

The latex-lab-firstaid package

Temporary patches to external packages needed for the tagging project

L^AT_EX Project*
v0.85h 2024-10-16

Abstract

1 Introduction

The followings contains small temporary changes to external packages to avoid errors with the new tagging code.

Similar to the main firstaid package the goal is to remove the patches once the packages have been updated.

2 Implementation

```
1 <*package>
2 <@@=tag>
3 \ProvidesPackage {latex-lab-testphase-firstaid} [%
4   \ltlabfirstaiddate\space v\ltlabfirstaiddate\space
5   Temporary patches to external packages needed for the tagging project]
\FirstAidNeededT This is a very simple help to ensure that we only apply first aid to an unmodified package
or class. It only works in the case the file has already been loaded and the csname
\ver@#1.#2 got defined (holding the current date, version, and short description info).
We then compare its content to a frozen string and make the modification #3 only if both
agree. If they differ we assume that the package/class in question got updated by its
maintainer.
6 \ExplSyntaxOn
7 \providecommand\FirstAidNeededT[3]{
8   \exp_args:Ncx\str_if_eq:onF{ver@#1.#2}{#3}
9     { \typeout{=>~ First~ Aid~ for~ #1.#2~ no~ longer~ applied!^^J
10       \@spaces Expected:^^J
11       \@spaces\@spaces #3^^J
12       \@spaces but~ found:^^J
13       \@spaces\@spaces \use:c{ver@#1.#2}^^J
14       \@spaces so~ I'm~ assuming~ it~ got~ fixed.
```

*Initial implementation done by Ulrike Fischer

```

15     } }
16     \exp_args:Ncx\str_if_eq:ont{ver@#1.#2}{#3}
17 }

```

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

2.1 ams classes

The `amsart`, `amsbook` and `amsproc` classes do not use `\@author` to store the author list but a command `\authors`. To be able to nevertheless use the authors in the xmp-metadata we map `\@author` to this new command.

```

18 \AddToHook{class/amsart/after}
19   {\def\@author{\authors}}
20 \AddToHook{class/amsbook/after}
21   {\def\@author{\authors}}
22 \AddToHook{class/amsproc/after}
23   {\def\@author{\authors}}

```

2.2 ams classes and amsthm

The `amsart`, `amsbook` and `amsproc` classes redefine the theorem code and this breaks the tagging added by the block code. The following reenables tagging. It does *not* give a completely identical output (similar to the new theorem code, see <https://github.com/latex3/tagging-project/issues/715>). The code also does not try to use sockets yet, as the theorem definitions in the block code don't do that yet either.

```

24 \AddToHook{class/amsart/after}[latex-lab-testphase-firstaid/amsthm]
25   {\tag_if_active:T{\__tag_firstaid_amsthm:}}
26 \AddToHook{class/amsbook/after}[latex-lab-testphase-firstaid/amsthm]
27   {\tag_if_active:T{\__tag_firstaid_amsthm:}}
28 \AddToHook{class/amsproc/after}[latex-lab-testphase-firstaid/amsthm]
29   {\tag_if_active:T{\__tag_firstaid_amsthm:}}
30 \AddToHook{package/amsthm/after}[latex-lab-testphase-firstaid/amsthm]
31   {\tag_if_active:T{\__tag_firstaid_amsthm:}}

32 \cs_new_protected:Npn \__tag_firstaid_amsthm:
33   {

```

`\@endtheorem` must use the endblock code

```

34   \def\@endtheorem{\endblockenv}

```

In `\@thm` we have to remove the `\trivlist`

```

35   \RenewDocumentCommand\@thm{mmmO{}}{-%
36     \ifhmode\unskip\unskip\par\fi
37     \normalfont
38     \let\thmheadnl\relax
39     \let\thm@swap\@gobble
40     \thm@notefont{\fontseries\mddefault\upshape}%
41     \thm@headpunct{.}% add period after heading
42     \thm@headsep 5\p@ plus\p@ minus\p@\relax
43     \thm@space@setup
44     ##1% style overrides
45     \@topsep \thm@preskip           % used by thm head
46     \@topseppadd \thm@postskip     % used by \@endparenv

```

We store the counter name so that the anchor can make use of it.

```

47     \tl_set:Nn \l__block_thm_current_counter_tl{##2}
48     \tl_if_empty:nTF{##2}
49     {
50       \@begintheorem{##3}{}[##4]
51     }
52     {
53       \@kernel@refstepcounter{##2}
54       \@begintheorem{##3}{\csname the##2\endcsname}[##4]
55     }
56   }

```

`\@begintheorem` has a larger number of changes

```

57   \def\@begintheorem##1##2[##3]{%

```

We use the theorem instance.

```

58     \UseInstance{blockenv}{theorem}{beginsep=\thm@preskip}

```

There is no working key to set the endskip, so we set the skip directly similar to what `amsthm` is doing after the `\trivlist`.

```

59     \skip_set:Nn\l__block_topsepadd_skip { \thm@postskip }

```

While create the caption/label we disable para-tagging.

```

60     \tagpdfparaOff
61     \mode_leave_vertical:

```

The anchor for links. `amsthm` allows for unnumbered theorems so we have to test for an empty counter.

```

62     \tl_if_empty:NTF \l__block_thm_current_counter_tl
63     {\MakeLinkTarget[theorem]{} }
64     {\MakeLinkTarget{\l__block_thm_current_counter_tl}}
65     \group_begin:
66     \normalfont
67     \the\thm@headfont \thm@indent
68     \@ifempty{##1}
69     {\let\thmname@gobble}

```

we insert the MC and the Lbl structure into `\thmname`, `\thmnumber` and `\thmnote`. This will also work with new theorem style as long as they use these command.

```

70     {\def\thmname####1{\tag_mc_begin:n {}####1\tag_mc_end:}}%
71     \@ifempty{##2}
72     {\let\thmnumber@gobble}
73     {\def\thmnumber####1
74       {\tag_struct_begin:n{tag=Lbl}\tag_mc_begin:n {}
75        ####1
76        \tag_mc_end:\tag_struct_end:}}%
77     \@ifempty{##3}
78     {\let\thmnote@gobble}
79     {\def\thmnote####1{\tag_mc_begin:n{}####1\tag_mc_end:}}%
80     \tag_struct_begin:n{tag=Caption}
81     \thm@swap\swappedhead\thmhead{##1}{##2}{##3}%
82     \tag_mc_begin:n{}\the\thm@headpunct\tag_mc_end:
83     \tag_struct_end:
84     \thmheadnl % possibly a newline.
85     \hskip\thm@headsep
86   \group_end:

```

Now we restart para tagging and start a paragraph. The socket is currently defined in tagpdf, so the code should only be used if tagging is active!

```

87     \tagpdfparaOn
88     \UseTaggingSocket{para/begin} %
89     \ignorespaces}

```

This redefines the standard styles for the theorem heads. `\thm@headpunct` has been moved into the head code to make tagging more easier.

```

90     \def\thmhead@plain##1##2##3{%
91         \thmname{##1}
92         \thmnumber{
93             \@ifnotempty{##1}{~}\@upn{##2}
94         }%
95         \thmnote{\pdfspacespace\space{\the\thm@notefont(##3)}}
96     }
97     \let\thmhead\thmhead@plain
98     \def\swappedhead##1##2##3{%
99         \thmnumber{##2}
100        \thmname{\@ifnotempty{##2}{\nobreakspace}##1}
101        \thmnote{\pdfspacespace\space{\the\thm@notefont(##3)}}
102    }
103    \let\swappedhead@plain=\swappedhead

```

At last some adjustments for the proof environment. The qed symbols use a drawn box by default. We add an actualtext.

```

104    \renewcommand{\openbox}{\leavevmode
105        \hbox to .77778em{\pdf_bdc:nn{Span}{/ActualText<FEFF220E>}}%
106        \pdfspacespace\hfil\vrule
107        \vbox to .675em{\hrule width .6em\vfil\hrule}%
108        \vrule\hfil\pdf_emc:}}

```

And redefine proof to no longer use a trivlist.

```

109    \renewenvironment{proof}[1][\proofname]{\par
110        \pushQED{\qed}%
111        \UseInstance{blockenv}{theorem}{beginsep=6\p@\@plus6\p@}
112        \normalfont
113        \tagpdfparaOff
114        \AddToHookNext{para/begin}
115        {\tag_struct_begin:n{tag=Caption}
116         \tag_mc_begin:n{}}%
117         \textit{##1\@addpunct{.}}}%
118        \tag_mc_end:
119        \tag_struct_end:
120        \tagpdfparaOn
121        \UseTaggingSocket{para/begin}
122        \pdfspacespace\hspace{\labelsep}}
123    \ignorespaces
124    }-%
125    \popQED\endblockenv\par
126    }
127 }
128 \ExplSyntaxOff

```

2.3 verse

The `verse` package has its own definition of the `verse` environment, which would tag correctly, except that it is overwritten by the block code in the hook `begindocument/before`. So the simplest way to make tagging work is to reinstall the package version afterwards, which is what we are doing here.

```

129 \AddToHook{package/verse/after}[latex-lab-firstaid]{%
130   \FirstAidNeededT{verse}{sty}{2014/05/10 v2.4b verse typesetting}%
131   {%
132     \AtBeginDocument{%
133       \renewenvironment{verse}[1][\linewidth]{%
134         \stepcounter{verse@envctr}%
135         \setcounter{poemline}{0}\refstepcounter{poemline}%
136         \setcounter{vslineno}{1}%
137         \let\@vscentercr
138         \list{}{\itemsep \z@
139             \itemindent -\vindent
140             \listparindent\itemindent
141             \parsep \stanzaskip
142             \ifdim #1 < \linewidth
143               \rightmargin \z@
144               \setlength{\leftmargin}{\linewidth}%
145               \addtolength{\leftmargin}{-#1}%
146               \addtolength{\leftmargin}{-0.5\leftmargin}%
147             \else
148               \rightmargin \leftmargin
149             \fi
150             \addtolength{\leftmargin}{\vindent}}%
151         \item[]%
152       }%
153       {\endlist}%
154     }%
155   }%
156 }

```

Of course, this means that the optional argument of the environment then only accepts a length value and not any more a key value list for altering the environment settings.

A more elaborate version could be something like this that allows key/val and legacy interface. Or one could extend the list template to support a `list-width` key.

```

\ExplSyntaxOn
\cs_new_protected:Npn \ExtractAndDropKey #1#2#3#4#5 {
  \tl_set_eq:NN #4 \c_novalue_tl % or empty?
  \keys_define:nn { #1 } { #2 .code:n = \tl_set:Nn #4{##1} }
  \keys_set_known:nnN { #1 } { #3 } #5
}
\ExplSyntaxOff

```

```

% Change the env definition for verse matching verse.sty
% This keeps the verse.sty interface as it is and only adjusts the
% main environment to use the basic list env with the verse.sty
% specific settings.
\makeatletter

```

```

\AddToHook{package/verse/after}{%
  \AtBeginDocument{%
    \RenewDocumentEnvironment{verse}={={verse-width}!0{\linewidth}}%
      {%
        \stepcounter{verse@envctr}%
        \setcounter{poemline}{0}\refstepcounter{poemline}%
        \setcounter{vslineno}{1}%
        \let\=\@vscentercr
      }%
    %
    \ExtractAndDropKey{verse}{verse-width}{#1}\@vswidth\@vsremainingkvlist
    % If other keys have been specified but not verse-width we have no
    % default for \@vswidth and need to set it again
    \ExpandArgs{o}\IfNoValueT \@vswidth
      {\def\@vswidth{\linewidth}}%
    %
    % This is a bit ugly but we can't stick \cs{@vsremainingkvlist} into
    % the instance argument as keys are expected to be visible on
    % top-level not hidden inside a macro. The alternative is to push
    % in \verb=#1= but then the key/value \verb/verse-width=.../ is
    % passed into the instance which is not known there (not harmful as
    % it will get ignored but noticeably more and unnecessary
    % processing).
    %
    \def\next##1{%
      \UseInstance{blockenv}{list}%
      {%
        item-indent =-\vindent,%
        parindent   =-\vindent,%
        par-skip    =\stanzaskip,%
        item-skip   =0pt,%
        leftmargin  = (\linewidth-\@vswidth)/2+\vindent,%
        rightmargin = \ifdim\@vswidth<\linewidth Opt
                      \else (\linewidth-\@vswidth)/2\fi,%
        ##1%
      }%
      \ExpandArgs{o}\next\@vsremainingkvlist
      \item\relax
    }\endblockenv}%
  }%
}
\makeatother

```

2.4 cleveref

The cleveref package redefines `\@makefn` and this means that the patches in the new footnote code fails. We use a hook instead.

```

157 \AddToHook{package/cleveref/after}
158 {
159   \let\@makefn\cref@old@makefn

```

```

160 \AddToHook{cmd/@makefnmark/before}{%
161 \cref@constructprefix{footnote}{\cref@result}%
162 \protected@edef\cref@currentlabel{%
163 [footnote][\arabic{footnote}][\cref@result]%
164 \p@footnote\@thefnmark}}
165 }

```

2.5 booktabs

In some cases booktabs inserts a `\multispan` into the table (through the commands `\@cmidruleb` and `\@cmidrulea` and this then errors with the tagging code. This affects both `tabular` and `longtable` (but `longtable` more as `booktabs` handles lines in `longtable` differently). See also issue <https://github.com/latex3/tagging-project/issues/69>

```

166 \ExplSyntaxOn
167 \AddToHook{package/booktabs/after}
168 {
169 \def\@cmidrulea{
170 \multispan\@cmidla
171 &\multispan\@cmidlb
172 \unskip\hskip\cmrkern@l
173 {
174 \tag_mc_begin:n{artifact}
175 \CT@arc@\leaders\hrule \@height\@thisrulewidth\hfill\kern\z@}
176 \hskip\cmrkern@r
177 \tag_mc_end: \int_gdecr:N \g__tbl_row_int
178 \cr}
179
180 \def\@cmidruleb{%
181 \multispan\@cmidlb
182 \unskip\hskip \cmrkern@l%
183 {
184 \tag_mc_begin:n{artifact}
185 \CT@arc@\leaders\hrule \@height\@thisrulewidth\hfill\kern\z@}
186 \hskip\cmrkern@r
187 \tag_mc_end: \int_gdecr:N \g__tbl_row_int
188 \cr}
189 }
190 \ExplSyntaxOff

```

2.6 fancyvrb

The `firstaid` adds first partial tagging support to the environments of `fancyvrb` (inline verbatim is untested). This supports then also packages like `minted` which internally uses `fancyvrb` and classes like `l3doc` (where currently the verbatim environment based on `fancyvrb` is overwritten by the block code). The environments are surrounded by a `verbatim` structure, every line by a `codeline` structure (this requires the block code, but `firstaid` should be used only with phase-III anyway). Line numbers are tagged as `Lbl`, currently outside of the `codeline` structure. The frame lines are marked as `artifact`.

`\FV@LeaveVMode` If we are in `vmode` we have to open a text-unit structure, if we are in `hmode` we have to set `para` mode to flattened before the `fancyhdr` code issues the `\par`. The closing of the text-unit structure is handled by the `doendpe` code in the block code.

```

191 \ExplSyntaxOn
192 \AddToHook{package/fancyvrb/after}
193 {
194   \def\FV@LeaveVMode{%
195     \if@noskipsec
196       \leavevmode
197     \else
198       \if\FV@ResetMargins\if@inlabel\leavevmode\fi\fi
199     \fi
200     \ifvmode
201       \@nparlisttrue
202       \tag_gincr_para_main_begin_int:
203       \tag_struct_begin:n{tag=\l__tag_para_main_tag_tl}
204     \else
205       \bool_set_true:N\l__tag_para_flattened_bool
206       \@nparlistfalse
207       \unskip\par
208     \fi
209   }

```

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

\FV@List At the begin of the list code we have to tag the frame as artifact and start the `verbatim` structure

```

210   \def\FV@List#1{%
211     \begingroup
212     \FV@UseKeyValues
213     \FV@LeaveVMode
214     \if@inlabel\else\setbox\@labels=\box\voidb@x\fi
215     \FV@ListNesting{#1}%
216     \FV@ListParameterHook
217     \FV@ListVSpace
218     \FV@SetLineWidth
219     \FV@InterLinePenalty
220     \let\FV@ProcessLine\FV@ListProcessLine@i
221     \FV@CatCodes
222     \FV@FormattingPrep
223     \FV@ObeyTabsInit
224     \cs_if_exist:NT \FV@BeginListFrame
225     {
226       \tag_mc_begin:n{artifact}
227       \FV@BeginListFrame
228       \tag_mc_end:
229     }
230     \tag_struct_begin:n{tag=verbatim}
231   }

```

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

\FV@endList At the end of the list code we close the `verbatim` structure and tag the frame as artifact.

```

232   \def\FV@endList{%
233     \FV@ListProcessLastLine
234     \tag_struct_end:
235     \cs_if_exist:NT \FV@endListFrame

```



```

236     {
237       \tag_mc_begin:n{artifact}
238       \FV@EndListFrame
239       \tag_mc_end:
240     }
241   \@endparenv
242   \endgroup
243   \@endpetrue
244 }

```

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

`\FV@ListProcessLine` At last the tagging of the code lines. Here we have to tag also numbers and frame parts if they exist.

```

245   \def\FV@ListProcessLine#1{%
246     \hbox to \hsize{%
247       \kern\leftmargin
248       \hbox to \linewidth{%
249         \cs_if_exist:NT \FV@LeftListNumber
250         {
251           \tag_struct_begin:n{tag=Lbl}
252           \tag_mc_begin:n{}
253           \FV@LeftListNumber
254           \tag_mc_end:
255           \tag_struct_end:
256         }
257         \cs_if_exist:NT \FV@LeftListFrame
258         {
259           \tag_mc_begin:n{artifact}
260           \FV@LeftListFrame
261           \tag_mc_end:
262         }
263         \tag_struct_begin:n{tag=codeline}
264         \tag_mc_begin:n{}%
265         \FancyVerbFormatLine{#1}%
266         \tag_mc_end:
267         \tag_struct_end:\hss
268         \cs_if_exist:NT \FV@RightListFrame
269         {
270           \tag_mc_begin:n{artifact}
271           \FV@RightListFrame
272           \tag_mc_end:
273         }
274         \cs_if_exist:NT \FV@RightListNumber
275         {
276           \tag_struct_begin:n{tag=Lbl}
277           \tag_mc_begin:n{}
278           \FV@RightListNumber
279           \tag_mc_begin:n{}
280           \tag_struct_end:
281         }
282       }
283     } \hss}}
284 }
285 \ExplSyntaxOff

```

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

```
286 </package>
287 <*latex-lab>
288 \ProvidesFile{firstaid-latex-lab-testphase.ltx}
289     [\ltmlabfirstaiddate\space v\ltmlabfirstaidversion\space
290     latex-lab wrapper firstaid]
291
292 \RequirePackage{latex-lab-testphase-firstaid}
293
294 </latex-lab>
```