LaTeX{} and Lua\LaTeX{} behaved different though: They either ignored \texttt{U+00AD} or interpreted it as an unconditional hyphen. This inconsistency is fixed now and \LaTeX{} always treats \texttt{U+00AD} as \texttt{\textasciitilde}. (github issue 323)
**Native Lua\TeX\ behaviour for \-**

\LaTeX\ changes \- to add a discretionary hyphen even if \texttt{\hyphenchar} is set to \texttt{-1}. This change is not necessary under Lua\TeX\ because in there \- is not affected by \texttt{\hyphenchar} in the first place. Therefore this behaviour has been changed to ensure that Lua\TeX\'s (language specific) hyphenation characters are respected by \-.

**Allow \par commands inside \texttt{\textbackslash typout}**

\texttt{\textbackslash typout} used to choke when seeing an empty line or a \texttt{\par} command in its argument. However, sometimes it is used to display arbitrary user input or code (wrapped, for example, in \texttt{\unexpanded}) which may contain explicit \texttt{\par} commands. This is now allowed.  

(github issue 335)

**Changes to packages in the graphics category**

*Generate a warning if existing color definition is changed*

If a color is defined twice using \texttt{\DefineNamedColor}, no info text \texttt{\textbf{Redefining color \ldots in named color model \ldots}} is written to the log file, because of a typo in the check. This has been corrected.

(gnats issue graphics/3635)

**Changes to packages in the tools category**

\texttt{array: Support stretchable glue in \texttt{w}-columns}

If stretchable glue, e.g., \texttt{\dotfill}, is used in \texttt{tabular} columns made with the \texttt{array} package, it stretches as it would in normal paragraph text. The one exception were \texttt{w}-columns (but not \texttt{W}-columns) were it got forced to its nominal width (which in case if \texttt{\hfill} or \texttt{\dotfill} is 0 pt). This has been corrected and now \texttt{w}-columns behave like all other column types in this respect.

(github issue 270)

**Changes to the babel package**

Multilingual typesetting has much evolved in the past years, and \textbf{babel}, like \LaTeX\ itself, has followed the footsteps of Unicode and the W3C consortiums to produce proper output in many languages.

Furthermore, the traditional model to define and select languages (which can be called “vertical”), based on closed files, which is still the preferred one in monolingual documents, is being extended with a new model (which can be called “horizontal”) based on services provided by \textbf{babel}, which allows to define and redefine locales with the help of simple \texttt{.ini} files based on key/value pairs. Babel provides about of 250 of these files, which have been generated with the help of the Unicode Common Language Data Repository.

Thanks to the recent advances in \texttt{lualatex} and \texttt{lualotfload}, babel currently provides \texttt{services} for bidi typesetting, line breaking for South East Asian and CJK scripts, non-standard hyphenation (like ff to ff-f), alphabetic and additive counters, automatic selection of fonts and languages based on the script, etc. This means babel can be used to typeset such a variety of languages as Russian, Arabic, Hindi, Thai, Japanese, Bangla, Amharic, Greek, and many others.

And since these \texttt{.ini} files they are easily parsable, they can serve as a source for other packages.

For further details take a look at the \textbf{babel} documentation [4].

**References**

[1] \LaTeX\ Project Team: \LaTeX\ 2e news 31.  
[2] LaTeX documentation on the LaTeX Project Website.  
https://latex-project.org/help/documentation/

https://github.com/latex3/latex2e/issues/

Localization and internationalization. 
https://www.ctan.org/pkg/babel