

L^AT_EX, GNU/Linux и русский стиль.

© Е.М. Балдин*



Эта статья была опубликована в майском номере русскоязычного журнала Linux Format (<http://www.linuxformat.ru>) за 2007 год. Статья размещена с разрешения редакции журнала на сайте <http://www.inp.nsk.su/~baldin/> и до конца октября месяца все вопросы с размещением статьи в других местах следует решать с редакцией Linux Format. Затем все права на текст возвращаются ко мне.

Текст, представленный здесь, не является точной копией статьи в журнале. Текущий текст в отличии от журнального варианта корректор не просматривал. Все вопросы по содержанию, а так же замечания и предложения следует задавать мне по электронной почте <mailto:E.M.Baldin@inp.nsk.su>.

Текст на текущий момент является просто *текстом*, а не книгой. Поэтому результирующая доводка в целях улучшения восприятия текста не проводилась.

*e-mail: E.M.Baldin@inp.nsk.su

Эмблемы T_EX и METAFont, созданные Дуайном Бибби, взяты со странички Д.Э. Кнута. Цветной пингвин взят из пакета ps2pdf от Ральфа Найпрашека (Rolf Niepraschk)

Оглавление

9. Справочно-поисковый аппарат издания	1
9.1. Рубрикация и оглавление	2
9.2. Ссылки и примечания	4
9.2.1. Механизм ссылок	4
9.2.2. Подстрочные примечания	5
9.3. Колонтитулы	6
9.4. Библиография	6
9.4.1. ВiVГeX	7
9.5. Алфавитный указатель	11

Справочно-поисковый аппарат издания

На этом же этаже располагалось книгохранилище. По поводу его размеров рассказывали, что в глубине, в полукилометре от входа, идёт вдоль стеллажей неплохое шоссе, оснащённое верстовые столбами.

«Понедельник начинается в субботу»
Аркадий и Борис Стругацкие.

Книги делятся на те, что читаются один раз и те, что многократно перечитываются. Наличие информации о структуре книги повышает ценность любого текста. Отсутствие этой информации — прямой намёк, что после прочтения произведение следует забыть и выбросить.

Справочно-поисковый аппарат издания позволяет читателю облегчить и ускорить поиск имеющихся в книге объектов. В качестве элементов из которых складывается справочно-поисковая система можно упомянуть рубрикации, оглавление, колонтитулы, ссылки, подстрочные примечания, алфавитный указатель и библиографию.

Этот аппарат существует исключительно для читателя и он достаточно трудоёмок при создании. Но сложности не должны пугать истинных энтузиастов в деле создания текстов, так как их преодоление значительно повышают ценность серьёзного произведения.

Пример рубрикации

А. В. Тор

Содержание

1. Раздел	1
1.1. Подраздел	1
1.1.1. Подподраздел	1
Заключение	1
А. Приложение	1

1. Раздел

Основной элемент рубрикации.

1.1. Подраздел

Вспомогательный элемент рубрикации.

1.1.1. Что-то более мелкое чем подраздел

Вспомогательный для вспомогательного. В содержании выводится краткая версия заголовка.

Параграф Важный параграф.

Подпараграф Параграф чуть менее важный.

Раздел, отсутствующий в содержании

Всяко бывает. Иногда и такое нужно.

Заключение

Заключение в отличие от, скажем, раздела 1.1 на странице 1 нумеровать не надо, но в содержании отразить необходимо.

А. Приложение

Рис. 9.1. Пример рубрикации и оглавления

9.1. Рубрикация и оглавление

Нужны ли книге оглавление или содержание? Любой скажет: что за вопрос, конечно, нужны. И не только в книге научной и деловой. В любой.

А.Э. Мильчин «Культура издания»

Для оформления разделов в основном используются команды секционирования `\section`, `\subsection`, `\subsubsection`, `\paragraph` и `\subparagraph`.

Команды перечислены в порядке убывания значимости при рубрикации. Обычно, любой сколько-нибудь сложный текст следует начинать с планирования структуры, то есть создать шаблон, примерно следующего вида:

```
\documentclass[a4paper,12pt]{ncc}
\usepackage[warn]{mathtext}
\usepackage[T2A]{fontenc}
\usepackage[koi8-r]{inputenc}
\usepackage[english,russian]{babel}
\usepackage{indentfirst}
\title{Пример рубрикации}
\author{А. \ , В. ~ Top}
\begin{document}
\maketitle{}
\tableofcontents{}
```

```
\section{Раздел}
\label{sec:section}
Основной элемент рубрикации.
```

```
\subsection{Подраздел}
\label{sec:subsection}
Вспомогательный элемент рубрикации.
```

```
\subsubsection[Подподраздел]{Что-то более мелкое чем подраздел}
\label{sec:subsubsection}
Вспомогательный для вспомогательного. В содержании выводится
краткая версия заголовка.
```

```
\paragraph{Параграф}
\label{sec:paragraph}
Важный параграф.
```

```
\subparagraph{Подпараграф}
\label{sec:subparagraph}
```

Параграф чуть менее важный.

```
\section*{Раздел, отсутствующий в содержании}
```

Всяко бывает. Иногда и такое нужно.

```
\section*{Заключение}
\label{sec:afterwords}
\addcontentsline{toc}{section}{Заключение}
```

Заключение в отличие от, скажем, раздела `\ref{sec:subsection}` на странице `\pageref{sec:subsection}` нумеровать не надо, но в содержание отразить необходимо.

```
\appendix
```

```
\section{Приложение}
\label{appendix}
```

```
\end{document}
```

Результат компиляции кода представлен на рис. 9.1. Кроме самих заголовков разделов, созданных с помощью команд секционирования, в начале создаётся оглавление. За создание оглавления отвечает команда `\tableofcontents`. При каждой компиляции информация о разделах собирается в файле с тем же именем, что и у `tex`-файла, но с расширением `toc`. При следующей компиляции обновлённая информация о разделах используется для создания оглавления.

Уровень до которого информация отображается в оглавлении можно поменять изменив значение переменной `tocdepth`, например, так:

```
\setcounter{tocdepth}{2}
```

В этом случае будет показана информация о разделах вплоть до второго уровня. Раздел типа `\section` соответствует первому уровню секционирования, `\subsection` второму и так далее.

Кроме оглавления также можно вывести список иллюстраций `\listoffigures` и таблиц `\listoftables`. Информация об иллюстрациях и таблицах автоматически собирается в файлах с расширениями `lof` и `lot`.

Для добавления какой-то информации в оглавления в обход команд секционирования можно воспользоваться командой

```
\addcontentsline{toc}{«уровень раздела»}{«строка в оглавлении»}
```

У этой команды три аргумента. Первый аргумент соответствует расширению файла (`toc`, `lof` или `lot`), куда добавляется «строка в оглавлении». Уровень раздела определяется именами команд секционирования, то есть `section`, `subsection` и так да-

лее. Команды \LaTeX при передаче в файлы списков следует защищать командой `\protect`, дабы избежать проблем с «хрупкими» инструкциями.

Команда `\appendix` отделяет приложение от основного текста. После её вызова правила нумерации разделов изменяются. `\appendix` тоже является командой секционирования.

9.2. Ссылки и примечания

Иногда не хочется разбивать канву повествования и в то же время необходимо как-то вставить пояснение. Это можно сделать просто сославшись на какой-то другой фрагмент текста с помощью ссылки или вставить пояснение на этой же странице с помощью подстрочного примечания.

9.2.1. Механизм ссылок

В примере, демонстрирующем работу команд секционирования вслед за каждой такой командой ставилась метка с помощью инструкции `\label`. Метка представляет из себя последовательность ASCII-символов. При компиляции документа информация о имеющихся метках добавляется в файл с расширением `aux`. Для того чтобы извлечь эту информацию, то есть номер раздела (команда `\ref`) или номер страницы (`\pageref`) необходимо пропустить текст через `latex` ещё раз.

Для того чтобы можно было сослаться на внешний документ следует воспользоваться пакетом `xr`. В этом случае в преамбуле необходимо добавить примерно следующие инструкции:

```
\usepackage{xr}
\externaldocument [EXT-]{externaldoc}
```

Это позволяет получить доступ к меткам файла `externaldoc.tex`. Обращение к меткам, как и обычно, осуществляется помощью команд `\ref`/`\pageref`, только перед именем метки добавляется префикс `EXT-`. Можно обойтись и без префикса, так как этот параметр является опциональным, но в этом случае повышается вероятность конфликта из-за одинаковых меток.

Ссылаться можно не только на разделы. Метки внутри нумерованных окружений, типа `equation` (выключенные математические формулы¹) или `theorem` (теоремы), принимают их номер. Это так же касается рисунков (окружение `figure`) и таблиц (окружение `tabular`). В этом случае `\label` должна следовать сразу за командой `\caption`, формирующей подпись к плавающему объекту.

Общего рецепта как делать ссылки на электронные ресурсы нет. Проще всего использовать команду `\url` из одноимённого пакета:

¹Для правильной ссылки на номера формул вместо команды `\ref` следует использовать инструкцию `\eqref`

Моя WWW-страничка находится по адресу
`\url{http://www.inp.nsk.su/~baldin/}`.

Моя WWW-страничка находится по адресу `http://www.inp.nsk.su/~baldin/`.

Ссылка печатается В адресной строке должны отсутствовать символы %, #, ^ и она не должна заканчиваться символом \. Если есть желание уйти и от этих ограничений, то аналогично команде `\verb` инструкцию можно использовать и так: `\url!http://www.адрес.ru!`.

В плане создания и управления гиперссылками также интересен пакет **hyperref** который предоставляет схожую функциональность и позволяет создавать гиперссылки в pdf-документах, но это уже совсем другая история:

```
%загрузка пакет hyperref
\usepackage [ unicode=true ] { hyperref }
```

9.2.2. Подстрочные примечания

Подстрочное примечание формируется с помощью команды `\footnote`. Правила оформления примечаний прописываются в определении класса. Без особых на то причин менять эти правила не стоит. Примечание, если позволяет место, печатается на той же странице где помещена ссылка. Текст и примечание отделяются разделительной линией.

Примечание можно добавлять и внутри окружения **minipage**, но тогда оно печатается внутри окружения:

```
\begin{minipage}{1.0\linewidth}
  Ссылки раз\footnote{Сноска.} и
  два\footnote[26]{Подстрочное примечание.}.
\end{minipage}
```

Ссылки раз^a и два^z.

^aСноска.

^zПодстрочное примечание.

Необязательный параметр `\footnote` позволяет присвоить примечанию значение по выбору пользователя.

Для того чтобы можно было сделать сноску внутри заголовка раздела необходимо защитить инструкцию `\footnote` командой `\protect`:

```
\section{Заголовок\protect\footnote{Подстрочное примечание.}}
```

Команда создания подстрочного примечания является «хрупкой».

Иногда в сложных ситуациях, например, когда требуется сделать подстрочное примечание внутри бокса, для формирования сноски требуется прибегнуть к независимым командам создания ссылки и создания примечания:

```
\footnotemark [ num ]
\footnotetext [ num ] { «сноска» }
```

Необязательный параметр `num` как и в случае `\footnote` позволяет формировать свою нумерацию. Для хранения текущего номера ссылки используется счётчик `footnote`.

9.3. Колонтитулы

Правила формирования колонтитулов целиком зависят от выбранного класса документа. Если же хочется изменить значения по умолчанию, то проще всего выбрать стиль страницы `myheadings` и сформировать колонтитулы:

```
\pagestyle{myheadings}
\markboth{«левый колонтитул»}{«правый колонтитул»}
```

Если печать односторонняя, то достаточно воспользоваться командой `\markright`, которая имеет только один аргумент.

Для полного управления содержимым колонтитулов лучше всего подходит пакет `fancyhdr`. Подробно об этом пакеты было рассказано в разделе 6.3.2.

9.4. Библиография

Книги создаются не в безвоздушном, или бескнижном, пространстве.

А.Э. Мильчин «Культура издания»

Хорошая книга представляет из себя ценность. На даже самая лучшая книга не в состоянии охватить абсолютно все аспекты рассматриваемой в ней темы. Книги существуют в книжном пространстве. Всегда можно найти что-то, на чём книга основывалась, что-то, что развивает основную идею и что-то, что позволяет взглянуть на главную тему с другой стороны. Список литературы книгу только украшает.

Для ссылок на литературу используется команда

```
\cite[«комментарий»]{«список меток»}
```

Метки либо формируются автором самостоятельно внутри окружения `thebibliography`

Полезно почитать книгу `\cite[Роженко]{rozenko-2005}`.

```
\begin{thebibliography}{9}
\bibitem{rozenko-2005}Роженко А.И. Искусство вёрстки в
\LaTeX'e. \newblock — Новосибирск: Изд. ИВМиМГ СО РАН,
2005. 398~с.
\end{thebibliography}
```

Полезно почитать книгу [1, Роженко].

Список литературы

- [1] Роженко А.И. Искусство вёрстки в \LaTeX 'е. — Новосибирск: Изд. ИВМиМГ СО РАН, 2005. 398 с.

Рис. 9.2. Пример библиографической ссылки.

либо используется механизм \bibTeX . Команда `\newblock` позволяет логически разделить разные по смыслу элементы. В качестве обязательный аргумента окружения `\thebibliography` требуется передать текст соответствующий самой широкой метки для выравнивания. То есть, если список литературы содержит меньше 10 записей, то достаточно передать однобуквенную фразу, например, «9», а в случае двузначного числа книг в списке уже потребуется «99» и так далее.

Как и в случае с перекрёстными ссылками информация о списке литературы заносится в файл с расширением `aux`. То есть для правильного его отображения необходимо два прохода \LaTeX .

Для того чтобы можно было использовать кириллицу в метках для цитирования литературы, а именно иметь возможность написать что-то вроде:

```
\cite{Котельников-2004}
...
\bibitem{Котельников-2004}
```

Следует воспользоваться пакетом `citehack`:

```
\usepackage{citehack}
```

Из названия пакета очевидно что это хак со всеми вытекающими. Не следует им злоупотреблять.

9.4.1. \bibTeX

Можно список литературы оформлять руками. Есть какой-никакой стандарт, например, тот-же ГОСТ 7.80-00² или ГОСТ 7.1-84. Берём и просто следуем подробной инструкции. Но, далеко не все издательства подчиняются этому стандарту в котором, например, нет информации как нужно оформлять `www`-ссылки. Да и вообще список сопутствующей литературы это нечто большее, чем просто довесок к статье или книге — это вполне самостоятельный фрагмент информации, который очень полезно уметь представлять по разному.

²ГОСТ на оформление библиографического указателя, принятый в 2000 году. Правила оформления могут нравиться или не нравиться, но это всё-таки хоть какой-то стандарт.

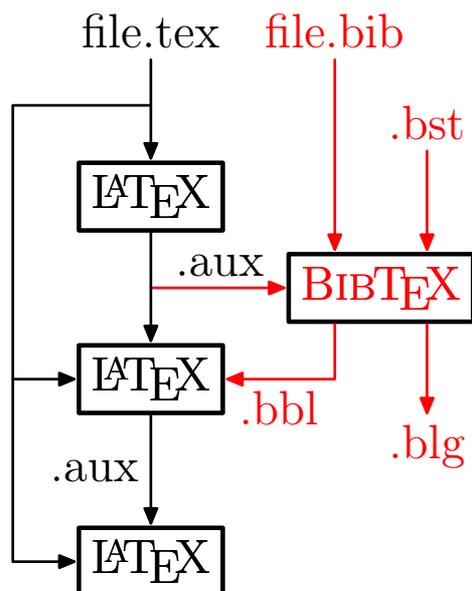


Рис. 9.3. Конвейер $\text{\LaTeX}+\text{\BibTeX}$. `tex` — \LaTeX -исходник, `bib` — библиографическая база, `bst` — стилевой файл для библиографии, `blg` — log-файл \BibTeX , `bbl` — отсортированный список литературы, `aux` — информация о ссылках.

Для решения этой проблемы Орен Поташник разработал программу \BibTeX , которая сама формирует окружение **thebibliography**, получая информацию из текстовой библиографической базы. Структура библиографической базы \BibTeX является довольно распространённым форматом, который использует в том числе и Google Scholar (<http://scholar.google.com/>), не говоря уж о том, что основной архив электронных препринтов <http://xxx.lanl.gov> предоставляет библиографическую информацию исключительно в виде записей \BibTeX .

Из программного обеспечения, позволяющего работать с \BibTeX следует упомянуть встроенный в Emacs пакет RefTeX и JabRef <http://jabref.sourceforge.net/>. Тот, кто не освоил Emacs и кому не нравится Java может поискать программные пакеты gBib и KBib для Gnome и KDE, соответственно. Простой конвертер **bibtex2html** позволяет получить список литературы и html-виде. Естественно, и простое редактирование текстового файла руками так же никто не отменял.

Как правило, библиографическая база в формате \BibTeX хранится в файле с расширением `bib`. Перед тем как с помощью команды `\bibliography` подключить базу к \LaTeX -исходнику нужно выбрать стиль сортировки библиографии:

```

\ bibliographystyle {«стиль»}
\ bibliography {«имя bib-файла»}

```

В \LaTeX есть четыре стандартных стиля для формирования списка литературы:

plain — открытый стиль. Библиографические записи помечаются порядковыми номерами и сортируются в алфавитном порядке. Чтобы правильно отсортировать библиографию на русском языке необходимо вместо **bibtex** воспользо-

Список литературы

- [1] *Гуссенс, М.* Путеводитель по пакету \LaTeX и его расширению $\LaTeX 2\epsilon$: Пер. с англ. / М. Гуссенс, Ф. Миттельбах, А. Самарин. — М.: Мир, 1999. — 606 с.
- [2] *Львовский, С. М.* Набор и вёрстка в системе \LaTeX . / С. М. Львовский. — М.: МЦНМО, 2003. — 448 с. — 3-е изд., испр. и доп.
- [3] *Котельников, И. А.* \LaTeX по русски. / И. А. Котельников, П. З. Чеботаев. — Новосибирск: Сибирский Хронограф, 2004. — 496 с. — 3-е изд., перераб. и доп.
- [4] *Роженко, А. И.* Искусство вёрстки в \LaTeX 'е. / А. И. Роженко; Под ред. А. Алексеева. — Новосибирск: Изд. ИВМиМГ СО РАН, 2005. — 398 с.

Рис. 9.4. Список литературы оформленный с помощью ViTeX . Стиль **gost780u**.

ваться командой **bibtex8** указав с помощью ключа `--csfile` соответствующее правило сортировки³.

unsrt — не сортирующий стиль. В отличии от **plain** порядок представления списка литературы определяется порядком цитирования библиографии в тексте.

alpha — алфавитный стиль. Вместо нумерации библиографии используются имена меток. Литература сортируется по меткам.

abbrv — аббревиатурный стиль. Вместо полных имён авторов, названий месяцев и журналов печатаются сокращения. Сортировка и нумерация соответствует стилю **plain**.

Максим Поляков разработал стили для ViTeX соответствующие ГОСТ 7.80-00 и ГОСТ 7.1-84: **gost780s/gost71s** аналогичные **plain** и **gost780u/gost71u** аналогичные **unsrt**. Описание этих стилей представлено в стандартной документации в виде файлов **gost780.pdf** и **gost71.pdf**. Многие журналы принимающие публикации в \LaTeX имеют свои собственные ViTeX -стили. В стандартной поставке TeXLive 2007 идёт более 200 различных библиографических стилей.

В дополнение к команде цитирования `\cite` в случае подключения библиографической базы можно использовать инструкцию `\nocite`. Команда `\nocite` не создаёт никакой ссылки в тексте, но упомянутая запись отображается в списке литературы.

Сама по себе база состоит из записей вида:

```
@book{Gussens – 1999,
  author = {М. Гуссенс and Ф. Миттельбах and А. Самарин},
  title = {Путеводитель по пакету \LaTeX{} и его
           расширению \LaTeXe: Пер. с англ.},
```

³В стандартной поставке \LaTeX есть правило для сортировки для кодовой страницы `sr866` `sr866rus.csf`. На основе этого файла можно создать правило для другой кодовой страницы.

```

year = {1999},
isbn = {5-03-003325-4},
publisher = {Мир},
address = {М.},
numpages = {606},
language = {russian},
OPTnote = {}
}

```

После знака «коммерческое at» @ идёт тип записи. В фигурных скобках вслед за меткой через запятую перечисляются пары ключ-значение. В_ИТ_ЕХ поддерживает определённый набор типов записей, каждому из которых соответствуют свои обязательные и необязательные поля⁴. Если не заполнено обязательное поле, то при компиляции В_ИТ_ЕХ генерирует ошибку. Имеются следующие стандартные типы записей (В_ИТ_ЕХ не чувствителен к регистру):

- **Article** — статья в журнале. Обязательные поля: author, title, journal, year. Необязательные поля: volume, number, pages, month, note, annote.
- **Book** — книга. Обязательные поля: author или editor, title, publisher, year. Необязательные поля: volume, number, series, address, edition, month, note, annote.
- **Booklet** — брошюра. Обязательное поле: title. Необязательные поля: author, howpublished, address, month, year, note, annote.
- **Conference** или **InProceedings** — статья опубликованная в трудах конференции. Обязательные поля: author, title. Необязательные поля: crossref, booktitle, pages, year, editor, volume, number, series, address, month, organisation, publisher, note, annote.
- **Proceedings** — труды конференции. Обязательные поля: title, year. Необязательные поля: booktitle, editor, volume, number, series, address, month, organisation, publisher, note, annote.
- **InBook** — ссылка на часть книги, то есть на её главу, раздел или просто на определённый набор страниц. Обязательные поля: author или editor, title, chapter, publisher year. Необязательные поля: volume или number, series, type, address, edition, month, pages, note, annote.
- **InCollection** — часть книги со своим заглавием. Обязательные поля: author, title, booktitle. Необязательные поля: crossref, pages, publisher, year, editor, volume или number, series, type, chapter, address, edition, month, note, annote.

⁴Если к названию необязательного поля добавить ОПТ (note→OPTnote), то такие поля игнорируются, даже если присутствуют в записи.

- **Manual** — техническая документация. Обязательное поле: `title`. Необязательные поля: `author`, `organistaion`, `address`, `edition`, `month`, `year`, `note`, `annotate`.
- **PhdThesis** — диссертация. Обязательные поля: `author`, `title`, `school`, `year`. Необязательные поля: `address`, `month`, `note`, `annotate`.
- **MastersThesis** — дипломная работа. Обязательные и необязательные поля копируют **PhdThesis**.
- **TechReport** — отчёт. Обязательные поля: `author`, `title`, `institution`, `year`. Необязательные поля: `type`, `numer`, `address`, `month`, `note`, `annotate`.
- **Unpublished** — неопубликованный авторский текст. Обязательные поля: `author`, `title`, `note`. Необязательные поля: `month`, `year`, `annotate`.
- **Misc** — то, что не подходит для других типов записей. Обязательные поля отсутствуют. Необязательные поля: `author`, `title`, `howpublished`, `month`, `year`, `note`, `annotate`.

Значение полей в большинстве случаев понятно из их названия. Исключением, пожалуй является поле **crossref**, в качестве значения которого можно передать ссылку на другую запись из которой ВІВТѢХ при трансляции возьмёт значения всех недостающих полей записи. Из особенностей следует упомянуть, что авторы в поле **author**, разделяются с помощью союза **and**.

Кроме перечисленных стандартных полей при использовании библиографических ГОСТ-стилей **gost780u/gost71u** и **gost780s/gost71s** можно использовать поля **numpages** — число страниц и **language** — влияет на сортировку при выборе стиля типа **plain**. Допустимые значения для поля **language** — `russian`, `ukrainian` и `english` по умолчанию.

9.5. Алфавитный указатель

Указатель значительно повышает ценность любой книги, если она из тех, что читают не только на сквозь, но и выборочно.

А.Э. Мильчин «Культура издания»

Для создания «полуфабриката» алфавитного указателя в преамбуле помещается инструкция

```
\makeindex
```

Слова, которые нужно поместить в указатель, отмечаются с помощью команды **\index**.

```
Указатель \index{Предметный указатель} ...
```

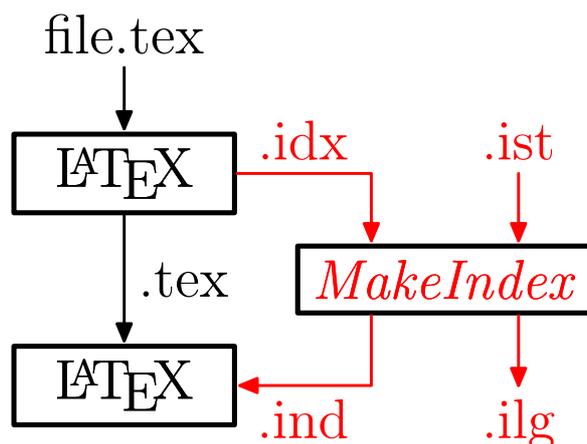


Рис. 9.5. Конвейер $\text{LATEX} + \text{MakeIndex}$. *tex* — LATEX -исходник, *idx* — не отсортированный индекс (полуфабрикат), *ist* — стилевой файл для указателя, *ilg* — log-файл *MakeIndex*, *ind* — отсортированный указатель.

Сама по себе команда `\index` игнорируется, но всё, что в ней отмечается вместе с информацией о положении команды, заносится в файл с расширением *idx*. Имя *idx*-файла по умолчанию соответствует имени основного документа. Полуфабрикатом *idx*-файл является потому, что записи хранятся в нём в не отсортированном виде. С помощью программ сортировки **rumakeindex** получается уже отформатированный правильным образом указатель в файле с расширением *ind*. Этот файл уже можно вставить в документ:

```
\input { «ind-файл» }
```

Это же делает и команда `\printindex` из пакета **makeidx**. В дополнение к подключению индекса команда `\printindex` проверяет существование индексного файла и не даёт LATEX генерировать ошибку в случае его отсутствия. Пакет **makeidx** содержит ещё несколько полезных команд для создания индекса, поэтому его в любом случае имеет смысл загрузить.

Программа **rumakeindex** является простейшим скриптом где с помощью **sed** кириллические буквы из внутреннего представления LATEX переводятся в *ko18*-г и правильным образом сортируются с использованием стандартного механизма **makeindex**. Если необходимо отсортировать индексный файл для включения в текст, использующий другую кодовую страницу или просто не устраивает правила сортировки, то этот скрипт легко переделывается.

Оригинальная программа сортировки индекса **makeindex** (подробная документация представлена в файле *makeindex.dvi*) была написана довольно давно и, естественно, не учитывала национальных особенностей других языков кроме английского и немецкого. К сожалению она оказалась достаточно гибка, чтобы полностью отказаться от неё в пользу другого механизма сортировки уже нормально поддерживающего интернационализацию. Наиболее вероятным претендентом на замену уже долгое время является **xindy** (<http://www.xindy.org/>), который поддержива-

Предметный указатель

MakeIndex, 110

Указатель, 110–111

makeindex, см. rumakeindex

rumakeindex, 110, 111

xindy, 111

Рис. 9.6. Пример готового предметного указателя.

ет множество языков из коробки вплоть до клинговского, но до сих пор отсутствует в основных L^AT_EX-дистрибутивах.

Аргумент команды `\index` может содержать любые символы кроме `!`, `"`, `@` и `|`. Их специальное значение проявляется только внутри команды. Чтобы убрать специальное значение этих символов внутри `\index` необходимо перед ними добавить символ `"` (двойную кавычку). На рис. 9.6 представлен пример простейшего указателя. Ниже будут раскрыты методы его создания.

Для формирования многоуровневых иерархических указателей используется разделитель в виде восклицательного знака `«!»`:

```
% на страницах 110 и 111
\index{Указатель!rumakeindex}
```

Команды `\index` с одинаковыми аргументами группируются в одну запись с полным списком страниц.

Символ вертикальной черты `«|»` используется для отделения видимого аргумента от управляющих знаков. Команда `\see` (перекрёстная ссылка на другую запись), определённая в пакете **makeidx**, в индексе должна идти вслед за этим разделителем:

```
\index{Указатель!makeindex|see{rumakeindex}}
```

Так же с помощью вертикальной черты можно сформировать указатель на диапазон страниц:

```
% на странице 110
\index{Указатель|({
  много текста
% на странице 111
\index{Указатель|)})}
```

Иногда номер страницы нужно как-то выделить. Команды выделения текста также должны идти после вертикальной черты:

```
% Выделяем страницу 110 жирным шрифтом
\index{MakeIndex@\textit{MakeIndex}textbf}
```

Конструкция «ключ»@«запись» используется для правильной сортировки внутри \index. По «ключу» производится сортировка, а «запись» выводится в предметном указателе.

Заключение

Написать любую книгу безумно тяжело. Сделать её полезной почти невозможно. Справочно-поисковый аппарат издания это лишь инструмент на этом пути. Но инструмент неоднократно проверенный грамотным человечеством. Можно путь пройти и без него, но с ним будет гораздо интереснее.