

The `latex-lab-sec` package

Changes related to the tagging of sectioning commands

L^AT_EX Project*

v0.84c 2024-07-05

Abstract

The following code implements a first draft for the tagging of sectioning commands.

1 Limitations

Sectioning commands are in not defined by the format but by the classes. Their implementation vary: some are defined with the help of `\@startsection`, some are like `\chapter` handcrafted, some build with the help of extension packages or as in the KOMA classes with class code that extends the `\@startsection` functionality.

The following code can therefore currently be used *only* with the standard classes or with classes which do not overwrite the changed definitions.

2 Introduction

Tagging of sectioning commands consist of two parts:

- The heading/title text of the section should be surrounded by a heading tag, typically `Hn` with some value of `n`. The number of the section command can optionally be put in a `Lb1`. The number of the `Hn` tag should reflect the “natural” level. So in an article `\section` will use `H1`, in a book `\chapter` will use `H1` and `\section` `H2`. Titles of `\part` are a bit out of this system as they are normally not part of the hierarchy: often only some chapters are grouped under a part. Their title is therefore tagged as `Title`.
- The whole section should normally be surrounded by a `Sect` tag. Parts should be surrounded by `Part`. It is a bit unclear if the headings should be inside or outside of these structures—the best practice guide puts them outside—but on the whole it sounds more logical to group the heading with the text inside the `Sect`. For the part this is actually required, as there can be only one `Title` in a structure, so the part title can’t be at the same level as the document `Title`.

Starting such an enclosing `Sect` structure is rather easy, but closing it requires code in various place, for example the commands `\mainmatter`, `\backmatter`, `\frontmatter` and `\appendix` should typically close everything. Following sectioning commands should close all previous structures with a level equal or higher than their own level.

*Initial implementation done by Ulrike Fischer

3 Technical details and problems

The implementation has to take care of various details.

- As sections in L^AT_EX are not environments, the <Sect> structures can be wrongly nested with other structures. For example if a document puts a sectioning command into a list or a trivlist or a minipage then it can no longer close previous <Sect> structures correctly. The problem can be detected by checking the structure stack and a warning can be issued, but the author then has to close the structures manually before the list or minipage.

Thus there have to be user interfaces to handle such cases. It should also be possible not to create all the <Sect> structures automatically but to tag only the headings so that the author can handle special cases manually.

- If hyperref is used, targets for links should be inserted, either with `\refstepcounter` or manually with `\MakeLinkTarget`. These targets must be in the correct structure for the structure destinations. They replace some of the current patches in hyperref.
- With lualatex the mc-commands set attributes *locally*, so the commands must be at the right grouping level.

3.1 Functions and keys

```
\tag_tool:n  
\tagtool
```

3.2 TODO

- A dedicated command to close a sectioning unit should be provided.
- A dedicated command to open a sectioning unit should be provided too.
- It should also be possible to suppress the sectioning unit in sectioning commands to allow e.g. to put an epigraph or similar in front.
- The number in `\part` and `\chapter` is currently not correctly tagged as a Lbl as this requires to redefine the internal (class dependent) commands too.

¹ `{*package}`

4 Implementation

```
2 \ProvidesExplPackage {latex-lab-testphase-sec} {\ltlabsecdate} {\ltlabsecversion}  
3 {Code related to the tagging of sectioning commands}
```

4.1 Surrounding by Sect structures

We use a stack to record the levels of the open `Sect`. The first item has level -100. A sectioning command will take a record from the stack. If its level is greater or equal it

closes this structure and takes the next record from the stack. If the record has a smaller level then it puts it back and stops. The stack is compared with the main structure stack, if they don't match it means we can't safely close the `Sect` and so we issue a warning and do nothing.

4.1.1 Loading general kernel changes

[kernel?] Also loaded in the toc-tagging code.

```

4 \RequirePackage{latex-lab-kernel-changes}
5 \end{package}

```

4.1.2 Glyphtounicode improvements

As lualatex runs with legacy encodings in the test files, we enable and load glyphtounicode. For the math we load additional definitions.

```

6 \kernelchange
7 \ifdefined\directlua
8 \ifnum\outputmode > 0
9 \pdfvariable gentounicode =1
10 \protected\def\pdfglyptounicode {\pdfextension glyphtounicode }
11 \protected\edef\pdfgentounicode {\pdfvariable gentounicode}
12 \input{glyphtounicode}
13 \fi
14 \fi
15 \ifdefined\pdfglyptounicode
16 \input{glyphtounicode-cmex}
17 \fi
18 \end{kernelchange}

```

4.1.3 updating \@currentHref

[kernel?]

We must ensure that manual targets (e.g. in unnumbered sections) correctly update `\@currentHref`. For this we extend the kernel definition of `\MakeLinkTarget`

```

19 \kernelchange
20 \ExplSyntaxOn
21 \int_new:N\g__kernel_target_int
22 \RenewDocumentCommand\MakeLinkTarget{s0{m}
23 {%
24 \ifvmode
25 \special{}%
26 \else
27 \savesf\spacefactor
28 \smash{}%
29 \spacefactor\@savs
30 \fi
31 \IfBooleanTF {#1}
32 {
33 \tl_gset:Ne \@currentHref {#3}
34 }
35 {
36 \int_gincr:N\g__kernel_target_int

```

```

37     \tl_gset:Ne \@currentHref {target*.\int_use:N\g__kernel_target_int}
38   }
39 }
40 \ExplSyntaxOff
41 \kernelchange)
42 (*package)

```

4.1.4 Tagging commands

\g__tag_sec_stack_seq The stack holds the tag and the level.

```

43 \seq_new:N \g__tag_sec_stack_seq
44 \seq_gpush:Nn\g__tag_sec_stack_seq {{Document}}{-100}}

```

\l__tag_sec_Sect_bool This boolean controls if a Sect structure is opened.

```

45 \bool_new:N \l__tag_sec_Sect_bool
46 \bool_set_true:N\l__tag_sec_Sect_bool

```

__tag_sec_begin:nn This starts a sectioning structure. Currently the tag is fix, either Sect or Part, depending on the level, but this will perhaps change. The second argument is currently unused.

```

47 \cs_new_protected:Npn\__tag_sec_begin:nn #1 #2 % #1 level #2 keyval
48 {
49   \tag_struct_begin:n
50   {
51     tag= {\int_compare:nNnTF {#1}={-1}}{Part}{Sect}}
52   ,#2
53   }
54   \seq_gpush:Ne \g__tag_sec_stack_seq {{\g__tag_struct_tag_tl}{\int_eval:n{#1}}}
55 }

```

(End of definition for __tag_sec_begin:nn.)

__tag_sec_end:n

```

56 \msg_new:nnn { tag } {wrong-sect-nesting}
57 {
58   The~structure~#1~can~not~be~closed.\\
59   It~is~not~equal~to~the~current~structure~#2~on~the~main~stack
60 }
61
62 \cs_new_protected:Npn\__tag_sec_end:n #1 % #1 level
63 {
64   \seq_get:NN \g__tag_sec_stack_seq \l__tag_tmpa_tl
65   \int_compare:nNnT {#1}<{\exp_last_unbraced:NV\use_ii:nn\l__tag_tmpa_tl+1}
66   {
67     \seq_get:NN\g__tag_struct_tag_stack_seq \l__tag_tmpb_tl
68     \exp_args:Nee
69     \tl_if_eq:nnTF
70     {\exp_last_unbraced:NV\use_i:nn\l__tag_tmpa_tl}
71     {\exp_last_unbraced:NV\use_i:nn\l__tag_tmpb_tl}
72     {

```

```

73         \seq_gpop:NN \g__tag_sec_stack_seq \l__tag_tmpa_tl
74         \tag_struct_end:
75         \__tag_sec_end:n {#1}
76     }
77     {
78         \msg_warning:nnee {tag}{wrong-sect-nesting}
79         { \exp_last_unbraced:NV\use_i:nn \l__tag_tmpa_tl }
80         { \exp_last_unbraced:NV\use_i:nn \l__tag_tmpb_tl }
81     }
82 }
83 }

```

(End of definition for __tag_sec_end:n.)

`__tag_tool_para_split:` Run-in-sectioning command must separate the heading from the following text.

```

84 \cs_new_protected:Npn \__tag_tool_para_split:
85 {
86     \tag_mc_end:
87     \tag_struct_end:
88     \tag_struct_begin:n{tag=\l__tag_para_tag_default_tl}
89     \tag_mc_begin:n{ }
90     \__tag_setup_restore_para_default:
91 }

```

(End of definition for __tag_tool_para_split:.)

`__tag_setup_restore_para_default:` We change the para tagging in the sectioning code. This here restores the default. Currently it only resets the the tag, but perhaps more will be needed later.

```

92 \cs_new_protected:Npn \__tag_setup_restore_para_default:
93 {
94     \tl_set:Nn \l__tag_para_main_tag_tl {text-unit}
95     \tl_set_eq:NN \l__tag_para_tag_tl \l__tag_para_tag_default_tl
96 }

```

(End of definition for __tag_setup_restore_para_default:.)

`__tag_sec_end_display:`

```

97 \cs_new_protected:Npn \__tag_sec_end_display:
98 {
99     \tag_struct_end: %P = Hn
100     \__tag_setup_restore_para_default:
101 }

```

(End of definition for __tag_sec_end_display:.)

Open sec structures should be closed at the end of the document. This should be done before tagpdf closes the Document structure.

```

102 \hook_gput_code:nnn{tagpdf/finish/before}{tagpdf/sec}{\__tag_sec_end:n{-10}}
103 \hook_gset_rule:nnnn {tagpdf/finish/before}{tagpdf/sec}{before}{tagpdf}

```

The commands `\mainmatter`, `\backmatter`, `\frontmatter` and `\appendix` close all **Sect** and **Part** structures.

```

104 \AddToHook{cmd/frontmatter/before}{\__tag_sec_end:n{-10}}
105 \AddToHook{cmd/mainmatter/before} {\__tag_sec_end:n{-10}}
106 \AddToHook{cmd/backmatter/before} {\__tag_sec_end:n{-10}}
107 \AddToHook{cmd/appendix/before}   {\__tag_sec_end:n{-10}}

```

4.2 Tagging tools

We need to provide user and package level commands

```

108 \cs_if_free:NT \tag_tool:n
109 {
110   \cs_new_protected:Npn \tag_tool:n #1
111     {
112       \tag_if_active:T { \keys_set:nn {tag / tool}{#1} }
113     }
114   \cs_set_eq:NN\tagtool\tag_tool:n
115 }
116 \keys_define:nn { tag / tool}
117 {
118   ,sec-start-part .code:n =
119     {
120       \bool_if:NT\l__tag_sec_Sect_bool
121         {
122           \__tag_sec_end:n {-1}
123           \__tag_sec_begin:nn{-1}{tag=Part}
124         }
125       \tag_struct_begin:n{tag=part,title=#1}

```

We remap here the text-unit from the paragraph to NonStruct. It would be better to suppress it completely as with the other sectioning commands, but this would require to redefine \@spart and \@part, as there is the grouping, and these commands are all slightly different in the standard classes. So this is delayed to the time when sectioning commands are redefined with templates.

```

126       \tl_set:Nn\l__tag_para_main_tag_tl {NonStruct}
127       \tl_set:Nn\l__tag_para_tag_tl {Span}
128     }
129   ,sec-stop-part .code:n = {\__tag_sec_end_display:}
130   ,sec-start-chapter .code:n =
131     {
132       \bool_if:NT\l__tag_sec_Sect_bool
133         {
134           \__tag_sec_end:n {0}
135           \__tag_sec_begin:nn{0}{tag=Sect}
136         }
137       \tag_struct_begin:n{tag=chapter,title=#1}

```

similar to part we remap to NonStruct for now ...

```

138       \tl_set:Nn\l__tag_para_main_tag_tl {NonStruct}
139       \tl_set:Nn\l__tag_para_tag_tl {Span}
140     }
141   ,sec-stop-chapter .meta:n = { sec-stop-part}
142   ,sec-start .code:n = % #1 is a name like "section"
143     {
144       \bool_if:NT\l__tag_sec_Sect_bool
145         {
146           \__tag_sec_end:n {\cs_if_exist_use:c{toclevel@#1}+0}
147           \__tag_sec_begin:nn {\cs_if_exist_use:c{toclevel@#1}+0}{tag=Sect}
148         }
149       \tl_set:Nn\l__tag_para_tag_tl{#1}
150     }

```

```

151 ,sec-start .value_required:n = true
152 ,sec-split-para .code:n = {\__tag_tool_para_split:}
153 ,restore-para .code:n = {\__tag_setup_restore_para_default:}
154 ,sec-stop .code:n =
155   {
156     \par\__tag_sec_end:n   {\cs_if_exist_use:c{toclevel@#1}+0}
157   }
158 ,sec-stop .value_required:n = true
159 ,sec-add-grouping .bool_set:N = \l__tag_sec_Sect_bool
160 }

```

5 Sectioning commands

5.1 \part and \chapter

\part and \chapter are defined by the classes. To tag them we redefine the user commands. This will probably break with various classes and with titlesec. The tagging inside relies on the para tagging. We do not yet use keyval in the optional argument, as this requires latex-dev and the naming of the keys and their key family is unclear.

```

161 \AddToHook{class/after}
162 {
163   \@ifundefined{chapter}
164   {

```

This redefines \part in article class.

```

165   \@ifundefined{part}{-}
166   {
167     \RenewDocumentCommand\part{ s O{#3} m }
168     {
169       \if@noskipsec \leavevmode \fi
170       \par
171       \addvspace{4ex}%
172       \@afterindentfalse

```

This are the tagging commands needed at the begin. They open a Part structure and the structure for the title of the heading.

```

173       % tagging start commands
174       \tag_tool:n {sec-start-part=#2}
175       % end tagging start commands

```

This adds a manual target if the part is unnumbered or starred. It replaces the hyperref patches.

```

176       \bool_lazy_any:nT
177       {
178         { #1 }
179         {
180           \int_compare_p:nNn {\c@secnumdepth}<{-1}
181         }
182       }
183       {
184         \MakeLinkTarget [part] {}
185       }

```

The main call to the underlying commands.

```
186     \IfBooleanTF
187     {#1}
188     { \@spart {#3} }
189     { \@part [#2]{#3} }
```

and now the closing command for the tagging of the title.

```
190     \tag_tool:n {sec-stop-part}
191     }
192   }
193 }
```

Redefinitions for book and report

```
194 {
195   \RenewDocumentCommand\chapter{ s O{#3} m }
196   {
197     \if@openright\cleardoublepage\else\clearpage\fi
198     \thispagestyle{plain}%
199     \global\@topnum\z@
200     \@afterindentfalse
```

This are the tagging commands needed at the begin. They open a Sect structure and the structure for the title of the heading.

```
201     \tag_tool:n { sec-start-chapter= #2 }
```

This adds a manual target if the chapter is unnumbered or starred. It replaces the hyperref patches.

```
202     \bool_lazy_any:nT
203     {
204       { #1 }
205       {
206         \int_compare_p:nNn {\c@secnumdepth}<{0}
207       }
208       {
209         %in book target also needed in frontmatter
210         \bool_lazy_and_p:nn
211         { \cs_if_exist_p:c { @mainmattertrue } }
212         { ! \legacy_if_p:n { @mainmatter } }
213       }
214     }
215     {
```

The relation target-struct is stored internally by the MakeLinkTarget commands

```
216     \MakeLinkTarget[chapter]{}
217   }
```

The main call to the underlying commands.

```
218     \IfBooleanTF
219     {#1}
220     { \@schapter {#3} }
221     { \@chapter [#2]{#3} }
```

and now the closing command for the tagging of the title.

```
222     \tag_tool:n {sec-stop-chapter}
223   }
```


and similar for `\part`

```
224     \RenewDocumentCommand\part{ s O{#3} m }
225     {
226         \if@openright
227             \cleardoublepage
228         \else
229             \clearpage
230         \fi
231         \thispagestyle{plain}%
232         \if@twocolumn
233             \onecolumn
234             \@tempwatrue
235         \else
236             \@tempwafalse
237         \fi
238         \null\vfil
```

This are the tagging commands needed at the begin. They open a Part structure and the structure for the title of the heading.

```
239     \tag_tool:n {sec-start-part=#2}
```

This adds a manual target if the part is unnumbered or starred. It replaces the hyperref patches.

```
240     \bool_lazy_any:nT
241     {
242         { #1 }
243         {
244             \int_compare_p:nNn {\c@secnumdepth}<{-1}
245         }
246         {
247             %in book target also needed in frontmatter
248             \bool_lazy_and_p:nn
249             { \cs_if_exist_p:c { @mainmattertrue } }
250             { ! \legacy_if_p:n { @mainmatter } }
251         }
252     }
253     {
254         \MakeLinkTarget [part] {}
255     }
```

The main call to the underlying commands.

```
256     \IfBooleanTF
257     {#1}
258     { \@spart {#3} }
259     { \@part [#2]{#3} }
```

and now the closing command for the tagging of the title.

```
260     \tag_tool:n{sec-stop-part}
261     }
262     }
263 }
```

5.2 Sectioning commands based on \@startsection

The tagging relies again on the para tagging: we simply exchange the tag name by the one given as #1. This assumes that a tag with the name of the sectioning type is defined. We don't try to pass the title, this will be done together with the new keyval handling in the user command.

5.2.1 Hyperref code

hyperref has to insert anchors. If the sectioning is numbered this is done by \refstepcounter (and so in vmode). For unnumbered section hyperref injects the anchor in hmode before the text, it also inserts a kern to compensate the indent.

This means that the target of numbered and unnumbered sectioning commands differ, both regarding the location and in relation to the tagging structure: The anchor from the \refstepcounter is outside of the structure created by the heading title if the para tags are used, while the other anchors are inside and so the structure destinations are different.

We unify this by suppressing the anchor from the refstepcounter. Also we only go back if the indent is positive.

At first suppress all hyperref patches related to sectioning:

```
264 \def\hyper@nopatch@sectioning{}
```

\@hyp@section@target@nnn A simple internal command. There is no need for something public, as packages defining their own version of \@startsection will probably need something slightly different based on \MakeLinkTarget.

```
265 \cs_new_protected:Npn \@hyp@section@target@nnn #1 #2 #3 % #1 optarg #2 name/counter, #3 indent
266 {
267   \makebox[0pt][l]
268   {
269     \skip_set:Nn \@tempskipa {#3}
270     \dim_compare:nNnF {\@tempskipa}<{0pt}{\kern-\@tempskipa}
271     \MakeLinkTarget#1{#2}
272   }
273 }
```

(End of definition for \@hyp@section@target@nnn. This function is documented on page ??.)

5.3 Adaption of the heading commands

We add to \@startsection the commands to open the Sect structure and to change the para tag.

```
274 \def\@startsection#1#2#3#4#5#6{%
275   \if@noskipsec \leavevmode \fi
276   \par
277   \@tempskipa #4\relax
278   \@afterindenttrue
279   \ifdim \@tempskipa <\z@
280     \@tempskipa -\@tempskipa \@afterindentfalse
281   \fi
282   \if@nobreak
283     \everypar{}%
284   \else
```

```

285 \addpenalty\@secpenalty\advspace\@tempkipa
286 \fi
287 \tag_tool:n { sec-start=#1}%new
288 \@ifstar
289   {\@ssect{#3}{#4}{#5}{#6}}%
290   {\@dblarg{\@sect{#1}{#2}{#3}{#4}{#5}{#6}}}}

```

To be able to correctly tag the number we need a special \@hangfrom variant. This is a bit tricky: As the paragraph starts after the \setbox luatex attributes are not set yet and numbers are unmarked if one doesn't pay attention. The code assumes that we are in vmode!

```

291 \cs_new_protected:Npn \@kernel@tag@hangfrom #1
292   {
293     \tagstructbegin{tag=\l__tag_para_tag_tl}
294     \cs_if_exist_use:N \__tag_gincr_para_begin_int:
295     \tagstructbegin{tag=Lbl}
296     \setbox\@tempboxa
297       \hbox
298       {

```

In lua mode we have to set the attributes inside the box!

```

299         \bool_lazy_and:nnT
300         {\tag_if_active_p:}
301         {\g__tag_mode_lua_bool}
302         {\tagmcbegin{tag=Lbl}}
303         {#1}
304       }

```

We stop tagging now, to avoid that the \noindent triggers the paratagging. We do not disable paratagging completely, to avoid that the numbering goes wrong.

```

305 \tag_stop:n{hangfrom}
306 \hangindent \wd\@tempboxa\noindent

```

Restart tagging and insert the box.

```

307 \tag_start:n{hangfrom}
308 \tagmcbegin{\box\@tempboxa\tagmccend\tagstructend\tagmcbegin{}}

```

This command is used to tag the numbers of runin. We do not try to avoid the empty container from the paratagging, this would require more changes.

```

309 \cs_new_protected:Npn \@kernel@tag@svsec
310   {
311     \tag_mc_end_push:
312     \tag_struct_begin:n{tag=Lbl}
313     \tag_mc_begin:n{}
314     \@svsec
315     \tag_mc_end:
316     \tag_struct_end:
317     \tag_mc_begin_pop:n{}
318   }

```

\@sect is only changed to replace the hyperref patches and to use the new \@kernel@tag@hangfrom and \@kernel@tag@svsec

```

319 \def\@sect#1#2#3#4#5#6[#7]#8{%
320   \ifnum #2>\c@secnumdepth
321     \def\@svsec{\hyp@section@target@nnn{[section]}-}{#3}}
322   \else

```

```

323 \LinkTargetOff
324 \refstepcounter{#1}%
325 \LinkTargetOn
326 \protected@edef\@svsec{\@hyp@section@target@nnn{#1}{#3}\@secntformat{#1}\relax}%
327 \fi
328 \@tempskipa #5\relax
329 \ifdim \@tempskipa>\z@
330 \begingroup
331 \tagtool{para-flattened=true} % or \bool_set_true\l__tag_para_flattened_bool
332 #6{%
333 \ifnum #2>\c@secnumdepth
334 \@hangfrom {\hskip #3\relax\@svsec}%
335 \else
336 \@kernel@tag@hangfrom{\hskip #3\relax\@svsec}%
337 \fi
338 \interlinepenalty \@M #8\@par}%
339 \endgroup
340 \csname #1mark\endcsname{#7}%
341 \addcontentsline{toc}{#1}{%
342 \ifnum #2>\c@secnumdepth \else
343 \protect\numberline{\csname the#1\endcsname}%
344 \fi
345 #7}%
346 \else
347 \def\@svsechd{%
348 #6{\hskip #3\relax
349 \ifnum #2>\c@secnumdepth
350 \@svsec
351 \else
352 \@kernel@tag@svsec
353 \fi #8}%
354 \csname #1mark\endcsname{#7}%
355 \addcontentsline{toc}{#1}{%
356 \ifnum #2>\c@secnumdepth \else
357 \protect\numberline{\csname the#1\endcsname}%
358 \fi
359 #7}}%
360 \fi
361 \@xsect{#5}}
similar for \@ssect
362 \def\@ssect#1#2#3#4#5{%
363 \@tempskipa #3\relax
364 \ifdim \@tempskipa>\z@
365 \begingroup
366 \tagtool{para-flattened=true}
367 #4{%
368 \@hangfrom{\hskip #1\relax\@hyp@section@target@nnn[[section]]{#1}}%
369 \interlinepenalty \@M #5\@par}%
370 \endgroup
371 \else
372 \def\@svsechd{#4{\hskip #1\relax\@hyp@section@target@nnn[[section]]{#3}\relax #5}}%
373 \fi
374 \@xsect{#3}}

```

At last `\@xsect` needs code in two places. For display headings it has to restore the default para code, for run in headings it has to separated the heading from the following text.

```

375 \def\@xsect#1{%
376   \@tempkipa #1\relax
377   \ifdim \@tempkipa>\z@
378     \par \nobreak
379     \vskip \@tempkipa
380     \tag_tool:n {restore-para}
381     \@afterheading
382   \else
383     \@nobreakfalse
384     \global\@noskipsectrue
385     \everypar{%
386       \if@noskipsec
387         \global\@noskipsecfalse
388         {\setbox\z@\lastbox}%
389         \clubpenalty\@M
390         \begingroup \@svsechd \endgroup
391         \unskip
392         \tag_tool:n {sec-split-para}
393         \@tempkipa #1\relax
394         \hskip -\@tempkipa
395       \else
396         \clubpenalty \@clubpenalty
397       \everypar{}}%
398     \fi}%
399   \fi
400   \ignorespaces}
401 \end{package}
402 \end{*latex-lab}
403 \ProvidesFile{sec-latex-lab-testphase.ltx}
404   [\ltxlabsecdate\space v\ltxlabsecversion\space latex-lab wrapper sec]
405 \RequirePackage{latex-lab-testphase-sec}
406 \end{*latex-lab}

```