The latex-lab-toc package
Changes related to the tagging of toc and similar lists

LaTeX Project*

v2026-08-12 0.85i

Abstract

Header for the testphase package

```
⟨*header⟩
\ProvidesExplPackage {latex-lab-testphase-toc} {\ltlabtocdate} {\ltlabtocversion}
  { Code related to the tagging of toc-like lists}
⟨/header⟩
```

1 Introduction

The following contains various functions related to the tagging of the table of contents and similar list.

The structure of tocs consist of nested TOC and TOCI structures. The code uses the first argument of the `\contentsline` command to detect the level and to decided if a structure should be closed. The structure that should be used in `/Ref` key to link to the heading is detected from the target name in the fourth argument – because of this with this code such a target name is created and stored also if hyperref is not loaded.

1.1 Manual toc additions

As the `/Ref` key relies on the target name, manual `\addcontentsline` commands must ensure that they reference the right structure. If an unnumbered heading command is used before this is normally the case, so the following should work fine:

```
\chapter*{Unnumbered}
\addcontentsline{toc}{chapter}{Unnumbered}
```

If there is no heading command a target name must be created manually *inside the right structure* with `\MakeLinkTarget`:

```
\noindent % start e.g. P-structure
\MakeLinkTarget*{unnumbered}% target inside the P-structure
Unnumbered
\addcontentsline{toc}{chapter}{Unnumbered}
```

```
⟨*package⟩
⟨@@=tag⟩
```

*Initial implementation done by Ulrike Fischer

2 Temporary variables

\l__tag_toc_tmpa_tl

```
\tl_new:N \l__tag_toc_tmpa_tl
```

(*End of definition for* `\l__tag_toc_tmpa_tl`.)

3 General struct commands

The following variables and commands are not restricted to toc, but probably will be need in other places too.

\g__tag_struct_dest_num_prop

This variable records for (some or all, not clear yet) destination names the related structure number to allow to reference them in a Ref. The key is the destination. Defined by tagpdf.
(*End of definition for* `\g__tag_struct_dest_num_prop`.)

4 Toc code

\g__tag_toc_level_int
\g__tag_toc_stack_seq

`\g_@@_toc_level_int` records in a toc the current absolute level. We must close open structures at the end of the toc, for this we maintain a stack `\g_@@_toc_stack_seq`.

```
\int_new:N \g__tag_toc_level_int
\seq_new:N \g__tag_toc_stack_seq
```

(*End of definition for* `\g__tag_toc_level_int` *and* `\g__tag_toc_stack_seq`.)

_tag_toc_starttoc_init:n

The init code clears the stack, and set the level to -100 and start to TOC structure. We also disable paratagging. The `/Title` is currently simply the type, but this could be done nicer. The argument is the argument of `\@starttoc`, so, e.g., `toc`.

```
\cs_new_protected:Npn \__tag_toc_starttoc_init:n #1
 {
    \bool_set_false:N \l__tag_para_bool
    \seq_gclear:N \g__tag_toc_stack_seq
    \int_gset:Nn  \g__tag_toc_level_int {-100}
    \tag_struct_begin:n{tag=\UseStructureName{toc},title=#1}
 }
```

(*End of definition for* `\_tag_toc_starttoc_init:n`.)

Now define the tagging plug and assign it to the socket.

`default` (*plug*)

```
\NewTaggingSocketPlug{toc/starttoc/before}{default}
 {
   \__tag_toc_starttoc_init:n{#1}
 }
\AssignTaggingSocketPlug{toc/starttoc/before}{default}
```

__tag_toc_starttoc_finalize:

```
\cs_new_protected:Npn \__tag_toc_starttoc_finalize:
 {
    \int_step_inline:nn
      {\seq_count:N \g__tag_toc_stack_seq }
      {\tag_struct_end:}
    \tag_struct_end:
    \seq_gclear:N \g__tag_toc_stack_seq
 }
```

(*End of definition for* `\__tag_toc_starttoc_finalize:`.)

Now define the tagging plug and assign it to the socket.

`default` (*plug*)

```
\NewTaggingSocketPlug{toc/starttoc/after}{default}
 {
   \__tag_toc_starttoc_finalize:
 }
\AssignTaggingSocketPlug{toc/starttoc/after}{default}
```

__tag_toc_end:n This commands ends all TOC on the stack with a level higher than the argument

```
\cs_new_protected:Npn \__tag_toc_end:n #1
 {
   \seq_get:NNT\g__tag_toc_stack_seq \l__tag_toc_tmpa_tl
     {
       \bool_lazy_and:nnT
         {
           \str_if_eq_p:ee{\tl_head:N\l__tag_toc_tmpa_tl}{TOC}
         }
         {
           \int_compare_p:nNn {#1}<{\tl_tail:N \l__tag_toc_tmpa_tl}
         }
         {
           \seq_gpop:NN\g__tag_toc_stack_seq \l__tag_toc_tmpa_tl
           \tag_struct_end:
           \__tag_toc_end:n{#1}
         }
     }
 }
\cs_generate_variant:Nn \__tag_toc_end:n {e}
```

(*End of definition for* `\__tag_toc_end:n`.)

__tag_toc_contentsline_begin:nnnn This is main command executed at the begin of a `\contentsline`.

```
\cs_new_protected:Npn \__tag_toc_contentsline_begin:nnnn #1 #2 #3 #4
  %#1 level, #2 content, #3 page number (unused) #4 destination
  {
```

We detect the intended level by checking the value of `toclevel@...` (currently only provided by hyperref, but should be there always). To be on the safe side we set it to 1 if not defined.

```
    \ExpandArgs{c}\providecommand { toclevel@#1 }{ 1 } %  just in case ...
    \int_compare:nNnF { \use:c{toclevel@#1} } > {\use:c{c@tocdepth}}
     {
```

if level goes up, start new sub TOC unless we are at the begin

```
        \bool_lazy_and:nnT
          { \int_compare_p:nNn { \g__tag_toc_level_int } > {-100} }
          { \int_compare_p:nNn { \use:c{toclevel@#1} }   > { \g__tag_toc_level_int } }
          {
            \seq_gpush:Ne \g__tag_toc_stack_seq {{TOC}\use:c{toclevel@#1}}
            \tag_struct_begin:n{tag=\UseStructureName{toc}}
          }
```

if level goes down close all TOC's with a higher level

```
        \int_compare:nNnT
          { \use:c{toclevel@#1} } < { \g__tag_toc_level_int }
          {
            \__tag_toc_end:e { \use:c{toclevel@#1} }
          }
```

if same level do nothing update toclevel to the current level.

```
        \int_gset:Nn \g__tag_toc_level_int { \use:c{toclevel@#1} }
```

now open the TOCI, the tagging of the inner structure is left to the `\l@xxx` commands. setting the title is not strictly necessary but looks nicer but we have to remove the `\numberline`

```
        \group_begin:
         \text_declare_expand_equivalent:Nn \numberline \use_none:n
         \pdf_purify:nN {#2}\l__tag_toc_tmpa_tl
         \tag_struct_begin:n{tag=\UseStructureName{toc/item},title-o=\l__tag_toc_tmpa_tl}
```

The TOCI structure should get a /Ref, we use a destination to retrieve it.

```
         \tag_struct_gput:nnn { \tag_get:n {struct_num} }{ref_dest}{#4}
         \seq_gpush:Ne \g__tag_toc_stack_seq {{\UseStructureName{toc/item}}\use:c{toclevel@#1
        \group_end:
    }
  }
```

(*End of definition for* `\__tag_toc_contentsline_begin:nnnn`.)

Now define the tagging plug and assign it to the socket.

`default` (*plug*)

```
\NewTaggingSocketPlug{toc/contentsline/before}{default}
  {
    \__tag_toc_contentsline_begin:nnnn #1
  }
\AssignTaggingSocketPlug{toc/contentsline/before}{default}
```

`\__tag_toc_contentsline_end:nnnn` This is the closing code of a `\contentsline`. If the contentsline was actually printed, the code has to close the TOCI structure and to update the stack.

```
\msg_new:nnn {tag}{toc-no-TOCI}{Missing~TOCI~structure~on~toc~stack}

\cs_new_protected:Npn \__tag_toc_contentsline_end:nnnn #1 #2 #3 #4
% #1 level, #2 content (unused), #3 page number (unused) #4 destination (unused)
  {
    \int_compare:nNnF { \use:c{toclevel@#1} } > {\use:c{c@tocdepth}}
      {
```

```
        \seq_gpop:NNT \g__tag_toc_stack_seq\l__tag_tmpa_tl
          {
            \str_if_eq:eeTF{\tl_head:N\l__tag_tmpa_tl}{\UseStructureName{toc/item}}
             {
               \tag_struct_end:
             }
             {
               \msg_warning:nn{tag}{toc-no-TOCI}
             }
          }
      }
  }
```

(*End of definition for* `\__tag_toc_contentsline_end:nnnn`.)

Now define the tagging plug and assign it to the socket.

`default` (*plug*)

```
\NewTaggingSocketPlug{toc/contentsline/after}{default}
  {
    \__tag_toc_contentsline_end:nnnn #1
  }
\AssignTaggingSocketPlug{toc/contentsline/after}{default}
```

4.1 Tagging of the content

This needs discussion.

```
\AddToHook{contentsline/text/before}[tagpdf]{%
  \tag_struct_begin:n{tag=\UseStructureName{toc/item/text}}%
  \tag_mc_begin:n{}}
\AddToHook{contentsline/text/after}[tagpdf]{%
  \tag_mc_end:}
\AddToHook{contentsline/page/before}[tagpdf]{%
  \tag_mc_begin:n{}}
\AddToHook{contentsline/page/after}[tagpdf]{%
  \tag_mc_end:
  \tag_struct_end:} %Reference
\AddToHook{contentsline/number/before}[tagpdf]{%
  \tag_mc_end:
  \tag_struct_begin:n{tag=\UseStructureName{toc/item/label}}%
  \tag_mc_begin:n{}}
\AddToHook{contentsline/number/after}[tagpdf]{%
  \tag_mc_end:
  \tag_struct_end:
  \tag_mc_begin:n{}}
```

`artifact` (*plug*)

```
\NewTaggingSocketPlug{toc/leaders/before}{artifact}
 {\tag_mc_begin:n{artifact}\nobreak}
\NewTaggingSocketPlug{toc/leaders/after}{artifact}
 {\nobreak\tag_mc_end:}
\AssignTaggingSocketPlug{toc/leaders/before}{artifact}
\AssignTaggingSocketPlug{toc/leaders/after}{artifact}
```

⟨/package⟩