Contents

\-\LaTeX

\section*{\-\LaTeX}

\section*{Default encodings in \-\LaTeX and Lua\-\LaTeX}

\section*{\texttt{\showhyphens} in \-\LaTeX}

\section*{The \texttt{fixltex2e} package}

\section*{The \texttt{latexbug} package}

\section*{Updates to \texttt{amsmath}}

\section*{Updates to \texttt{tools}}

\section*{An addendum to the release changes in 2015: page breaks and vertical spacing}

\section*{\-\LaTeX}

In \-\LaTeX News 16 (December 2003) the team announced

\par

\begin{quote}
We expect that within the next two years, releases of \-\LaTeX will change modestly in order to run best under an extended \TeX engine that contains the \-\TeX primitives, e.g., \-\TeX or pdf\-\TeX.
\end{quote}

\par

and also said

\begin{quote}
Although the current release does not require \-\TeX features, we certainly recommend using an extended \TeX, especially if you need to debug macros.
\end{quote}

\par

For many years the team have worked on the basis that users will have \-\TeX available but had not revisited the above statements formally. As of the January 2017 release of \-\LaTeX 2e, \-\TeX is required to build the format, and attempting to build a format without the extensions will fail.

Practically, modern \TeX distributions provide the extensions in all engines other than the “pure” Knuth \texttt{tex}, and indeed parts of the format-building process already require \-\TeX, most notably some of the UTF-8 hyphenation patterns. As such, there should be no noticeable effect on users of this change.

The team expect to make wider use of \-\TeX within the kernel in future; details will be announced where they impact on end users in a visible way.

\section*{Default encodings in \-\LaTeX and Lua\-\LaTeX}

The default encoding in \-\LaTeX has always been the original 128-character encoding \texttt{OT1}. For Unicode based \TeX engines, this is not really suitable, and is especially problematic with \-\LaTeX as in the major distributions this is built with Unicode based hyphenation patterns in the format. In practice this has not been a major problem as documents use the contributed \texttt{fontspec} package in order to switch to a Unicode encoded font.

In this release we are adding \texttt{TU} as a new supported encoding in addition to the previously supported encodings such as \texttt{OT1} and \texttt{T1}. This denotes a Unicode based font encoding. It is essentially the same as the \texttt{TU} encoding that has been on trial with the experimental \texttt{tuenc} option to \texttt{fontspec} for the past year.

\par

The \-\LaTeX and Lua\-\LaTeX formats will now default to \texttt{TU} encoding and \texttt{lmr} (Latin Modern) family. In the case of Lua\-\LaTeX the contributed \texttt{luaotfload} Lua module will be loaded at the start of each run to enable the loading of OpenType fonts.

\par

The \texttt{fontspec} package is being adjusted in a companion release to recognise the new encoding default arrangements.

\par

Note that in practice no font supports the full Unicode range, and so \texttt{TU} encoded fonts, unlike fonts specified for \texttt{T1}, may be expected to be incomplete in various ways. In the current release the file \texttt{tuenc.def} that implements the \texttt{TU} encoding-specific commands has made some basic assumptions for (for example) default handling of accent commands, and the set of command names is derived from the command names used for the UTF-8 support in the \texttt{inputenc} package, restricted roughly to the character ranges classically provided by the \texttt{T1} and \texttt{TS1} encodings, but is part of a longer term plan seen over recent releases to increase support for Unicode based \TeX engines into the core \-\LaTeX support.

\par

If for any reason you need to process a document with the previous default \texttt{OT1} encoding, you may switch encoding in the usual ways, for example
\begin{verbatim}
\usepackage[OT1]{fontenc}
\end{verbatim}

or you may roll back all the changes for this release by starting the document with
\begin{verbatim}
\RequirePackage[2016/12/31]{latexrelease}
\end{verbatim}
\showhyphens in \LaTeX

Due to the way \LaTeX interfaces to font libraries, the standard definition of \showhyphens does not work. A variant definition has been available in the contributed `xctextra` package, however a (slightly different) definition for \showhyphens is now included in \LaTeX by default. As usual this change will be undone if an earlier release is specified using the `latexrelease` package.

The `fixltx2e` package

As described in \LaTeX News 22, the `fixltx2e` package has become obsolete with the new update policy. Since 2015 it has just made a warning and exited. In this release we have re-introduced all the code from the original fixes in the 2014 \LaTeX but guarded by `\IncludeInRelease{2015/01/01}{}`. So for current releases `fixltx2e` still just displays a warning but for old releases, whether that is an old format, or a format with the version date reset via the `latexrelease` package, the fixes in the original `fixltx2e` will be applied.

This improves the ability to run old documents in a way that is compatible with contemporary formats. If you have a 2014 document that used `\usepackage{fixltx2e}` and you add `\RequirePackage[2014/01/01]{latexrelease}` and process it with the current format then `latexrelease` will undo most changes made since 2014, but now when the document includes `fixltx2e` it will act like a 2014 version of the package and apply the code fixes, not just give a warning that the package is obsolete.

The `latexbug` package

As explained in more detail at the \LaTeX Project website\(^1\) a new package, `latexbug`, has been produced to help produce test files to accompany bug reports on the core \LaTeX distribution. This is being published separately to CTAN at the same time as this release. By using the `latexbug` package you can easily check that the packages involved in the test are all part of the core release. The \LaTeX project cannot handle bug reports on contributed packages, which should be directed to the package maintainer as given in the package documentation.

Updates to `amsmath`

The `amsmath` package has two updates at this release.

- The large delimiters around generalised fractions (for example in the `\binom` construct) did not work in previous releases if using Lua\TeX or \LaTeX with OpenType math fonts. This is related to the lack of specific metrics for this use in the OpenType Math table. In principle Lua\TeX has two additional named metrics to control the delimiters but these are not initialised by default, and in \LaTeX it does not seem possible to make them work at all. So for Unicode \TeX systems, a new implementation of `\genfrac` is used at this release that uses `\left\right` internally but parameterised to give spacing as close to the original as possible. The implementation in (pdf)\TeX is unaffected.

- The spacing to the left of the `aligned` and `gathered` environments has been fixed: the spurious thin space is no longer added by default. Package options control this to revert to the original behaviour where required; see the `amsldoc` guide for further details.

---

\(^1\)https://www.latex-project.org/bugs/