

L^AT_EX, GNU/Linux и русский стиль.

© Е.М. Балдин*



Эта статья была опубликована в октябрьском номере русскоязычного журнала Linux Format (<http://www.linuxformat.ru>) за 2006 год. Статья размещена с разрешения редакции журнала на сайте <http://www.inp.nsk.su/~baldin/> и до марта 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

2. Базовые элементы	1
2.1. «Командная логика»	1
2.2. Логика документа	3
2.2.1. Структура L ^A T _E X-файла	3
2.2.2. Класс документа	4
2.2.3. Стили	5
2.2.4. Тело документа	6
2.3. Логика набора	6
2.3.1. Печатаем текст	6
2.3.2. Пунктуация	7
2.4. Структурная логика	9
2.4.1. Титульный лист	9
2.4.2. Секционирование	10
2.4.3. Перекрёстные ссылки	11
2.4.4. Сложные документы	11

Базовые элементы

Мы говорим на русском языке и пользуемся операционной системой GNU/Linux — подобное на нашей планете не так уж и часто встретишь. Хотите стать ещё более элитарным подмножеством? Используйте ЛАТЭХ ☺!

2.1. «Командная логика»

Щенок должен понять, что быстрое и четкое выполнение команд хозяина всегда вознаграждается лакомством или лаской.

Для набора кода в ЛАТЭХ знание английского языка приходится очень кстати. Основных команд немного и их можно запомнить и так, но для совершенствования английский необходим, хотя бы для чтения документации к пакетам. Названия у команд, как правило, вполне осмыслены, что очень помогает при поиске чего-нибудь необходимого в алфавитном указателе. «Правильный» текстовый редактор тоже не является лишним.

Спецсимволы Не все символы одинаково равноправны. За частью символов в ЛАТЭХ зарезервированы специальные значения.

Это: «\», «\$», «%», «_», «{», «}», «&», «#», «^» и «~». В процессе изложения их роль будет со временем раскрыта.

Чтобы отобразить эти символы при печати необходимы дополнительные усилия. Предыдущий параграф в текстовом редакторе выглядел бы так:

Это: <<\textbackslash>>, <<\\$>>, <<\%>>, <<_>>¹,
<<\{>>, <<\}>>, <<\&>>, <<\#>>, <<\^{}>> и <<\~{}>>.
В процессе изложения их роль будет раскрыта.

¹Можно воспользоваться пакетом **underscore** — в этом случае необходимость экранировать знак «_» в текстовой моде отпадает.

Группировка Группировка осуществляется с помощью фигурных скобок: { группа }. Фигурные скобки при печати не отображаются.

Сложные конструкции, которые имеют открывающую и закрывающую команды (например, окружения) тоже группируют текст.

Построение команды Команды в \LaTeX начинаются с символа «\» (backslash или обратная косая черта) и продолжаются комбинацией, состоящей только из стандартных латинских букв². Команды завершаются пробелом, цифрой или не латинской буквой. Все пробельные символы после команды игнорируются. Для того чтобы пробел после команды не игнорировался, достаточно вставить «пустую группу»: `\command{}`. Например, чтобы лого \LaTeX (команда `\LaTeX`) не слилось со следующим за ним словом следует написать `\LaTeX{}`.

Аргументы Командам \LaTeX могут передаваться внешние аргументы:

```
\command [param 1] [param 2] {param 3} {param 4}
```

В квадратные скобки заключаются не обязательные параметры (`param1` и `param2`), а в фигурные — обязательные (`param3` и `param4`).

Некоторые из команд \LaTeX влияют только на свои аргументы. Например, команда `\textbf{текст}` печатает «текст» жирным шрифтом.

Декларативные команды Часть команд \LaTeX являются своеобразными переключателями режимов.

Область действия декларативной команды может ограничиваться логической группой или единицей структуры печатного документа, например, страницей.

```
%дальнейший текст будет печататься жирным шрифтом
\bfseries
%убрать заголовки и нумерацию для текущей страницы
\thispagestyle{empty}
```

Окружения Сложные конструкции, которые имеют открывающую и закрывающую команды вида `\begin{имя}` и `\end{имя}` называют окружениями. Вместо слова «имя» подставляется название соответствующего окружения

```
\begin{center}
Это строка будет центрирована
\end{center}
```

Окружения могут вкладываться друг в друга как матрёшки, но их область действия не может перекрываться.

²Это верно для стандартной кириллизации. Существуют варианты русификации, например, русификация Шеня, где русский буквы так же могут входить в имена команд.

2.2. Логика документа

Вид документа определяется выбором класса и стилей. Хорошим приёмом является наличие личных готовых шаблонов с уже предопределёнными предпочтениями, которые могут меняться по мере развития документа.

2.2.1. Структура \LaTeX -файла

Текстовый `tex`-файл состоит из двух частей: заголовка или преамбулы и, собственно, самого текста, и выглядит примерно следующим образом:

```
%—начало заголовка—
%выбор класса документа, например article или book
\documentclass{...}
%минимальная кириллизация
\usepackage[koi8-r]{inputenc}
\usepackage[english,russian]{babel}
\usepackage[indentfirst}
%загрузка пакетов по выбору
\usepackage{...}
...
%определение своих команд или переопределение уже существующих
\newcommand{\mycommand}{...}
\renewcommand{\oldcommand}{...}
...
%локальные настройки
...
%—конец заголовка—
\begin{document}

%тело документа

\end{document}
```

Первым делом с помощью инструкции `\documentclass` осуществляется выбор класса документа. Далее загружаются стилевые пакеты.

Для того чтобы можно было набирать русский текст, необходимо с помощью пакета `inputenc` указать кодировку текстового файла, например, `koi8-r`. Далее нужно подключить пакет `babel`, который отвечает за локализацию, в частности за настройку переносов и «национальные особенности» набора. Например, при включении русского языка доопределяется символ номера № (`\No`), символ параграфа § (`\S`) и многое другое. Для формирования отступа или красной строки у первого параграфа, как это принято в России, необходимо загрузить пакет `indentfirst`. По идее это должно относиться к «национальным особенностям», но в `babel` по умолчанию не подгружается.

К вопросу у переносах Пожалуй единственное, что возможно потребует настройки в свежесталновленном дистрибутиве ЛАТЭХ, это включение переносов. Этой проблемы заведомо не возникает при установке ТЭХ Live и в большинстве современных дистрибутивах GNU/Linux, но всякое бывает.

Для установки переносов можно воспользоваться утилитой `texconfig`³. После запуска программы следует выбрать меню `HYPHENATION`, а затем меню `latex`. Далее будет предложено отредактировать⁴ файл переносов `language.dat`. Обычно достаточно таких настроек:

```
english hyphen.tex
russian ruhyphen.tex
```

Всё остальное по желанию можно закомментировать. За переносы отвечает пакет **babel**. По умолчанию, когда включаются переносы для определённого языка, все остальные правила переносов отключаются. Но в случае английского и русского языков это можно обойти, воспользовавшись русско-английской таблицей переносов:

```
ruseng ruenhyph.tex
=russian
=english
```

Следует понимать, что подобная настройка с точки зрения философии ЛАТЭХ не совсем корректна. Для гарантированно одинакового результата компиляции не зависимо от платформы лучше поступиться с некоторыми удобствами.

2.2.2. Класс документа

С помощью обязательной инструкции `\documentclass [опции] {класс}` можно задать к какому классу будет относиться рабочий текст.

Класс документа следует выбирать в зависимости от того, что должен из себя представлять текст. Для начала можно остановиться на стандартном классе **article**. Этот класс разработан специально для статей и небольших отчётов. Для отчётов побольше можно использовать класс **report**, а для книг класс **book**.

Перечисленные стандартные классы сложились очень давно и многие производные классы документов основаны на них. Как следствие стандартные базовые классы абсолютно статичны. Поэтому для специализированных вещей используются свои классы. Например, для научных статей популярны различные модификации **revtex4**. Вячеслав Фёдоров разработал класс **eskd** (стандарт ЕСКД), который

³`texconfig` — это простенькое dialog-подобное консольное приложение. Некоторые настройки могут потребовать привилегии суперпользователя. После изменения настроек через `texconfig` автоматически регенерируются необходимые форматные файлы. В противном случае может потребоваться сделать это в ручную, например, с помощью инструкции вида: `texconfig init`.

⁴Редактор можно определить с помощью переменной окружения `$EDITOR`. Если переменная не определена, то вызывается редактор `vi`. В случае отсутствия опыта работы с `vi` следует выйти из него с помощью последовательности `:q` и настроить переменную окружения.

можно найти на CTAN или в стандартной поставке T_EX Live. Тех, кого «напрягают» большие поля в стандартных L^AT_EX-классах, могут обратить внимание на набор классов KOMA-script (**scrartcl**, **scrreprt** и **scrbook** вместо **article**, **report** и **book**, соответственно).

По началу в выбранном классе ничего менять не следует. То, что кажется с непривычки не удобным, на самом деле может улучшать восприятие от печатной копии. Например, относительно узкая ширина текста в стандартных классах (следствие больших полей) позволяет при прочтении охватывать взглядом всю строку целиком, что увеличивает скорость чтения.

Обычно, разумные модификации можно выбирать с помощью передачи параметров при выборе класса, например, так:

```
\documentclass[a4paper,12pt,oneside]{scrbook}
```

a4paper — размер листа бумаги (можно выбрать другой стандарт, например, **a5paper**), **12pt** — базовый размер шрифта (в стандартных классах доступны размеры в **10pt** и **11pt**), а **oneside** — односторонняя печать (удобнее при просмотре электронной версии).

В заключении хотелось бы отметить набор классов *NC* активно разрабатываемых А.И. Роженко. Класс **ncc** автором позиционируется как «русскоязычная статья». Класс можно взять на CTAN и он присутствует в стандартной поставке T_EX Live.

2.2.3. Стили

Стилевой файл (**.sty**) или пакет представляет из себя набор макросов и определений, созданных для решения какой-то определённой задачи. Для подключения стилового файла используется команда `\usepackage[опции]{стиль}`.

Основное отличие классов от пакетов, что на документ может быть ровно один класс и сколько угодно стилиевых пакетов. Фактически на любую задачу в L^AT_EX находится ответ в виде соответствующего пакета. В стандартной поставке T_EX Live присутствует свыше двух тысяч **.sty**-файлов, кроме того ничего не мешает создать свой, заточенный под свои локальные проблемы.

К вопросу о кириллизации Чтобы кириллизировать L^AT_EX необходимы шрифты. Благодаря Ольге Лапко на свете есть шрифты семейства **lh**, которые отлично согласуются с базовыми шрифтами Computer Modern. Мало иметь просто кириллические буквы — надо чтобы их начертания соответствовали и другим шрифтам, в том числе и математическим. В 2001 году Владимир Волович проделал огромную работу по переводу METAFONT-шрифтов в формат TrueType, что теперь позволяет создавать не только хорошие печатные копии, но и вполне качественные электронные pdf-версии документов.

За перевод из кодировки во внутреннюю кодировку L^AT_EX отвечает пакет **inputenc**. В качестве опции при загрузке с ним передаётся текущая восьмибитная кодовая

страница документа. Для кириллицы могут оказаться интересны следующие варианты: `koi8-r`, `koi8-u`, `cp866`, `cp1251` и `8859-5`. Собственно говоря, всё. Единственное неудобство, которое возникает из-за этого, заключается в том, что сообщения об ошибке `LaTeX` выдаёт в своей внутренней T2A кодировке⁵. Для исправления этого неудобства можно воспользоваться простейшим фильтром. Для начало его надо собрать:

```
> locate t2filter.c
{TEXMF}/texmf-dist/doc/generic/t2/etc/t2filter.c
> cd {TEXMF}/texmf-dist/doc/generic/t2/etc/
> gcc -Wall -O2 -s -o ~/bin/t2filter t2filter.c
> latex {файл}.tex | t2filter
```

2.2.4. Тело документа

Всё, что заключено внутри окружения `document`, является телом документа. Если у вас есть какие-то куски текста, которые печатать не хочется, а выкинуть жалко, то их достаточно вынести в конец за инструкцию `\end{document}`.

2.3. Логика набора

Объявление в газете: Ищу работу машинистки.
Печатаю со скоростью 4000 знаков в минуту.
Правда, такая белиберда получается!

Мало открыть файл в текстовом редакторе и начать набирать. Нажимать на клавиши надо осмысленно.

2.3.1. Печатаем текст

При наборе книги/статьи/заметки основное вовсе не команды, а сам текст. Правила очень просты.

Комментарии Всё что следует за знаком «`%`» включительно является комментарием.

Большие закомментированные сегменты мешают работать с основным текстом, и поэтому их следует исключать из рабочего файла. Но при желании можно можно воспользоваться окружением `comment` из пакета `verbatim`.

⁵Расположение букв похоже на расположение букв в `cp1251`, но полностью не совпадает — чистая случайность.

Разделение слов Пробельные символы используются в \LaTeX для разделения слов. Пробелы в начале строки игнорируются. Символ перевода строки так же воспринимается как пробел. Если в конце строки сразу за последним словом вставить знак комментария:

```
экранировка перевода стр%
оки
```

то разделения слов не происходит. Иногда этот приём может оказаться полезным.

Разделение абзацев Для того чтобы начать следующий абзац необходимо оставить пустую строку:

```
текущий абзац закончился
```

```
следующий абзац начался
```

Число пустых строк между абзацами не имеет значения.

2.3.2. Пунктуация

Напечатанный текст обезличивается. Нет эмоций — только буквы. Единственное что остаётся — это знаки пунктуации и, возможно, смайлики ☺.

Запятую, точку, точку с запятой, двоеточие, многоточие, скобки, кавычки, восклицательный и вопросительные знаки следует «прижимать» к словам. Не надо оставлять пробелов, а то \LaTeX «подумает», что так и надо.

Пробелы Расстояние между словами \LaTeX выбирает по своему усмотрению для максимально равномерного заполнения страницы. Но иногда необходимо сделать указать размер пробела руками:

$\tilde{\square}$ — неразрывный пробел, т. е. по этому пробелу не производится перенос предложения на другую строку,

$\square,$ — маленький нерастяжимый пробел,

$\square\lrcorner$ — нормальный нерастяжимый пробел.

В основном, указывать размеры пробелов надо в случае набора каких-либо сокращений, например, так следует набирать ФИО:
 $\Phi.\backslash, A.\tilde{\square}$ Милия "—— негоже <<отрывать>> ИЮ от $\Phi\backslash@$. Ещё примеры:
 $\Gamma.\backslash, e.\tilde{\square}$, $\Gamma.\tilde{\square}$ Новосибирск, $\text{рис.}\tilde{\square}1$ и $\tilde{\square}\Gamma.\backslash, д.\backslash$ и $\tilde{\square}\Gamma.\backslash, п.$

\LaTeX считает, что после точки предложение заканчивается, если эта точка стоит не после заглавной буквы. Растяжимость пробелов между предложениями и между словами существенно разная. Поэтому если точка случается в середине предложения, то после неё следует явно вставить пробел « \lrcorner » или неразрывный пробел « $\tilde{\square}$ ».

Может случиться, что точка следует сразу за заглавной буквой и означает именно конец предложения (как в примере происходит с буквой Φ). Для этого перед такой точкой следует добавить коррекцию в виде команды «\@».

Дефисы, минусы и тире В издательских системах, основанных на \TeX 'е различают дефис «-» (hyphen), короткое тире «-» (en-dash), длинное тире «—» (em-dash) и знак минуса «-».

Чтобы получить на печати дефис, короткое или длинное тире, надо набрать один, два или три знака «-», соответственно.

При подключении пакета **babel** с опцией **russian** появляются дополнительные команды позволяющие более строго следовать русским печатным традициям.

Дефис используют в составных словах (кто-то, где-нибудь), короткое тире рекомендуется для указания диапазона чисел (10--15, 2001--2006), длинное тире означает обычное тире (\LaTeX "---- это круто), минус может существовать только в формулах ($a-b=c$).

Пакет `\textbf{babel}` вводит дополнительные команды для написания тире. Для составных/двойных фамилий следует использовать конструкцию "—~, например, уравнение Клайперона—~Менделеева, композитор Римский—~Корсаков. Чтобы длинное тире не отрывалось от предыдущего слова и вокруг него создавались правильные пробелы вместо — следует употреблять "----, т.\,е. к трём тире надо добавить двойную кавычку. Прямая речь должна начинаться с команды "—*:

"—* Я сказал.

Правила могут показаться немного запутанными, но к ним быстро привыкаешь, и они того стоят.

Переносы В большинстве случаев \LaTeX грамотно переносит слова. Но в случае сложных слов, которые пишутся через дефис, перенос происходит только по дефису. Аналогично проблемы возникают когда слово частично состоит из английских букв, а частично из кириллицы.

Прямо в тексте перенос можно указать с помощью команды \-, например: дель\—та-функ\—ция, \TeXно\—ло\—гия.

При наличии русского языка в `\textbf{babel}` вместо дефиса в сложном слове можно поставить команду "=", например, дельта"=функция. В этом случае переносы будут сделаны корректно без подсказки.

Для часто упоминаемых слов можно задать шаблон переноса с помощью команды `\hyphenation`{образ—цы пе—ре—но—са дель—та=—функ—ция}. Обычно, образцы переноса лучше определять в заголовке документа. Следует понимать, что образцы автоматически не склоняются, поэтому надо предусмотреть всевозможные варианты окончаний.

С помощью команды `\hyphenation` можно запретить перенос слова в нежелательных местах, просто не указав место разрыва. В тексте запрет переноса можно оформить с помощью инструкции `\mbox`{нет переноса}.

Многоточие Многоточие печатается с помощью команды `\ldots`. Если многоточие идёт после точки, то необходимо вставить неразрывный пробел `~`.

Ударение В русском языке длительность ударного гласного примерно в 1.5–2 раза длиннее безударного. Если ударение поставить не в том месте, то слово будет звучать совсем по другому.

В корне `\textbf`{зар—} "—— `\textbf`{зор—} под ударением пишется гласная в соответствии с произношением, без ударения "—— `\textbf`{a}.

`\emph`{Исключения:} зор \ 'янка, озар \ 'ять.

„Лапки“ и «Ёлочки». В пакете `babel` кроме всего прочего определены традиционные русские кавычки.

Если в начале или в конце текста встречаются внутренние и внешние кавычки, то они должны различаться между собой рисунком.

Он сказал: <<А пойду—ка я и подпишусь на „Linux Format“>>.

2.4. Структурная логика

Л^AT_EX ориентирован на логическую разметку документа. Можно конечно «сказать», что данный кусок текста следует напечатать размером 20 пунктов, выровнять по левому краю и сделать отступ после него в два интервала, но проще указать, что это заголовок раздела.

2.4.1. Титульный лист

Создания титульного листа это отдельная задача в которой визуальная составляющая обычно превалирует над структурной. В этом случае следует воспользоваться окружением `titlepage`. При инициализации этого окружения создаётся чистая

страница, которой присваивается номер один, а содержание этой странице полностью определяется фантазией автора. Но, в любом случае, это следует делать после написания самого текста. Обычно, достаточно стандартного заголовка:

```
\title{\LaTeX, Unix и русский стиль}
\author{Е. \,М.~Балдин\thanks{e-mail: E.M.Baldin@inp.nsk.su}}
\date{2006}
\maketitle
```

Команда `\maketitle` создаёт стандартный титульный заголовок, используя информацию о названии документа (`\title`), авторе (`\author`) и даты написания текста (`\date`). Команда (`\thanks`) правильным образом позволяет оформить подстрочное примечание на титульной странице. Если авторов более чем один, то их можно перечислять разделяя командой (`\and`) — в этом случае список авторов печатается в виде таблицы.

В статьях (производные от класса `article`) вслед за заголовком следует обязательная аннотация, которая оформляется с помощью окружения `abstract`.

2.4.2. Секционирование

Часто бывает полезно сразу за титульной страницы вывести оглавление с помощью команды `\tableofcontents`. Но для этого в тексте должно присутствовать логическое разбиение на разделы.

```
\subsection{Секционирование}
\label{sec:base:sec}
```

Часто бывает ...

Команды секционирования образуют строгую иерархию. Самыми старшими по «званию» являются разделы `\part{Часть}` и `\chapter{Глава}`. Это большие куски текста и, соответственно, их применение обосновано только в книгах, поэтому они не определены в классах производных от `article` и `report`, зато определены в классе `book`.

Далее по старшинству следуют:

```
\section{Раздел}
\label{ex:section}
```

```
\subsection{Подраздел}
\label{ex:subsection}
```

```
\subsubsection[Подподраздел]{Что-то более мелкое чем подраздел}
\label{ex:subsubsection}
```

```
\paragraph{Параграф}
```

```
\label{ex:paragraph}
```

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

Если воспользоваться необязательным параметром команды секционирования, то он замещает основной заголовок при печати оглавления и создания колонтитулов.

Команды секционирования печатают заголовок необходимым шрифтом и нумеруют раздел. Если нет желания, что-бы название раздела попало в оглавления и нумерация без надобности, то к команде секционирования следует добавить символ «*», например, `\section*{Приложение}`.

2.4.3. Перекрёстные ссылки

Одной из основных причин по которой ЛАТ_EX вытеснил обычный Т_EX из текстовых редакторов Т_EXников является механизм нумерации и создания ссылок.

Чтобы сослаться на раздел в нём необходимо оставить метку `\label{метка}`. А затем можно на этот раздел сослаться:

```
В разделе \ref{ex:section} на странице \pageref{ex:section} ...
```

Когда ссылки идут через метку, то номер раздела и номер страницы определяется ЛАТ_EX автоматически. Причём автоматическая нумерация свойственна не только командам секционирования точно так же можно сослаться на формулы, таблицы, картинки и листинги программ. Для этого необходимо оставить метку `\label` в соответствующем окружении.

2.4.4. Сложные документы

Всё можно хранить в одном файле — это ничему не противоречит. Более того само понятие файл для пользователя не так уж и необходимо. Но уж если файл есть, то почему бы не разделить большой текст на несколько частично независимых кусков.

С помощью команды `\input{имя файла}` можно вставлять другой tex-файл в документ. ЛАТ_EX просто добавляет содержимое по месту команды, считывая файл либо до конца, либо до первой встретившейся инструкции `\endinput`. Совершенно не важно в какой части документа встречается `\input`. Бывает довольно удобно вынести преамбулу в отдельный файл. В имени файла можно опустить расширение `.tex`.

Для включения текста можно применить другой способ:

```
\input{preheader}
\includeonly{
% intro ,
  base ,
% presentation
```

```
}  
\begin{document}  
%введение  
\include{intro}  
%базовые команды  
\include{base}  
%Презентация  
\include{presentation}  
\end{document}
```

Декларация `\include` позволяет включить только `tex`-файл (при написании имени расширение `.tex` опускается). В преамбуле с помощью команды `\includeonly` можно перечислить какие части надо подключить при текущей сборке. При этом сохраняется правильная нумерация страниц и можно сослаться на *не* включённые в эту сборку разделы. Довольно актуально в случае больших текстов в процессе их создания, так как значительно ускоряет компиляцию.

За букровку «ё» замолвите слово

Пара слов в поддержку буквы «ё». Эту букву незаслуженно забывают при наборе текстов. Более того, некоторые деятели ратуют за её полное упразднение. Однако, коль уж Вам довелось узнать русский язык, то говорить и писать на нём следует правильно. Наличие буквы «ё» в тексте значительно облегчает процесс чтения. Особенно это актуально при быстром чтении.

Поставь букву «ё» на её место!

Для проверки правописания следует использовать словарь Александра Лебедева. Этот словарь построен на основе словаря русского языка для `ispell`, первоначально составленного Нилом Далтоном (Neal Dalton) в 1992 г. После тщательной проверки и исправления примерно 4000 ошибок в словаре Нила Далтона, в словарь были добавлены отсутствовавшие в нём правила образования форм существительных, прилагательных, причастий, наречий, изменены правила формирования окончаний глаголов, так что `affix`-файл можно считать переписанным заново. Одновременно в словарь было добавлено большое число слов.

Отличительной чертой данного словаря является то, что в него включена полноценная поддержка буквы «ё». В современных дистрибутивах GNU/Linux словарь Александра Лебедева является основным русским словарём для `ispell`. К сожалению в подавляющем большинстве случаев словарь «собирается» без поддержки «ё». Мантейнерами пакета ошибочно предполагается, что «е» и «ё» это одинаковые буквы. При проверке правильными являются такие слова как «ежик», «елка» и тому подобное. Правильный выход: убедить мантейнера/самому стать мантейнером пакета. Не правильный, но гораздо более простой: локально пересобрать пакет с полноценной поддержкой «ё».

Словарь постоянно совершенствуется, дополняется и корректируется. Последнюю версию словаря можно найти на авторской страничке Александра Лебедева: <http://semiconductors.phys.msu.su/~swan/orthography.html>

Врезка: T_EX-лого

Когда Д.Э.Кнут создавал T_EX — он много думал. Причём думал не только об алгоритмах и кодировке. В частности он нашёл время подумать о том, как назвать своё произведение. T_EX читается как «тех». Последняя буква вовсе не английская буква «икс», а греческая «хи». Так же он продумал и правила изображения этого названия. С тех пор в T_EX-сообществе возникла мода на создание T_EX-лого.

T _E X	<code>\TeX</code>
L ^A T _E X	<code>\LaTeX</code> или <code>\LATEX</code>
L ^A T _E X 2 _ε	<code>\LaTeXe</code>
A _M S-T _E X	<code>\AMSTeX</code> или <code>\AmSTeX</code>
МЕТAFONT	<code>\METAFONT</code> или <code>\MF</code>
B _I B T _E X	<code>\BIBTeX</code> или <code>\BibTeX</code>
N _C C	<code>\NCC</code>

Таблица 2.1. Распространённые T_EX-лого

Команда `\NCC` определена в пакете **ncclatex**. Остальные команды заведомо определены в пакете **texnames**.