Headers and footers with titleps

Javier Bezos†

2025/01/04

Contents

1. Introduction

The titleps package provides tools for one-stage setting of page styles (headlines and footlines). A higher-level interface is used, where the mark mechanism is hidden and there is no need to deal with `\leftmark`s and `\rightmark`s – just use a command or variable registered as a "mark" as the expected value will be returned, i.e., those when the mark was emitted, either by a sectioning command or explicitly with `\chaptermark`, `\sectionmark`, etc. A simple example, whose meaning should be obvious, is:

```
\newpagestyle{main}{
  \sethead[\thepage][\chaptertitle][(\thesection]  % even
         {\thesection)}}{\sectiontitle}{\thepage}} % odd
\pagestyle{main}
```

Other features are:

- Working top marks, compatible with floats (unlike the standard `\topmark`, which does not work correctly in LaTeX).

- Access to top, first and botttom marks in a single headline/footline (e.g., the first and last section numbers).

- Marks for more than 2 sectioning levels.

- Auto-defining of suitable `\sectionmark` and the like.

- Simple (and not so simple) headrules and footrules.

- Headlines and footlines for pages with floats.

- Headlines and footlines for specific floats (a sort of `\thispagestyle` for floats).

- Multiple sets of marks (named here *markset*s and *extra marks*).

It can be used without titlesec, but you will get most of it when used together. To load it as a separate package, use the customary `\usepackage{titleps}`, but with titlesec you have to load it with:

```
\usepackage[pagestyles]{titlesec}
```

Specific tools for sectioning command marks are provided, which are considered the main *markset* (unnamed) and carry out some internal task. However, titleps can be used as well for headers with the first an last entry of a page in a dictionary or index, for example, thank to the *extra marks* described below.

It works with standard classes and many others; however, additional LaTeX layout adjustments could be necessary in some of them (for instance the AMS classes) and several packages emitting marks directly with `\markboth` and `\markleft`.[1]

Once loaded the package, you can define your own page styles and then activate them, but note page styles so defined cannot be used with `\thispagestyle` except if the surrounding page style has been defined with titleps, too.

2.   Defining Page Styles

```
\newpagestyle{⟨name⟩}[⟨global-style⟩]{⟨commands⟩}
\renewpagestyle{⟨name⟩}[⟨global-style⟩]{⟨commands⟩}
```

Defines a new style or redefines an existing one, named ⟨*name*⟩. For convenience, spaces and end of lines are ignored, and you need not "hide" them with `%`.[2]

⟨*global-style*⟩ is any command to be applied to both footlines and headlines. Fragile commands are allowed.

In ⟨*commands*⟩ you can use:

```
\sethead[⟨even-left⟩][⟨even-center⟩][⟨even-right⟩]
        {⟨odd-left⟩}{⟨odd-center⟩}{⟨odd-right⟩}
\setfoot[⟨even-left⟩][⟨even-center⟩][⟨even-right⟩]
        {⟨odd-left⟩}{⟨odd-center⟩}{⟨odd-right⟩}
```

Set the corresponding part in headlines and footlines (note the "visual" layout of parameters). The optional arguments are the three or none. If no optional arguments are given the odd-page settings applies to both even and odd pages. A couple of starred variant (`\setfoot*`, `\sethead*`) reverses the order of the settings for even pages (obviously, with them the optional arguments are not allowed). If `\sethead` is not given, then the headline it is left empty, and same for `\setfoot` and the footline.

In the arguments of `\sethead`/`\setfoot` and because of its one-stage mechanism, we must distinguish between two groups of commands:

- The first one includes those saved when the marks are emitted, with information related to sections, and includes:
    - `\thechapter`, `\thesection`, `\thesubsection`...
    - `\chaptertitle`, `\sectiontitle`, `\subsectiontitle`... which print the chapter, section... title.
    - (Only titlesec. Not in titleps.) `\ifthechapter{⟨true⟩}{⟨false⟩}`, `\ifthesection{⟨true⟩}{⟨false⟩}`, `\ifthesubsection{⟨true⟩}{⟨false⟩}`... which expands to ⟨*true*⟩ except if the corresponding title lacks of label or there is no title yet after the superior level (for example, between `\chapter` and a subsequent `\section`).
    - Any other command or value "registered" as "mark." (See below.)

    They are ordinary commands, with no internal marks at all, and you can test if, for example, they are empty.

- The second group are those expanded on every page, and includes:
    - `\thepage`.
    - Any other command not included in the previous items.

---

[1] Two columns layout requires the fix2col package by David Carlisle.

[2] Even if marks should be reserved to pass information to headers, some classes, like the AMS ones, introduce additional code. When redefining the `plain` style of AMS classes, you must add the following line: `\global\topskip\normaltopskip`.

```
\settitlemarks{⟨level-name⟩,⟨sublevel-name⟩,⟨subsublevel-name⟩...}
\settitlemarks*{⟨level-name⟩,⟨sublevel-name⟩,⟨subsublevel-name⟩...}
```

Sets which `\...title` commands are to be defined, and when the marks are emitted; any number of levels is allowed (1, 2, 3...). For example `\settitlemarks{chapter,section}` means that:

- `\chaptertitle` and `\sectiontitle` are the allowed titles in headlines,

- `\sectiontitle` is reset in `\chapter`,

- (Only titlesec. Not in titleps.) `\ifthechapter` and `\ifthesection` are valid tests.

- Marks are updated in `\chapter` and `\section`.

Default is `\settitlemarks{chapter,section}`, but `\settitlemarks{section,subsection}` in article class. `\settitlemarks` may be used outside `\(re)newpagestyle` to set the default to be used in newly defined and redefined page styles – it is used in `\pagestyle`.[3]

The starred version `\settitlemarks*` creates parallel extra marks as described below.

```
\headrule              \footrule
\setheadrule{⟨length⟩}  \setfootrule{⟨length⟩}
```

If you want a rule below the headline and above the footline. You may also set their width directly with the commands `\setheadrule` and `\setfootrule`. (For instance, `\setheadrule{.4pt}` which is, by the way, the default value.)

These commands are local to a page style, i.e., defining them directly in the preamble will not work because the page style settings will override them.

```
\makeheadrule \makefootrule
```

These commands are used by titleps to typeset the rules. If there is no rule, both commands are empty (this value is the default). `\setheadrule{dim}` just stands for

```
\renewcommand{\makeheadrule}{\rule[-.3\baselineskip]{\linewidth}{⟨dim⟩}}
```

unless if *dim* is 0 pt, which empties `\makeheadrule` (and similarly `\setfootrule`).

You can access the total header/footer width with `\linewidth` but the box enclosing the rule is in fact dimensionless. Hence, you must be concerned with the rule placement only. Its baseline is the same as that of the header/footer. That means the material to be used as rule should be raised or lowered with `\raisebox`, the lifting argument in `\rule` or by using sensible coordinates in `picture`, for instance. That makes easy to place material above *and* below. For instance, the following code creates a headline with a black bold line above and a red rule below (the color package is required):

```
\renewcommand{\makeheadrule}{%
  \makebox[0pt][l]{\rule[.7\baselineskip]{\linewidth}{0.8pt}}%
  \color[named]{Red}%
  \rule[-.3\baselineskip]{\linewidth}{0.4pt}}
```

Of course, the material to be used as rule is not restricted to actual rules; pictures and leaders, for instance, are allowed, too.

3.   On `\markboth` and `\markleft`

As *The LATEX Companion explains* "Leslie Lamport had a specific marking scheme in mind when he designed those commands". They are clearly intended for the `heading` and `myheading` pagestyles, with a *main-mark* on the left page and a *sub-mark* on the right one.

In order to provide the functionality offered by titleps, a more general, less specific mechanism is devised, which means `\markboth` here does not make sense, because it doesn't work like in standard LATEX, and in fact, can lead to unpredictable results.

---

[3]The command `\setmarks` is deprecated. Now `\setmarks{<primary>}{<secondary>}`is `\settitlemarks{<primary>, <secundary>}`.

Therefore, using directly the marking commands is discouraged (see `titlesec.pdf`), but if you need them, you may write:

```
\chapter*{My Chapter}
\chaptermark{My Chapter}
```

But some issues remain, because there are a number of packages assuming the pagestyle is `heading` or `myheading`. This is, of course, a reasonable guess, but not always correct (even if titleps is not used at all). Most of them do the right thing – `\markboth` is encapsulated into a macro easily redefinable so that the user can modify it if necessary. This is true, for example, for biblatex, but unfortunately not for endnotes. When this happens, you can resort to the following macro.

`\setmarkboth{⟨code-to-use⟩}`

From this point on, `\markboth` behaves as defined. It doesn't redefine directly the command, but rather provides a replacement code (you can use `#1` and `#2` for the two arguments). With `\resetmarkboth` (which can be used inside `\setmarkboth`) the original definition will be again in force. For example, to cancel the next `\markboth`:

```
\setmarkboth{\resetmarkboth\markboth{}{}}
```

4. Headline/footline width

`\widenhead[⟨even-left⟩][⟨even-right⟩]{⟨odd-left⟩}{⟨odd-right⟩}`
`\widenhead*{⟨even-right/odd-left⟩}{⟨even-left/odd-right⟩}`

Makes the headlines/footlines lines wider. The extra width is asymmetrically added and hence the two [four] arguments. Like `\sethead`, the starred version reverses the setting for even pages—for example, `\widenhead*{0pt}{6pc}` is the same as `\widenhead[6pc][0pt]{0pt}{6pc}`.

To be used outside `\(re)newpagestyle`.

`nopatches`   (package option)

By default, sectioning commands are patched slightly for top marks and `\@mkboth` to work, so that page styles behave like standard `headings`. However, if you want to be explicit, like with standard `myheadings`, and emit premarks and, if necessary, marks by yourself, set this package option. Remenber you have to use `\sectionmark` and the like, and not `\markboth` or `\markright`.

You might also prefer to patch the sectioning commands by yourself, either with low-level definitions or with the provided `\TitlepsPatchSection` (with a starred version which adds no premark and intended mainly for chapters). For example:

```
\TitlepsPatchSection*{chapter}  % patches \chapter
\TitlepsPatchSection{section}   % patches \section
```

(In standard classes `\part` emits no mark, and therefore it is not patched by default.)

5. Marks

Marks can be retrieved in 4 ways:

top marks If a page begins with text, they are the marks belonging to the sectioning unit of that text. If a page begin with a sectioning title, they are the marks of that title. In other words, top marks refer to the section immediately adjacent to the headline.

first marks The marks of the very first sectioning title.

botttom marks The marks of the very last (bottom) sectioning title (also known as *bot marks*).

next top marks A botttom mark with the values of the top mark of the next page (only `straight` class or equivalent); by picking some value from it and comparing it with the corresponding value from the botttom marks you may know if a section continues in the next page. Of course, it will not be used very often.

Standard LaTeX provides only first and bottom marks. As you see, with titleps you have two additional marks, namely, top and next top marks.

It should be stressed top marks are not the same as the primitive `\topmark` (which, in fact, is not used internally at all). They are true top marks, returning the expected result when the title is at the top of the page. Further, they are compatible with floats, which is not true with `\topmarks`. However, they fail with explicit page breaks, but you can use `\pretitlemark` as described below.

`outermarks innermarks topmarks botmarks`   (package options)

`innermarks` is the default in LaTeX, with bottom marks in even pages and first marks in odd pages. More convenient is `outermarks`, with top marks in even pages and bottom marks in odd pages; it is the more usual system in classic scientific literature and it is described in *The TeXbook*, p. 259.[4] Both of them are intended for two side printing; `topmarks`/`botmarks` are intended for one side printing with top/bot marks in every page (they can be used in two side printing as well).

`\bottitlemarks  \toptitlemarks  \firsttitlemarks \nexttoptitlemarks`
`\outertitlemarks  \innertitlemarks`

These commands set which mark the values of the subsequent `\thesection`, `\sectiontitle`, etc., will be taken from. These macros are switches, and therefore subsequent commands and variables are taken from the selected mark. The following is valid:

```
\toptitlemarks\textbf{\thesection.} \sectiontitle --
\bottitlemarks\textbf{\thesection.} \sectiontitle
```

You can use them freely in your headers;[5] just for fun, the following header shows the section label from three of these marks:

```
\newpagestyle{funny}{
  \sethead{\toptitlemarks Top is \thesection}
          {\firsttitlemarks First is \thesection}
          {\bottitlemarks Bot is \thesection}}
```

In multilingual documents, the language in force is that corresponding to the package options – command versions does not switch the language, but you can do it easily with the help a new title mark and then switching the language, as explained below.

`\newtitlemark{⟨`*command-name*`⟩}`
`\newtitlemark*{⟨`*variable-name*`⟩}`

Add a command or a variable to the list of "marks" to be emitted at a sectioning command. The macro must be parameterless, and the variable is either a length, if it has the form \⟨*name*⟩, or a counter, if it has the form ⟨*name*⟩ (i. e., a string). For example, if you write (with babel):

```
\newtitlemark{\languagename}
```

then a `\languagename` in the head/foot will contain the language when the corresponding mark was emitted; unfortunately, babel does not allow `\languagename` as the first argument of `\foreignlanguage` (it enters in a infinite loop), but there is a simple trick – just use an intermediate command:

```
\newcommand\titlelanguage{\languagename}
\newtitlemark{\titlelanguage}
```

---

[4]`outermarks` may not work well in conjunction with explicit `\<section>mark` commands
[5]Not exactly: top marks must not be used in the page where a chapter begins, except by overriding the default definition and making it of `top` class.

(What is saved is the expansion of \titlelanguage, which is turn is the expansion of \languagename. Alternatively, you can create the intermediate command with \let in the header itself.)

Marks to be emitted at other places (not sectioning commands) are handled with extra marks and *marksets* (see below).

Commands must be robust, but not protected, or the value returned will not be the right one. Remember there are two kinds of robust commands – those containing in turn robust commands and those protected with, say, \DeclareRobustCommand.[6] An example of how to use protected commands is a layout used sometimes if sections are long as well as their titles, where the title is splitted across the left and the right pages. Here is how this can be carried out:

```
\DeclareRobustCommand\splitheader[2]{#1 #2}
\def\contsplitheader{}

\def\dosplitheader{%
  \gdef\contsplitheader{}%
  \DeclareRobustCommand\splitheader[2]{##1\gdef\contsplitheader{##2}}}

\renewpagestyle{headings}{
  \sethead[\thepage]
          []
          [\dosplitheader\sectiontitle]
          {\contsplitheader}
          {}
          {\thepage}
}

\pagestyle{headings}
```

(Note the header also uses \DeclareRobustCommand.)

\pretitlemark{⟨*section*⟩}{⟨*text*⟩}
\pretitlemark*{⟨*section*⟩}{⟨*text*⟩}

Top marks require in fact two marks, one before the page break and other after it. This is done by titlesec, while titleps patches the sectioning commands as described above.

However, if you need an explicit page break you can emit the mark with \pretitlemark. For example:

```
\pretitlemark{section}{Performing Brahms}
\newpage
\section{Performing Brahms}
```

Note the text in the premark is the title of the next section. The counter is advanced temporarily, so that it has the right value, but if for some reason you do not want that, just use \pretitlemark*.

\ifsamemark{⟨*group*⟩}{⟨*command*⟩}{⟨*true*⟩}{⟨*false*⟩}

One of the disavantaged of the switching mechanism is that comparing a command from, say, the top marks and the botttom marks is not trivial. This macro just makes it simpler. If you want to compare the currect \thesection with that in the top mark, write:

\ifsamemark\toptitlemarks\thesection{⟨*true*⟩}{⟨*false*⟩}

---

[6]Unfortunately, \DeclareRobustCommand has a misleading name, because a command created with \newcommand can be robust, too. A better name would have been \DeclareProtectedCommand.

6.　Running heads with floats

```
psfloats        (package option)
```

This package option activates the commands described in this section.[7]

```
\setfloathead*{.}{.}{.}{⟨extra⟩}[⟨which-floats⟩]
\setfloathead[.][.][.]{.}{.}{.}{⟨extra⟩}[⟨which-floats⟩]
(Similarly \setfloatfoot.)
```

The arguments with a dot are similar to those of \sethead. The final argument says the head is used when there is a float of the specified types (default is tp in headers, and bp in footers). For example:

```
\newpagestyle{main}{%
  \sethead ... your definition
  \setfoot ... your definition
  \headrule
  \footrule
  \setfloathead{}{}{}{\setheadrule{0pt}}[p]
  \setfloatfoot{}{}{}{\setfootrule{0pt}}[p]}
```

removes the header/footer on float pages, including the rules.

```
\nextfloathead*{.}{.}{.}{⟨extra⟩}[⟨which-floats⟩]
\nextfloathead[.][.][.]{.}{.}{.}{⟨extra⟩}[⟨which-floats⟩]
(Similarly \nextfloatfoot.)
```

You may force a header (respectively footer) when a certain float is the first top one (respectively last bottom) with those commands just before the corresponding float, where the arguments with a dot are similar to those of \sethead. Those commands are used by themselves, without putting them inside \(re)newpagestyle. So, to empty the headline when a certain figure is placed at the top (because it is a bleed, for example):

```
\nextfloathead{}{}{}{}[t]
\begin{figure}
...
\end{figure}
```

7.　Extra marks

Both titlesec and titleps provide tools for handling extra marks, if you are using an ε-TeX based engine (currently most of them) and load the etex package or equivalent. The way they work are conceptually similar to the title marks described above – you set which commands or variables are to get synchronized in heads and foots. To use extra marks, you have to load them with the following package option:

```
extramarks        (package option)
```

Extra marks are subsidiary. If page styles are modified internally by another package and the document does not make use of sectioning commands, you might still need an empty title mark (like \sectionmark{}), as well as to synchronize the language by hand.

In many cases, the following "short" marks mechanism will be enough. With it, you simply say with \newshormark the name of a command to be used in heads or foots, and then emit the mark at appropiated places (short marks for variables are not available). An example illustrates how this is carried out:

---

[7]It redefines some internal LaTeX commands and there could be incompatibilities with other packages modifying the same commands. That's the very reason these commands are loaded only optionally. As of version 2.6 the syntax was extended with a new argument to allow extra stuff in running heads/foots. This change is backward compatible—just preserve the old name floatps, which is deprecated, if you want the pre-2.6 behaviour.

```
\newcommand\lemmatitle{}

\newshortmark\lemmatitle

\newcommand\lemma[1]{%
  \renewcommand\lemmatitle{#1}%
  \preshortmark\lemmatitle % if you want top marks
  \pagebreak[2]%
  \vspace{1em}%
  \textbf{#1}\enspace
  \shortmark\lemmatitle}

\newpagestyle{main}{
  \sethead{}{\topshortmark\lemmatitle
          --\firstshortmark\lemmatitle
          --\botshortmark\lemmatitle
          --\nexttopshortmark\lemmatitle}{}}

\pagestyle{main}
```

Top marks require emitting the marks before the page break – just use the command `\preshortmark` as shown above. Otherwise, you may omit it. Unlike title marks and extra marks, the selector of short marks (`\topshortmark`, `\firstshortmark`, `\botshortmark`, `\nexttopshortmark`), is *not* a switch and must be always immediately followed by the command to be retrieved.

For a more general approach use the following tools.

`\newmarkset{`⟨*markset*⟩`}`

A *markset* contains a set of related commands and variables intended to be emitted simultaneously as marks, so that their values can be retrieved in heads/foots. A typical *markset* could be a title and its counter (and perhaps, in a multilingual document, the active language, i.e., `\languagename` in babel, or an intermediate command). There is always a special unnamed *markset* for sectioning marks, which is the main one and handled with the tools described above (with `title` in their names).

`\newextramark{`⟨*markset*⟩`}{`⟨*macro-name*⟩`}`
`\newextramark*{`⟨*markset*⟩`}{`⟨*variable-name*⟩`}`

Add to a *markset* a command or a variable to be synchronized . They are identical to `\newtitlemark` and `\newtitlemark*`, except for the additional parameter with the *markset* name.

`\botextramarks{`⟨*markset*⟩`}`   `\topextramarks{`⟨*markset*⟩`}`
`\firstextramarks{`⟨*markset*⟩`}`   `\nexttopextramarks{`⟨*markset*⟩`}`
`\outerextramarks{`⟨*markset*⟩`}`   `\innerextramarks{`⟨*markset*⟩`}`

The `extra` counterparts of the `title` commands described above, with the same behaviour. For example:

```
\newcommand{\lemmatitle}{}
\newcounter{lemma}

\newmarkset{lemma}
\newextramark{lemma}{\lemmatitle}
\newextramark*{lemma}{lemma} % ie, the counter 'lemma'

\newcommand{\lemma}[1]{%
  \renewcommand{\lemmatitle}{#1}%
  \stepcounter{lemma}%
  \preextramark{lemma}%
  \pagebreak[2]%
```

```
   \vspace{1em}%
   \textbf{(\arabic{lemma}) #1}\enspace
   \extramark{lemma}}

\newpagestyle{main}{
  \sethead{}
         {\topextramarks{lemma}(\arabic{lemma}) \textbf{\lemmatitle}}
         {}}

\pagestyle{main}
```

| \settitlemarks*{⟨*level-name*⟩,⟨*sublevel-name*⟩,⟨*subsublevel-name*⟩...} |

By default, LATEX and titleps use a single *markset* for all sectioning levels. This is problematic, for example, in the article class, because sections does not start a new page. Consider the following piece of code:

```
\section{A section}
\subsection{A subsection}
```

On odd pages with first marks and subsection titles, the head has no title, because \subsection is the second mark, not the first one.

To overcome this limitation, you can use the starred version of \settitlemarks, which creates parallel *marksets* (one per level) synchonized with the main *markset*. They are retrieved like other extra marks, but they are special in that you cannot emit them by yourself – they are always emitted by the corresponding sectioning mark. So, with

```
\settitlemarks*{section,subsection}
```

and a page style containing

```
\firstextramarks{subsection}\subsectiontitle
```

you retrieve the mark emitted by the first \subsection, even if preceded by a \section. (Top marks are best retrieved from the main *markset*.)

They are extra marks, and therefore language synchronization has to be done by hand – add \languagename or an intermediate command to the main *markset* with \newtitlemark and, if you want, define a command. For example:

```
% use an intermediate command because we'll need \foreignlanguage
\newcommand\titlelanguage{\languagename}
\newtitlemark{\titlelanguage}

% define a command to ease defining page styles
\newcommand{\toplang}[1]{%
  {\topextramarks{section}%
   \foreignlanguage{\titlelanguage}{#1}}}
```

8.   A couple of examples

Because of the way marks are handled, special constructs like this are allowed:

```
\newpagestyle{main}[\small\sffamily]{
   \sethead[\textbf{\thepage}]
          [\textsl{\chaptertitle}]
          [[\toptitlemarks\thesection--\bottitlemarks\thesection]
          {\toptitlemarks\thesection--\bottitlemarks\thesection]}
          {\textsl{\sectiontitle}}
          {\textbf{\thepage}}}
```

As you can see, the range of sections in a page with \thesection is printed in both left and right heads. Of course, this example should be fine tuned to collapse the ranges if there is a single section in a page (with \ifsamemark), but it gives a hint of how to get stunning headers.

In this document, the following styles are used:

```
\widenhead{2.1pc}{0pc}
\renewpagestyle{plain}[\small\sffamily\slshape]{
  \footrule
  \setfoot{}{\thepage}{}}
\newpagestyle{myps}[\small\sffamily\slshape]{
  \headrule
  \sethead{titlesec -- titleps}{\sectiontitle}{\thepage}}
```

The following definitions provide page styles similar to those in Lamport's *The LATEX book* (with the calc package):

```
\widenhead*{0pt}{\marginparsep + \marginparwidth}  % symmetrically
\renewpagestyle{plain}{}
\newpagestyle{latex}[\bfseries]{
  \headrule
  \sethead[\thepage][][\chaptertitle]
          {\thesection\ \sectiontitle}{}{\thepage}}
\pagestyle{latex}
```