

# The latex-lab-firstaid package

## Temporary patches to external packages needed for the tagging project

L<sup>A</sup>T<sub>E</sub>X Project\*

v0.85u 2026-04-24

### 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   \ltxlabfirstaiddatetime\space v\ltxlabfirstaidversion\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:Nce\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:Nce\str_if_eq:ont{ver@#1.#2}{#3}
17 }

```

(End of definition for `\FirstAidNeededT`.)

## 2.1 tikz/pgf

`tikz` inputs libraries with the primitive `\input` command. This means that these libraries are not listed in the file list written by `\listfiles` and the new tagging status report created with the `check-tagging-status` key.

We therefore redefine one `pgf` command to use the L<sup>A</sup>T<sub>E</sub>X `\input` command. Check <https://github.com/pgf-tikz/pgf/issues/1424> for changes.

```

18 \AddToHook{package/pgfrcs/after}
19 {\def\pgfutil@InputIfFileExists#1#2#3{\pgfutil@IfFileExists{#1}{\input{#1}\relax#2}{#3}}}

```

## 2.2 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.

The authors are set with a `trivlist`, this leads to faulty spacing and wrong tagging. This is not yet handled here, but an example how to correct this is in the `test-amsart-title` test.

```

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

```

The classes redefine `\@startsection` and define their own heading commands which are not tagging compatible. The template code will reinstate the new `\@startsection` and so handle the headings using it. But we also need to redefine `\chapter` and `\part`. The following does not try to create exactly the same design, only near enough. This will not properly handle `\specialsection`.

```

26 \cs_new_protected:Npn \__tag_firstaid_amsbook_heading:
27 {
28   \DeclareDocumentCommand \chapter {s = {shorttitle} o m}
29     { \ParseLaTeXeHeading {chapter} {##1} {##2} {##3} }
30   \DeclareInstance{heading}{chapter}{display}
31   {
32     , name          = chapter
33     , level         = 0
34     , placement     = top
35     , after-penalty-sep = 32pt
36     , after-sep     = 22pt
37     , number-format = \MakeUppercase{\chaptername}\enspace \thechapter
38     , heading-decls = \centering
39     , number-decls  = \normalsize\mdseries
40     , title-decls   = \fontsize{\@xivpt}{18}\bfseries
41     , headformat-instance = chapter

```

```

42     , mark-cmd = \chaptermark{##1}
43     , para-indent = true
44     , contents-extra= \addtocontents{lof}{\protect\addvspace{10\p@}}%
45                       \addtocontents{lot}{\protect\addvspace{10\p@}}%
46   }
47   \DeclareInstance{headformat}{chapter}{display}
48   {
49     , heading-indent = 0pt
50     , number-title-sep = 18pt
51   }
52   \DeclareDocumentCommand \part {s = {shorttitle} o m}
53   { \ParseLaTeXeHeading {part} {##1} {##2} {##3} }
54   \DeclareInstance{heading}{part}{display}
55   {
56     , name = part
57     , level = -1
58     , placement = page
59     , start-code= {\cleardoublepage\thispagestyle{empty}}%
60                  \null\vfil\markboth{}{}}
61     , final-code= \vfil\vfil\newpage\newpage\thispagestyle{empty}
62     , number-format = \partname\nobreakspace\thepart
63     , heading-decls = \centering\bfseries
64     , number-decls = \huge
65     , title-decls = \Huge
66     , headformat-instance = display
67     , mark-cmd = \partmark {##1}
68   }
69 }
70 \cs_new_protected:Npn \__tag_firstaid_amsart_heading:
71 {
72   \DeclareInstance{heading}{part-@startsection}{display}
73   {
74     name=part,
75     level=-1,
76     mark-cmd=\partmark {##1},
77     para-indent = true,
78     before-sep = 12.0pt plus 12.0pt,
79     after-penalty-sep = 0pt,
80     after-sep = 6.0pt,
81     heading-decls = \normalfont \bfseries \raggedright,
82     headformat-instance = part-@startsection,
83     number-format = \partname\space\theheading.
84   }
85   \DeclareInstance{headformat}{part-@startsection}{hang}
86   {heading-indent = \z@, number-title-sep = 0.5em}
87 }

```

The redefinitions for amsbook:

```

88 \AddToHook{class/amsbook/after}[latex-lab-testphase-firstaid/amschap]
89   {\__tag_firstaid_amsbook_heading:}

```

amsart and amsproc can use the same definition:

```

90 \AddToHook{class/amsproc/after}[latex-lab-testphase-firstaid/amschap]
91   {\__tag_firstaid_amsart_heading:}

```

```

92 \AddToHook{class/amsart/after}[latex-lab-testphase-firstaid/amschap]
93   {\_tag_firstaid_amsart_heading:}

```

## 2.3 ams classes and amsthm

In these classes the abstract is stored in a `\vbox`, as it ends with a list an explicit `\par` is needed.

```

94 \AddToHook{class/amsart/after}[latex-lab-testphase-firstaid/abstract]
95   {
96     \_tag_firstaid_ams_abstract:
97   }
98 \AddToHook{class/amsbook/after}[latex-lab-testphase-firstaid/abstract]
99   {
100     \_tag_firstaid_ams_abstract:
101   }

102 \cs_new_protected:Npn \_tag_firstaid_ams_abstract:
103 {
104   \renewenvironment{abstract}{%
105     \ifx\maketitle\relax
106       \ClassWarning{@classname}{Abstract~ should~ precede~
107         \protect\maketitle\space in~ AMS~ document~ classes;~ reported}%
108     \fi
109     \global\setbox\abstractbox=\vtop \bgroup
110     \normalfont\Small
111     \list{}{\labelwidth\z@
112       \leftmargin3pc \rightmargin\leftmargin
113       \listparindent\normalparindent \itemindent\z@
114       \parsep\z@ \@plus\p@
115       \let\fullwidthdisplay\relax
116     }%
117     \item[\hskip\labelsep\scshape\abstractname.]%
118   }{%
119     \endlist
120     \par           % <--- added
121     \egroup
122     \ifx\@setabstract\relax \@setabstracta \fi
123   }
124 }
125 \ExplSyntaxOff

```

## 2.4 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.

```

126 \AddToHook{package/verse/after}[latex-lab-firstaid]{%
127   \FirstAidNeededT{verse}{sty}{2014/05/10 v2.4b verse typesetting}%
128   {%
129     \AtBeginDocument{%
130       \renewenvironment{verse}[1][\linewidth]{%
131         \stepcounter{verse@envctr}%

```

```

132         \setcounter{poemline}{0}\refstepcounter{poemline}%
133         \setcounter{vslineno}{1}%
134         \let\@=\@vscentercr
135         \list{}{\itemsep \z@
136             \itemindent -\vindent
137             \listparindent\itemindent
138             \parsep \stanzaskip
139             \ifdim #1 < \linewidth
140                 \rightmargin \z@
141                 \setlength{\leftmargin}{\linewidth}%
142                 \addtolength{\leftmargin}{-#1}%
143                 \addtolength{\leftmargin}{-0.5\leftmargin}%
144             \else
145                 \rightmargin \leftmargin
146             \fi
147             \addtolength{\leftmargin}{\vindent}}%
148         \item[]%
149     }%
150     {\endlist}%
151 }%
152 }%
153 }

```

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,%
    para-indent   = -\vindent,%
    para-vspace   = \stanzaskip,%
    item-skip      = Opt,%
    left-margin   = (\linewidth-\@vswidth)/2+\vindent,%
    right-margin  = \ifdim\@vswidth<\linewidth Opt
                    \else (\linewidth-\@vswidth)/2\fi,%

    ##1%
  }}%
  \ExpandArgs{o}\next\@vsremainingkvlist
  \item\relax
} {\endblockenv}%
}%
}
\makeatother

```

## 2.5 cleveref

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

```

154 \AddToHook{package/cleveref/after}
155 {
156   \let\@makefnfntext\cref@old@makefnfntext
157   \AddToHook{cmd/@makefnfntext/before}{%
158     \cref@constructprefix{footnote}{\cref@result}%
159     \protected@edef\cref@currentlabel{%
160       [footnote][\arabic{footnote}][\cref@result]%
161       \p@footnote\@thefnmark}}
162 }

```

## 2.6 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>

```

163 \ExplSyntaxOn
164 \AddToHook{package/booktabs/after}
165 {
166   \def\@cmidrulea{
167     \multispan\@cmidla
168     &\multispan\@cmidlb
169     \unskip\hskip\cmrkern@l
170   {
171     \UseTaggingSocket{tbl/leaders/begin}
172     \CT@arc@\leaders\hrule \@height\@thisrulewidth\hfill\kern\z@}
173     \hskip\cmrkern@r
174     \UseTaggingSocket{tbl/leaders/end}
175     \tbl_gdecr_row_count:
176     \cr}
177
178   \def\@cmidruleb{%
179     \multispan\@cmidlb
180     \unskip\hskip \cmrkern@l%
181   {
182     \UseTaggingSocket{tbl/leaders/begin}
183     \CT@arc@\leaders\hrule \@height\@thisrulewidth\hfill\kern\z@}
184     \hskip\cmrkern@r
185     \UseTaggingSocket{tbl/leaders/end}
186     \tbl_gdecr_row_count:
187     \cr}
188   }
189 \ExplSyntaxOff

```

## 2.7 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. Line numbers are tagged as Lbl, inside 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 fancyvrb code issues the **\par**. The closing of the text-unit structure is handled by the doendpe code in the block code.

```

190 \ExplSyntaxOn
191 \AddToHook{package/fancyvrb/after}
192 {
193   \def\FV@LeaveVMode{%
194     \if@noskipsec
195       \leavevmode
196     \else
197       \if\FV@ResetMargins\if@inlabel\leavevmode\fi\fi
198       \fi
199       \ifvmode
200         \@noparlisttrue

```

```

201      \UseTaggingSocket{para/semantic/begin}{}
202    \else
203      \tagpdfsetup{para/flattened=true}
204      \@noparlistfalse
205      \unskip\par
206    \fi
207  }

```

(End of definition for \FV@LeaveVMode.)

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

```

208    \def\FV@List#1{%
209      \begingroup
210      \FV@UseKeyValues
211      \FV@LeaveVMode
212      \if@inlabel\else\setbox\@labels=\box\voidb@x\fi
213      \FV@ListNesting{#1}%
214      \FV@ListParameterHook
215      \FV@ListVSpace
216      \FV@SetLineWidth
217      \FV@InterLinePenalty
218      \let\FV@ProcessLine\FV@ListProcessLine@i
219      \FV@CatCodes
220      \FV@FormattingPrep
221      \FV@ObeyTabsInit
222      \cs_if_exist:NT \FV@BeginListFrame
223      {
224        \UseTaggingSocket{mc}{artifact}{\FV@BeginListFrame}
225      }
226      \UseTaggingSocket{struct/begin}{tag=\UseStructureName{block/verbatim}}
227    }

```

(End of definition for \FV@List.)

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

```

228    \def\FV@EndList{%
229      \FV@ListProcessLastLine
230      \UseTaggingSocket{struct/end}
231      \cs_if_exist:NT \FV@EndListFrame
232      {
233        \UseTaggingSocket{mc}{artifact}{\FV@EndListFrame}
234      }
235      \@endparenv
236      \endgroup
237      \@endpetrue
238    }

```

(End of definition for \FV@EndList.)

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

```

239    \def\FV@ListProcessLine#1{%
240      \hbox to \hsize{%
241        \kern\leftmargin

```



```

242 \hbox to \linewidth{%
243 \UseTaggingSocket{struct/begin}{tag=\UseStructureName{block/verbatim/codeline}}
244 \cs_if_exist:NT \FV@LeftListNumber
245 {
246 \UseTaggingSocket{struct-mc}{tag=\UseStructureName{block/verbatim/linenumber}}
247 {
248 \FV@LeftListNumber
249 }
250 }
251 \cs_if_exist:NT \FV@LeftListFrame
252 {
253 \UseTaggingSocket{mc}{artifact}{\FV@LeftListFrame}
254 }
255 \UseTaggingSocket{mc}{}{\FancyVerbFormatLine{#1}}%
256 \UseTaggingSocket{struct/end}\hss
257 \cs_if_exist:NT \FV@RightListFrame
258 {
259 \UseTaggingSocket{mc}{artifact}{\FV@RightListFrame}
260 }
261 \cs_if_exist:NT \FV@RightListNumber
262 {
263 \UseTaggingSocket{struct-mc}{tag=\UseStructureName{block/verbatim/linenumber}}
264 {
265 \FV@RightListNumber
266 }
267 }
268 }
269 \hss}}
270 }
271 \ExplSyntaxOff

```

*(End of definition for \FV@ListProcessLine.)*

## 2.8 The float package first aid

The float package has a number of incompatibilites:

- Restyling floats with `\restylefloat` breaks the tagging tagging support.
- New float types created with `\newfloat` are not properly tagged (tagging issue 890)
- Non-floating floats using the H placement are not properly tagged.

The following firstaid addresses the last problem. A flaw is that if the float is inside a paragraph and followed by text there the text is tagged as a new text-unit structure. TODO

```

272 \ExplSyntaxOn
273 \AddToHook{package/float/after}[firstaid]
274 {
275 \__tag_firstaid_float_patch_H:
276 }
277 \cs_new_protected:Npn \__tag_firstaid_float_patch_H:
278 {
279 \NewTaggingSocket{float/H/begin}{1}
280 \NewTaggingSocketPlug{float/H/begin}{default}

```

```

281 {
282   \mode_if_horizontal:T {\tag_mc_end_push: \tag_struct_end:}
283   \tag_struct_begin:n {\tag=\UseStructureName{float/##1}}
284 }
285 \AssignTaggingSocketPlug{float/H/begin}{default}
286
287 \NewTaggingSocket{float/H/end}{0}
288 \NewTaggingSocketPlug{float/H/end}{default}
289 {
290   \tag_struct_end:
291   \mode_if_horizontal:T
292   {
293     \tag_struct_begin:n {\tag=\UseStructureName{para/textblock}}
294     \tag_mc_begin_pop:n {}
295   }
296 }
297 \AssignTaggingSocketPlug{float/H/end}{default}

```

and here the redefinitions of the commands of float

```

298 \def\@float@HH##1[H]{\UseTaggingSocket{float/H/begin}{##1}%
299   \expandafter\let\csname end##1\endcsname\float@endH
300   \let\@currbox\float@box
301   \def\@capttype{##1}\setbox\@floatcapt=\vbox{}%
302   \expandafter\ifx\csname fst##1\endcsname\relax
303   \@flstylefalse\else\@flstyletrue\fi
304   \setbox\@currbox\color@vbox\normalcolor
305     \vbox\bgroup \hsize\columnwidth \@parboxrestore
306     \@floatboxreset \setnobreak
307     \ignorespaces}
308 \renewcommand\float@endH{\@endfloatbox\UseTaggingSocket{float/H/end}
309   \vskip\intextsep
310   \if@flstyle\setbox\@currbox\float@makebox\columnwidth\fi
311   \box\@currbox\vskip\intextsep\relax}
312 }
313 \ExplSyntaxOff
314 \end{package}

```