The latex-lab-bib package
Changes and additions to the kernel related to tagging and links in citations and bibliography entries

LaTeX Project*

v0.81f 2026-06-26

Abstract

1 Introduction

The following contains small changes to improve tagging of bibliography entries and citations.

The tagging of the standard bibliography is actually quite straightforward: A bibliography is typically a list with a heading and the code which tags sectioning commands and lists handles that.

There are here only two problems:

- The structure number of the `LI` element created by a `\bibitem` must be recorded somehow to allow to reference it in a `\cite`.
- hyperref redefines the item command and so breaks the list structure see https://github.com/latex3/latex2e/discussions/1010#discussioncomment-5565418

Both problems are rather easy to resolve, but it must be checked if other packages interfere again by redefining the commands.

More difficult is the tagging of citation commands. Citations should be inside a Reference structure and contain a /Ref entry pointing to the relevant item in the bibliography. For simple citations like "[1]" or "Doody (2023)" this is easy, but it is not obvious how to handle combined citations like "Doody (2003,2018)" (or even compressed citations like "[1-3]"). The current implementation follows the links: whatever hyperref would link is set as the reference.

There exist various packages which over the years tried to improve and extend the bibliography commands. We discuss here three: natbib, chapterbib and biblatex.

natbib It is rather easy to support natbib: it has hooks for links and the tagging code can follow. Only a bit coordination with hyperref is needed to avoid that hyperref remove the tagging code again.

*Initial implementation done by Ulrike Fischer

chapterbib In standard LaTeX every bib entry has an unique label which points to the (mandatory and unique) bibliography and the target created by hyperref has the simple form `cite.`⟨*key*⟩. If a package that support multiple bibliographies is used (e.g. chapterbib) this is no longer works: a bib entry `doody` can in one chapter get the label "[1]" and in the other "[5]" or even "Doo19" and naturally links should jump to the relevant chapter bibliography. chapterbib solves this by creating bib keys with a suffix: when reading the `.aux` files it will create the keys `doody@-1` and `doody@-2` where the number is related to the chapter/include, and in the document and in the document `\cite{doody}` will look for `doody@-1` and `doody@-2` depending on the number of the current include. For some unknown reason chapterbib uses two commands to handle the suffix: the command `\@extra@binfo` is written to the aux-files and used when processing the `\bibcite` commands, but in the document `\@extra@b@citeb` is used. Supporting this is straightforward: one only has to take care that the tagging code uses `\@extra@b@citeb` in the relevant places too.

biblatex biblatex supports multiple bibliographies out-of-the-box. It numbers the link target by refsection and uses then the name `\the\c@refsection @`⟨*key*⟩.

Printing a bibliography is not required, in this case you get an engine warning and links jump to the begin of the document:

```
name{cite.0@doody} has been referenced but does not exist
```

Bibliographies can be printed more than once by refsection. To avoid duplicated target, biblatex stores the names of the targets in a list and if later it detects that a target name has already been used in a bibliography no new target is created for this item. This means a citation will normally jump to the first bibliography which shows the entry.

The tagging code has to mimic this code. This means that it can't label every item, but has to test if this anchor is already known.

2 Provided or redefined commands

`\@extra@binfo`
`\@extra@b@citeb`

These are taken from hyperref, they are for chapterbib compatibility (and also signal to chapterbib not to change the citation commands)

`\@bibitem`
`\@lbibitem`

The internal item commands.

3 Implementation

```
⟨*package⟩
⟨@@=tag⟩

\ProvidesExplPackage {latex-lab-testphase-bib} {\ltlabbibdate} {\ltlabbibversion}
  {Code related to the tagging of bibliography and cite command}
```

\@extra@binfo
\@extra@b@citeb

These are taken from hyperref, they are for chapterbib compatibility (and also signal to chapterbib not to change the citation commands)

```
\providecommand*\@extra@binfo{}%
\providecommand\@extra@b@citeb{}
```

(*End of definition for* `\@extra@binfo` *and* `\@extra@b@citeb`. *These functions are documented on page* 2.)

3.1 Handling the bibliography

\@lbibitem

This is the command used if `\bibitem` has an optional argument.
The link target is now added by the block code in the `\item` command (as long as the bibliography is built as a list). We only need to change the name. This also works if hyperref is not loaded as the kernel `\MakeLinkTarget` also contains the `target/setname/after` hook.

```
\AddToHookWithArguments{cmd/@lbibitem/before}[latex-lab-testphase-bib/target]
 {
   \ExpandArgs{e}\NextLinkTarget{cite.#2\@extra@b@citeb}
 }
```

We make a copy to be able to reinstate the definition. This is, e.g., currently needed with hyperref.

```
\let\@kernel@copy@lbibitem\@lbibitem
```

(*End of definition for* `\@lbibitem`. *This function is documented on page* 2.)

\@bibitem

Similar for `\@bibitem`.
The target is added by the block code. We only need to change the name.

```
\AddToHookWithArguments{cmd/@bibitem/before}[latex-lab-testphase-bib/target]
 {
   \ExpandArgs{e}\NextLinkTarget{cite.#1\@extra@b@citeb}
 }

\let\@kernel@copy@bibitem\@bibitem
```

(*End of definition for* `\@bibitem`. *This function is documented on page* 2.)

3.2 Handling citation commands

We redefine similar to hyperref the `\bibcite` command to inject link and structure. Even if it looks a bit odd it is now used for many years and so hopefully compatible with various packages. But differently to hyperref we use the new hooks with arguments. TODO: consider hook name.

```
\NewMirroredHookPairWithArguments{bibcite/before}{bibcite/after}{2}
\def\bibcite#1#2{%
   \@newl@bel{b}{#1\@extra@binfo}{%
      \UseHookWithArguments{bibcite/before}{2}{#1}{#2}
       #2
      \UseHookWithArguments{bibcite/after}{2}{#1}{#2}
       }%
     }%
\let\@kernel@copy@bibcite\bibcite
```

Now we add the tagging structure. The target is in the structure of the label, so we use the tagpdf command that retrieve the parent structure, i.e., the LI.

```
\NewTaggingSocket{bib/cite/before}{1}
\NewTaggingSocketPlug{bib/cite/before}{kernel}
 {
    \tag_mc_end_push:
    \tagstructbegin{tag=\UseStructureName{cite}}
    \tag_struct_gput:nne{\tag_get:n{struct_num}}{ref_dest_parent}{#1}
    \tagmcbegin{}
 }
\AssignTaggingSocketPlug{bib/cite/before}{kernel}
\AddToHookWithArguments{bibcite/before}
  {
    \UseTaggingSocket{bib/cite/before}{cite.#1\@extra@b@citeb}
  }
\AddToHookWithArguments{bibcite/after}
  {
    \UseTaggingSocket{inline/end}
  }
```

The package hyperref uses the hook too and the link will be inside the reference.

3.3 natbib and biblatex support

Both natbib and biblatex use `\hyper@natlinkstart`, `\hyper@natlinkend` and `\hyper@natanchorstart` to handle the links and anchors. The commands can be also used to handle tagging, both if hyperref is loaded and if not. We only need to ensure that they use `\MakeLinkTarget` also if hyperref is not active (and that hyperref does not overwrite that again if used).

natbib calls `\hyper@natanchorstart{key\@extra@b@citeb}`. With hyperref loaded this gives the anchor `cite.key\@extra@b@citeb`. biblatex calls `\hyper@natanchorstart{\the\c@refse` and this gives `cite.\the\c@refsection @\abx@field@entrykey`.

We assume that no document loads both package – that will probably break.

```
\newcommand\hyper@natlinkstart[1]{}
\newcommand\hyper@natlinkend{}
\newcommand\hyper@natanchorstart[1]{}
```

natbib patches `\@lbibitem` (and doesn't use `\@bibitem`) and adds anchors directly, so we undo the patches to avoid to get duplicated destination warnings if hyperref is used:

```
\AddToHook{package/natbib/after}
  {
   \RemoveFromHook{cmd/@lbibitem/before}[latex-lab-testphase-bib/target]
   \RemoveFromHook{cmd/@bibitem/before}[latex-lab-testphase-bib/target]
  }
```

We overwrite the anchor command at begin document so that this is also used with hyperref.

```
\AtBeginDocument{\renewcommand\hyper@natanchorstart[1]{\MakeLinkTarget*{cite.#1}}}
```

The code for the citations can be shared by both packages, as they both use the natbib hooks. The structure this references is currently the itemlabel for natbib, and the itembody for biblatex. TODO: move this up one level.

```
\AddToHookWithArguments{cmd/hyper@natlinkstart/before}
```

```
 {
   \leavevmode
   \UseTaggingSocket{bib/cite/before}{cite.#1}
 }
\AddToHook{cmd/hyper@natlinkend/after}
 {
   \UseTaggingSocket{inline/end}
 }

⟨/package⟩
```