

The **latex-lab-text** package

Changes and additions to the kernel related to the tagging of various small text commands

L^AT_EX Project^{*}
v0.85j 2026-09-30

Abstract

1 Introduction

This module contains small changes to improve the tagging of (inline) text commands.

While the tagging of the L^AT_EX logo is quite straightforward, tagging of `\emph` as **EM** can have unwanted side-effect as both tags are inline tags and so not every inner structure is allowed. Probably both commands will need an optional argument or a starred variant which suppress the tagging.

2 Implementation

```
1 <*package>
2 <@@=tag>

3 \ProvidesExplPackage {latex-lab-testphase-text} {\ltlabtextdate} {\ltlabtextversion}
4 {Code related to the tagging of various small text commands}
```

2.1 The L^AT_EX and T_EX logo

This uses the generic hooks, so that it catches also other definitions. For `\LaTeX` we stop tagging to avoid that the inner `\TeX` leads to a double tagging.

```
5 \AddToHook{cmd/LaTeX/before}
6 {
7   \mode_leave_vertical:
8   \UseTaggingSocket{inline/begin}
9   {tag=\UseStructureName{span},actualtext=LaTeX,phoneme=['la:tɛx]}
10  \tag_suspend:n{\LaTeX}
11 }
12 \AddToHook{cmd/LaTeX/after}
13 {
14   \tag_resume:n{\LaTeX}
15   \UseTaggingSocket{inline/end}
```

^{*}Initial implementation done by Ulrike Fischer

```

16 }
17 \AddToHook{cmd/TeX/before}
18 {
19   \mode_leave_vertical:
20   \UseTaggingSocket{inline/begin}
21     {tag=\UseStructureName{span},actualtext=TeX}
22 }
23 \AddToHook{cmd/TeX/after}
24 {
25   \UseTaggingSocket{inline/end}
26 }

```

2.2 Emphasizing

We tag `\emph` and `\strongemph`. Explicit font changes like `\textit` or `\textbf` are left alone, as they have no defined semantic meaning. If one is wanted `\emph` or `\strongemph` should be used,

For now we use hooks to add the code but this should be changed.

```

27 \AddToHook{cmd/emph/before}
28 { \mode_leave_vertical:
29   \UseTaggingSocket{inline/begin}
30     {tag=\UseStructureName{text/emph}}
31 }
32 \AddToHook{cmd/emph/after}
33 { \UseTaggingSocket{inline/end} }

```

Instead we offer `\strongemph` for this purpose.

```

34 \AddToHook{cmd/strongemph/before}
35 { \mode_leave_vertical:
36   \UseTaggingSocket{inline/begin}
37     {tag=\UseStructureName{text/strong}}
38 }
39 \AddToHook{cmd/strongemph/after}
40 { \UseTaggingSocket{inline/end} }

```

2.3 `\textsuperscript` and `\textsubscript`

```

41 \sys_if_engine_luatex:TF
42 {
43   \cs_new:Npn \__tag_space_ignore:n #1 {\tag_spacechar_off:#1\tag_spacechar_on:}
44 }
45 {
46   \cs_set_eq:NN \__tag_space_ignore:n \use:n
47 }
48 \protected\def\@textsuperscript#1%
49 {%
50   \check@mathfonts
51   \leavevmode
52   \begingroup
53   \UseTaggingSocket{inline/begin}{tag=\UseStructureName{span},attribute-class=sup}
54   \sbox\z@{\fontsize\sf@size\sf@size#1}%
55   \raise\dimexpr\textsuperscript@offset\relax\box\z@

```

```

56   \UseTaggingSocket{inline/end}
57   \_tag_space_ignore:n{\textsuperscript@space}
58   \endgroup
59 }
60 \protected\def\@textsubscript#1%
61 {%
62   \check@mathfonts
63   \leavevmode
64   \begin@group
65   \UseTaggingSocket{inline/begin}{tag=\UseStructureName{span},attribute-class=sub}
66   \sbox\z@{\fontsize\sf@size\sf@size#1}%
67   \lower\dimexpr\textsubscript@offset\relax\box\z@
68   \UseTaggingSocket{inline/end}
69   \_tag_space_ignore:n{\textsuperscript@space}
70   \endgroup
71 }

```

2.4 \mbox

When used in math and when `luamml` is active `\mbox` has to be correctly annotated. For this we add a socket that `luamml` can activate. We test for the socket until 2025-06-01.

```

72 \DeclareRobustCommand\mbox[1]
73 {
74   \leavevmode
75   \tag_socket_use:nnn {math/luamml/hbox}
76   {}{ \hbox{#1} }
77 }

```

And the same for `\makebox`

```

78 \long\def\@makebox[#1][#2]#3{%
79   \tag_socket_use:nnn {math/luamml/hbox}{ }
80   {
81     \begin@tempboxa\hbox{#3}%
82     \setlength\@tempdima{#1}%          support calc
83     \hb@xt@\@tempdima{%
84       \expandafter\ifx\csname bm@#2\endcsname\relax
85         \bm@c
86         \latex@warning{Unexpected-alignment~#2}%
87       \else
88         \csname bm@#2\endcsname
89       \fi}%
90     \end@tempboxa
91   }}

```

2.5 phantom commands

In text mode we do not want the content of a phantom command to create a structure, so we suspend tagging TODO: consider if it makes sense to set `measuring` to true instead. `\mathsm@sh` is redefined in `latex-lab-math`.

`\makeph@nt`

```

92 \def\makeph@nt#1{%
93   \setbox\z@\hbox

```

```

94     {\SuspendTagging{\makeph@nt}}%
95     \color@begingroup#1\color@endgroup\ResumeTagging{\makeph@nt}}\finph@nt}

```

(End of definition for \makeph@nt.)

\@kernel@mathph@nt
\mathph@nt

```

96 \def\@kernel@mathph@nt#1#2{%
97   \setbox\z@\hbox{\SuspendTagging{\mathph@nt}}$\m@th#1{#2}$\ResumeTagging{\mathph@nt}}\finph@nt}
98 \def\mathph@nt#1#2
99 {
100   \setbox \z@ = \hbox {
101     $
102     \m@th
103     #1
104     {
105 % nested phantoms should not be processed by luamml
106       \let\mathph@nt\@kernel@mathph@nt
107       #2
108     }
109     \UseTaggingSocket{math/luamml/save/nNn}{{\mathphant} #1 {\mphantom}}
110     $
111   }
112   \UseTaggingSocket{math/luamml/finph@nt}{{}\{\finph@nt}}
113 }

```

(End of definition for \@kernel@mathph@nt and \mathph@nt.)

2.6 The \verb command

```

114 \def\verb{\relax\ifmmode\hbox\else\leavevmode\null\fi
115   \bgroup
116   \UseTaggingSocket{inline/begin}{tag=\UseStructureName{text/verb}}%
117   \verb@eol@error \let\do\@makeoother \dospecials
118   \verbatim@font\@noligs
119   \language\l@nohyphenation
120   \@ifstar\@sverb\@verb}
121 \def\verb@egroup{\global\let\verb@balance@group\@empty\UseTaggingSocket{inline/end}\egroup}

```

2.7 The \rule command

Rules supplied as ornaments in the chapter designs and many other places should be tagged as artifacts. The is what the following code takes care of. It could be either done with Artifact SE's or with MCs that are artifacted (current version) — using MCs has the advantage that the PDF readers have less work to do, using SE the advantage that one can see the artifact in the structure tree. Not sure what is better. For now I provided two plugs so that one can switch.

```

122 \NewStructureName{text/rule}
123 \AssignStructureRole{text/rule}{Artifact}

124 \NewTaggingSocket{rule}{2}

125 \NewTaggingSocketPlug{rule}{artifact-MC}{
126   \tag_mc_end_push:
127   \tag_mc_begin:n{artifact}

```

```

128   #2
129   \tag_mc_end:
130   \tag_mc_begin_pop:n{ }
131 }

132 \NewTaggingSocketPlug{rule}{artifact-SE}{
133   \UseTaggingSocket{inline/begin}
134     {tag=\UseStructureName{text/rule}}
135   #2
136   \UseTaggingSocket{inline/end}
137 }

138 \AssignTaggingSocketPlug{rule}{artifact-MC}

```

`\@rule`

L^AT_EX's `\rule` command should also generate rules that are artifacted by default. This doesn't belong here, but before moving into `latex-lab-text.dtx` it may need some further adjustments and a decision on whether to use Artifact SEs or MCs.

```

139 \protected\def\@rule[#1]#2#3{
140   \leavevmode
141   \begingroup

```

Save all dimensions in variables before using them so that the process is calc safe.

```

142   \setlength\@tempdima{#1}
143   \setlength\@tempdimb{#2}
144   \setlength\@tempdimc{#3}
145   \advance\@tempdimc\@tempdima

```

Don't tag if the rule is known to be invisible.

```

146   \ifdim \@tempdimb=\z@
147   \hbox{ \vrule \@width \@tempdimb
148           \@height \@tempdimc
149           \@depth -\@tempdima }
150   \else
151   \UseTaggingSocket{rule}{-}{-}{
152     \hbox{ \vrule \@width \@tempdimb
153             \@height \@tempdimc
154             \@depth -\@tempdima }
155   }
156   \fi
157   \endgroup
158 }

```

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

```

159 \endpackage

```