The lualatex-math package*

Philipp Stephani
p.stephani2@gmail.com
2021/07/05

Contents

1 Introduction 1

2 Interface 2

3 Implementation of the \LaTeX\ 2e package 2

3.1 Requirements ............................................. 2

3.2 Messages .................................................. 3

3.3 Initialization ............................................. 3

3.4 Patching .................................................. 3

3.5 \LaTeX\ 2e kernel ........................................ 5

3.6 amsmath .................................................. 5

3.7 mathtools ............................................... 8

3.8 icomma .................................................. 9

4 Implementation of the Lual\TeX\ module 10

1 Introduction

Lual\TeX\ brings major improvements to all areas of \TeX\ typesetting and programming. They are made available through new primitives or the embedded Lua interpreter, and combining them with existing \LaTeX\ 2e packages is not a task the average \LaTeX\ user should have to care about. Therefore a multitude of \LaTeX\ 2e packages have been written to bridge the gap between documents and the new features. The lualatex-math package focuses on the additional possibilities for mathematical typesetting. The most eminent of the new features is the ability to use Unicode and OpenType fonts, as provided by Will Robertson’s unicode-math package. However, there is a smaller group of changes unrelated to Unicode: these are to be dealt with in this package. While in principle most \TeX\ documents written for traditional engines should work just fine with Lual\TeX, there is a small number of breaking changes that require the attention of package authors. The lualatex-math package tries to fix some of the issues encountered while porting traditional macro packages to Lual\TeX.

The decision to write patches for existing macro packages should not be made lightly: monkey patching done by somebody different from the original package author ties the patching package to the implementation details of the patched functionality and breaks all rules of encapsulation. However, due to the lack of

*This document corresponds to lualatex-math v1.11, dated 2021/07/05.
alternatives, it has become an accepted way of providing new functionality in \LaTeX. To keep the negative impact as small as possible, the lualatex-math package patches only the \LaTeX\ 2\epsilon kernel and a small number of popular packages. In general, this package should be regarded as a temporary kludge that should be removed once the math-related packages are updated to be usable with Lua\LaTeX. By its very nature, the package is likely to cause problems; in such cases, please refer to the issue tracker\(^1\).

2 Interface

The lualatex-math package can be loaded with \usepackage or \RequirePackage, as usual. It has no options and no public interface; the patching is always done when the package is loaded and cannot be controlled. As a matter of course, the lualatex-math package needs Lua\LaTeX\ to function; it will produce error messages and refuse to load under other engines and formats. The package depends on the expl3 bundle, the etoolbox package and the filehook package. The lualatex-math package is independent of the unicode-math package; the fixes provided here are valid for both Unicode and legacy math typesetting.

Currently patches for the \LaTeX\ 2\epsilon kernel and the amsmath, mathtools and icomma packages are provided. It is not relevant whether you load these packages before or after lualatex-math. They should work as expected (and ideally you shouldn’t notice anything), but if you load other packages that by themselves overwrite commands patched by this package, bad things may happen, as it is usual with \LaTeX.

One user-visible change is that the new \mathstyle primitive should work in all cases after the lualatex-math package has been loaded, provided you use the high-level macros \frac, \binom, and \genfrac. The fraction-like \TeX primitives like \over or \atopwithdelims and the plain \TeX leftovers like \brack or \choose cannot be patched, and you shouldn’t use them.

3 Implementation of the \LaTeX\ 2\epsilon package

3.1 Requirements

\begin{verbatim}
\@_restore_catcode:N
\end{verbatim}

Executing the exhaustive expansion of \@_restore_catcode:N(\textit{character token}) restores the category code of the \textit{character token} to its current value.

\begin{verbatim}
\cs_new_nopar:Npn \@@_restore_catcode:N \#1 \{
  \char_set_catcode:NN \int_eval:n {`\#1} \char_value_catcode:n {`\#1}
}\directlua{require("lualatex-math")}
\end{verbatim}

\(^1\)https://github.com/phst/lualatex-math/issues
We use the macro defined above to restore the category code of the dollar sign. There are packages that make the dollar sign active; hopefully they get loaded after the packages we are trying to patch.

\exp_args:Nx \AtEndOfPackage {
\@@_restore_catcode:N \$
}
\char_set_catcode_math_toggle:N $

3.2 Messages

\texttt{luatex-required} Issued when not running under \texttt{LuaTeX}.

\msg_new:nnn { lualatex-math } { luatex-required } {
I- will- stop- loading- now.
}

\texttt{macro-expected} Issued when trying to patch a non-macro. The first argument must be the detokenized macro name.

\msg_new:nnn { lualatex-math } { macro-expected } {
I've- expected- that- \#1- is- a- macro,- but- it- isn't.
}

\texttt{wrong-meaning} Issued when trying to patch a macro with an unexpected meaning. The first argument must be the detokenized macro name; the second argument must be the actual detokenized meaning; and the third argument must be the expected detokenized meaning.

\msg_new:nnn { lualatex-math } { wrong-meaning } {
I've- expected- \#1- to- have- the- meaning \"
#3, \"
but- it- has- the- meaning \"
#2.
}

\texttt{patch-macro} Issued when a macro is patched. The first argument must be the detokenized macro name.

\msg_new:nnn { lualatex-math } { patch-macro } {
I'm- going- to- patch- macro- \#1.
}

3.3 Initialization

Unless we are running under \texttt{LuaTeX}, we issue an error and quit immediately.

\sys_if_engine_luatex:F {
\msg_error:nn { lualatex-math } { luatex-required }
\endinput
}

3.4 Patching

\@@_temp:w A scratch macro.

\cs_new_eq:NN \@@_temp:w \prg_do_nothing:
\@@_patch:NNnnn The auxiliary macro \@@_patch:NNnnn\texttt{(command)}\texttt{(factory command)}\texttt{(parameter text)}\texttt{(expected replacement text)}\texttt{(new replacement text)} tries to patch \texttt{(command)}. If \texttt{(command)} is undefined, do nothing. Otherwise it must be a macro with the given \texttt{(parameter text)} and \texttt{(expected replacement text)}, created by the
given *(factory command)* or equivalent. In this case it will be overwritten using the *(parameter text)* and the *(new replacement text)*. Otherwise issue a warning and don't overwrite.

\begin{verbatim}
40 \cs_new_protected_nopar:Npn \@@_patch:NNnnn #1 #2 #3 #4 #5 {
41 \cs_if_exist:NT #1 {
42 \token_if_macro:NTF #1 {
43 \group_begin:
44 #2 \@@_temp:w #3 #4 
45 \cs_if_eq:NNTF #1 \@@_temp:w {
46 \msg_info:nnx { lualatex-math } { patch-macro }
47 \{ \token_to_str:N #1 }
48 \group_end:
49 #2 #1 #3 { #5 }
50 } {
51 \msg_warning:nnxxx { lualatex-math } { wrong-meaning }
52 \{ \token_to_str:N #1 } \{ \token_to_meaning:N #1 }
53 \{ \token_to_meaning:N \@@_temp:w }
54 \group_end:
55 }
56 } {
57 \msg_warning:nxxx { lualatex-math } { macro-expected }
58 \{ \token_to_str:N #1 }
59 }
60 }
61 }
62 \cs_generate_variant:Nn \@@_patch:NNnnn { c }
\end{verbatim}

The macro \@@_set_mathchar:NN *(control sequence)*(token) defines the *(control sequence)* as an extended mathematical character shorthand whose mathematical code is given by the mathematical code of the character *(token)*. We cannot use the \Umathcharnumdef primitive here since we would then rely on the \Umathcodenum primitive which is currently broken.\footnote{\url{http://tug.org/pipermail/luatex/2012-October/003794.html}}

\begin{verbatim}
63 \cs_new_protected_nopar:Npn \@@_set_mathchar:NN #1 #2 {
64 \Umathchardef #1
65 \lua_now:e {
66 lualatex.math.print_class_fam_slot( \int_eval:n { `#2 } )
67 }
68 \scan_stop:
69 }
\end{verbatim}

The macro \@@_before_package:nn *(package)*(code) executes the *(code)* before the *(package)* is loaded. Accordingly, \@@_after_package:nn *(package)*(code) executes the *(code)* after the *(package)* is loaded. If the *(package)* is already loaded, nothing happens. We prefer using native \LaTeX\ 2\epsilon hooks if possible.

\begin{verbatim}
70 \ifl@t@r \fmtversion { 2020/10/01 } {
71 \cs_new_protected_nopar:Npn \@@_before_package:nn #1 #2 {
72 \AddToHook { package/before/#1 } { #2 }
73 }
74 \cs_new_protected_nopar:Npn \@@_after_package:nn #1 #2 {
75 \AddToHook { package/after/#1 } { #2 }
76 }
77 }
78 \RequirePackage { filehook } [ 2011/03/09 ]
79 \cs_new_protected_nopar:Npn \@@_before_package:nn #1 #2 {
80 \AtBeginOfPackageFile { #1 } { #2 }
81 }
\end{verbatim}
The macro \@after_package_or_now:nn{(package)}{(code)} executes the (code) after the (package) is loaded. If the (package) is already loaded, the (code) is executed immediately.

3.5 \LaTeX{} 2ε kernel

\LaTeX{} enables access to the current mathematical style via the \texttt{\mathstyle} primitive. For this to work, fraction-like constructs (e.g., \texttt{\over\text{numerator}}) have to be enclosed in a \texttt{\Ustack} group. \texttt{\frac} can be patched to do this, but the plain \LaTeX{} remnants \texttt{\choose}, \texttt{\brack} and \texttt{\brace} should be discouraged.

\texttt{\frac}

Here we assume that nobody except \texttt{amsmath} redefines \texttt{\frac}. This is obviously not the case, but we ignore other packages (e.g., \texttt{nath}) for the moment. We only patch the \LaTeX{} 2ε kernel definition if the \texttt{amsmath} package is not loaded; the corresponding patch for \texttt{amsmath} follows below. Since \texttt{\frac} is declared by \texttt{\DeclareRobustCommand}, we must patch the macro \texttt{\frac}.

\texttt{\AtEndPreamble}

\texttt{\EndPreamble}

To do: do we need the additional set of braces around \texttt{\Ustack}?

3.6 \texttt{amsmath}

The popular \texttt{amsmath} package is subject to three \LaTeX{}-related problems:

- The \texttt{\mathcode} primitive is used several times, which fails for Unicode math characters. \texttt{\Umathcode} should be used instead.

- Legacy font dimensions are used for constructing stacks in the \texttt{\substack} command and the \texttt{\subarray} environment. This doesn’t work if a Unicode math font is selected.

- The fraction commands \texttt{\frac} and \texttt{\genfrac} don’t use the \texttt{\Ustack} primitive.

These problems have been fixed in version 2.17i of \texttt{amsmath}, so we don’t attempt to patch it if that version is loaded.
These constants contain the standard \TeX mathematical codes for the minus and the equal signs. We temporarily set the math codes to these constants before loading the \texttt{amsmath} package so that it can request the legacy math code without error.

\begin{verbatim}
\int_const:Nn \c_@@_std_minus_mathcode_int { "2200 }
\int_const:Nn \c_@@_std_equal_mathcode_int { "303D }
\end{verbatim}

These mathematical characters are saved before \texttt{amsmath} is loaded so that we can temporarily assign the \TeX values to the mathematical codes of the minus and equals signs. The \texttt{amsmath} package queries these codes, and if they represent Unicode characters, the package loading will fail. If \texttt{amsmath} has already been loaded, there is nothing we can do, therefore we use the non-starred version of \texttt{\texttt{AtBeginOfPackageFile}}.

\begin{verbatim}
\tl_new:N \l_@@_minus_mathchar
\tl_new:N \l_@@_equal_mathchar
@@_before_package:nn { amsmath } {
    \@ifpackagelater { amsmath } { 2020/08/24 } { } {
        \@@_set_mathchar:NN \l_@@_minus_mathchar \-
        \@@_set_mathchar:NN \l_@@_equal_mathchar =
    }
\@@_after_package:nn { amsmath } {
    \std@minus
    \std@equals
}
\end{verbatim}

Now we temporarily reset the mathematical codes.

\begin{verbatim}
\char_set_mathcode:nn { \`\- } { \c_@@_std_minus_mathcode_int }
\char_set_mathcode:nn { \`\=} { \c_@@_std_equal_mathcode_int }
\@@_after_package:nn { amsmath } {
    \std@minus
    \std@equals
}
\end{verbatim}

The \texttt{amsmath} package defines the control sequences \texttt{\std@minus} and \texttt{\std@equal} as mathematical character shorthands while loading, but uses our restored mathematical codes, which must be fixed.

\begin{verbatim}
\cs_set_eq:NN \std@minus \l_@@_minus_mathchar
\cs_set_eq:NN \std@equal \l_@@_equal_mathchar
\end{verbatim}

Finally, we restore the original mathematical codes of the two signs.

\begin{verbatim}
\Umathcodenum \`\- \l_@@_minus_mathchar
\Umathcodenum \`\= \l_@@_equal_mathchar
\end{verbatim}

All of the following fixes work even if \texttt{amsmath} is already loaded.

\begin{verbatim}
\@begindocumenthook
amsmath repeats the definition of \texttt{\std@minus} and \texttt{\std@equal} at the beginning of the document, so we also have to patch the internal kernel macro \texttt{\begindocumenthook} which contains the hook code.
\end{verbatim}

\begin{verbatim}
\@@_after_package_or_now:nn { amsmath } {
    \@ifpackagelater { amsmath } { 2020/08/24 } { } {
        \tl_replace_once:Nnn \@begindocumenthook {
            \mathchardef \std@minus \mathcode `\- \relax
            \mathchardef \std@equal \mathcode `\= \relax
        }
    }
}\end{verbatim}

\texttt{subarray} The \texttt{subarray} environment uses legacy font dimensions. We simply patch it to use \texttt{Lua\TeX} font parameters (and \texttt{\LaTeX3} expressions instead of \TeX arithmetic). Since subscript arrays are conceptually vertical stacks, we use the sum of top and bottom
shift for the default vertical baseline distance (\baselineskip) and the minimum vertical gap for stack for the minimum baseline distance (\lineskip).

\@ifpackagelater { amsmath } { 2020/09/23 } { } { 130
\@ifpackagelater { amsmath } { 2020/09/23 } { } { 131
\@patch:Nnnn \subarray \cs_set:Npn { #1 } { 132
\vcenter \bgroup \Let@ \restore@math@cr \default@tag \baselineskip \fontdimen 10\scriptfont 2 \advance \baselineskip \fontdimen 12\scriptfont 2 \langle @\rangle \lineskip \thr@@ \fontdimen 8\scriptfont 3 \langle @\rangle \lineskiplimit \lineskip \ialign \bgroup \ifx c #1 \hfil \fi \m@th \scriptstyle ## \hfil \crcr } { 134
\vcenter \c_group_begin_token \Let@ \restore@math@cr \default@tag \skip_set:Nn \baselineskip { \Umathstacknumup \scriptstyle + \Umathstackdenomdown \scriptstyle } \lineskip \Umathstackvgap \scriptstyle \lineskiplimit \lineskip \ialign \c_group_begin_token \token_if_eq_meaning:NNT c #1 { \hfil } \Ustartmath \m@th \scriptstyle \alignmark \alignmark \Ustopmath \hfil \crcr } { 157
\Ualign { \group_begin: #1 \group_end: \@@over #2 } { 179
\Ustack { \group_begin: #1 \group_end: \@@over #2 } { 180
\U@ =\textmath
\frac

Since \frac is declared by \DeclareRobustCommand, we must patch the macro \frac. 172 \@patch:cNnnn { frac- } \cs_set:Npn { #1 \#2 } { 173 { 174 (\@=) \begingroup \#1 \endgroup \@@over \#2 } 175 } { 176 } { 177 } { 178 } { 179 \Ustack { \group_begin: #1 \group_end: \@@over #2 } { 180 (\@=\textmath) 181 } 182 }
\genfrac Generalized fractions are typeset by the \texttt{\genfrac} command. Since \texttt{\genfrac} is declared by \texttt{\DeclareRobustCommand}, we have to patch the macro \texttt{\genfrac}.

\begin{verbatim}
\@@_patch:cNnnn { genfrac- } \cs_set:Npn { \#1 \#2 \#3 \#4 \#5 \#6 }
\{
\@mathstyle \{ \#4 \}
\genfrac@choice o { \#1 }
\{
\begingroup \#5 \endgroup
\langle \@= \rangle
\ifx @ \#3 @ \@@over \else \@@above \fi \#3 \relax \#6
\}
\genfrac@choice c { \#2 }
\}
\{
\@mathstyle \{ \#4 \}
\genfrac@choice o { \#1 }
\{
\Ustack { \group_begin: \#5 \group_end: }
\tl_if_empty:nTF { \#3 } { \@@over }
\{
\@@above \#3 \scan_stop:
\}
\Utmath}
\#6
\}
\{
\genfrac@choice c { \#2 }
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\}
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
\)
Here the additional set of braces is absolutely necessary, otherwise the changed mathematical style would be applied to the material after the \mathchoice construct. As the original command works in both text and math mode, we use \ensuremath here.

\begin{verbatim}
{ \ensuremath { \use:c { cramped \cs_to_str:N #1 } #2 } }
\end{verbatim}

\subsection*{3.8 icomma}

The \texttt{icomma} package uses \texttt{mathchardef} to save the mathematical code of the comma character. This breaks for Unicode fonts. The incompatibility was noticed by Peter Breitfeld.\footnote{https://groups.google.com/forum/#!topic/de.comp.text.tex/Cputk-AJS5I/discussion}

\texttt{icomma} defines the mathematical character shorthand \texttt{icomma} at the beginning of the document, therefore we again patch \texttt{@begindocumenthook}.

\begin{verbatim}
\@@_after_package_or_now:nn { icomma } { \\
\@ifl@t@r \fmtversion { 2020/10/01 } { \\
\hook_gput_code:nnn { begindocument } { lualatex-math } { \\
\@@_set_mathchar:NN \mathcomma \, \\
\mathcode `\, = "8000 ~
\end{verbatim}

\footnote{https://groups.google.com/forum/#!topic/de.comp.text.tex/Cputk-AJS5I/discussion}
\hook_gset_rule:nnnn\begindocument { lualatex-math } { voids } { icomma }
{ }
\tl_replace_once:Nnn \@begindocumenthook {
\mathchardef \mathcomma \mathcode `,
}
\@@_set_mathchar:NN \mathcomma ,
}
⟨/package⟩

\section{Implementation of the \LaTeX{}\module{ module}}

For the Lua module, we use the standard \luatexbase-modutils template.

\begin{verbatim}
unpack The function \texttt{unpack} needs to be treated specially as it got moved around in Lua 5.2.
local unpack = unpack or table.unpack

local cctb = \tex\catcodetables or
{\string = \registernumber{\catcodetable@string}}

\end{verbatim}

\begin{verbatim}
print\_class\_fam\_slot\ The function \texttt{print\_class\_fam\_slot} takes one argument which must be a number.
It interprets the argument as a Unicode code point whose mathematical code
is printed in the form \texttt{\textbackslash\Ummathchardef\textbackslash\class\textbackslash\family\textbackslash\slot},
suitable for the right-hand side of \\texttt{\Ummathchardef}.

function lualatex.math.print\_class\_fam\_slot(char)
local code = \tex\mathcode(char)
local class, family, slot = unpack(code)
local result = \string\format("\%i \%i \%i ", class, family, slot)
tex.sprint(cctb.string, result)
end

return lualatex.math
⟨/lua⟩
\end{verbatim}

\section*{Change History}

\begin{verbatim}
v0.1
General: Initial version ........................................ 1

v0.2
General: Added patch for the icomma package .................... 9
\end{verbatim}
v0.3
   General: Patched math group allocation to gain access to all families .......... 5
v0.3a
   General: Updated for changes in l3kernel ........................................ 1
v0.3b
   \@begindocumenthook: Another update for a change in l3kernel .................. 6
v0.3c
   \@@_set_mathchar:NN: l3kernel renamed \lua_now:x to \lua_now_x:n .............. 4
v1.0
   General: Switched to l3docstrip ....................................................... 1
v1.1
   \@@_set_mathchar:NN: Update reasoning why \Umathcharnumdef is not used here ....... 4
   General: Add fix and unit test for amsopn ......................................... 8
v1.1a
   \@@_set_mathchar:NN: l3kernel has (currently) dropped \lua_now_x:n .............. 4
v1.4
   \MT_cramped_internal:Nn: Added \ensuremath to work around issue 11 ............. 9
   General: Removed patch for math group allocation; the kernel itself now supports all available math families ................................................. 5
v1.4a
   \@@_set_mathchar:NN: \lua_now_x:n is back .......................................... 4
   General: Avoid \RequireLuaModule ..................................................... 2
   Load luatexbase only if required .................................................. 2
   Load all of luatexbase ..................................................................... 10
   Pick up new name for string catcode table where available ...................... 10
   Use expl3 versions of LuaTEX math primitives .................................... 2
v1.5
   General: Removed patch for \Mathstrutbox@; amsmath now has a definition usable in Lua\LaTeXX ......................................................... 6
   Removed unused helper macro \@@_char_dim:NN ...................................... 6
   Removed unused Lua function print_fam_slot ..................................... 10
v1.6
   General: Removed patch for \newmcodes@; amsmath now has a definition usable in Lua\LaTeXX ......................................................... 8
v1.7
   \genfrac: Adapt patch to changes in amsmath .................................... 8
v1.8
   \@@_set_mathchar:NN: \lua_now_x:n is now called \lua_now_e .................... 4
   Stop using \.:D control sequences ................................................. 4
   \frac: Stop using \.:D control sequences ............................................ 5, 7
   \genfrac: Stop using \.:D control sequences ....................................... 8
   General: Stop using \.:D control sequences ........................................ 6
   subarray: Stop using \.:D control sequences ...................................... 7
v1.9
   \@begindocumenthook: Don’t patch newer versions of amsmath ................. 6
   \MT_cramped_internal:Nn: Stop using \.:D control sequences .................. 9
   \frac: Adapt to changes in \LaTeXX 2\v kernel .................................. 5
   \\@_00_equal_mathchar: Don’t patch newer versions of amsmath .............. 6
   General: Require 2020 version of \LaTeXX 2\v .................................. 2
Use built-in \TeX \LaTeX hooks if available ........................................... 2
subarray: Don’t patch newer versions of amsmath ................................ 7
Stop using $:D control sequences ...................................................... 7

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.

<table>
<thead>
<tr>
<th>Symbols</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>16, 18</td>
</tr>
<tr>
<td>,</td>
<td>263, 264, 270, 272</td>
</tr>
<tr>
<td>-</td>
<td>108, 110, 115, 123, 126</td>
</tr>
<tr>
<td>&gt;</td>
<td>109, 111, 116, 124, 127</td>
</tr>
<tr>
<td>\afterpackage</td>
<td>70, 87, 112</td>
</tr>
<tr>
<td>\afterpackageorname</td>
<td>86, 126, 218, 260</td>
</tr>
<tr>
<td>\beforepackage</td>
<td>70, 106</td>
</tr>
<tr>
<td>\patchchnnn</td>
<td>40, 131, 220</td>
</tr>
<tr>
<td>\patchchnnn</td>
<td>40, 191, 172, 183</td>
</tr>
<tr>
<td>\restorecatcodex</td>
<td>11, 16</td>
</tr>
<tr>
<td>\setmathcharNN</td>
<td>63, 108, 109, 126, 127, 263, 272</td>
</tr>
<tr>
<td>\temp:w</td>
<td>39, 44, 45, 53</td>
</tr>
<tr>
<td>\above</td>
<td>192, 207</td>
</tr>
<tr>
<td>\over</td>
<td>175, 179, 192, 205</td>
</tr>
<tr>
<td>\begindocumenthook</td>
<td>120, 269</td>
</tr>
<tr>
<td>\ifinner</td>
<td>70, 261</td>
</tr>
<tr>
<td>\ifpackage\later</td>
<td>107, 121, 130, 219</td>
</tr>
<tr>
<td>\ifpackage\loaded</td>
<td>87, 90</td>
</tr>
<tr>
<td>\mathsize</td>
<td>187, 199</td>
</tr>
<tr>
<td>\</td>
<td>20, 27, 28, 29</td>
</tr>
</tbody>
</table>

A

| \addtolength | 72, 75                      |
| \advance | 138, 232, 246               |
| \alignmark | 167                         |
| amsmath (package) | 1, 2, 3, 5, 6, 11, 12    |
| amsopn (package) | 11                         |
| \AtBeginPackageFile | 80                      |
| \AtEndPackage \ AtEndPackageFile | 15               |
| \AtEndPackageFile | 83                         |

B

| \baselineskip | 137, 138, 155             |
| \begin{group} | 93, 175, 190            |
| \begin{group} | 133, 144                 |
| \binom | 2                         |
| \box | 250                        |
| Breitfeld, Peter | 9                      |

C

<p>| \c_@std_equal_machcode_int | 102, 111               |
| \c_@std_minus_machcode_int | 102, 110             |
| \c_group_begin_token | 151, 162              |
| \char_set_machcode:nn | 12                       |
| \char_set_machcode_mach_toggle:x | 18                   |
| \char_set_mathcode:nn | 110, 111               |</p>
<table>
<thead>
<tr>
<th>\textbf{P}</th>
</tr>
</thead>
<tbody>
<tr>
<td>packages:</td>
</tr>
<tr>
<td>amsmath</td>
</tr>
<tr>
<td>amsopn</td>
</tr>
<tr>
<td>etoolbox</td>
</tr>
<tr>
<td>expl3</td>
</tr>
<tr>
<td>filehook</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>\textbf{K}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\kern</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>\textbf{L}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\l3docstrip (package)</td>
</tr>
<tr>
<td>\l3kernel (package)</td>
</tr>
<tr>
<td>\l_@@_equal_mathchar</td>
</tr>
<tr>
<td>\l_@@_minus_mathchar</td>
</tr>
<tr>
<td>\leavevmode</td>
</tr>
<tr>
<td>\lua_now:e</td>
</tr>
<tr>
<td>\luatexbase (package)</td>
</tr>
<tr>
<td>\luatexbase-modutils (package)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>\textbf{M}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\math</td>
</tr>
<tr>
<td>macro-expected (message)</td>
</tr>
<tr>
<td>\mathchardef</td>
</tr>
<tr>
<td>\mathcode</td>
</tr>
<tr>
<td>\mathcomma</td>
</tr>
<tr>
<td>\mathstyle</td>
</tr>
<tr>
<td>mathtools (package)</td>
</tr>
<tr>
<td>messages:</td>
</tr>
<tr>
<td>\luatexbase (package)</td>
</tr>
<tr>
<td>\macro-expected</td>
</tr>
<tr>
<td>\patch-macro</td>
</tr>
<tr>
<td>\wrong-meaning</td>
</tr>
<tr>
<td>module (function)</td>
</tr>
<tr>
<td>\msg_error:n</td>
</tr>
<tr>
<td>\msg_info:n</td>
</tr>
<tr>
<td>\msg_new:n</td>
</tr>
<tr>
<td>\msg_warning:n</td>
</tr>
<tr>
<td>\msg_warning:nn</td>
</tr>
<tr>
<td>\Mt_{cramped internal:Ln}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>\textbf{N}</th>
</tr>
</thead>
<tbody>
<tr>
<td>nath (package)</td>
</tr>
<tr>
<td>\NeedsTeXFormat</td>
</tr>
<tr>
<td>\nuulabbytecode</td>
</tr>
<tr>
<td>\multidelimspace</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>\textbf{O}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\over</td>
</tr>
<tr>
<td>Package</td>
</tr>
<tr>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>icomma</td>
</tr>
<tr>
<td>l3docstrip</td>
</tr>
<tr>
<td>l3kernel</td>
</tr>
<tr>
<td>luatexbase</td>
</tr>
<tr>
<td>luatexbase-modutils</td>
</tr>
<tr>
<td>mathtools</td>
</tr>
<tr>
<td>nath</td>
</tr>
<tr>
<td>unicode-math</td>
</tr>
<tr>
<td>patch-macro (message)</td>
</tr>
<tr>
<td>\prog_do nothing</td>
</tr>
<tr>
<td>print_class_fam_slot (function)</td>
</tr>
<tr>
<td>print_fam_slot (function)</td>
</tr>
<tr>
<td>\ProvidesExpPackage</td>
</tr>
<tr>
<td>\radical</td>
</tr>
<tr>
<td>\relax</td>
</tr>
<tr>
<td>\RequirePackage</td>
</tr>
<tr>
<td>\restoremathtr</td>
</tr>
<tr>
<td>Robertson, Will</td>
</tr>
<tr>
<td>\scan_stop:</td>
</tr>
<tr>
<td>\scriptfont</td>
</tr>
<tr>
<td>\scriptscriptfont</td>
</tr>
<tr>
<td>\scriptstyle</td>
</tr>
<tr>
<td>\setbox</td>
</tr>
<tr>
<td>\skip_net:N</td>
</tr>
<tr>
<td>\str@Equal</td>
</tr>
<tr>
<td>\str@equals</td>
</tr>
<tr>
<td>\str@minus</td>
</tr>
<tr>
<td>\subarray</td>
</tr>
<tr>
<td>\subarray (environment)</td>
</tr>
<tr>
<td>\sys_if_engine_luatex:F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Environment</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>T	extfont</td>
<td>231, 232, 236</td>
</tr>
<tr>
<td>T\textstyle</td>
<td>235</td>
</tr>
<tr>
<td>T\thr@@</td>
<td>140</td>
</tr>
<tr>
<td>T\if_empty:aTF</td>
<td>204</td>
</tr>
<tr>
<td>T\new:N</td>
<td>104, 105</td>
</tr>
<tr>
<td>T\replace_one:Mon</td>
<td>122, 269</td>
</tr>
<tr>
<td>T\token_eq_meaning:IFT</td>
<td>163</td>
</tr>
<tr>
<td>T\token_if_macro:IFT</td>
<td>42</td>
</tr>
<tr>
<td>T\token_to_meaning:IFT</td>
<td>52, 53</td>
</tr>
<tr>
<td>T\token_to_attr:N</td>
<td>47, 52, 58</td>
</tr>
<tr>
<td>T\tw@</td>
<td>137, 138</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Environment</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>U\mathchardef</td>
<td>64</td>
</tr>
<tr>
<td>U\mathcode</td>
<td>115, 116</td>
</tr>
<tr>
<td>U\mathstackdendown</td>
<td>157</td>
</tr>
<tr>
<td>U\mathstacknumup</td>
<td>156</td>
</tr>
<tr>
<td>U\mathstackvapp</td>
<td>159</td>
</tr>
<tr>
<td>unicode-math (package)</td>
<td>I, 2</td>
</tr>
<tr>
<td>unpack (function)</td>
<td>10, 288</td>
</tr>
<tr>
<td>use-c</td>
<td>254</td>
</tr>
<tr>
<td>\usetack</td>
<td>97, 179, 202</td>
</tr>
<tr>
<td>\usetartmath</td>
<td>164</td>
</tr>
</tbody>
</table>