The latex-lab-title package
Changes related to the tagging of the title

LaTeX Project*

v0.85g 2026-05-27

Abstract

1 Introduction

This module contains changes to improve the tagging (in the standard classes) of the title created with the `\maketitle` command. It also improves the setting of the metadata related to the title and the author.

For basic tagging of the printed title there are basically three things to do:

- The actual title should be tagged with the `Title` tag.
- The tabular used to format the author list should *not* be tagged as a tabular.
- `\maketitle` redefines footnote internals. These must be made tagging aware.

A second task related to title is to store the authors and the title text (or a shorter version) inside the XMP-metadata and (in PDF 1.7 or lower) in the Info dictionary. Currently this can only be set if hyperref is loaded and requires the use of the `pdftitle` and `pdfauthor` keys. The new code therefore extends the `\title` and `\author` commands: They stores their argument and use them at the end of the document for the PDF metadata if the data hasn't been given in another way. The code also gives `\title` and `\author` an optional argument where the PDF title or author can be given with in a key-value syntax. As with hyperref it is possible to store titles in more than one language:

```
\title
  [pdftitle =
     {[en]English Title,[de] Deutscher Titel,[fr]{titre français, avec comma}}]
  {Document title}
```

It is also possible to set a subtitle which is then stored in the XMP-metadata:

```
\title
  [pdfsubtitle =
     {[en]English Subtitle,[de] Deutscher Subtitel,[fr]{subtitre français, avec comma}
  {Document title}
```

*Initial implementation done by Ulrike Fischer

If using the `pdfauthor` key authors should be separated by commas, and to hide commas in a name inside braces if needed:

```
\author[pdfauthor = {Bär, Peter Anteater, {Riley, the sloth}}]{\ldots}
```

If hyperref is loaded there is no difference to the `pdftitle` and `pdfauthor` key used in `\hypersetup`. Both can be used (and the last key used will win).

1.1 Open questions and TODOs

- Writing into the Info dictionary needs to convert the input into a PDF string. This is here done with a simple version of hyperref's `\pdfstringdef`, similar code exist also in the generic hyperref driver. This should be moved into a better place module.
- Is it sensible to enhance `\author` and `\title` with an optional argument as done here? An advantage is that it is rather light-weight and doesn't require to decide how this values should be set in `\DocumentMetadata` (and would also work without `\DocumentMetadata`. But a problem could be that various classes and packages already extend this commands with other optional arguments.
- Some of the definitions related to metadata should perhaps be moved into l3pdfmeta.
- Are the names `pdftitle` and `pdfauthor` ok?
- The patch for `\thanks` to get a rlap-footnotemarker looks wrong. This probably means that some configuration option is missing in the footnote code.

2 Implementation

```
⟨*package⟩
⟨@@=tag⟩
\ProvidesExplPackage {latex-lab-testphase-title} {\ltlabtitledate} {\ltlabtitleversion}
  {Changes related to the tagging of the title}
```

2.1 `\maketitle` in article class

```
\cs_new_protected:Npn \__tag_patch_thanks:n #1
  {
    \rlap{\footnotemark}
    \protected@xdef\@thanks{\@thanks
      \protect\footnotetext[\the\c@footnote]{#1}}
  }
```

The no-titlepage version of article, report and book

```
\cs_new_protected:Npn \__tag_patch_maketitle:
  {
    \par
    \begingroup
```

Disable table tagging

```
      \tagpdfsetup{table/tagging=false}%
      \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
```

the original definition redefines `\@makefnmark` and `\@makefntext` to get an rlap-mark in the text without affecting the mark in the note (which gives by the way a wrong link area with hyperref). There seem to be currently no good way in the footnote to configure this, so we redefine `\thanks` instead

```
      \cs_set_eq:NN \thanks \__tag_patch_thanks:n
      \if@twocolumn
        \ifnum \col@number=\@ne
          \@maketitle
        \else
          \twocolumn[\@maketitle]%
        \fi
      \else
      \newpage
        \global\@topnum\z@   % Prevents figures from going at top of page.
        \@maketitle
      \fi
      \thispagestyle{plain}\@thanks
    \endgroup
    \setcounter{footnote}{0}%
    \global\let\thanks\relax
    \global\let\maketitle\relax
    \global\let\@maketitle\relax
    \global\let\@thanks\@empty
    \global\let\@author\@empty
    \global\let\@date\@empty
    \global\let\@title\@empty
    \global\let\title\relax
    \global\let\author\relax
    \global\let\date\relax
    \global\let\and\relax
  }
```

We must also change `\@maketitle` to insert a `Title` tag. We have here two options: We can surround the whole block with a `Title` or only the title text (but not both). As default the title text has the `Title` tag and the block is a `Part` (from the symbolic name `para/semantic`), but this can be changed by reassigning the symbolic name `title/block` and changing the plug of the tagging socket to the transparent plug, e.g.,
`\AssignStructureRole{title/block}{Title}`
`\AssignTaggingSocketPlug{title}{transparent}`

Note that in PDF 1.7 the `Title` tag is not a standard tag, it is a user defined and by default rolemapped to `P`. If `Title` should be used for the title block the rolemapping must be adjusted, e.g., to `Part` or `Sect`.

```
\AssignStructureRole{title/block}{\UseStructureName{para/semantic}}
\IfSocketExistsF{tagsupport/title}
 {
  \NewTaggingSocket{title}{2}
 }
\NewTaggingSocketPlug{title}{kernel}
 {
   \pdf_version_compare:NnTF > {1.7}
    {\tag_struct_begin:n{tag=\UseStructureName{title/text}}#2\tag_struct_end:}
    {\AssignStructureRole{para/textblock}{\UseStructureName{title/text}}#2}%
 }
```

```
\AssignTaggingSocketPlug{title}{kernel}
\cs_new_protected:Npn \__tag_patch_@maketitle:
 {
  \newpage
  \null
  \vskip 2em%
  \begin{center}%
   [tagging-suppress-paras=true,
    tagging-recipe=standalone,
    tag-name=\UseStructureName{title/block}]
  \let \footnote \thanks
```

By default this socket adds `Title` around the title text. In PDF 1.7 `Title` is rolemapped to P and the socket adapts the `para/textblock` to avoid that we get an `P` inside a `P`. One can assign the `transparent` plug to the socket if `Title` should be used for the title block instead.

```
  {\LARGE\UseTaggingSocket{title}{}{\@title}\par}
   \vskip 1.5em%
    {\large
      \lineskip .5em%
      \begin{tabular}[t]{c}%
        \@author
      \end{tabular}\par}%
    \vskip 1em%
    {\large \@date}%
  \end{center}%
  \par
  \vskip 1.5em
  }

```

The titlepage variant:

```
\cs_new_protected:Npn \__tag_patch_maketitle_page:
  {\begin{titlepage}%
```

disable table tagging

```
  \tagpdfsetup{table/tagging=false}%
  \let\footnotesize\small
  \let\footnoterule\relax
  \let \footnote \thanks
  \null\vfil
  \vskip 60\p@
```

```
  \begin{center}%
   [tagging-suppress-paras=true,
    tagging-recipe=standalone,
    tag-name=\UseStructureName{title/block}]
```

By default this socket adds `Title` around the title text. In PDF 1.7 `Title` is rolemapped to P and the socket adapts the `para/textblock` to avoid that we get an `P` inside a `P`. One can assign the `transparent` plug to the socket if `Title` should be used for the title block instead.

```
   {\LARGE\UseTaggingSocket{title}{}{\@title}\par}
   \vskip 3em%
    {\large
```

```
     \lineskip .75em%
      \begin{tabular}[t]{c}%
        \@author
      \end{tabular}\par}%
      \vskip 1.5em%
    {\large \@date \par}%       % Set date in \large size.
  \end{center}\par
  \@thanks
  \vfil\null
  \end{titlepage}%
  \setcounter{footnote}{0}%
  \global\let\thanks\relax
  \global\let\maketitle\relax
  \global\let\@thanks\@empty
  \global\let\@author\@empty
  \global\let\@date\@empty
  \global\let\@title\@empty
  \global\let\title\relax
  \global\let\author\relax
  \global\let\date\relax
  \global\let\and\relax
  }

```

Map the new commands onto `\maketitle`:

```
\AddToHook{class/article/after}
  {
    \if@titlepage
     \cs_set_eq:NN \maketitle \__tag_patch_maketitle_page:
    \else
     \cs_set_eq:NN \maketitle \__tag_patch_maketitle:
     \cs_set_eq:NN \@maketitle \__tag_patch_@maketitle:
    \fi
  }
\AddToHook{class/report/after}
  {
    \if@titlepage
     \cs_set_eq:NN \maketitle \__tag_patch_maketitle_page:
    \else
    \cs_set_eq:NN \maketitle \__tag_patch_maketitle:
    \cs_set_eq:NN \@maketitle \__tag_patch_@maketitle:
    \fi
  }
\AddToHook{class/book/after}
  {
    \if@titlepage
     \cs_set_eq:NN \maketitle \__tag_patch_maketitle_page:
    \else
    \cs_set_eq:NN \maketitle \__tag_patch_maketitle:
    \cs_set_eq:NN \@maketitle \__tag_patch_@maketitle:
    \fi
  }
```

2.2 Helper commands to set metadata

Some temp variables

```
\str_new:N \g__tag_title_tmpa_str
\str_new:N \l__tag_title_tmpa_str
\tl_new:N  \l__tag_title_tmpa_tl
\seq_new:N \l__tag_title_tmpa_seq
```

We ensure that the default definitions of `\@title` and `\@author` are robust:

```
\protected\def\@title{\@latex@error{No~\noexpand\title given}\@ehc}
\protected\def\@author{\@latex@warning@no@line{No~\noexpand\author given}}
```

A helper command to convert the title into a pdfstring similar to `\pdfstringdef`.

```
\cs_new_protected:Npn \__tag_title_pdfstring:nnN #1 #2 #3 % #1 text, #2 e.g. utf16/hex
  {
    \group_begin:
    \pdf_purify:nN {#1}\l__tag_title_tmpa_tl
    \pdf_string_from_unicode:nVN { #2 } \l__tag_title_tmpa_tl \l__tag_title_tmpa_str
    \str_gset_eq:NN \g__tag_title_tmpa_str\l__tag_title_tmpa_str
    \group_end:
    \str_set_eq:NN #3 \g__tag_title_tmpa_str
  }
\cs_generate_variant:Nn\__tag_title_pdfstring:nnN {e}
```

2.3 Extend title to set metadata

At first a variable to store the title, as `\@title` is emptied by LaTeX.

```
\tl_new:N \g__tag_title_title_tl
```

Now we redefine `\title` so that it stores the title, and processes keys in the optional argument. We use `hyp` as module name for the key as this means that if hyperref is loaded its definition of `pdftitle` will be used – at some time probably this should be moved out of hyperref so that we have only one definition.

```
\RenewDocumentCommand\title{O{}m}
 {
    \gdef\@title{#2}
    \tl_gset_eq:NN\g__tag_title_title_tl\@title
    \keys_set:nn {hyp}{#1}
 }
```

Now we define the `pdftitle` key. This is more or less the same definition as in the generic hyperref driver.

```
\regex_new:N\l__tag_title_optlang_regex
\regex_set:Nn\l__tag_title_optlang_regex {\A\[([A-Za-z\-]+)\](.*)}
\cs_generate_variant:Nn \clist_item:nn {on}
```

and now the keys.

```
\keys_define:nn { hyp }
   {
     pdftitle  .code:n =
       {
         \tl_if_blank:nTF {#1}
           {
             \pdfmanagement_remove:nn {Info}{Title}
           }
```

```
            {
              \tl_set:Ne\l__tag_title_tmpa_tl {\clist_item:on{#1}{1}}
              \regex_extract_once:NVN
                 \l__tag_title_optlang_regex
                 \l__tag_title_tmpa_tl
                 \l__tag_title_tmpa_seq
              \seq_if_empty:NTF\l__tag_title_tmpa_seq
               {
                 \__tag_title_pdfstring:nnN {#1}{utf16/hex}\l__tag_title_tmpa_str
               }
               {
                 \__tag_title_pdfstring:enN
                    {\seq_item:Nn \l__tag_title_tmpa_seq{3}}{utf16/hex}\l__tag_title_tmpa_str
               }
              \str_if_eq:VnF\l__tag_title_tmpa_str{<FEFF>}
                {
                  \pdfmanagement_add:nne {Info}{Title}{\l__tag_title_tmpa_str}
                }
            }
           \AddToDocumentProperties[hyperref]{pdftitle}{#1}
       }
   ,pdfsubtitle .code:n = { \AddToDocumentProperties[hyperref]{pdfsubtitle}{#1} }
   }
```

2.4 Extend \author to set metadata

At first a variable to store the authors, as `\@author` is emptied by LaTeX.

```
\tl_new:N \g__tag_title_author_tl
```

Now we redefine `\author` so that it stores the authors, and processes keys in the optional argument. We use `hyp` as module name for the key as this means that if hyperref is loaded its definition of `pdfauthor` will be used – at some time probably this should be moved out of hyperref so that we have only one definition.

```
\RenewDocumentCommand\author{O{}m}
 {
    \gdef\@author{#2}
    \tl_gset_eq:NN\g__tag_title_author_tl\@author
    \keys_set:nn {hyp}{#1}
 }
```

Now we define the `pdfauthor` key. This is more or less the same definition as in the generic hyperref driver.

```
\keys_define:nn { hyp }
   {
     pdfauthor  .code:n =
       {
         \tl_if_blank:nTF {#1}
           {
             \pdfmanagement_remove:nn {Info}{Author}
           }
           {
             \tl_set:Ne\l__tag_title_tmpa_tl {\clist_item:on{#1}{1}}
             \regex_extract_once:NVN
```

```
                \l__tag_title_optlang_regex
                \l__tag_title_tmpa_tl
                \l__tag_title_tmpa_seq
              \seq_if_empty:NTF\l__tag_title_tmpa_seq
               {
                 \__tag_title_pdfstring:nnN {#1}{utf16/hex}\l__tag_title_tmpa_str
               }
               {
                 \__tag_title_pdfstring:enN
                    {\seq_item:Nn \l__tag_title_tmpa_seq{3}}{utf16/hex}\l__tag_title_tmpa_str
               }
              \str_if_eq:VnF\l__tag_title_tmpa_str{<FEFF>}
                {
                  \pdfmanagement_add:nne {Info}{Author}{\l__tag_title_tmpa_str}
                }
            }
           \AddToDocumentProperties[hyperref]{pdfauthor}{#1}
       }
   }
```

2.5 Fallback for classes and packages that redefine \title or \author

If a class redefines **\author** and **\title** again, we try to retrieve at least the values.

```
\AddToHook{cmd/maketitle/before}
  {
   \tl_gset_eq:NN \g__tag_title_author_tl\@author
   \tl_gset_eq:NN \g__tag_title_title_tl\@title
  }
```

2.6 Finalize document

At last we set the title and the author at the end of document if that hasn't happened yet:

```
\AddToHook{shipout/lastpage}
 {
   \tl_if_empty:eT{\GetDocumentProperty{hyperref/pdftitle}}
      {
        \group_begin:
        \cs_set_eq:NN\thanks \use_none:n
        \str_set:Ne \l__tag_title_tmpa_str {\text_purify:n { \g__tag_title_title_tl } }
        \keys_set:ne{hyp}{pdftitle={\exp_not:V\l__tag_title_tmpa_str}}
        \group_end:
      }
   \tl_if_empty:eT{\GetDocumentProperty{hyperref/pdfauthor}}
      {
        \group_begin:
        \cs_set_eq:NN\thanks \use_none:n
        \cs_set:Npn \and {,}
        \str_set:Ne \l__tag_title_tmpa_str {\text_purify:n { \g__tag_title_author_tl } }
        \keys_set:ne{hyp}{pdfauthor={\exp_not:V\l__tag_title_tmpa_str}}
        \group_end:
      }
```

force display title, if an UA-standard is detected.

```
   \tl_if_empty:eF{\GetDocumentProperty{document/pdfstandard-UA}}
     {
       \pdfmanagement_add:nnn {Catalog / ViewerPreferences } { DisplayDocTitle } { true }
     }
 }
\DeclareHookRule{shipout/lastpage}{latex-lab-testphase-title}{before}{pdfmanagement-testphase
⟨/package⟩
⟨*latex-lab⟩
\ProvidesFile{title-latex-lab-testphase.ltx}
        [\ltlabtitledate\space v\ltlabtitleversion\space
         Changes related to the tagging of the title]

\RequirePackage{latex-lab-testphase-title}

⟨/latex-lab⟩
```