

Пакеты `titlesec` и `titletoc`*

Javier Bezos[†]

January 22, 2005

Техническое редактирование¹

Команды разрывов или другие дополнительные настройки перед заголовками	8	Команда <code>\cleardoublepage</code> с пустым колонтитулом (опция <code>clearempty</code>)	9
Пакет для заголовков <code>titlesec</code>		Ненумерованные заголовки разделов	12
Заголовки со специальными значками	13	Новый уровень заголовка	10
Задание/отмена абзацного отступа после заголовков (опции <code>intentafter</code> и <code>nointentafter</code>)	6	Отбивки и поля вокруг заголовков (команда <code>\titlespacing</code>)	5
Задание разного оформления для нумерованного и ненумерованного заголовка	10	Оформление заголовков (команда <code>\titleformat</code>)	4
Задание разного оформления для разворотов	10	Оформление заголовочных линеек (команда <code>\titleline</code>)	7
Замена команды <code>\\</code> в заголовках для записи колонтитулов и оглавления (опция <code>newlinetospace</code>)	9	Спускосые полосы глав	6
Запись заголовка части в оглавление в новой/старой форме (опции <code>newparttoc</code> и <code>oldparttoc</code>)	9	Стили (размещение) заголовков	4
Запись полной формы заголовков в оглавление (опция <code>toctitles</code>)	9	Управление заголовками внизу полосы	6
Классы заголовков (<code>page</code> , <code>top</code> , <code>straight</code>)	10	Пакет для оглавления <code>titletoc</code>	
Классы заголовков, команда <code>\titleclass</code>	10	Миниоглавления	23
Колонтитулы по-русски (опция <code>outermarks</code>)	17	Опции выравнивания нумерации заголовков	23
		Основные команды пакета	20
		Текст разделов в подбор	22
		Точки в нумерации заголовков	23
		Заголовки со звёздочками	25

Содержание

1. Введение 2
2. На скорую руку 2
 - 2.1. Оформление, 2. — 2.2. Отбивки, 3. — 2.3. Инструменты, 3.
3. Расширенный интерфейс 3
 - 3.1. Оформление, 4. — 3.2. Отбивки, 5. — 3.3. Инструменты выключки, 6. — 3.4. Линейки, 7. — 3.5. Разрывы, 8. — 3.6. Другие опции пакета, 9. — 3.7. Дополнительные настройки, 9. — 3.8. Создание нового уровня и изменение класса заголовка, 10.
4. Дополнительные замечания 11
 - 4.1. Нумерационные заголовки фиксированной ширины, 11. — 4.2. «Звёздные?» версии, 12. — 4.3. Варианты (специальные знаки в заголовках), 13. — 4.4. Поставить точку после текста заголовка, 13.
5. Стили колонтитулов 14
 - 5.1. Определение стилей колонтитула, 14. — 5.2. Дополнительные настройки, 16. — 5.3. Колонтитулы с плавающими объектами, 16. — 5.4. Маркировка, 17. — 5.5. Парочка примеров, 17.

*Пакет `titlesec` версии 2.6. © 1998–2005 Javier Bezos. Пакет `titletoc` версии 1.5. © 1999–2005 Javier Bezos. All Rights Reserved.

[†]Просьба высылать сообщения об ошибках в пакете (и небольшой файл примера), комментарии и предложения по адресу `jbezoz@wanadoo.es`, или по традиционной почте: Apartado 116.035, E-28080 Madrid, Spain. Английский язык не является моей сильной стороной, пожалуйста, сообщите об ошибках в моём руководстве. Другие пакеты автора: `gloss` (совместно с José Luis Díaz), `accents`, `tensind`, `esindex`, `dotless`.

¹Добавлено при переводе.

- 6. **Заключительные комментарии** 18
- 7. **Оглавление: пакет titletoc** 19
 - 7.1. Десятиминутный гид по titletoc, 19.— 7.2. И ещё, 21.— 7.3. Миниоглавления, 23.—
 - 7.4. Минисписки, 24.— 7.5. Примеры, 24.— 7.6. Добавление плавающего рисунка в оглавление, 25.—
 - 7.7. Задание заголовков со спецзнаками, 25.
- 8. **Философия пакета titlesec** 25
- 9. **Приложение** 25

1. Введение

Данный пакет является заменой — частичной или полной — макрокоманд \LaTeX 'а связанных с заголовками: собственно заглавия, колонтитулов и оглавления. Целью пакета является предоставление возможностей, отсутствующих в данной версии \LaTeX 'а; если вам нужен более дружественный интерфейс чем предоставленный стандартным \LaTeX 'ом но без изменений работы самого \LaTeX 'а вы, возможно, предпочтёте использование пакета fancyhdr (Piet van Oostrum), sectsty (Rowland McDonnell), и tocloft (Peter Wilson), позволяющие делать замечательные вещи.

Новые возможности предлагаемые пакетом:

- Различные классы «формы» заглавий, с инструментами для необычных оформлений. Вы можете определить разные оформления для левой и правой страниц, или нумерованных или ненумерованных заголовков, вычислить ширину заголовка, добавить новый уровень заголовка, использовать графику, и многое другое. Приложение предлагает множество примеров, смотрите примеры прямо сейчас!
- Верхние и нижние колонтитулы определённые без посредства команд `\. . . mark`, и, возможно являющиеся `top`, `first` и `bot marks` одновременно. Top marks синхронизированы с заголовками, исключая несовмещения с механизмом плавающих элементов. Декоративные элементы легко можно добавить, используя окружения рисунков.
- Различные варианты оформления оглавления, с возможностью сгруппировать заголовки различного уровня в одном абзаце или изменение оформления заголовков в середине документа.

Пакет titlesec работает со стандартными классами и со многими другими, включая классы AMS'а, и может работать с пакетом hyperref.²

Для начала, загрузите пакет как обычно, используя команду `\usepackage`. Затем, переопределите заголовки команд с помощью простых, предопределённых установок (см. разд. «Quick Reference») или предлагаемых команд, если вы желаете создать более сложное оформление (см. разд. «Advanced Interface»). В конце концов, если вам нужно только переопределить команды, пожалуйста. Оба способа можно использовать одновременно, но поскольку `\part` определена не так, как другие заголовки, она остаётся нетронутой при простых изменениях и её оформление может быть изменено только с помощью команд и настроек раздела «Расширенный интерфейс».

2. На скорую руку

Простейший способ изменить оформление — это задание опций пакета и несколько команд. Если вам достаточно изменений, предлагаемых данными инструментами, вам достаточно прочесть только данный раздел, игнорируя остальные.

2.1. Оформление

В пакете имеются три группы опций пакета отвечающие за шрифт, размер и выключку заголовков. Вам не нужно задавать все группы, поскольку они заданы установками по умолчанию; однако, вам наверняка понадобится хотя бы одна из опций, если вам нужны «простые установки».

²Однако, обратите внимание что классы AMS'а переопределяют некоторые внутренние команды. Эти изменения теряются. Совместимость с пакетом hyperref были протестированы с dvips, dvipdfm и pdftex опциями но эта возможность не поддерживается. Пожалуйста, проверьте что версия hyperref совместима с titlesec.

```
rm sf tt md bf up it sl sc
```

Определите соответствующие family/series/shape. По умолчанию задана опция bf.

```
big medium small tiny
```

Задание размер заголовков. По умолчанию задана опция big, которая задаёт размеры соответствующие стандартному L^AT_EX’у. С опцией tiny, разделы (кроме частей) набираются размером основного шрифта. Опции medium и small задают промежуточные варианты.

```
raggedleft center raggedright
```

Определяют выключку заголовков.

2.2. Отбивки

```
compact
```

Эта опция независима от предыдущих и уменьшает отбивки над и под заголовками.

2.3. Инструменты

```
\titlelabel{<label-format>}
```

Изменяет формат нумерационной части разделов. Команда \thetitle заменяет соответственно \thesection, \thesubsection, и т. д. По умолчанию в стандартных классах задано

```
\titlelabel{\thetitle\quad}
```

и вы можете задать точку после нумерации

```
\titlelabel{\thetitle.\quad}
```

Что и было сделано в данном документе.

```
\titleformat*{<command>}{<format>}
```

Данная команда позволяет изменить оформление (<format>) раздела, например:

```
\titleformat*{\section}{\itshape}
```

3. Расширенный интерфейс

Для изменения оформления предназначены две команды. Одна используется для «внутреннего» оформления, т. е., задания формы, шрифта, вида нумерационной части. . . , другая определяет «внешнее» оформление — отбивку до и после, втяжку слева (и справа). Данная схема должна упростить настройки, поскольку в большинстве случаев вам нужно изменить либо отбивку либо оформление.³ Команды переопределяют команды существующих разделов, но не создают новые. Новые уровни заголовков могут быть добавлены командой \titleclass, как объясняется ниже, и только после этого оформление может быть определено командами, описанными здесь.

³Информация для модификации «извлечена» из команд рубрикации классов, исключая главу и часть. Подразумеваются стандартные определения \@startsection, а если имеется специальная процедура, то предлагается специальное значение, которое потом можно изменить. (К сожалению, нет возможности для этих целей подцепить из стандартных классов оформление глав, частей и заголовков, им подобных.)

3.1. Оформление

Предлагаются установки оформления, которые определяют размещение заголовка и его нумерационной части:

hang опция заголовка по умолчанию, задающая номер заголовка на весу. (Как стандартный заголовок `\section`.)

block печатает весь заголовок как блок (абзац), без дополнительного оформления [нумерационной части]. Нужен для заголовков по центру⁴ и специального форматирования (использующего графические инструменты, такие как `picture`, `pspicture`, и т. д.)

display помещает нумерационную часть заголовка на отдельной строке. (Как в стандартной главе, `\chapter`.)

runin Заголовок в подбор. (Как стандартный `\paragraph`.)

leftmargin помещает заголовок (у него есть ещё название «фонарик») в левом поле. Заголовки в самом низу страницы будут перемещены на следующую и не будут выходить за нижний край полосы, но это означает, что могут создаваться неполные полосы.⁵ В таком случае вы можете увеличить растяжимость элементов страницы, задать команду `\raggedbottom` или использовать опцию пакета `nobottomtitles` описанную ниже. Поскольку используемый механизм создания таких заголовков независим от механизма создания заметок на полях (`\marginpar`), то возможны наложения текста. Опция-синоним — `margin`.

rightmargin то же, что `leftmargin` но помещает заголовок с правой стороны.

drop заголовок в оборку (или «форточка»), предполагается, что первый абзац раздела длиннее заголовка (в противном случае текст накладывается). Комментарии к опции заголовка `leftmargin` для этой опции также в силе.

wrap то же что и `drop`. Единственное отличие в том, что ширина «блока», зарезервированного для заголовка в опции заголовка `drop` зафиксирована, а в опции `wrap` автоматически рассчитывается по самой длинной строке. Ограничения описанные ниже и относящиеся к опции пакета `calwidth` также действуют и здесь.

frame то же, что и `display`, но заголовок обводится рамкой.

Обратите внимание, что некоторые опции заголовков не действуют в оформлении глав и частей.

```
\titleformat{<command>}[<shape>]{<format>}{<label>}{<sep>}{<before>}[<after>]
```

Здесь:

- `<command>` команда заголовка, которую нужно переопределить, т. е., `\part`, `\chapter`, `\section`, `\subsection`, `\subsubsection`, `\paragraph` или `\subparagraph`.
- Форма заголовка определяется опцией `<shape>`, возможные варианты описаны выше [в переводе данной документации называется опцией заголовка].
- `<format>` оформление для заголовка целиком и нумерационной части и текста по отдельности. Эта часть может содержать вертикальный материал (но не всегда горизонтальный) который печатается сразу после отбивки сверху.
- Нумерационная часть определяется в `<label>`. Вы можете опустить это оформление [оставить аргумент пустым], если нумерация заголовков для этого уровня не нужна, но обратите внимание, что удаление кода не удалит нумерацию из оглавления и колонтитулов.

⁴Нумерационная часть заголовка смещается влево, если у него две и более строк и используется опция `hang`, если только не задана команда `\`.

⁵Однако, плавающие объекты, идущие через несколько строк после заголовка могут вмешаться в алгоритм разрыва страницы и заголовок вылезет за нижний край полосы.

- `<sep>` горизонтальный разделитель между нумерационной частью и текстом заголовка, и должно быть задано как длина (аргумент не может быть пустым). Разделитель становится вертикальным если используется опция заголовка `display`; в опции `frame` это расстояние от текста до окружающей рамки. Как `<label>` так и `<sep>` игнорируются в «звёздных» формах заголовков. Если вы используете окружения `picture` и аналогичные, задайте этот параметр равным 0 pt.
- `<before>` код предшествующий собственно тексту заголовка. Самая последняя команда может использовать аргумент, которым является текст заголовка.⁶
- `<after>` код после заголовка. Это вертикальный материал при опциях заголовков `hang`, `block` и `display`; и горизонтальный при `runin`. В других случаях игнорируется.

`\chaptertitlename`

По умолчанию это `\chaptername`, а в приложениях `\appendixname`. Используется вместо `\chaptername` при оформлении глав.

3.2. Отбивки

`\titlespacing*{<command>}{<left>}{<beforesep>}{<aftersep>}[<right>]`

«Звёздные» заголовки уничтожают втяжку, кроме `drop`, `wrap` и `runin` где этот аргумент не имеет значения.

- `<left>` увеличивает левое поле, кроме опций заголовков `...margin` и `drop`, где этот параметр определяет ширину, в опции `wrap`, максимальную величину, а в `runin`, отступ первой строки заголовка. При отрицательной величине получается заголовок «на весу».⁷
- `<beforesep>` вертикальная отбивка до заголовка.
- `<aftersep>` отбивка заголовка от текста: вертикальная при `hang`, `block`, и `display`, и горизонтальная при `runin`, `drop`, `wrap` и `...margin`. Отрицательная величина возможна, но её абсолютное значение [по модулю] не может превышать величину `\parskip`.⁸
- Опции `hang`, `block` и `display` имеют возможность изменить правое (`<right>`) поле при помощи дополнительного аргумента.

Если вам не нравится задание отбивок с использованием параметров `plus` и `minus`, предлагается аббревиатура `*n`. В аргументе `<beforesep>` это эквивалентно `n ex` с небольшим растяжением и сжатием. В аргументе `<aftersep>` небольшое растяжение и нет сжатия.⁹ Итак, можно написать

```
\titlespacing{\section}{0pt}{*4}{*1.5}
```

Длины `\beforetitleunit` и `\aftertitleunit` используются как единицы в установках `*` и вы можете их изменить, если вам не нравятся предопределённые величины (или, например, для настроек в макете).

Примечания. `\titlespacing` не работает ни с `\chapter` ни с `\part` если не было определено оформление с использованием команды `\titleformat`, элементарными настройками [из раздела «Quick Reference»(?)], или команды `\titleclass`. Аргументами в команде `\titlespacing` должны быть длины; команда `\stretch` — это набор кодов и поэтому вызовет ошибку.

⁶Помните, что размер шрифта может быть изменён только между абзацами и что такие изменения должны быть в группе, иначе можно получить неверный интерлиньяж — слишком большой или слишком маленький.

⁷Этот параметр не эквивалентен `<indent>` из `\@startsection`, который работает некорректно. При отрицательной величине и если `<indent>` больше ширины нумерационной части заголовка, первая строка начинается на поле, как и ожидалось, но последующие строки нет, что ещё хуже, эти строки укорачиваются справа.

⁸См. Goossens, Mittelbach and Samarin: *The L^AT_EX Companion*, Reading, Addison Wesley, 1993, p. 25.

⁹Они расшифровываются как `n умноженное на 1ex plus .3ex minus .06ex` и `1ex plus .1ex`, соответственно.

3.3. Инструменты выключки

Эти команды предлагаются как инструменты для команд `\titleformat` и `\titlespacing`.

```
\filright \filcenter \filleft \fillast \filinner \filouter
```

Варианты команд `\ragged...`, с небольшими отличиями. В частности команды `\ragged...` уничтожают втяжки заданные командой `\titlespacing`. `\fillast` создаёт выключной абзац, но последняя строка выключается по центру.¹⁰ Эти команды работают также в обрамлённых (`frame`) заголовках.

Команды `\filinner` и `\filouter` — это `\filleft` или `\filright` зависящие от чётности страницы. Поскольку разрыв страницы в `TeX`'е асинхронный, эти команды можно использовать только в командах `\chapter`. Если вам нужен инструмент форматирования зависящий от чётности страницы, см. «Extended settings» далее.

```
\wordsep
```

Междусловный пробел текущего шрифта.

```
indentafter noindentafter (Package options)
```

Установки для всех команд разделов.¹¹

```
rigidchapters rubberchapters (Package options)
```

С опцией пакета `rigidchapters` пробел на спусковых полосах остаётся постоянным, а `\aftersep` в команде `\titlespacing` определяет не отбивку снизу после заголовка, а спуск от верха заголовка до текста, т. е., `\beforesep` + `\aftersep` теперь фиксированный спуск от верха полосы [без учёта колонтитула!] до текста. По умолчанию задаётся опция `rubberchapters` где `\aftersep` является как обычно разделителем между заголовком и текстом. В действительности, название опций неточно, поскольку это относится не только к главам, но и ко всем заголовкам, имеющим класс заголовка `top`. (Подробнее о классах заголовков ниже.)

```
bottomtitles nobottomtitles nobottomtitles* (Package options)
```

Если задана опция пакета `nobottomtitles`, заголовки, оказывающиеся близко к низу полосы переносятся на следующую полосу оставляя предыдущую неполной. Минимальное (приблизительное) пространство внизу полосы, в котором заголовок остаётся на текущей полосе, задаётся командой

```
\renewcommand{\bottomtitlespace}{\langle length \rangle}
```

величина по умолчанию `.2\textheight`. Отбивка на полосе до заголовка установлена как 0 pt. Опция `bottomtitles` задана по умолчанию, что задаёт `\bottomtitlespace` как отрицательную величину.

[Примечание при переводе. Если параметр `\bottomtitlespace` составляет несколько строк (к примеру `4-6\baselineskip`), этот параметр нужно комбинировать с командами `\sectionbreak` и т. д. (см. в документации ниже), в которых задаётся форсирование разрыва, как минимум `\pagebreak[1]`. В противном случае можно получить незаполненную полосу до заголовка а перед заголовком на следующей полосе «прилеплены» одна-две строки текста.]

Опция пакета `nobottomtitles*` предлагает более настроенные расчёты, но опции заголовков `margin`, `wrap` или `drop` могут создать плохую вёрстку. В остальном, лучше использовать «звёздную» форму.

¹⁰Странноватое имя, но зато краткое.

¹¹В предыдущих версиях `indentfirst` и `nonindentfirst`, сейчас не используются.

```
aftersep largestsep (Package options)
```

По умолчанию, когда идут два заголовка подряд, используется отбивка $\langle aftersep \rangle$ после первого заголовка. Иногда этот макет нежелателен, особенно если отбивка $\langle beforesep \rangle$ следующего заголовка значительно больше, чем $\langle aftersep \rangle$ предыдущего (в противном случае общепринятый макет предпочтительнее). Опция пакета `largestsep` задаёт использование большей отбивки. По умолчанию задана опция `aftersep`.

```
\ \ \*
pageatnewline (Package option)
```

2.6 Опция пакета `pageatnewline` создана для совместимости с предыдущими версиями. В версии 2.6, команда `\ \` не позволяет разрыв страницы, что эквивалентно команде `\ *`. Поскольку я предполагаю, что никому не захочется, чтобы страница разрывалась посреди заголовка, это задано по умолчанию. Если по какой-то прихоти вам нужно разрешить разрыв страниц внутри заголовков, используйте эту опцию пакета.

3.4. Линейки

Пакет предлагает некоторые инструменты для задания линеек и других элементов сверху или снизу заголовка. Поскольку поля могут быть изменены, эти макрокоманды учитывают локальные изменения полей и корректно размещаются. Также учитываются поля используемые заголовками на полях.

```
\titleline[align]{horizontal material}
\titlerule[height]
\titlerule*[width]{text}
```

Команда `\titleline` позволяет задать линию, которая может содержать и текст и другой «горизонтальный» материал. Эта команда предназначена для линеек, но, фактически, предназначена и для других целей. Линия имеет фиксированную длину и потому должна быть заполнена, т. е., `\titleline{CHAPTER}` создаст незаполненный бокс. В таких случаях используется дополнительный аргумент $\langle align \rangle$ (l, r or c), теперь можно просто задать, скажем, `\titleline[c]{CHAPTER}`.¹²

Использование `\titleline` в местах, где не нужен вертикальный материал может создать неожиданный результат. Другими словами, вы можете использовать в аргументах $\langle format \rangle$ (всегда) и $\langle after \rangle$ (в опциях заголовков `hang`, `display` и `block`); также в опции `display` в самом начале аргументов $\langle before \rangle$ и $\langle label \rangle$. Но рисковать можно, поскольку весьма вероятно это работает и в других местах.

Команда `\titlerule`, которая помещается автоматически внутрь `\titleline` при необходимости[??], может использоваться для задания линеек и заполнений. «Незвёздные» версии рисуют линейки линейки толщиной .4 pt, или толщиной $\langle height \rangle$ если она задана. Например:

```
\titlerule[.8pt]%
\vspace{1pt}%
\titlerule
```

рисует две линейки различной толщины разделённые на 1 pt.

«Звёздная» версия создаёт заполнители с аргументом $\langle text \rangle$ повторяемым в боксах натуральной величины. Величина боксов может быть изменена на $\langle width \rangle$, но первый бокс остаётся своей натуральной величины поэтому $\langle text \rangle$ выравнивается по левому и правому полю заполняемого поля.

К примеру

```
\titleformat{\section}[leftmargin]
{\titlerule*[1pc]{.}%
```

¹²По умолчанию задан дополнительный аргумент `s` команды `\makebox`.

```
\vspace{1ex}%
\bfseries}
{... definition follows
```

создаёт отгочия сверху заголовка раздела.

`calwidth` (Package option)

Опция `wgar` имеет возможность измерения длины строк в заголовке при форматировании абзаца. Эта возможность может быть распространена на три другие опции заголовков — `display`, `block` и `hang` — при использовании данной опции пакета. Длина максимальной строки задаётся в параметре `\titlewidth`.

В \TeX 'е предполагается, что любой бокс содержит в себе какой-то наборный материал. Если бокс был увеличен пробельным материалом или, наоборот сжат, величина `\titlewidth` может оказаться неверной (по нашему разумению). Опция `hang`, к примеру, использует такой вид бокса, но в данном случае такое поведение желательно, когда используется выключка `вправо[??]`; в противном случае опция заголовка `block` создаст лучший результат. Другими словами, использование боксов ширины которых была подавлена может дать неверный результат.¹³ Более того, некоторые команды могут сбить с толку \TeX и остановить анализ заголовка. Но если вы зададите `\` и `\[. . .]` (другие команды разбивки на строки крайне нежелательны), то проблем не будет. Другой важный момент состоит в том, что аргументы `\langle before \rangle`, `\langle label \rangle`, `\langle sep \rangle`, и `\langle title \rangle` (но не `\langle after \rangle`) локально используются дважды; если вы используете *глобальную* команду изменения счётчика [это `\addtocounter`]. В большинстве случаев, при помещении проблемного определения в аргумент `\langle after \rangle` всё будет в порядке.¹⁴

Теперь опишем использование `\titleline`.

`\titleline*[\langle align \rangle]{\langle horizontal material \rangle}`

Текст будет помещён в бокс шириной `\titlewidth`; этот бокс в свою очередь помещён в бокс с определённой выключкой. Аналога `\titlerule` нет, и поэтому, чтобы использовать параметр `\titlewidth`, можно её поместить в `\titleline*`:

```
\titleline*[c]{\titlerule[.8pc]}
```

3.5. Разрывы

`\sectionbreak` `\subsectionbreak` `\subsubsectionbreak`
`\paragraphbreak` `\subparagraphbreak`

Определяя эти команды, используя `\newcommand` можно определить различные разрывы на странице для различных уровней. Если они не определены, используется штраф по умолчанию, предлагаемый классом (обычно `-300`, соответствует параметру штрафа `[\@medpenalty]`). Например,

```
\newcommand{\sectionbreak}{\clearpage}
```

«заставляет» начать раздел с новой полосы. При некоторых вариантах оформления, нужно создать отбивку сверху, даже в случае размещения его вверху полосы; этого можно достигнуть:

```
\newcommand{\sectionbreak}{%
  \addpenalty{-300}%
  \vspace*{0pt}}
```

¹³Который создаст выключные строки, с увеличенным междусловным пробелом.

¹⁴Есть ещё два параметра, `\titlewidthfirst` и `\titlewidthlast`, которые возвращают длину первой и последней строк. Для их использования не определены специальные инструменты, но вы можете внести их значения в параметр `\titlewidth` а затем использовать `\titleline*`.

2.6 До сих пор `\...break` была возможна только для класса заголовков `straight`, но теперь это возможно и для заголовков класса `top`. Вполне возможно определить `\cleardoublepage` (это используется по умолчанию с опцией класса документа `openright`) и `\clearpage` (что используется при опции `openany`). Теперь, можно изменить установки опции `openright`, определив `\chapterbreak` как `\clearpage`, определив класс главы как `top` (в данном примере, части будут использовать параметры опции класса документа `openright`).

`\chaptertolists`

2.6 Если определена эта команда, отбивка, обычно появляющаяся в списках рисунков и таблиц будет заменена на код заданный в этой команде. Если вам нужно отменить отбивки при переходе к другой главе, определите её как пустую. Эта команда не единственный инструмент для создания отбивок в списках и действует только в заголовках класса `top`, и не действует в главах созданных по умолчанию, только если не был изменён их класс (с другой стороны, `\...tolists` может быть использован для любого заголовка класса `top`).

3.6. Другие опции пакета

`newparttoc oldparttoc` (Package options)

Стандартные части записывают номер части в оглавлении не так, как в других заголовках. Вы можете изменить это задав `newparttoc`, в результате `titletoc` или подобный пакет может манипулировать с нумерационным заголовком. (Это работает, если оформление `\part` было переопределено.)

`cleareempty` (Package options)

Изменяет поведение `\cleardoublepage`, так, что на пустых полосах используется стиль колонтитула `empty`.

`toctitles` (Package option)

2.6 Изменяет поведение дополнительного аргумента заголовков так, что он задаётся только в колонтитулах, а в оглавление отправляется полная версия заголовка.

`newlinetospace` (Package option)

2.6 Заменяет команду `\\` или `*` в заголовках на пробел в колонтитулах и оглавлении. В результате нет необходимости повторять заголовок только для того, чтобы удалить команды новых строк.

3.7. Дополнительные настройки

Первый аргумент команд `\titleformat` и `\titlespacing` позволяет использовать расширенный синтаксис задания опций заголовков, позволяющий использовать различные оформления в зависимости от контекста.¹⁵ Этот аргумент должен быть перечнем пар `key/value` в виде:

`\langle key \rangle = \langle value \rangle, \langle key \rangle = \langle value \rangle, \langle key \rangle, \langle key \rangle, \dots`

В данный момент, только страницы и нумерованные заголовки могут понадобиться, помимо команды заголовка. Итак, существующие ключи (keys):

- `name`. Возможные опции `\chapter`, `\section`, и т. д.
- `page`. Возможные опции `odd` или `even`.

¹⁵Для этого требуется пакет `keyval`.

- `numberless`. Ключ не имеет опций. В нём нет необходимости, только если вы не захотите создать различные варианты оформления для нумерованного (без этого ключа) и ненумерованного (с ключом `numberless`) заголовков.

Основная форма начала описания заголовка, скажем

```
\titleformat{\section} ...
```

фактически является аббревиатурой для

```
\titleformat{name=\section} ...
```

Предположим что нам нужен макет заголовков на внешнем поле. Мы можем задать, например

```
\titleformat{name=\section,page=even}[leftmargin]
  {\filleft\scshape}{\thesection}{.5em}{}
```

```
\titleformat{name=\section,page=odd}[rightmargin]
  {\filright\scshape}{\thesection}{.5em}{}
```

Поскольку информация о странице записывается в `aux`-файл, то требуется прокрутить L^AT_EX не менее двух раз, чтобы получить желаемый результат.

Нумерованной версии обычно достаточно, если при генерации ненумерованной версии достаточно только удалить номер заголовка, но если вам нужно специальное форматирование, используется ключ `numberless` с которым определяется альтернативное оформление заголовка (это относится к заголовкам «за пределами» `secnumdepth`, во вступительной и заключительной частях книги и, естественно, к «звёздным» заголовкам). Например

```
\titleformat{name=\section}{...% The normal definition follows
\titleformat{name=\section,numberless}{...% The unnumbered
% definition follows
```

Опции `<label>` и `<sep>` игнорируются в варианте `numberless`.

Эти ключи можно использовать как в `\titleformat`, так и в `\titlespacing`. Используя `page` в одной (или обеих) командах, чётная и нечётная страницы будут оформлены по разному. Фактически, «чётный» и «нечётный», уже установившиеся термины L^AT_EX'а, но они обманчивы. Вдобавок при односторонней печати «нечётными» в оформлении становятся и «чётные» страницы (ср. `\oddsidemargin`).

Если вы предполагаете создать разные чётные/нечётные и разные нумерованные/ненумерованные версии, рекомендуется определить четыре варианта.

Если был удалён определитель полосы в команде заголовка, то нужно удалить и `.aux`-файл.

3.8. Создание нового уровня и изменение класса заголовка

Если опции заголовков изменяют их поведение относительно окружающего текста, классы заголовков позволяют изменять их генерацию. С помощью классов вы можете добавить новый уровень заголовка `subchapter` между `chapter` и `section`, или создать свою собственную схему. *За создание последовательной схемы и определение полноценного инструментария, такого, как счётчики, макрокоманды, оформление и отбивки отвечает пользователь.* Созданы три класса: `page` для части книги `\part`, на отдельной полосе, `top` для главы `\chapter`, которая начинает полосу и помещает заголовок вверху полосы, и `straight` предназначен для заголовков внутри текста.¹⁶

```
\titleclass{<name>}{<class>}
\titleclass{<name>}{<class>}[<sup-level>]
```

Если не используется дополнительный аргумент, то изменяется только класс `<class>` заголовка `<name>`. Например:

¹⁶Есть ещё один класс `part` для эмуляции `\part` в классе `\part`, но лучше его не использовать. Используйте класс `straight`. Remember that some features rely in these classes and that `titlesec` does not change by default the definition of `\part` and `\chapter`.

```
\titleclass{\part}{straight}
```

задаёт части `part` класс `straight` (внутри текста).

Если же дополнительный аргумент используется, устанавливается уровень заголовка следующим за заданным в аргументе $\langle sup-level \rangle$. Например:

```
\titleclass{\subchapter}{straight}[\chapter]
\newcounter{subchapter}
\renewcommand{\thesubchapter}{\Alph{subchapter}}
```

создаёт уровень заголовка следующий за частью (в примере добавлен ещё дополнительный код, но нужно не забыть определить `\titleformat` и `\titlespacing`).¹⁷ Если уровень главы равен 0, то `subchapter` получает 1; и уровни последующих заголовков увеличатся на единицу (`section = 2`, `subsection = 3`, и т. д.). [*Примечание при переводе.* Возможно понадобится проконтролировать и изменить глубину выводимых заголовков в оглавлении с помощью счётчика `tocdepth`.]

Существуют две команды которые привязаны к конкретным заголовкам, вне зависимости от заданного их класса:

- `\chapter` использует определение, заданное в `\chaptertitlename`.
- `\part` не выводит уровень в оглавление, только если вы не задали опцию пакета `newparttoc`.

`loadonly` (Package option)

Итак, вы желаете создать свою рубрикацию с самого начала. Эта опция пакета игнорирует любые предопределения, если они есть, и следовательно аннулирует опции пакета, описанные в разделе «Quick Reference». Теперь вы используете инструменты `titlesec`, и создаёте соответствующие счётчики и рубрикацию.

`\titleclass{\langle name \rangle}[\langle top-num \rangle]{\langle class \rangle}`

Здесь, аргумент $\langle name \rangle$ предполагает определение верхнего уровня в аргументе $\langle top-num \rangle$ (обычно 0 или -1). Такая запись используется в случае, если команды заголовки задаются с нуля, с использованием опции пакета `loadonly`, и должна использоваться только *одна* (не более, не менее) декларация подобного рода. После этого остальные заголовки определяются, как описано выше.

4. Дополнительные замечания

Этот раздел описывает вкратце некоторые команды \LaTeX 'а, полезные при определении заголовков.

4.1. Нумерационные заголовки фиксированной ширины

Команда `\makebox` позволяет создать нумерационные заголовки фиксированной ширины, что делает левое поле заголовка (не номер) постоянным. Например (приведён только необходимый код для примера):

```
\titleformat{\section}
{. .}
{\makebox[2em]{\thesection}}{. .}{. .}
```

См. подробнее о боксах в вашем руководстве по \LaTeX 'у.

¹⁷Что касается счётчиков, то будет полезен пакет `remreset`.

4.2. «Звёздные?» версии

Использование команд рубрикации в «звёздной» версии *настоятельно не рекомендуется*.

Вместо этого, можно использовать команды макетирования, которые легко определить и изменять. Поэтому, можно протестировать различные установки и выбрать нужную.

Вначале нужно запомнить, что если задать

```
\setcounter{secnumdepth}{0}
```

заголовки разделов не будут нумероваться но будут попадать и в колонтитулы и в оглавление.

Теперь, предположим что вам не нужна нумерация у некоторых разделов, например, у разделов в упражнениях. Мы можем использовать окружение, названное `exercises`, которое может использоваться так:

```
\section{A section}
Text of a normal section.
```

```
\begin{exercises}
\section{Exercises A}
Some exercises
```

```
\section{Exercises B}
Some exercises
\end{exercises}
```

Следующее определение отменяет нумерацию, но не строки в оглавлении или в колонтитулах.

```
\newenvironment{exercises}
{\setcounter{secnumdepth}{0}}
{\setcounter{secnumdepth}{2}}
```

Следующее окружение добавляет заголовок в оглавление но не в колонтитул:

```
\newenvironment{exercises}
{\setcounter{secnumdepth}{0}%
\renewcommand\sectionmark[1]{}}
{\setcounter{secnumdepth}{2}}
```

Следующее окружение изменяет колонтитулы, но не добавляет строк в оглавление:

```
\newenvironment{exercises}
{\setcounter{secnumdepth}{0}%
\addtocontents{toc}{\protect\setcounter{tocdepth}{0}\ignorespaces}}
{\setcounter{secnumdepth}{2}%
\addtocontents{toc}{\protect\setcounter{tocdepth}{2}\ignorespaces}}
```

(Я нашёл последний вариант несколько странным; первый и второй вариант выглядят более реально. Команда `\ignorespaces` не очень важна, и вам она не понадобится, если только не хотите появления ненужных пробелов в оглавлении.)

Это работает со стандартными классами, но если вы используете `fancyhdr` или `titlesec` при определении колонтитулов то потребуются дальнейшая настройка, чтобы удалить номер раздела. В пакете `titlesec` это можно осуществить с помощью команды `\ifthesection` (см. ниже).

Как можно увидеть, в примерах нет команд `\addcontentsline`, `\markboth` или `\section*`, только логическая структура. Естественно можно изменить окружение как пожелаете; если вы решите набрать эти разделы уменьшенным кеглем, можно включить команду `\small`, и если обнаружите что этого не нужно, просто удалите.

Поскольку стандартные команды \LaTeX 'а проще и прямо созданы для решения простых проблем, я думаю предлагаемый метод в примерах выше наиболее предпочтителен в больших документах.

4.3. Варианты (специальные знаки в заголовках)

Предположим, что мы хотим пометить некоторые разделы как усложнённые со звёздочкой после номера. Следующий код решит эту проблему:

```
\newcommand{\secmark}{}
\newenvironment{advanced}
  {\renewcommand{\secmark}{*}}
  {}
\titleformat{\section}
  {...}
  {\thesection\secmark\quad}{.}{.}
```

Теперь поместите усложнённые разделы в окружение

```
\begin{advanced}
\section{...}
...
\end{advanced}
```

Такие марки пометят разделы `\section`, но не `\subsection`. Если вам нравятся излишества и пометить уровень заголовков `\subsection`, вы должны их пометить непосредственно.

[Если звёздочки нужно ставить не в группе разделов, а в отдельных разрозненных заголовках, пригодится создание команды, например, `\Star`, обычно пустой

```
\def\Star{}
```

в команде настройки заголовка после нумерационной части нужно задать эту команду и повторить её переопределение как пустой с помощью команды `\gdef`, действие которой выходит за все группы. В результате переопределение будет распространяться только на один заголовок.

```
\titleformat{\section}
  {...}
  {\thesection\Star\gdef\Star{}}
  ...
```

и команды, которая будет использоваться перед заголовком

```
\def\starred{\def\Star{$^\star$}}
```

]

4.4. Поставить точку после текста заголовка

Сегодня такой стиль не используется, но в прошлом был очень популярен. Основная техника была описана выше, Здесь только напоминание:

```
\newcommand{\periodafter}[1]{#1.}
\titleformat{\section}
  {...}
  {\thesection}{.}{.}\periodafter}
```

Если вам нужно совместить задание точки с другой командой (к примеру подчёркиванием), можно задать:

```
\newcommand{\periodafter}[2]{#1{#2.}}
\titleformat{\section}
  {...}
  {\thesection}{.}{.}\periodafter{\ul}} % \ul from soul package
```

Команда `\ul` из пакета `soul`.

[Приём, описанный выше можно применить и здесь определив например, `\PutDot`, в виде точки

```
\def\PutDot{.}
\newcommand{\periodafter}[1]{#1\PutDot\gdef\PutDot{.}0}
}
```

А перед заголовком, который заканчивается, например, вопросительным знаком можно переопределить `\PutDot`, и это переопределение будет работать только один раз.

```
\def\PutDot{}
]
```

5. Стили колонтитулов

Как часть пакета ряд команд предложен для верхних и нижних колонтитулов. Эти стили колонтитулов были созданы для работы с заголовками; например, их не стоит использовать для создания колонтитулов по статьям словаря (вы можете это сделать, но это потребует более сложных определений чем с пакетом `fancyhdr`).

Команды работают со стандартными классами и многими другими; однако, в некоторых из них необходимы дополнительные настройки L^AT_EX'a (к примеру, для классов AMS).¹⁸ Для использования их, задайте опцию пакета:¹⁹

```
pagestyles
```

Задав эту опцию, вы теперь можете задать свои стили колонтитулов, а затем активировать их.²⁰

5.1. Определение стилей колонтитула

```
\newpagestyle{<name>}[<global-style>]{<commands>}
\renewpagestyle{<name>}[<global-style>]{<commands>}
```

Определяет новый или переопределяет существующий, названный `<name>`. Для удобства, концы строк игнорируются, и вам не нужно «прятать» их знаком процента %.²¹

Аргумент `<global-style>` — это любая команда, которая добавляется как в верхний, так и в нижний колонтитулы. Можно использовать хрупкие команды.

В аргументе `<commands>` можно использовать (эти команды локальные в стиле колонтитула, т. е., команды, определённые непосредственно в преамбуле не будут работать, поскольку установки стиля колонтитула переопределяют их):

```
\headrule          \footrule
\setheadrule{<length>} \setfootrule{<length>}
```

Если вам нужна линейка под верхним колонтитулом или над нижним. Можно также задать их толщину с помощью команд `\setheadrule` и `\setfootrule`. (Например, `\setheadrule{.4pt}` что является значением по умолчанию.)

```
\makeheadrule \makefootrule
```

Эти команды используются пакетом `titlesec` для создания линеек. Если линейка не нужна, обе команды пустые (это задано по умолчанию). Команда `\setheadrule{d}` заменяет код

```
\renewcommand{\makeheadrule}{\rule[-.3\baselineskip]{\linewidth}{d}}
```

¹⁸ Двухколонный набор требует подключения пакета `fix2col` (David Carlisle).

¹⁹ Поддержка стилей колонтитулов полностью осуществлена в версии 2.3. Поэтому, в большинстве случаев вы не заметите это вообще.

²⁰ Стили колонтитулов, определённые таким способом не могут использоваться вместе с командой `\thispagestyle` если только окружающие стили колонтитула определены с помощью стилей пакета `titlesec`.

²¹ Даже если марки предназначены для вывода информации в колонтитулы, некоторые классы, как классы AMS, предлагают дополнительный код. При переопределении стиля `plain` классов AMS, вы должны добавить следующее: `\global\topskip\normaltopskip`.

если `d` равно `0 pt`, отменяет `\makeheadrule` (и аналогично `\setfootrule`).

Вы можете определить толщину линеек с помощью команды `\linewidth` но бокс, включающий линейку фактически «безразмерный». Отсюда, ваша задача только разместить линейку. Её базовая линия та же, что и для верхнего и нижнего колонтитулов. Это означает что материал, используемый как линейка может быть поднят или опущен с командой `\raisebox`, смещающий аргумент в команде `\rule` или использовать, например, координаты в окружении `picture`. Это делает простым размещение материала сверху и снизу. Например, следующий код создаёт линейку верхнего колонтитула как чёрной жирной линейки сверху и красной линией снизу (для этого понадобится пакет `color`):

```
\renewcommand{\makeheadrule}{%
  \makebox[0pt][1]{\rule[.7\baselineskip]{\linewidth}{0.8pt}}%
  \color[named]{Red}%
  \rule[-.3\baselineskip]{\linewidth}{0.4pt}}
```

Конечно материал используемый как линейка не запрещён для линеек; картинки и заполнения, к примеру, возможны тоже.

<pre>\sethead[<i>(even-left)</i>][<i>(even-center)</i>][<i>(even-right)</i>]{<i>(odd-left)</i>}{<i>(odd-center)</i>}{<i>(odd-right)</i>} \setfoot[<i>(even-left)</i>][<i>(even-center)</i>][<i>(even-right)</i>]{<i>(odd-left)</i>}{<i>(odd-center)</i>}{<i>(odd-right)</i>}</pre>
--

Задаёт соответствующий элемент в колонтитулы. Дополнительные аргументы могут быть заданы все или ни одного. Если не заданы дополнительные аргументы, установки с колонтитула на нечётной полосе распространяются на чётную. «Звёздные» команды (`\setfoot*`, `\sethead*`) переворачивают установки для чётных полос (очевидно, если используются эти команды, дополнительных аргументов здесь нет). В аргументах команд `\sethead`/`\setfoot` и из-за одношагового механизма, нужно сделать различие между двумя группами команд. В первой заданы те, которые сохранены, когда марки посланы, с информацией относящейся к разделам, и включает:

- `\thechapter`, `\thesection`...
- `\chaptertitle`, `\sectiontitle`... который печатает заголовки разделов `chapter`, `section`...
- `\ifthechapter{(true)}{(false)}`, `\ifthesection{(true)}{(false)}`... которые являются `<true>` кроме случая, когда заголовок не того уровня или ещё не было заголовка нужного уровня (например, между `\chapter` и соответствующим заголовком `\section`).
- Любая команда или значение «зарегистрированное» как «марка». (См. ниже.)

Следующая группа действуют на каждой странице, и включают:

- `\thepage`.
- Любая другая команда, не включённая в предыдущие элементы.

<pre>\setmarks{<i>(primary)</i>}{<i>(secondary)</i>}</pre>
--

Определяет которые команды `\dots title` должны быть определены, и где марки обновляются. Например `\setmarks{chapter}{section}` означает:

- `\chaptertitle` и `\sectiontitle` теперь будут помещаться в колонтитулы,
- `\sectiontitle` переустанавливается в команде `\chapter`,
- тесты `\ifthechapter` и `\ifthesection` будут действующими и
- марки обновляются в командах `\chapter` и `\section`.

По умолчанию заданы `\setmarks{chapter}{section}`, но `\setmarks{section}{subsection}` в классе `article`. Команды `\setmarks` могут быть использованы вне `\(re)newpagestyle` для задания установок по умолчанию для заданных новых стилей колонтитулов. Используйте это до `\pagestyle`.

Обратите внимание, что команда `\markboth` устанавливает колонтитулы для марок стиля `myheadings` и здесь не действует. Фактически, её использование может привести к неожиданным результатам. В любом случае, используя непосредственно команды маркировки бездействуют (см. разд. 4.2) но если они вам нужны, вы можете использовать в виде:

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

5.2. Дополнительные настройки

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

Делают head/foot шире. Дополнительная ширина асимметрично добавляется и здесь два [четыре] аргумента. Также как `\sethead`, «звёздные» версии оборачивают установки для чётных полос — например, `\widenhead*{Opt}{6pc}` то же, что и `\widenhead[6pc][Opt]{Opt}{6pc}`.

5.3. Колонтитулы с плавающими объектами

```
psfloats (Package option.)
```

Эта опция пакета активирует команды описанные в данном разделе.²²

2.6 Команды ниже используют новый аргумент, который позволяет дополнительные команды в колонтитулах. Это изменение обратно совместимое — используйте старую опцию пакета `floatps`, которая теперь устарела.

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

Аргументы с точкой аналогичны аргументам в `\sethead`. Последний аргумент говорит, что этот колонтитул используется, если есть плавающие объекты определённого расположения на странице (по умолчанию `tp` для верхний и `bp` для нижних колонтитулов). Например:

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

убирает колонтитул на страницах только с плавающими объектами, включая линейки.

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

Вы можете форсировать верхний колонтитул (и соответственно нижний) когда соответствующий плавающий объект является первым сверху (и, соответственно последний внизу) с соответствующими командами, где аргументы с точкой аналогичны аргументам из `\sethead`, перед соответствующим объектом. Эти команды используются самостоятельно, без размещения их внутри `\(re)newpagestyle`.

²²Она переопределяет некоторые внутренние команды \LaTeX 'а и возможны несовместимости с другими пакетами, изменяющими те же команды. Это единственная причина, по которой эти команды загружаются «по требованию».

5.4. Маркировка

```
outermarks innermarks topmarks botmarks (Package options)
```

Опция `innermarks` задаётся по умолчанию в \LaTeX 'е, с нижними марками на чётных полосах и верхними марками на нечётных. Более подходящей опцией пакета будет опция `outermarks`, с верхними марками на чётной полосе и нижними марками на нечётных; это более обычная система колонтитула для технических работ и описана в *The TeXbook*, с. 259. Обе эти опции предлагаются для двусторонней печати; опции `topmarks/botmarks` предназначены для верхних/нижних марок на каждой полосе (хотя их можно использовать и для двусторонней печати).²³

```
\bottitlemarks \toptitlemarks \firsttitlemarks \nexttoptitlemark
\outertitlemarks \innertitlemarks
```

Одна из дополнительных возможностей, добавленных в стилях колонтитула — это возможность задания марок в соответствующих макрокомандах. Эти команды задают какая марка из `\thesection`, `\sectiontitle`, и т. д. берётся. Вы можете использовать их для колонтитулов;²⁴ как пример, следующий колонтитул показывает нумерационную часть из одной из трёх марок:

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

Команда `\outertitlemarks` является командой по умолчанию, если задана опция пакета `outermarks`, команда `\toptitlemarks` — если задана `topmarks`, и т. д. Команда `\innertitlemarks` является командой по умолчанию в стандартных установках \LaTeX 'а и в данном пакете, если не задана другая опция пакета. Следующая установка марок — команда `\nexttoptitlemarks` определяет нижнюю марку как верхнюю на следующей странице (только класс `straight`); взяв значение и сравнив с соответствующим значением из `\bottitlemarks` вы можете узнать, продолжается ли раздел на следующей странице. Естественно, такие команды не должны использоваться слишком часто.

```
\newtitlemark{<macro-name>}
\newtitlemark*{<variable-name>}
```

Добавить макрокоманду или переменную в перечень «марок» которые нужно сохранить. Макрокоманда не должна иметь параметры, и переменная должна быть задана в форме \TeX 'а (что то же в длинах но не в счётчиках).

5.5. Парочка примеров

Поскольку марками можно управлять, можно создать следующие конструкции:

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

Как видно, интервал разделов на странице `\thesection` печатается в обоих, правом и левом колонтитулах. Естественно, этот пример должен быть настроен так, чтобы собрать интервал,

²³Опция `outermarks` не работает, если используются прямые команды `\<section>mark`. Механизм верхних марок провалится сразу после использования команд прямого разрыва на странице.

²⁴Не совсем так: верхние марки не будут использоваться на странице, где начинается глава, если только переопределить установки по умолчанию и сделать их классом `top`.

если на странице один раздел, но это даёт подсказку как получить такие потрясающие колонтитулы.

В данном документе, используются следующие стили:

```
\widenhead{2.1pc}{0pc}
\renewpagestyle{plain}[\small\sffamily\slshape]{
  \footrule
  \setfoot{}{\thepage}{}}
\newpagestyle{mysp}[\small\sffamily\slshape]{
  \headrule
  \sethead{Titlesec}{\sectiontitle}{\thepage}}
```

Следующие определения создают стиль колонтитула, аналогичный книге Лампорта \LaTeX book (с пакетом calc):

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

6. Заключительные комментарии

- Изменение в последний момент — but of course! — [представляет] ошибку версии 2.0 который добавляет нежелательный пробел перед `\titleline`. Пример, предложенный

```
\titleline{\hrulefill}%
\vspace{1pt}%
\titleline{\hrulefill}%
```

как вариант добавить линейки, разделённые отбивкой в 1 pt, но фактически пробел оказывается гораздо больше. Это было исправлено в версии 2.1, что в результате создало несовместимость с предыдущими версиями. Как решение, я настоятельно рекомендую изменить команды, чтобы перенастроить отбивки, но если по каким-то причинам файлы должны быть сохранены, вы можете создать стиль со следующим кодом:

```
\def\ttl@line[#1]#2{%
  \par\hrule height \z@
  \nobreak
  \begingroup
    \parindent\z@
    \leftskip\z@
    \rightskip\z@
    \makebox[\hsize][#1]{\ttl@makeline{#2}}%
  \par
  \endgroup
  \hrule height \z@
  \nobreak}
```

и через `\usepackage` загрузить его. Не используйте это в новых документах.

- В заголовках можно использовать команду `\footnote`, но это работает. Однако, знак сноски не удаляется из оглавления и колонтитула; для этого используйте пакет с опцией

```
\usepackage[stable]{footmisc}
```

- Стили колонтитулов полностью переопределены полностью в версии 2.3. Большинство предыдущих определений будут работать с текущим кодом, даже если многие вещи были изменены. В частности, колонтитулы, использующие команды `\the...` и `\...title`, помимо `\usepage` и форматизирующих команд до сих пор работают. Теперь, стили колонтитула могут быть заданы прямо с опцией пакета `pagestyles`, но если она не была задана и три базовых команды использовались, макрокоманды стилей колонтитулов загружаются с предупреждением. Новый код преодолевает ограничения предыдущих релизов, фиксирует некоторые недочёты (например, несовместимость между верхними марками и плавающими объектами), и добавляет новые возможности.

7. Оглавление: пакет `titletoc`

Этот пакет сопровождает пакет `titlesec` и управляет элементами оглавления. Однако он является независимым пакетом, и его можно использовать самостоятельно. Философия пакета аналогична пакету `titlesec` — помимо настроек стандартных команд, определённых в \LaTeX 'е и классах, заданы новые команды, позволяющие оформить элементы оглавления. Это означает что вам нужно изучить две базовых команды и пару инструментов, не более, и вы должны открыть доступ другим опциям пакета. Оформление абзаца и шрифты задаются командами типа `\`, `\makebox`, `\large`, `\itshape`, и т. д., а в элементах оглавления оформление отсутствует, поскольку они имеют свободную форму.

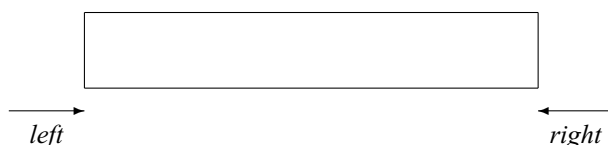
Поведение элементов определённое с пакетом `titletoc` отлично в некоторых пунктах, по сравнению со стандартными командами. В частности:

- Разрыва на страницы никогда не будет между элементами, если первый из них имеет старший уровень по отношению ко второму, например, между разделом и подразделом. Если оба элемента одного уровня то разрыв разрешён, если же первый элемент младше последующего, то у разрыва будет предпочтение.
- Символы в заполнениях не центрируются, а выключаются вправо. Обычно это более предпочтительно.

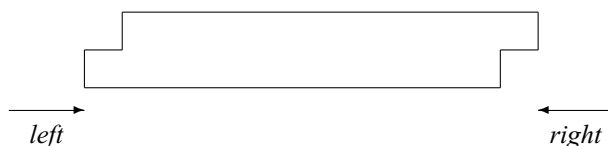
Нужно заметить, что попытки настроить оглавление не могут быть полными, поскольку стандартные команды \LaTeX 'а записывают непосредственно часть форматирования, которое не может быть удалено. Это, в частности, важно в списках рисунков и таблиц, и в команде `\part`.²⁵

7.1. Десятиминутный гид по `titletoc`

Элементы оглавления создаются как прямоугольники, где печатаются текст и, возможно, заполнитель. Теперь нарисуем такой прямоугольник (конечно сами строки не печатая):

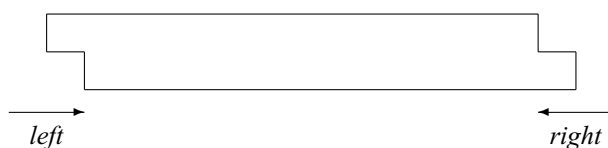


Отбивка между левым полем страницы и левым краем элемента назовём $\langle left \rangle$; аналогично мы имеем и $\langle right \rangle$. Вы можете изменить начало первой строки и завершение последней строки. Например, «заняв» оба места `\hspace*{2pc}` прямоугольник станет таким:



А если «очистить» пространство в обоих местах с помощью `\hspace*{-2pc}` прямоугольник станет следующим:

²⁵Но некоторые из этих установок можно изменить с помощью пакета `titlesec`.



Если вы видели оглавления, то последнее окажется вам знакомым — номер в самом начале, страница в самом конце:

```
3.2 This is an example showing that toc
    entries fits in that scheme . . . . 4
```

Всё, что вам нужно — это поместить эти элементы нужным образом. Если вы зарезервировали пространство через `\hspace*{-2pc}`, просто добавьте бокс шириной в 2 pc содержащий номер заголовка или номер страницы и это пространство будет заполнено соответствующим образом; такое оформление используется так часто, что для этого созданы две специальные команды:

- `\contentslabel{<length>}` создаёт пространство в начале элемента оглавления и печатает номер раздела.
- `\contentspage` создаёт пространство в конце и печатает номер страницы.

Теперь продемонстрируем три основные команды

```
\dottedcontents{<section>}[<left>]{<above>}
                 {<label width>}{<leader width>}
```

Здесь:

- `<section>` название раздела без бэкслэша: `part`, `chapter`, `section`, и т. д. элементы `figure` и `table` тоже можно использовать. (Бэкслэш опущен, поскольку мы не имеем дело с командами `\part`, `\section`, и т. д. Кроме того, `figure` и `table` являются окружениями.)
- `<above>` задаёт код для глобального форматирования элемента. Можно использовать вертикальный материал. В этом месте значение команда `\thecontentslabel` (см. ниже) уже известно, что позволяет вам задать установки на основе её значения (с помощью пакета `ifthen`). Вы можете использовать команды из пакета `titlesec` `\filleft`, `\filright`, `\filcenter` и `\fillast`.
- `<left>` хотя и задан как дополнительный, но аргумент является обязательным и задаёт левое поле элемента.
- `<label width>` задаёт ширину для нумерационной части элемента, как описано выше.
- `<leader width>` ширина бокса, содержащего знак, используемый как заполнитель, как описано ниже.

Определения для элементов `section` и `subsection` в классе `book` эквивалентны следующему:

```
\contentsmargin{2.55em}
\dottedcontents{section}[1.5em]{-}{2.3em}{1pc}
\dottedcontents{subsection}[3.8em]{-}{3.2em}{1pc}
```

```
\titlecontents{<section>}[<left>]{<above>}
                {<before with label>}{<before without label>}
                {<filler and page>}[<after>]}
```

Здесь аргументы `<section>`, `<left>` и `<above>` те же, что и выше, а

- `<before with label>` задаётся в горизонтальном режиме и используется перед заголовком. Как в команде `\titleformat`, последняя команда может оперировать аргументом, которым является текст заголовка.

- `<before without label>` то же что и предыдущий, но если нет нумерационной части.
- `<filler and page>` ясно из названия. Заполнители создаются командой `\titlerule` которая используется и пакетом `titlesec`. Однако, когда используется в данном контексте, её поведение несколько изменяется, чтобы удовлетворить потребностям заполнителей оглавления.²⁶ Вместо неё вы можете предпочесть команду `\hspace`.
- И наконец `<after>` — это код, для, скажем, вертикальной отбивки.

При определении элементов, используйте `\addvspace` если вам нужно добавить вертикальную отбивку и `*` вместо `\\` для разрывов строк.

Эта команда может быть задана в любом месте документа для изменения оформления элементов `toc/lot/lof`. Новый формат будет записан в файл оглавления `toc` и чтобы увидеть изменения нужно документ прокрутить два раза.

`\contentsmargin{<right>}`

Заданная величина будет использоваться всеми элементами. Если вы спросите почему, ответ простой: в большинстве случаев правое (`<right>`) поле должно быть одинаковым. Однако, вы можете изменить его в аргументе `<before>`. Обратите внимание, что пространство в стандартных классах не предусмотрено для полужирных номеров выше 100 и поэтому вы пожелаете увеличить поле с помощью этой команды.

Класс `book` оформляет раздел аналогично (но не полностью) как:

```
\titlecontents{section}
    [1.5em]
    {}
    {\contentslabel{2.3em}}
    {\hspace*{-2.3em}}
    {\titlerule*[1pc]{.}\contentspage}
```

Сравните это определение с заданным выше и вы поймёте, как определяется команда `\dottedcontents`.

Хотя стандартные классы используют единицы измерения, зависящие от шрифта (обычно `em`), всё же рекомендуется использовать абсолютные единицы (`pc`, `pt`, и т. д.)

7.2. И ещё

Типографские правила требуют, чтобы полные строки заголовков не заканчивались правее последней точки заполнителя; в идеале они строки с текстом и отточия должны быть выровнены. К сожалению, `TeX` не имеет инструмента, чтобы сделать это автоматически — когда вы заполняете бокс точками, они могут быть отценрованы с помощью примитивы `\cleaders`, «выключены» на формат с помощью `\xleaders` или выровнены по самому последнему боксу с помощью `\leaders`, но нет возможности выровнять их по заданному «текущему» полю.

Поэтому единственный путь получить нужный результат, установить его вручную. Чтобы сделать это, вы можете использовать дополнительный аргумент в команде `\contentsmargin` полный синтаксис которой таков:

`\contentsmargin[<correction>]{<right>}`

Длина `<correction>` добавляется справа ко всем строкам кроме последней, где помещаются заполнители. Например, если текстовые строки длиннее на 6 `pt`, вы можете перезадать `\contentsmargin` добавив `<correction>` в 6 `pt`.²⁷ В отличие от стандартных команд `LaTeX`'а, команда `\titlerule*` настроена так, что требуется минимум коррекции.

²⁶Для `TeX`нарей, команда `\xleaders` превращается в `\leaders`.

²⁷Большинство программ просмотра `dvi`файлов позволяют установить координаты на странице.

```
\thecontentslabel \thecontentspage
```

Содержит текст с номером заголовка и номером страницы без дополнительного форматирования, более того, что уже определено в классе документа.

```
\contentslabel[format]{space}
\contentspage[format]
```

Как объяснено выше, но с различными аргументами *format*. По умолчанию это `\thecontentslabel` и `\thecontentspage`, соответственно.

```
\contentspush{text}
```

Печатает текст *text* и увеличивает втяжку *left* на ширину текста *text*. Аналогично опции заголовка `hang` в пакете `titlesec`.

```
\titlecontents*{section}[left]{above}
    {before with label}{before without label}
    {filler and page}[separator]
    or ...{filler and page}[separator][end]
    or ...{filler and page}[begin][separator][end]
```

«Звёздная» форма группирует элементы в один абзац. Аргумент *separator* является разделителем между элементами, здесь есть также ещё один дополнительный аргумент с завершающей пунктуацией. Например, в данном документе задано:

```
\titlecontents*{subsection}[1.5em]
    {\small}
    {\thecontentslabel. }
    {}
    {, \thecontentspage}
    [---][.]
```

результат которого можно увидеть в самом начале документа. Обратите внимание, что оформление абзаца должно быть задано в аргументе *above*.

Теперь объясним, как работает дополнительный аргумент. Вначале обратите внимание, что число их определяет их значение — поскольку есть разделитель между элементами он всегда будет присутствовать; с другой стороны, *begin* используется редко и поэтому имеет наименьшее «предпочтение». Простейший случай — когда элементы имеют один уровень; в этом случае параметры *sep* и *end* (который обычно пустой) помещаются между соответствующими элементами и в конце блока, соответственно. Аргумент *before* задаётся один раз вначале блока и его установки локальны для всего блока элементов.

Теперь представим, что мы группируем элементы двух уровней; в этом случае включается принцип вкладывания. Для иллюстрации идеи, используем элементы `section` и `subsection`. Когда элементы `subsection` начинаются после `section`, аргумент *before* включается и добавляется аргумент *begin* из `subsection`, который содержит только команды оформления. Затем элементы `subsection` добавляются через разделители, как описано выше. Когда появляется элемент `section`, ставится завершающая пунктуация элемента `subsection` и добавляется разделитель элементов `section` (если только блок не заканчивается на `subsection`, где в таком случае ставится завершение элемента `section`). Мы сказали «после `section`» потому что элемент `subsection` никогда не начинает блок.²⁸ Элементы `subsection` помещены внутрь элементов `section`, и декларации вновь будут локальными.

Иллюстрирующий пример:

```
\titlecontents*{section}[0pt]
    {\small\itshape}{-}{-}
```

²⁸В редких случаях это может быть необходимо, ещё[???].

```
{}[ \textbullet\ ] [.]
```

```
\titlecontents*{subsection}[Opt]
  {\upshape}{-}{-}
  {, \thecontentspage}[ ( ] [ . ] [ ) ]
```

создаёт нечто похожее на:

The first section • The second one • The third one (A subsection in it, 1. Another, 2) • *A fourth section* (A subsection in it, 1. Another, 2).

```
\contentsuse{<name>}{<ext>}
```

Даёт знать titletoc о существовании файла-оглавления с расширением *<ext>*. В основном, это подтверждает, что команда \contentsfinish будет добавлена в соответствующее оглавление (и которую нужно задать в конце оглавления, созданного вручную). Пакет уже определил

```
\contentsuse{figure}{lof}
\contentsuse{table}{lot}
```

```
leftlabels rightlabels (Package options)
```

Данные опции пакета определяют как выравнивается нумерация в команде \contentslabel. по умолчанию задано rightlabels. При leftlabels оформление по умолчанию (*<format>*) для \contentslabel становится \thecontentstlabel\enspace.

```
dotinlabels (Package option)
```

С этой опцией добавляется точка после номера в команде \contentslabel.

7.3. Миниоглавления

```
\startcontents[<kind>]
```

В том месте, где эта команда задана, начинается миниоглавление. Аргумент *<kind>* позволяет создать различные варианты оглавления и имеет установки по умолчанию (default). Эти установки могут смешиваться, но обычно они вкладываются. Например, вы хотите два типа миниоглавлений: для части и для главы (помимо полного оглавления, естественно). Когда начинается часть, запишите \startcontents[parts], а когда глава— \startcontents[chapters]. В этом случае новое оглавление начнётся с каждой частью и главой.²⁹

```
\stopcontents[<kind>]
\resumecontents[<kind>]
```

Завершает миниоглавление вида *<kind>*, которое может быть возобновлено. Поскольку миниоглавления завершаются со следующей \startcontents если нужно, вам не нужно использовать эти макрокоманды слишком часто.

```
\printcontents[<kind>]{<prefix>}{<top>}{<init-code>}
```

Печатает текущее миниоглавление вида *<kind>*. Используется оформление основного оглавления, если только не задан *<prefix>*. В этом случае используется оформление *<prefix><level>*, если оно определено. Например, если задан префикс l и определено оформление для lsection, то используется это определение; в противном случае, используется оформление section. Параметр *<top>* задаёт верхний уровень оглавления — для части это может быть 0 (chapter), для главы — 1 (section), и т. д. Наконец, аргумент *<init-code>* задаёт локальный код для данного

²⁹ Все миниоглавления помещаются в один файл с расширением .ptc.

оглавления; это может быть, к примеру, изменение счётчика `tocdepth` или величины поля `\contentsmargin`.

Простое использование может выглядеть так (предполагается, что вы используете и пакет `titlesec`):

```
\titleformat{\chapter}[display]
  {...}{...}{...} % Your definitions come here
  [\vspace*{4pc}]%
  \startcontents
  \printcontents{1}{1}{\setcounter{tocdepth}{2}}

\titlecontents*{lsection}[0pt]
  {\small\itshape}{-}{-}
  {}[ \textbullet\ ]{.}
```

Включены элементы уровня 1 и 2.

7.4. Минисписки 2.6

Вы можете захотеть создать минисписки рисунков и таблиц. Синтаксис аналогичен синтаксису при создании миниоглавлений. Команды:

```
\startlist[⟨kind⟩]{⟨list⟩}
\stoplist[⟨kind⟩]{⟨list⟩}
\resumelist[⟨kind⟩]{⟨list⟩}
\printlist[⟨kind⟩]{⟨list⟩}{⟨prefix⟩}{⟨init-code⟩}
```

Здесь аргументом `⟨list⟩` может быть либо `lof` либо `lot`. Обратите внимание, что `\printlist` не имеет аргумента `⟨top⟩`, поскольку рисунки и таблицы [обычно] не имеют уровней. В настоящее время (version 2.6), поддерживаются списки только этих флотов, но в будущем будет осуществлена поддержка и других списков. К сожалению, многие классы создают собственные команды оформления для таких списков (точнее, добавляют `\addvspace` в командах `\chapter`); Я до сих пор не знаю, как удалить эти команды без удаления других желательных команд, но пока быстрый трюк по удалению этих отбивок — это переопределение команды `\addvspace` в аргументе `⟨init-code⟩` как `\renewcommand\addvspace[1]{}`.

7.5. Примеры

```
\titlecontents{chapter}
  [0pt]
  {\addvspace{1pc}%
   \itshape}%
  {\contentsmargin{0pt}%
   \bfseries
   \makebox[0pt][r]{\huge\thecontentslabel\enspace}%
   \large}
  {\contentsmargin{0pt}%
   \large}
  {\quad\thepage}
  [\addvspace{.5pc}]
```

Номер главы выходит на поле, задан шрифтом бóльшим чем текст заголовка. Если глава не имеет номера (потому, скажем, что это предисловие или библиография) выделение полужирным отменено. Номер страницы ставится сразу после заголовка (без заполнителей), через кегельную.

```
\titlecontents{chapter}
  [3pc]
  {\addvspace{1.5pc}%
   \filcenter}
  {CHAPTER \thecontentslabel\*\ [.2pc]%
   \huge}
  {\huge}
```

```
{ } % That is, without page number
[\addvspace{.5pc}]
```

Текст заголовка отцентрирован, и нумерационный заголовок помещён над ним. Номер страницы отсутствует.

7.6. Добавление плавающего рисунка в оглавление

Команда `\addtocontents` действует и её можно использовать для определения таких специальных операций, как включение плавающего рисунка перед или после элемента оглавления. К сожалению, хрупкие аргументы нельзя использовать и запись сложного кода может сбить с толку. Трюк заключается в том, что команда определяется вовне и затем записывается с помощью `\protect`.

Предположим, мы хотим добавить рисунок до элемента оглавления.

```
\newcommand{\figureintoc}[1]{
  \begin{figure}
    \includegraphics{#1}%
  \end{figure}}
```

делает чёрную работу.

В месте где вставляется рисунок добавляется:

```
\addtocontents{\protect\figureintoc{myfig}}
```

7.7. Задание заголовков со спецзнаками

Теперь вспомним проблему иллюстрируемую в разделе о `titlesec`: отметить разделы, «повышенной трудности» звёздочками не только в тексте, но и в оглавлении. Приводим код:

```
\newcommand{\secmark}{}
\newcommand{\marktoc}[1]{\renewcommand{\secmark}{#1}}
\newenvironment{advanced}
  {\renewcommand{\secmark}{*}}%
  {\addtocontents{toc}{\protect\marktoc{*}}}
  {\addtocontents{toc}{\protect\marktoc{}}}
\titleformat{\section}
  {...}
  {\thesection\secmark}{...}{...}
\titlecontents{section}[...]{...}
  {\contentslabel[\thecontentslabel\secmark]{1.5pc}}{...}{...}
```

8. Философия пакета titlesec

При первом же чтении документации становится ясно, что это не пакет для случайного пользователя, который любит стандартное оформление и которому нужны лишь простые модификации. Это инструмент для серьёзного полиграфиста, которые имеет ясное представление, что ему нужно, но не имеет достаточной квалификации, как это сделать. Здесь не было попытки повысить вашу квалификацию в оформлении книги.

9. Приложение

Следующие примеры служат хорошей иллюстрацией возможностей. В данной части, параметр `\parskip` задан как 0 pt.

■

10 Пример определения раздела и его иллюстрация, повторяю, пример определения раздела и его иллюстрация

```
\titleformat{\section}[block]
  {\normalfont\bfseries\filcenter}{\fbox{\itshape\thesection}}{1em}{}
```

■

РАЗДЕЛ 11

Заголовок окружённый рамкой

```
\titleformat{\section}[frame]
  {\normalfont}
  {\filright
   \footnotesize
   \enspace РАЗДЕЛ \thesection\enspace}
  {8pt}
  {\Large\bfseries\filcenter}
```

■

12. Заголовок с линейкой

```
\titleformat{\section}
  {\titlerule
   \vspace{.8ex}%
   \normalfont\itshape}
  {\thesection.}{.5em}{}
```

■

13

Другой пример заголовка с линейкой

```
\titleformat{\section}[block]
  {\normalfont\sffamily}
  {\thesection}{.5em}{\titlerule\[\.8ex]\bfseries}
```

■

.....

14 Длина линейки сверху превышает самую длинную
строку заголовка на 2 pc

.....

15 Этот пример короче

```
\titleformat{\section}[block]
  {\filcenter\large
   \addtolength{\titlewidth}{2pc}%
   \titleline*[c]{\titlerule* [.6pc]{\tiny\textbullet}}}%
  \advspace{6pt}%
  \normalfont\sffamily}
  {\thesection}{1em}{}
```

```
\titlespacing{\section}
  {5pc}{*2}{*2}[5pc]
```

■

РАЗДЕЛ 16

Пример определения раздела и его иллюстрация, повторяю: пример определения раздела и его иллюстрация и снова определения раздела и его иллюстрация и опять пример определения раздела и его иллюстрация, итак, это пример определения раздела и его иллюстрация или, ещё раз, пример определения раздела и его иллюстрация

```
\titleformat{\section}[display]
  {\normalfont\fillast}
  {\scshape раздел \oldstylenums{\thesection}}
  {1ex minus .1ex}
```


Заголовок на полях выше был задан:

```
\titleformat{\section}[leftmargin]
  {\normalfont
   \titlerule* [.6em]{\bfseries.}%
   \vspace{6pt}%
   \sffamily\bfseries\filleft}
  {\thesection}{.5em}{
\titlespacing{\section}
  {4pc}{1.5ex plus .1ex minus .2ex}{1pc}
```

■

Следующие примеры предложены для оформления глав. Однако, в данном документе не может быть глав (команд `\chapter`) и будут показаны с помощью заголовков `\section` с небольшими изменениями.

ГЛАВА 20

Заголовок главы

```
\titleformat{\chapter}[display]
  {\normalfont\Large\filcenter\sffamily}
  {\titlerule[1pt]%
   \vspace{1pt}%
   \titlerule
   \vspace{1pc}%
   \LARGE\MakeUppercase{\chaptertitlename} \thechapter}
  {1pc}
  {\titlerule
   \vspace{1pc}%
   \Huge}
```

■

ГЛАВА XXI

Заголовок главы

```
\renewcommand{\thechapter}{\Roman{chapter}}
\titleformat{\chapter}[display]
  {\bfseries\Large}
  {\filleft\MakeUppercase{\chaptertitlename} \Huge\thechapter}
  {4ex}
  {\titlerule
   \vspace{2ex}%
   \filright}
  [\vspace{2ex}%
   \titlerule]
```

9.1. Полный пример

Далее полный комплекс переопределений заголовка.

```

\documentclass[twoside]{report}
\usepackage[sf,sl,outermarks]{titlesec}

% \chapter, \subsection...: no additional code

\titleformat{\section}
  {\LARGE\sffamily\slshape}
  {\thesection}{1em}{ }
\titlespacing{\section}
  {-6pc}{3.5ex plus .1ex minus .2ex}{1.5ex minus .1ex}

\titleformat{\paragraph}[leftmargin]
  {\sffamily\slshape\filright}
  {}{}{ }
\titlespacing{\paragraph}
  {5pc}{1.5ex minus .1ex}{1pc}

% 5+1=6, ie, the negative left margin in section

\widenhead{6pc}{0pc}

\renewpagestyle{plain}{}

\newpagestyle{special}[\small\sffamily]{
  \headrule
  \sethead[\usepage] [\textsl{\chaptertitle}] []
  {}{\textsl{\chaptertitle}}{\usepage}}

\newpagestyle{main}[\small\sffamily]{
  \headrule
  \sethead[\usepage] [\textsl{\thechapter. \chaptertitle}] []
  {}{\textsl{\thesection. \sectiontitle}}{\usepage}}

\pagestyle{special}

\begin{document}

---TOC

\pagestyle{main}

---Body

\pagestyle{special}

---Index
\end{document}

```

9.2. Стандартные классы

Далее демонстрируется, как команды рубрикации стандартных классов могут быть определены.

```

\titleformat{\chapter}[display]
  {\normalfont\huge\bfseries}{\chaptertitlename\ \thechapter}{20pt}{\Huge}
\titleformat{\section}
  {\normalfont\Large\bfseries}{\thesection}{1em}{ }
\titleformat{\subsection}
  {\normalfont\large\bfseries}{\thesubsection}{1em}{ }
\titleformat{\subsubsection}

```

```

    {\normalfont\normalsize\bfseries}{\thesubsubsection}{1em}{ }
\titleformat{\paragraph}[runin]
    {\normalfont\normalsize\bfseries}{\theparagraph}{1em}{ }
\titleformat{\subparagraph}[runin]
    {\normalfont\normalsize\bfseries}{\thesubparagraph}{1em}{ }

\titlespacing*{\chapter}      {0pt}{50pt}{40pt}
\titlespacing*{\section}     {0pt}{3.5ex plus 1ex minus .2ex}{2.3ex plus .2ex}
\titlespacing*{\subsection}  {0pt}{3.25ex plus 1ex minus .2ex}{1.5ex plus .2ex}
\titlespacing*{\subsubsection}{0pt}{3.25ex plus 1ex minus .2ex}{1.5ex plus .2ex}
\titlespacing*{\paragraph}   {0pt}{3.25ex plus 1ex minus .2ex}{1em}
\titlespacing*{\subparagraph}{\parindent}{3.25ex plus 1ex minus .2ex}{1em}

```

9.3. Пример главы

Последний пример показывает, как использовать окружение `picture` в необычных оформлениях заголовков. Даже используя простые инструменты, предлагаемые стандартными командами ЛАТЭХ'а можно создать впечатляющее оформление, или более сложные макеты, например, `rpicture` (пакет `PSTricks`) или включая графику созданную специальными графическими программами.

```

\usepackage[dvips]{color}
\usepackage[rigidchapters]{titlesec}

\DeclareFixedFont{\chapterfont}{T1}{phv}{bx}{n}{11cm}

\newcommand{\chapterformat}[1]{%
  \put(0,190){\parbox[t]{300pt}{\Huge\sffamily\filright#1}}}

\titlespacing{\chapter}{0pt}{0pt}{210pt}
% Most of titles have some depth. The total space is
% a bit larger than the picture box.

\titleformat{\chapter}[block]
  {\begin{picture}(330,200)}
  {\put(450,80){%
    \makebox(0,0)[rb]{%
      \chapterfont\textcolor[named]{SkyBlue}{\thechapter}}}}
  \put(0,230){%
    \makebox(0,0)[lb]{%
      \Huge\sffamily\underline{Chapter \thechapter}}}}
  {0pt}
  {\chapterformat}
  [\end{picture}]

```

(Точные значения зависят от количества текста, класса [заголовка?], `\unitlength`, размера бумаги, и т. д.)