The `latex-lab-amsmath` code*

LaTeX Project

May 4, 2025

Abstract

Contents

1 Introduction **1**

2 The Implementation **1**
2.1 File declaration . . . 2
2.2 Tagpdf support . . . 2
2.3 Measuring . . . 2
2.4 Display environments . . . 2
2.4.1 Tag . . . 2
2.4.2 align & friends . . . 3
2.4.3 `gather` and `gather*` . . . 5
2.4.4 gathered, aligned and alignedat . . . 7
2.4.5 multline and multline* . . . 8
2.5 Cases . . . 11
2.5.1 smallmatrix . . . 11
2.6 The split environment . . . 12
2.7 `\intertext` . . . 17
2.8 `\text` . . . 18
2.9 `\pmb` . . . 19

Index **20**

1 Introduction

This file implements adaptations to the amsmath package needed for the tagging project.

*

2 The Implementation

Better no @@ expansion here . . . ...

```
⟨@@=⟩
⟨∗kernel⟩
```

2.1 File declaration

```
\ProvidesFile{latex-lab-amsmath.ltx}
        [2025-03-07 v0.1f amsmath adaptions]
```

2.2 Tagpdf support

To make the code independent from tagging being loaded and active we load the tagpdf-base package:

```
\RequirePackage{tagpdf-base}
\ExplSyntaxOn
```

2.3 Measuring

When measuring we neither want tagging nor the luamml processing.

`\measuring@true`

```
\def\measuring@true{\let\ifmeasuring@\iftrue\tag_suspend:n{\measuring}\luamml_ignore:}
```

(*End of definition for* `\measuring@true`*. This function is documented on page* ??.)

2.4 Display environments

2.4.1 Tag

The tag/label must be saved, so that it can be reinserted later.

TODO: `\maketag@@@` is perhaps used in places where tagging/luamml handling is not wanted. This must be checked and handled.

`\maketag@@@`

```
\def\maketag@@@#1
  {%
    \ifmeasuring@
      \hbox{\m@th\normalfont#1}%
    \else
      \UseTaggingSocket{math/display/tag/begin}
      \hbox{\m@th\normalfont#1
      \UseTaggingSocket{math/luamml/mtable/tag/save}
      }%
      \UseTaggingSocket{math/display/tag/end}
    \fi
   }
```

(*End of definition for* `\maketag@@@`*. This function is documented on page* ??.)

`\eqref` uses `\tagform@` and so `\maketag@@@` but we do not want this tagging there.

\eqref
\maketag@@@notog

```
\def\maketag@@@notag#1{\hbox{\m@th\normalfont#1}}
\DeclareRobustCommand{\eqref}[1]
  {\textup{\let\maketag@@@\maketag@@@notag\tagform@{\ref{#1}}}}
```

(*End of definition for* `\eqref` *and* `\maketag@@@notog`*. These functions are documented on page* **??**.)

2.4.2 align & friends

Most display environment uses a common command for the end which contains the luamml socket to finalize the mtable.

\common@align@ending

```
\def\common@align@ending {
  \math@cr \black@\totwidth@
  \UseExpandableTaggingSocket {math/luamml/mtable/finalize} {\@currenvir}
  \egroup
  \ifingather@
    \restorealignstate@
    \egroup
    \nonumber
    \ifnum0=‘{\fi\iffalse}\fi
  \else
    \dollardollar@end
  \fi
  \ignorespacesafterend
}
```

(*End of definition for* `\common@align@ending`*. This function is documented on page* **??**.)

Now we redefine the display alignments to use these ending.

```
\renewenvironment{align}{%
  \start@align\@ne\st@rredfalse\m@ne
}{%
  \common@align@ending
}
\renewenvironment{align*}{%
  \start@align\@ne\st@rredtrue\m@ne
}{%
  \common@align@ending
}
\renewenvironment{alignat}{%
  \start@align\z@\st@rredfalse
}{%
  \common@align@ending
}
\renewenvironment{alignat*}{%
  \start@align\z@\st@rredtrue
}{%
  \common@align@ending
}
\renewenvironment{flalign}{%
  \start@align\tw@\st@rredfalse\m@ne
}{%
  \common@align@ending
```

```
}
\renewenvironment{flalign*}{%
  \start@align\tw@\st@rredtrue\m@ne
}{%
  \common@align@ending
}
\renewenvironment{xalignat}{%
  \start@align\@ne\st@rredfalse
}{%
  \common@align@ending
}
\renewenvironment{xalignat*}{%
  \start@align\@ne\st@rredtrue
}{%
  \common@align@ending
}
\renewenvironment{xxalignat}{%
  \start@align\tw@\st@rredtrue
}{%
  \common@align@ending
}
```

And register these environments for the math grabbing.

```
 \math_register_halign_env:nn {align}{}
 \math_register_halign_env:nn {align*}{}
 \math_register_halign_env:nn {alignat}{}
 \math_register_halign_env:nn {alignat*}{}
 \math_register_halign_env:nn {flalign}{}
 \math_register_halign_env:nn {flalign*}{}
 \math_register_halign_env:nn {xalignat}{}
 \math_register_halign_env:nn {xalignat*}{}
 \math_register_halign_env:nn {xxalignat}{}
```

The align preamble (used in `\align@`) needs code for luamml to save the cells.

`\align@preamble`

```
\def\align@preamble{%
   &\hfil
    \strut@
    \setboxz@h
      {
        \@lign
         $
          \m@th\displaystyle{##}
         \ifmeasuring@
          \luamml_ignore:
         \else
          \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
         \fi
         $
      }%
    \ifmeasuring@
      \savefieldlength@
    \else
      \UseTaggingSocket{math/luamml/mtable/finalizecol}{box}
```

```
    \fi
    \set@field
    \tabskip\z@skip
   &\setboxz@h
      {
        \@lign
         $
           \m@th\displaystyle{{}##}
           \ifmeasuring@
           \luamml_ignore:
           \else
           \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
           \fi
         $
      }%
    \ifmeasuring@
      \savefieldlength@
    \else
      \UseTaggingSocket{math/luamml/mtable/finalizecol}{box}
    \fi
    \set@field
    \hfil
    \tabskip\alignsep@
}
```

(*End of definition for* `\align@preamble`. *This function is documented on page* ??.)

`\math@cr@@@align`

```
\def\math@cr@@@align{%
  \ifst@rred\nonumber\fi
  \if@eqnsw \global\tag@true \fi
  \global\advance\row@\@ne
  \add@amps\maxfields@
  \omit
  \kern-\alignsep@
  \iftag@
    \setboxz@h{\@lign\strut@{\make@display@tag}}%
    \place@tag
  \fi
  \UseTaggingSocket{math/luamml/mtable/tag/set}
  \ifst@rred\else\global\@eqnswtrue\fi
  \global\lineht@\z@
  \cr
}
```

(*End of definition for* `\math@cr@@@align`. *This function is documented on page* ??.)

2.4.3 gather and gather*

`\gather@`

```
\def\gather@#1{%
    \ingather@true \let\split\insplit@
    \let\tag\tag@in@align \let\label\label@in@display
    \chardef\dspbrk@context\z@
    \intertext@ \displ@y@ \Let@
```

```
    \let\math@cr@@@\math@cr@@@gather
    \gmeasure@{#1}%
    \global\shifttag@false
    \tabskip\z@skip
    \global\row@\@ne
    \halign to\displaywidth\bgroup
        \strut@
        \setboxz@h
          {
           $\m@th\displaystyle{##}
            \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
           $
          }%
        \UseTaggingSocket{math/luamml/mtable/finalizecol}{box}
        \calc@shift@gather
        \set@gather@field
        \tabskip\@centering
       &\setboxz@h{\strut@{##}}%
        \place@tag@gather
        \UseTaggingSocket{math/luamml/mtable/tag/set}
         \tabskip \iftagsleft@ \gdisplaywidth@ \else \z@skip \span\fi
         \crcr
         #1%
}
```

(*End of definition for* `\gather@`*. This function is documented on page* **??**.)

`\endgather`

```
\def\endgather{
      \math@cr
      \black@ \totwidth@
      \UseExpandableTaggingSocket{math/luamml/mtable/finalize} {gather}
  \egroup
  \dollardollar@end
  \ignorespacesafterend
}
```

(*End of definition for* `\endgather`*. This function is documented on page* **??**.)

The original definition of `gather*` uses `\endgather` but this redirection doesn't work if we alter `gather` so we use the real meaning and add the socket.

`gather*` (*env.*)

```
\renewenvironment{gather*}
 {
   \start@gather\st@rredtrue
 }
 {
    \math@cr
    \black@\totwidth@
    \UseExpandableTaggingSocket{math/luamml/mtable/finalize} {gather}
   \egroup
   \dollardollar@end
   \ignorespacesafterend
 }
```

Register both environments

```
\math_register_halign_env:nn {gather}{}
\math_register_halign_env:nn {gather*}{}
```

2.4.4 gathered, aligned and alignedat

These environments are not grabbed as they are inside other display environments but they need various sockets for luamml support.

`\start@aligned`

```
\renewcommand{\start@aligned}[2]{
  \RIfM@
  \else
    \nonmatherr@ {\begin{\@currenvir}}
  \fi
  \savecolumn@ % Assumption: called inside a group
  \UseTaggingSocket{math/luamml/annotate/false}{}{ \alignedspace@left }
  \ams@start@box {#1} \bgroup
    \maxfields@ #2 \relax
    \ifnum \maxfields@ > \m@ne
      \multiply \maxfields@ \tw@
      \let \math@cr@@@ \math@cr@@@alignedat
      \alignsep@ \z@skip
    \else
      \let \math@cr@@@ \math@cr@@@aligned
      \alignsep@ \minalignsep
    \fi
    \Let@ \chardef \dspbrk@context \@ne
    \default@tag
    \spread@equation % no-op if already called
    \global \column@ \z@
    \ialign \bgroup
      & \column@plus
        \hfil
        \strut@
        $
          \m@th
          \displaystyle
          {##}
          \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
        $
        \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
        \tabskip \z@skip
      & \column@plus
        $
          \m@th
          \displaystyle
          {
            {}
            ##
          }
          \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
        $
        \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
```

```
          \hfil
          \tabskip\alignsep@
        \crcr
        \ams@return@opt@arg
}
```

(*End of definition for* `\start@aligned`. *This function is documented on page* **??**.)

`gathered` (*env.*)

```
\renewenvironment{gathered}[1][c]{%
    \RIfM@\else
        \nonmatherr@{\begin{gathered}}%
    \fi
```

This annotates the space

```
    \UseTaggingSocket{ math/luamml/annotate/false } {}{ \alignedspace@left }
    \ams@start@box{#1}\bgroup
        \Let@ \chardef\dspbrk@context\@ne \restore@math@cr
        \spread@equation
        \ialign\bgroup
            \hfil\strut@$\m@th\displaystyle##
```

This save the cell and then finalize it.

```
            \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
            $
            \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
            \hfil
            \crcr
    \ams@return@opt@arg
}{%
  \endaligned
}
```

`\endaligned`

```
\def\endaligned
 {
      \crcr
      \UseExpandableTaggingSocket{math/luamml/mtable/innertable/save}
    \egroup
    \restorecolumn@
  \egroup
  \UseTaggingSocket{math/luamml/mtable/innertable/finalize}
 }
```

(*End of definition for* `\endaligned`. *This function is documented on page* **??**.)

2.4.5 multline and multline*

`multline*` needs a redefinition before it is registered as it uses `\endmultline`, this must be replaced by the true code.

`multline*` (*env.*)

```
\renewenvironment{multline*}{\start@multline\st@rredtrue}
 {%
  \iftagsleft@ \@xp\lendmultline@ \else \@xp\rendmultline@ \fi
```

```
 \ignorespacesafterend
}
```

And now we register both

```
\math_register_halign_env:nn {multline}{}
\math_register_halign_env:nn {multline*}{}
```

In the internal commands we have to add sockets for alignment attributes

`\multline@`

```
\def\multline@#1{%
    \Let@
    \@display@init{\global\advance\row@\@ne \global\dspbrk@lvl\m@ne}%
    \chardef\dspbrk@context\z@
    \restore@math@cr
    \let\tag\tag@in@align
    \global\tag@false \global\let\raise@tag\@empty
    \mmeasure@{#1}%
    \let\tag\gobble@tag \let\label\@gobble
    \tabskip \if@fleqn \@mathmargin \else \z@skip \fi
    \totwidth@\displaywidth
    \if@fleqn
        \advance\totwidth@-\@mathmargin
    \fi
    \halign\bgroup
        \hbox to\totwidth@{%
            \if@fleqn
                \hskip \@centering \relax
            \else
                \hfil
            \fi
            \strut@
            $\m@th\displaystyle{}##\endmultline@math
            \hfil
        }%
        \crcr
        \if@fleqn
            \hskip-\@mathmargin
            \def\multline@indent{\hskip\@mathmargin}% put it back
        \else
            \hfilneg
            \def\multline@indent{\hskip\multlinegap}%
        \fi
        \iftagsleft@
            \iftag@
                \begingroup
                    \ifshifttag@
                        \rlap{\vbox{%
                                \normalbaselines
                                \hbox{%
                                    \strut@
                                    \make@display@tag
                                }%
                                \vbox to\lineht@{}%
                                \raise@tag
```

```
                        }}%
                        \multline@indent
                    \else
                        \setbox\z@\hbox{\make@display@tag}%
                        \dimen@\@mathmargin \advance\dimen@-\wd\z@
                        \ifdim\dimen@<\multlinetaggap
                          \dimen@\multlinetaggap
                        \fi
                        \box\z@ \hskip\dimen@\relax
                    \fi
                \endgroup
            \else
                \multline@indent
            \fi
        \else
            \multline@indent
        \fi
    \ifmeasuring@ \else
      \UseTaggingSocket{math/luamml/mtable/aligncol} {left}
    \fi
    #1%
    \ifmeasuring@ \else
      \UseTaggingSocket{math/luamml/mtable/aligncol} {right}
    \fi
}
```

(*End of definition for* `\multline@`. *This function is documented on page* **??**.)

Luckily, `multline` uses `\endmultline@math` in exactly the spot where we have to set the flag. Less luckily, `\endmultline@math` sometimes get overwritten for the last line. But that isn't much of a problem since we want special behavior there anyway.

`\endmultline@math`

```
\def\endmultline@math
 {
   \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
   $
   \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
 }
```

(*End of definition for* `\endmultline@math`. *This function is documented on page* **??**.)

`\rendmultline@`

```
\def\rendmultline@{%
    \iftag@
```

we need to surround the math token with tagging sockets.

```
      \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }
      $
      \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
        \let\endmultline@math\relax
            \ifshifttag@
                \hskip\multlinegap
                \llap{\vtop{%
                    \raise@tag
                    \normalbaselines
```

```
                    \setbox\@ne\null
                    \dp\@ne\lineht@
                    \box\@ne
                    \hbox{\strut@\make@display@tag}%
                }}%
            \else
                \hskip\multlinetaggap
                \make@display@tag
            \fi
```

Here we set the tag TODO: is that sensible with multline? Where is the tag saved?

```
    \UseTaggingSocket{math/luamml/mtable/tag/set}
    \else
        \hskip\multlinegap
    \fi
    \hfilneg
        \math@cr
```

Now we finalize the mtable.

```
        \UseExpandableTaggingSocket {math/luamml/mtable/finalize} {multline}
  \egroup
  \dollardollar@end
}
```

(*End of definition for* `\rendmultline@`. *This function is documented on page* **??**.)

And something similar for the left version

`\lendmultline@`

```
\def\lendmultline@{%
        \hfilneg
        \hskip\multlinegap
        \math@cr
        \UseExpandableTaggingSocket {math/luamml/mtable/finalize} {multline}
    \egroup
    \dollardollar@end
}
```

(*End of definition for* `\lendmultline@`. *This function is documented on page* **??**.)

2.5 Cases

`env@cases`

```
\def\env@cases{%
  \let\@ifnextchar\new@ifnextchar
  \left\lbrace
  \def\arraystretch{1.2}%
  \array{@{}l@{\quad}l@{\luamml_ignore:}}%
}
```

(*End of definition for* `env@cases`. *This function is documented on page* **??**.)

2.5.1 smallmatrix

`smallmatrix` (*env.*)

```
\renewenvironment {smallmatrix} {
  \UseTaggingSocket{ math/luamml/annotate/false } {} { \null\, }
  \vcenter \bgroup
    \Let@
    \restore@math@cr
    \default@tag
    \baselineskip 6 \ex@
    \lineskip 1.5 \ex@
    \lineskiplimit \lineskip
    \ialign \bgroup
        \hfil
        $
        \m@th
        \scriptstyle
        ##
        % No \scriptsize here since we want to add the mstyle nodes
        \UseTaggingSocket{math/luamml/save/nn}{ {} {mtd} }
        $
        \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
        \hfil
      &&
        \thickspace
        \hfil
        $
        \m@th
        \scriptstyle
        ##
        % No \scriptsize here since we want to add the mstyle nodes
        \UseTaggingSocket{math/luamml/save/nn}{ {} {mtd} }
        $
        \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}
        \hfil
      \crcr
}{%
      \crcr
      \UseExpandableTaggingSocket{math/luamml/mtable/smallmatrix/save}
    \egroup
  \egroup
  \UseTaggingSocket{math/luamml/mtable/innertable/finalize}
  \UseTaggingSocket{ math/luamml/annotate/false } {}{ \, }
}
```

2.6 The split environment

The split environment is not trivial to handle as it has a rather convoluted implementation in amsmath: depending on in which display environment it is embedded it takes different branches, which makes it difficult to finalize the mtable.

The following patches work with leqno and reqno if the (default) `centertags` are used. The currently fail (the structure is wrong) if the option `tbtags` is used. The

alignment of the cells is currently not handled. A simple debug command until everything is sorted.

__math_split_debug_typeout:n

```
\cs_new:Npn\__math_split_debug_typeout:n#1{\use_none:n{#1}}
```

(*End of definition for* `\__math_split_debug_typeout:n`.)

We need to detect if `\gather@split` has been used or not

\l__math_gathersplit_bool

```
\bool_new:N\l__math_gathersplit_bool
```

(*End of definition for* `\l__math_gathersplit_bool`.)

At first a redefinition of the main environment. Here we only have to add the saving command for the inner table:

split (*env.*)

```
\renewenvironment{split}{%
  \__math_split_debug_typeout:n {begin~split}
  \if@display
    \ifinner
       \@xp\@xp\@xp\split@aligned
    \else
      \ifst@rred \else \global\@eqnswtrue \fi
    \fi
  \else
    \let\endsplit\@empty \@xp\collect@body\@xp\split@err
  \fi
  \collect@body\gather@split
}{\__math_split_debug_typeout:n{end~split}%
     \crcr
    \UseExpandableTaggingSocket{math/luamml/mtable/innertable/save}%
    \egroup
  \egroup
  \iftagsleft@ \@xp\lendsplit@ \else \@xp\rendsplit@ \fi
}
```

In `\gather@split` we have to add the finalization socket. We also set the boolean to true so that we can detect if the finalization has already happened. Perhaps this could be done in the luamml code instead?

\gather@split

```
\def\gather@split#1#2#3{
  \__math_split_debug_typeout:n{gather@split}%
  \@xp\endgroup \reset@equation % math@cr will handle equation numbering
  \iftag@
     \toks@\@xp{\df@tag}%
     \edef\split@tag{%
       \gdef\@nx\df@tag{\the\toks@}%
       \global\@nx\tag@true \@nx\nonumber
     }%
  \else \let\split@tag\@empty
  \fi
  \bool_set_true:N\l__math_gathersplit_bool
  \spread@equation
```

```
  \vcenter\bgroup
    \gather@{\split@tag
    \begin{split}#1\end{split}}%
    \def\endmathdisplay@a{%
      \__math_split_debug_typeout:n{endmathdisplay@a}
      \__math_split_debug_typeout:n{finalize~innertable~endmathdisplay@a}
      \math@cr
      \black@ \totwidth@
      \egroup
      \egroup
      \UseExpandableTaggingSocket{math/luamml/mtable/innertable/finalize}%
    }%
}
```

(*End of definition for* `\gather@split`*. This function is documented on page* **??**.)

`\insplit@` In `\insplit@` we have to add the sockets which store the cell content.

```
\def\insplit@{\__math_split_debug_typeout:n{insplit@}%
  \global\setbox\z@\vbox\bgroup
    \Let@ \chardef\dspbrk@context\@ne \restore@math@cr
    \default@tag % disallow use of \tag here
    \ialign\bgroup
      \hfil
      \strut@
      $\m@th\displaystyle {##}%
      \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }%
      $%
      \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}%
      &$\m@th\displaystyle {{}##}%
      \UseTaggingSocket{math/luamml/save/nNn}{ {} \displaystyle {mtd} }%
      $%
      \UseTaggingSocket{math/luamml/mtable/finalizecol}{last}%
      \hfill % Why not \hfil?---dmj, 1994/12/28
      \crcr
}
```

(*End of definition for* `\insplit@`*. This function is documented on page* **??**.)

And now the difficult part. Depending on the options leqno/reqno `\lendsplit@` or `\rendsplit@` are used for the typesetting and the inner table must be finalized here in case this hasn't happen yet. This must be tested with the boolean from `\gather@split`

```
\def\lendsplit@{%
    \global\setbox9\vtop{\unvcopy\z@}%
    \ifinalign@
        \setbox\@ne\vbox{%
            \unvcopy\z@
            \global\setbox8\lastbox
        }%
        \setbox\@ne\hbox{%
            \unhcopy8%
            \unskip
            \setbox\tw@\lastbox
            \unskip
            \global\setbox\thr@@\lastbox
        }%
```

```
        \__math_split_debug_typeout:n{lendsplit@/aligncase}
        \ifctagsplit@
         \__math_split_debug_typeout:n{lendsplit@/aligncase/centertags}
            \gdef\split@{%
                \hbox to\wd\thr@@{}%
               &\vcenter{\vbox{\moveleft\wd\thr@@\box9}}%
               \__math_split_debug_typeout:n{finalize~innertable~aligncase}
               \UseTaggingSocket{math/luamml/mtable/innertable/finalize}%
            }%
        \else
        %TODO, not correct yet.
         \__math_split_debug_typeout:n{lendsplit@/aligncase/tbtags}
            \gdef\split@{%
                \hbox to\wd\thr@@{}%
               &\vbox{\moveleft\wd\thr@@\box9}%
               \__math_split_debug_typeout:n{finalize~innertable~aligncase}
               \UseTaggingSocket{math/luamml/mtable/innertable/finalize}%
            }%
        \fi
    \else
```

If not in align we need to test for `\gather@split`

```
        \ifctagsplit@
           \bool_if:NTF \l__math_gathersplit_bool
            {
              \__math_split_debug_typeout:n{lendsplit/equationcase/centertags}
              \gdef\split@%
               {\UseTaggingSocket{math/luamml/annotate/false}{}{\vcenter{\box9}}}
            }
            {
              \__math_split_debug_typeout:n {lendsplit/gathercase/centertags}
              \gdef\split@{\vcenter{\box9}%
              \__math_split_debug_typeout:n {finalize~innertable~gathercase}
              \UseTaggingSocket{math/luamml/mtable/innertable/finalize}}
            }
        \else
          % TODO tbtags not correct yet.
           \bool_if:NTF \l__math_gathersplit_bool
            {
              \__math_split_debug_typeout:n {lendsplit/equationcase/tbtags}
              \gdef\split@%
               {\UseTaggingSocket{math/luamml/annotate/false}{}{\box9}}
            }
            {
              \__math_split_debug_typeout:n {lendsplit/gathercase/tbtags}
              \gdef\split@{
                \box9%
                \__math_split_debug_typeout:n {finalize~innertable~gathercase}
                \UseTaggingSocket{math/luamml/mtable/innertable/finalize}}
            }
        \fi
    \fi
    \aftergroup\split@
}
```

\rendsplit@ And more or less the same for the \rendsplit@ environment.

```
\def\rendsplit@{%
    \ifinalign@
        \global\setbox9 \vtop{%
            \unvcopy\z@
            \global\setbox8 \lastbox
            \unskip
        }%
        \setbox\@ne\hbox{%
            \unhcopy8
            \unskip
            \global\setbox\tw@\lastbox
            \unskip
            \global\setbox\thr@@\lastbox
        }%
        \ifctagsplit@
            \gdef\split@{%
                \hbox to\wd\thr@@{}
                &\vcenter{\vbox{\moveleft\wd\thr@@\boxz@}}%
            \__math_split_debug_typeout:n {rendsplit/aligncase/centertags}
            \__math_split_debug_typeout:n {finalize~innertable~aligncase}
            \UseTaggingSocket{math/luamml/mtable/innertable/finalize}
            }%
        \else
           \__math_split_debug_typeout:n{rendsplit@/aligncase/tbtags}
           %TODO tbtags is not correct yet
            \global\setbox7 \hbox{\unhbox\tw@\unskip}%
            \gdef\split@{%
                \global\@tempcnta\column@
               &\setboxz@h{}%
                \savetaglength@
                \global\advance\row@\@ne
                \vbox{\moveleft\wd\thr@@\box9}%
               \crcr
                \noalign{\global\lineht@\z@}%
                \add@amps\@tempcnta
                \UseTaggingSocket{math/luamml/annotate/false}{}{\box\thr@@}
               &\box7
                 \__math_split_debug_typeout:n {finalize~innertable~aligncase}
                 \UseTaggingSocket{math/luamml/mtable/innertable/finalize}%
            }%
        \fi
```

and again the if we are not in align we need to test for \gathersplit

```
    \else
        \ifctagsplit@
          \bool_if:NTF \l__math_gathersplit_bool
            {
              \__math_split_debug_typeout:n {rendsplit/equationcase/centertags}
              \gdef\split@%
               {\UseTaggingSocket{math/luamml/annotate/false}{}{\vcenter{\boxz@}}}
            }
            {
              \__math_split_debug_typeout:n  {rendsplit/gathercase/centertags}
```

```
              \gdef\split@{\vcenter{\boxz@}%
               \__math_split_debug_typeout:n {finalize~innertable~gathercase}
               \UseTaggingSocket{math/luamml/mtable/innertable/finalize}}
            }
        \else
           \bool_if:NTF \l__math_gathersplit_bool
            {
              \__math_split_debug_typeout:n {rendsplit/equationcase/tbtags}
              \gdef\split@{%
                \UseTaggingSocket{math/luamml/annotate/false}{}{\boxz@}}
            }
            {
              \__math_split_debug_typeout:n {rendsplit/gathercase/tbtags}
              \gdef\split@{%
                \boxz@
                \UseTaggingSocket{math/luamml/mtable/innertable/finalize}}
            }
        \fi
    \fi
    \aftergroup\split@
}
```

(*End of definition for* `\rendsplit@`*. This function is documented on page* **??**.)

2.7 \intertext

The `\intertext` command errors with active tagging as it is processed twice which leads to duplicated structures.

`\intertext@` TODO: review and add sockets!

```
\def\intertext@{%
  \def\intertext##1{%
    \ifvmode\else\\\@empty\fi
    \noalign{%
      \penalty\postdisplaypenalty\vskip\belowdisplayskip
      \vbox{
```

Stop tagging when measuring:

```
       \ifmeasuring@\tag_suspend:n{\measuring}\fi
       \normalbaselines
        \ifdim\linewidth=\columnwidth
        \else \parshape\@ne \@totalleftmargin \linewidth
        \fi
```

End the previous mc:

```
        \tag_mc_end_push:
```

We are already in a par so we change now to text:

```
        \tagpdfsetup{para/tag=P}%
```

TODO why `\tagpdfparaOn` needed?

```
        \tagpdfparaOn
        \noindent\ignorespaces##1\par
```

Restart the MC

```
          \tag_mc_begin_pop:n{}%
        \penalty\predisplaypenalty\vskip\abovedisplayskip%
      }%
}}
```

(*End of definition for* `\intertext@`. *This function is documented on page* ??.)

2.8 \text

The `\text` command uses `\mathchoice` which "typesets" the argument four times. This makes it quite problematic for tagging. Without precautions structure objects would be created four times and would get MC-chunks as kids that don't really exist. amsmath contains a switch that allows to execute code only in the first (displaymath) branch, but that isn't usable here. At first because we don't know if the first branch creates the same structure as the one that is actually used. At second because the engines executes some commands like `\label` and `\pdfannot` only at shipout from the branch that really was used. So we would get structure data from one `\mathchoice`-branch and MC-labels and links from another one and that gets very messy.

We therefore have to avoid that tagging is active in unused branches. In pdflatex it is not possible to detect the mathstyle before, so we use a label. With lualatex is is possible to redefine `\text` not to use `\mathchoice`

`\text@`

```
\AddToHook{package/amstext/after}
 {
```

currently amsmath is loaded in a begindocument hook, so this test is fine. If amstext is loaded earlier (in the kernel), this needs perhaps a change.

```
   \tag_if_active:T
    {
      \sys_if_engine_luatex:TF
       {
         \def\text@#1{
          \tag_socket_use:nnn {math/luamml/hbox}{}
          {{%
           \ifcase\mathstyle
           \hbox{{#1}}\or
           \hbox{{#1}}\or
           \hbox{{#1}}\or
           \hbox{{#1}}\or
           \hbox{{\let\f@size\sf@size\selectfont#1}}\or
           \hbox{{\let\f@size\sf@size\selectfont#1}}\or
           \hbox{{\let\f@size\ssf@size\selectfont#1}}\or
           \hbox{{\let\f@size\ssf@size\selectfont#1}}\or
           \ERROR
           \fi
           \check@mathfonts
         }}}
       }
       {
         \def\text@#1
          {{
```

```
            \int_gincr:N\g__math_mathchoice_int
            \tag_suspend:n{\text@}
            \mathchoice
             {
              \__math_tag_if_mathstyle:en{mathchoice-\int_use:N\g__math_mathchoice_int}{0}
              \textdef@\displaystyle\f@size{#1}
             }
             {
              \__math_tag_if_mathstyle:en{mathchoice-\int_use:N\g__math_mathchoice_int}{2}
              \textdef@\textstyle\f@size{\firstchoice@false #1}
             }
             {
              \__math_tag_if_mathstyle:en{mathchoice-\int_use:N\g__math_mathchoice_int}{4}
              \textdef@\textstyle\sf@size{\firstchoice@false #1}
             }
             {
               \__math_tag_if_mathstyle:en{mathchoice-\int_use:N\g__math_mathchoice_int}{6}
               \textdef@\textstyle \ssf@size{\firstchoice@false #1}
             }
             \check@mathfonts
           }}
       }
    }
}
```

(*End of definition for* `\text@`. *This function is documented on page* **??**.)

2.9 \pmb

`\pmb` prints its argument three times. For tagging we must mark two of occurrences as artifact. For luatex the attributes in the box must be reset, for this we switch to expl3-boxes.

`\pbm@@`
`\pmb@`

```
\AddToHook{package/amsbsy/after}
 {
   \def\pmb@@#1#2#3{\leavevmode\hbox_set:Nn\l_@@_tmpa_box{xxx#3}
      \dimen@-\box_wd:N\l_@@_tmpa_box
      \kern-.5\ex@\box_use:N\l_@@_tmpa_box
      \tag_mc_end:\tag_mc_begin:n{artifact}
      \tag_mc_reset_box:N\l_@@_tmpa_box
      \kern\dimen@\kern.25\ex@\raise.4\ex@\box_use:N\l_@@_tmpa_box
      \kern\dimen@\kern.25\ex@\box_use_drop:N\l_@@_tmpa_box
      \tag_mc_end:\tag_mc_begin:n{}
    }
   \def\pmb@#1#2{\hbox_set:Nn\l_@@_tmpa_box{$\m@th#1{#2}$}
     \setboxz@h{$\m@th#1\mkern.5mu$}\pmbraise@\wdz@
     \binrel@{#2}
     \dimen@-\box_wd:N\l_@@_tmpa_box
     \binrel@@{
       \mkern-.8mu\box_use:N\l_@@_tmpa_box
       \tag_mc_end:\tag_mc_begin:n{artifact}
       \tag_mc_reset_box:N\l_@@_tmpa_box
       \kern\dimen@\mkern.4mu\raise\pmbraise@\box_use:N\l_@@_tmpa_box
```

```
738         \kern\dimen@\mkern.4mu\box_use_drop:N\l_@@_tmpa_box
739         \tag_mc_end:\tag_mc_begin:n{}
740         }
741     }
742   }
```

(*End of definition for* `\pbm@@` *and* `\pmb@`. *These functions are documented on page* **??**.)

```
743 \ExplSyntaxOff
```

744 ⟨/kernel⟩

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

Symbols

`\,` 406, 444
`@@` commands:
`\l_@@_tmpa_box` . 720, 721, 722, 724, 725, 726, 729, 732, 734, 736, 737, 738
`\\` 651

A

`\abovedisplayskip` 665
`\AddToHook` 668, 718
`\advance` 137, 287, 297, 334, 607
`\aftergroup` 575, 647
`\array` 403
`\arraystretch` 402

B

`\baselineskip` 411
`\begin` 204, 252, 482
`\begingroup` 320
`\belowdisplayskip` 653
`\bgroup` 160, 208, 222, 255, 258, 299, 407, 414, 480, 494, 497
bool commands:
`\bool_if:NTF` 546, 560, 620, 633
`\bool_new:N` 447
`\bool_set_true:N` 478
bool internal commands:
`\l__math_gathersplit_bool` 447, 478, 546, 560, 620, 633
`\box` 338, 374, 530, 539, 550, 554, 564, 569, 608, 612, 613
box commands:
`\box_use:N` 722, 725, 734, 737
`\box_use_drop:N` 726, 738
`\box_wd:N` 721, 732

C

`\chardef` 153, 218, 256, 288, 495
`\columnwidth` 657
`\cr` 148
`\crcr` 176, 247, 264, 271, 310, 437, 439, 461, 509, 609
cs commands:
`\cs_new:Npn` 446

D

`\DeclareRobustCommand` 21
`\def` 7, 8, 20, 23, 91, 134, 150, 179, 269, 285, 313, 316, 355, 361, 391, 399, 402, 467, 483, 493, 511, 577, 649, 650, 674, 692, 720, 729
`\displaystyle` . . 98, 102, 117, 121, 164, 165, 228, 230, 237, 242, 259, 260, 307, 357, 363, 500, 501, 504, 505, 699
`\displaywidth` 160, 295
`\dp` 373

E

`\edef` 472
`\egroup` 26, 29, 183, 195, 273, 275, 388, 396, 441, 442, 463, 464, 488, 489
`\else` 12, 32, 101, 108, 120, 127, 146, 175, 203, 214, 251, 280, 294, 303, 314, 332, 341, 344, 347, 351, 377, 382, 453, 454, 456, 465, 476, 534, 544, 558, 599, 618, 632, 651, 658
`\end` 482
`\endaligned` 267, 269
`\endgather` *6*, 179
`\endgroup` 340, 469
`\endmultline` *8*
`\endsplit` 457

environments:
gather* 187
gathered 250
multline* 278
smallmatrix 405
split 448
\eqref 2, 20
\ERROR 686
\ExplSyntaxOff 743
\ExplSyntaxOn 6

F

\fi 18, 31, 34, 103, 110, 122, 129, 135, 136, 144, 146, 175, 205, 217, 253, 280, 294, 298, 305, 317, 337, 339, 343, 346, 349, 353, 380, 384, 454, 455, 458, 465, 477, 543, 573, 574, 617, 645, 646, 651, 655, 659, 687

G

gather* (env.) 187
gathered (env.) 250
\gathersplit 16
\gdef 473, 528, 537, 549, 554, 563, 568, 592, 603, 623, 628, 636, 641
\global 136, 137, 146, 147, 157, 159, 221, 287, 291, 454, 474, 494, 512, 516, 523, 579, 581, 587, 589, 602, 604, 607, 610

H

\halign 160, 299
\hbox 11, 14, 20, 300, 324, 333, 375, 518, 529, 538, 584, 593, 602, 678, 679, 680, 681, 682, 683, 684, 685
hbox commands:
\hbox_set:Nn 720, 729
\hfil 92, 131, 224, 245, 259, 263, 304, 308, 415, 424, 427, 436, 498, 508
\hfill 508
\hfilneg 315, 385, 392
\hskip 302, 312, 313, 316, 338, 368, 378, 383, 393

I

\ialign 222, 258, 414, 497
\ifcase 677
\ifdim 335, 657
\iffalse 31
\ifinner 451
\ifnum 31, 210
\iftrue 7
\ifvmode 651
\ignorespaces 663
\ignorespacesafterend . . 35, 185, 197, 281
int commands:
\int_gincr:N 694
\int_use:N 698, 702, 706, 710
int internal commands:
\g__math_mathchoice_int 694, 698, 702, 706, 710
\intertext 17, 650

K

\kern 140, 722, 725, 726, 737, 738

L

\label 18, 152, 293
\lastbox 516, 521, 523, 581, 587, 589
\lbrace 401
\leavevmode 720
\left 401
\let 7, 22, 151, 152, 155, 212, 215, 290, 291, 293, 366, 400, 457, 476, 682, 683, 684, 685
\lineskip 412, 413
\lineskiplimit 413
\linewidth 657, 658
\llap 369
luamml commands:
\luamml_ignore: 7, 100, 119, 403

M

math commands:
\math_register_halign_env:nn 82, 83, 84, 85, 86, 87, 88, 89, 90, 199, 200, 283, 284
math internal commands:
__math_split_debug_typeout:n 446, 446, 449, 460, 468, 484, 485, 493, 525, 527, 531, 536, 540, 548, 553, 555, 562, 567, 570, 595, 596, 600, 614, 622, 627, 629, 635, 640
__math_tag_if_mathstyle:nn 698, 702, 706, 710
\mathchoice 18, 696
\mathstyle 677
\measuring 7, 655
\minalignsep 216
\mkern 730, 734, 737, 738
\moveleft 530, 539, 594, 608
\multiply 211
multline* (env.) 278
\multlinegap 316, 368, 383, 393
\multlinetaggap 335, 336, 378

N

\noalign 610, 652
\noindent 663
\nonumber 30, 135, 474

\normalbaselines 323, 371, 656
\normalfont 11, 14, 20
\null 372, 406

O

\omit 139
\or . . 678, 679, 680, 681, 682, 683, 684, 685

P

\par 663
\parshape 658
\pdfannot *18*
\penalty 653, 665
\pmb *19*
\postdisplaypenalty 653
\predisplaypenalty 665
\ProvidesFile 3

Q

\quad 403

R

\raise 725, 737
\ref 22
\relax 209, 302, 338, 366
\renewcommand 201
\renewenvironment . 37, 42, 47, 52, 57, 62, 67, 72, 77, 187, 250, 278, 405, 448
\RequirePackage 5
\rlap 322

S

\scriptsize 420, 432
\scriptstyle 418, 430
\selectfont 682, 683, 684, 685
\setbox 333, 372, 494, 512, 514, 516, 518, 521, 523, 579, 581, 584, 587, 589, 602
smallmatrix (env.) 405
\span 175
split (env.) 448
\split 151
sys commands:
\sys_if_engine_luatex:TF 672

T

\tabskip 112, 132, 158, 171, 175, 233, 246, 294
\tag 152, 290, 293, 496
tag commands:
\tag_if_active:TF 670
\tag_mc_begin:n . . . 723, 727, 735, 739
\tag_mc_begin_pop:n 664
\tag_mc_end: 723, 727, 735, 739
\tag_mc_end_push: 660
\tag_mc_reset_box:N 724, 736
\tag_socket_use:nnn 675
\tag_suspend:n 7, 655, 695
\tagpdfparaOn *17*, 662
\tagpdfsetup 661
TEX and LATEX 2ε commands:
\@centering 171, 302
\@currenvir 25, 204
\@display@init 287
\@empty 291, 457, 476, 651
\@eqnswtrue 146, 454
\@gobble 293
\@ifnextchar 400
\@lign 96, 115, 142
\@mathmargin . . 294, 297, 312, 313, 334
\@ne 38, 43, 68, 73, 137, 159, 218, 256, 287, 372, 373, 374, 495, 514, 518, 584, 607, 658
\@nx 473, 474
\@tempcnta 604, 611
\@totalleftmargin 658
\@xp 280, 452, 457, 465, 469, 471
\add@amps 138, 611
\align@ *4*
\align@preamble 91
\alignedspace@left 207, 254
\alignsep@ 132, 140, 213, 216, 246
\ams@return@opt@arg 248, 265
\ams@start@box 208, 255
\binrel@ 731
\binrel@@ 733
\black@ 24, 181, 193, 487
\boxz@ 594, 624, 628, 637, 642
\calc@shift@gather 169
\check@mathfonts 688, 713
\collect@body 457, 459
\column@ 221, 604
\column@plus 223, 234
\common@align@ending 23, 40, 45, 50, 55, 60, 65, 70, 75, 80
\default@tag 219, 410, 496
\df@tag 471, 473
\dimen@ 334, 335, 336, 338, 721, 725, 726, 732, 737, 738
\displ@y@ 154
\dollardollar@end 33, 184, 196, 389, 397
\dspbrk@context 153, 218, 256, 288, 495
\dspbrk@lvl 287
\endmathdisplay@a 483
\endmultline@math . . *10*, 307, 355, 366
\env@cases 399
env@cases 399
\ex@ 411, 412, 722, 725, 726
\f@size . . . 682, 683, 684, 685, 699, 703
\firstchoice@false 703, 707, 711
\gather@ 150, 481

\gather@split *12–15*, 459, 467
\gdisplaywidth@ 175
\gmeasure@ 156
\gobble@tag 293
\if@display 450
\if@eqnsw 136
\if@fleqn 294, 296, 301, 311
\ifctagsplit@ 526, 545, 591, 619
\ifinalign@ 513, 578
\ifingather@ 27
\ifmeasuring@ 7, 10, 99, 106, 118, 125, 347, 351, 655
\ifshifttag@ 321, 367
\ifst@rred 135, 146, 454
\iftag@ 141, 319, 362, 470
\iftagsleft@ 175, 280, 318, 465
\ingather@true 151
\insplit@ *14*, 151, 493
\intertext@ 154, 649
\label@in@display 152
\lendmultline@ 280, 391
\lendsplit@ *14*, 465, 511
\Let@ 154, 218, 256, 286, 408, 495
\lineht@ 147, 328, 373, 610
\m@ne 38, 43, 58, 63, 210, 287
\m@th 11, 14, 20, 98, 117, 164, 227, 236, 259, 307, 417, 429, 500, 504, 729, 730
\make@display@tag 142, 326, 333, 375, 379
\maketag@@@ *2*, 8, 22
\maketag@@@notag 20, 22
\maketag@@@notog 20
\math@cr .. 24, 180, 192, 386, 394, 486
\math@cr@@@ 155, 212, 215
\math@cr@@@align 134
\math@cr@@@aligned 215
\math@cr@@@alignedat 212
\math@cr@@@gather 155
\maxfields@ 138, 209, 210, 211
\measuring@true 7
\mmeasure@ 292
\multline@ 285
\multline@indent 313, 316, 331, 342, 345
\new@ifnextchar 400
\nonmatherr@ 204, 252
\pbm@@ 718
\place@tag 143
\place@tag@gather 173
\pmb@ 718
\pmb@@ 720
\pmbraise@ 730, 737
\raise@tag 291, 329, 370
\rendmultline@ 280, 361
\rendsplit@ *14*, *15*, 465, 577
\reset@equation 469
\restore@math@cr ... 256, 289, 409, 495
\restorealignstate@ 28
\restorecolumn@ 274
\RIfM@ 202, 251
\row@ 137, 159, 287, 607
\savecolumn@ 206
\savefieldlength@ 107, 126
\savetaglength@ 606
\set@field 111, 130
\set@gather@field 170
\setboxz@h 94, 113, 142, 162, 172, 605, 730
\sf@size 682, 683, 707
\shifttag@false 157
\split@ 528, 537, 549, 554, 563, 568, 575, 592, 603, 623, 628, 636, 641, 647
\split@aligned 452
\split@err 457
\split@tag 472, 476, 481
\spread@equation 220, 257, 479
\ssf@size 684, 685, 711
\st@rredfalse 38, 48, 58, 68
\st@rredtrue 43, 53, 63, 73, 78, 189, 278
\start@align 38, 43, 48, 53, 58, 63, 68, 73, 78
\start@aligned 201
\start@gather 189
\start@multline 278
\strut@ 93, 142, 161, 172, 225, 259, 306, 325, 375, 499
\tag@false 291
\tag@in@align 152, 290
\tag@true 136, 474
\tagform@ *2*, 22
\text@ 668
\textdef@ 699, 703, 707, 711
\thr@@ 523, 529, 530, 538, 539, 589, 593, 594, 608, 612
\toks@ 471, 473
\totwidth@ 24, 181, 193, 295, 297, 300, 487
\tw@ 58, 63, 78, 211, 521, 587, 602
\wdz@ 730
\z@ 48, 53, 147, 153, 221, 288, 333, 334, 338, 494, 512, 515, 580, 610
\z@skip ... 112, 158, 175, 213, 233, 294
\text *18*
\textstyle 703, 707, 711
\textup 22
\the 473
\thickspace 426

U

\unhbox 602
\unhcopy 519, 585
\unskip 520, 522, 582, 586, 588, 602
\unvcopy 512, 515, 580
use commands:
  \use_none:n 446
\UseExpandableTaggingSocket 25, 182, 194, 272, 387, 395, 440, 462, 490
\UseTaggingSocket 13, 15, 17, 102, 109, 121, 128, 145, 165, 168, 174, 207, 230, 232, 242, 244, 254, 260, 262, 276, 348, 352, 357, 359, 363, 365, 381, 406, 421, 423, 433, 435, 443, 444, 501, 503, 505, 507, 532, 541, 550, 556, 564, 571, 597, 612, 615, 624, 630, 637, 643

V

\vbox 322, 328, 494, 514, 530, 539, 594, 608, 654
\vcenter 407, 480, 530, 550, 554, 594, 624, 628
\vskip 653, 665
\vtop 369, 512, 579

W

\wd . . 334, 529, 530, 538, 539, 593, 594, 608