# The latex-lab-firstaid package
# Temporary patches to external packages needed for the tagging project

LaTeX 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

```
⟨*package⟩
⟨@@=tag⟩
```

```
\ProvidesPackage {latex-lab-testphase-firstaid} [%
   \ltlabfirstaiddate\space v\ltlabfirstaiddate\space
   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.

```
\ExplSyntaxOn
\providecommand\FirstAidNeededT[3]{
  \exp_args:Ncx\str_if_eq:onF{ver@#1.#2}{#3}
      { \typeout{==>~ First~ Aid~ for~ #1.#2~ no~ longer~ applied!^^J
          \@spaces Expected:^^J
          \@spaces\@spaces #3^^J
          \@spaces but~ found:^^J
          \@spaces\@spaces \use:c{ver@#1.#2}^^J
          \@spaces so~ I'm~ assuming~ it~ got~ fixed.
```

*Initial implementation done by Ulrike Fischer

```
    } }
  \exp_args:Ncx\str_if_eq:onT{ver@#1.#2}{#3}
}
```

(*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.

```
\AddToHook{class/amsart/after}
 {\def\@author{\authors}}
\AddToHook{class/amsbook/after}
 {\def\@author{\authors}}
\AddToHook{class/amsproc/after}
 {\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 completly 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.

```
\AddToHook{class/amsart/after}[latex-lab-testphase-firstaid/amsthm]
 {\tag_if_active:T{\__tag_firstaid_amsthm:}}
\AddToHook{class/amsbook/after}[latex-lab-testphase-firstaid/amsthm]
 {\tag_if_active:T{\__tag_firstaid_amsthm:}}
\AddToHook{class/amsproc/after}[latex-lab-testphase-firstaid/amsthm]
 {\tag_if_active:T{\__tag_firstaid_amsthm:}}
\AddToHook{package/amsthm/after}[latex-lab-testphase-firstaid/amsthm]
 {\tag_if_active:T{\__tag_firstaid_amsthm:}}

\cs_new_protected:Npn \__tag_firstaid_amsthm:
  {
```

`\@endtheorem` must use the endblock code

```
    \def\@endtheorem{\endblockenv}
```

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

```
    \RenewDocumentCommand\@thm{mmmO{}}{%
      \ifhmode\unskip\unskip\par\fi
      \normalfont
      \let\thmheadnl\relax
      \let\thm@swap\@gobble
      \thm@notefont{\fontseries\mddefault\upshape}%
      \thm@headpunct{.}% add period after heading
      \thm@headsep 5\p@ plus\p@ minus\p@\relax
      \thm@space@setup
      ##1% style overrides
      \@topsep \thm@preskip               % used by thm head
      \@topsepadd \thm@postskip           % used by \@endparenv
```

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

```
      \tl_set:Nn \l__block_thm_current_counter_tl{##2}
      \tl_if_empty:nTF{##2}
       {
        \@begintheorem{##3}{}[##4]
       }
       {
         \@kernel@refstepcounter{##2}
         \@begintheorem{##3}{\csname the##2\endcsname}[##4]
       }
      }
```

`\@begintheorem` has a larger number of changes

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

We use the theorem instance.

```
      \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`.

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

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

```
      \tagpdfparaOff
      \mode_leave_vertical:
```

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

```
     \tl_if_empty:NTF \l__block_thm_current_counter_tl
       {\MakeLinkTarget[theorem]{}}
       {\MakeLinkTarget{\l__block_thm_current_counter_tl}}
      \group_begin:
      \normalfont
      \the\thm@headfont \thm@indent
        \@ifempty{##1}
         {\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.

```
         {\def\thmname####1{\tag_mc_begin:n {}####1\tag_mc_end:}}%
        \@ifempty{##2}
         {\let\thmnumber\@gobble}
         {\def\thmnumber####1
           {\tag_struct_begin:n{tag=Lbl}\tag_mc_begin:n {}
             ####1
            \tag_mc_end:\tag_struct_end:}}%
        \@ifempty{##3}
         {\let\thmnote\@gobble}
         {\def\thmnote####1{\tag_mc_begin:n{}####1\tag_mc_end:}}%
        \tag_struct_begin:n{tag=Caption}
        \thm@swap\swappedhead\thmhead{##1}{##2}{##3}%
        \tag_mc_begin:n{}\the\thm@headpunct\tag_mc_end:
        \tag_struct_end:
        \thmheadnl % possibly a newline.
        \hskip\thm@headsep
      \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!

```
       \tagpdfparaOn
       \UseTaggingSocket{para/begin}  %
      \ignorespaces}
```

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

```
    \def\thmhead@plain##1##2##3{%
      \thmname{##1}
      \thmnumber{
        \@ifnotempty{##1}{~}\@upn{##2}
       }%
      \thmnote{\pdffakespace\space{\the\thm@notefont(##3)}}
      }
    \let\thmhead\thmhead@plain
    \def\swappedhead##1##2##3{%
      \thmnumber{##2}
      \thmname{\@ifnotempty{##2}{\nobreakspace}##1}
      \thmnote{\pdffakespace\space{\the\thm@notefont(##3)}}
     }
    \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.

```
    \renewcommand{\openbox}{\leavevmode
      \hbox to.77778em{\pdf_bdc:nn{Span}{/ActualText<FEFF220E>}%
      \pdffakespace\hfil\vrule
      \vbox to.675em{\hrule width.6em\vfil\hrule}%
      \vrule\hfil\pdf_emc:}}
```

And redefine proof to no longer use a trivlist.

```
    \renewenvironment{proof}[1][\proofname]{\par
      \pushQED{\qed}%
      \UseInstance{blockenv}{theorem}{beginsep=6\p@\@plus6\p@}
      \normalfont
      \tagpdfparaOff
      \AddToHookNext{para/begin}
       {\tag_struct_begin:n{tag=Caption}
        \tag_mc_begin:n{}%
          \textit{##1\@addpunct{.}}%
        \tag_mc_end:
        \tag_struct_end:
        \tagpdfparaOn
        \UseTaggingSocket{para/begin}
        \pdffakespace\hspace{\labelsep}}
      \ignorespaces
    }{%
      \popQED\endblockenv\par
     }
  }
\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.

```
\AddToHook{package/verse/after}[latex-lab-firstaid]{%
  \FirstAidNeededT{verse}{sty}{2014/05/10 v2.4b verse typesetting}%
   {%
     \AtBeginDocument{%
       \renewenvironment{verse}[1][\linewidth]{%
         \stepcounter{verse@envctr}%
         \setcounter{poemline}{0}\refstepcounter{poemline}%
         \setcounter{vslineno}{1}%
         \let\\=\@vscentercr
         \list{}{\itemsep \z@
                 \itemindent  -\vindent
                 \listparindent\itemindent
                 \parsep       \stanzaskip
                 \ifdim #1 < \linewidth
                   \rightmargin        \z@
                   \setlength{\leftmargin}{\linewidth}%
                   \addtolength{\leftmargin}{-#1}%
                   \addtolength{\leftmargin}{-0.5\leftmargin}%
                 \else
                   \rightmargin        \leftmargin
                 \fi
                 \addtolength{\leftmargin}{\vindent}}%
         \item[]%
       }%
       {\endlist}%
     }%
   }%
}
```

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 elabroate 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}!O{\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 0pt
                          \else (\linewidth-\@vswidth)/2\fi,%
            ##1%
           }}%
      \ExpandArgs{o}\next\@vsremainingkvlist
      \item\relax
    }{\endblockenv}%
  }%
}
\makeatother
```

## 2.4 cleveref

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

```
\AddToHook{package/cleveref/after}
{
  \let\@makefntext\cref@old@makefntext
```

```
   \AddToHook{cmd/@makefntext/before}{%
    \cref@constructprefix{footnote}{\cref@result}%
    \protected@edef\cref@currentlabel{%
     [footnote][\arabic{footnote}][\cref@result]%
    \p@footnote\@thefnmark}}
 }
```

## 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

```
\ExplSyntaxOn
\AddToHook{package/booktabs/after}
 {
  \def\@cmidrulea{
   \multispan\@cmidla
   &\multispan\@cmidlb
   \unskip\hskip\cmrkern@l
  {
   \tag_mc_begin:n{artifact}
   \CT@arc@\leaders\hrule \@height\@thisrulewidth\hfill\kern\z@}
   \hskip\cmrkern@r
   \tag_mc_end: \int_gdecr:N \g__tbl_row_int
   \cr}

  \def\@cmidruleb{%
    \multispan\@cmidlb
    \unskip\hskip \cmrkern@l%
   {
    \tag_mc_begin:n{artifact}
    \CT@arc@\leaders\hrule \@height\@thisrulewidth\hfill\kern\z@}
    \hskip\cmrkern@r
    \tag_mc_end: \int_gdecr:N \g__tbl_row_int
    \cr}
  }
\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.

```
\ExplSyntaxOn
\AddToHook{package/fancyvrb/after}
{
    \def\FV@LeaveVMode{%
      \if@noskipsec
        \leavevmode
      \else
        \if@FV@ResetMargins\if@inlabel\leavevmode\fi\fi
      \fi
      \ifvmode
        \@noparlisttrue
        \__tag_gincr_para_main_begin_int:
        \tag_struct_begin:n{tag=\l__tag_para_main_tag_tl}
      \else
        \bool_set_true:N\l__tag_para_flattened_bool
        \@noparlistfalse
        \unskip\par
      \fi
      }
```

(*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

```
    \def\FV@List#1{%
      \begingroup
      \FV@UseKeyValues
      \FV@LeaveVMode
      \if@inlabel\else\setbox\@labels=\box\voidb@x\fi
      \FV@ListNesting{#1}%
      \FV@ListParameterHook
      \FV@ListVSpace
      \FV@SetLineWidth
      \FV@InterLinePenalty
      \let\FV@ProcessLine\FV@ListProcessLine@i
      \FV@CatCodes
      \FV@FormattingPrep
      \FV@ObeyTabsInit
      \cs_if_exist:NT \FV@BeginListFrame
        {
         \tag_mc_begin:n{artifact}
         \FV@BeginListFrame
         \tag_mc_end:
        }
      \tag_struct_begin:n{tag=verbatim}
      }
```

(*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.

```
    \def\FV@EndList{%
      \FV@ListProcessLastLine
      \tag_struct_end:
      \cs_if_exist:NT \FV@EndListFrame
```

```
          {
           \tag_mc_begin:n{artifact}
           \FV@EndListFrame
           \tag_mc_end:
          }
        \@endparenv
        \endgroup
        \@endpetrue
        }
```

(*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.

```
      \def\FV@ListProcessLine#1{%
        \hbox to \hsize{%
          \kern\leftmargin
          \hbox to \linewidth{%
            \cs_if_exist:NT \FV@LeftListNumber
             {
               \tag_struct_begin:n{tag=Lbl}
               \tag_mc_begin:n{}
               \FV@LeftListNumber
               \tag_mc_end:
               \tag_struct_end:
             }
            \cs_if_exist:NT \FV@LeftListFrame
             {
               \tag_mc_begin:n{artifact}
               \FV@LeftListFrame
               \tag_mc_end:
             }
            \tag_struct_begin:n{tag=codeline}
            \tag_mc_begin:n{}%
            \FancyVerbFormatLine{#1}%
            \tag_mc_end:
            \tag_struct_end:\hss
            \cs_if_exist:NT \FV@RightListFrame
             {
               \tag_mc_begin:n{artifact}
               \FV@RightListFrame
               \tag_mc_end:
             }
            \cs_if_exist:NT \FV@RightListNumber
             {
               \tag_struct_begin:n{tag=Lbl}
               \tag_mc_begin:n{}
               \FV@RightListNumber
               \tag_mc_begin:n{}
               \tag_struct_end:
             }
            }
            \hss}}
  }
\ExplSyntaxOff
```

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

```
⟨/package⟩
⟨*latex-lab⟩
\ProvidesFile{firstaid-latex-lab-testphase.ltx}
        [\ltlabfirstaiddate\space v\ltlabfirstaidversion\space
         latex-lab wrapper firstaid]

\RequirePackage{latex-lab-testphase-firstaid}

⟨/latex-lab⟩
```