

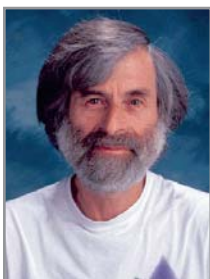
TeXнология текста

TeX/LaTeX

Для верстки технических или математических текстов обычные программы не очень-то удобны, поэтому студенты и инженеры уже давно отдают предпочтение средствам, предназначенным специально для этих целей. Самое известное из них — система TeX/LaTeX.



◀ **Рональд Кнут** — американский математик и программист, создатель TeX



◀ **Лесли Лэмпорт** — разработчик системы LaTeX

Вам когда-нибудь приходилось набирать на компьютере тексты с математическими формулами? Наверняка приходилось — курсовые, дипломы, статьи... Согласитесь, нелегко было заставить текстовый процессор сначала правильно расположить все компоненты на экране, а потом еще и красиво вывести формулу на печать. А сколько ресурсов требуется тому же, например, Microsoft Equation? В свое время мой старенький 486-й с 16 Мбайт памяти так и не справился с отчетом по теории вероятности. Апгрейд процессора и наращивание памяти решили часть проблемы — стало возможно набирать практически любые формулы и вставлять их в текст (естественно, после долгих попыток понять, где и что в Equation находится), но как эти формулы выглядели в тексте! А на печати — так еще хуже. Разумеется, есть еще и другие редак-

торы, например MathType. Но они стоят денег, как, в принципе, и апгрейд.

Тогда-то и встал вопрос: есть ли какое-то решение проблемы, позволяющее верстать качественные математические тексты, одинаково хорошо отображаемые и на экране, и на бумаге? Конечно же, хотелось иметь систему бесплатную и платформонезависимую.

Решение пришло позже, когда я всерьез обратилась к Linux. Тогда-то и оказалось, что существует система, отвечающая всем вышеперечисленным требованиям, причем создана она была задолго до появления не только Equation, но и самой Windows. Называется эта система TeX (читается как «тех»).

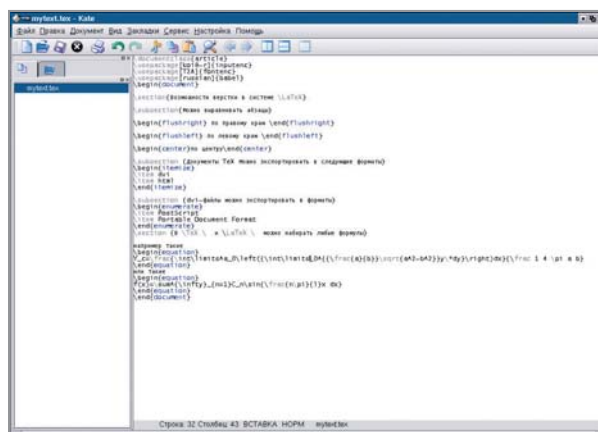
В конце 70-х годов известный программист и математик Дональд Кнут, столкнувшись с проблемой создания сложных математических текстов, придумал систему, а точнее язык для верстки »

» текстов с формулами, который назвал TeX, объясняя название сокращением от греческого слова — предшественника термина «*technologia*». Таким образом, последняя буква в названии TeX — это не заглавная «икс», а греческая буква «хи».

TeX не является готовой системой для подготовки и верстки текстов, это лишь набор правил, язык для разметки документов плюс компилятор, способный транслировать размеченные документы, то есть отображать их в готовом, сверстанном виде.

TeX довольно сложен для изучения и работы, кроме того, он в большей степени направлен на физическую структуру документа, нежели на логическую. Эта ситуация во многом исправилась, когда Лесли Лэмпорт разработал надстройку над TeX, которую назвал LaTeX. LaTeX ориентирована прежде всего на создание логической структуры документа и включает в себя большой набор макросов TeX, предназначенных для создания оформления документа. При этом, в отличие от TeX, LaTeX содержит целую систему классов документов, что позволяет применять различные стили оформления к одной логической структуре.

На базе TeX/LaTeX впоследствии было создано множество различных систем верстки, работающих практически на всех платформах и под любыми операционными системами. Соответственно, создаваемые в этих системах документы легко переносимы, скажем, из Windows в Linux, или наоборот. Для Windows наи-



◀ Пример размеченного TeX-документа в редакторе Kate, входящем в состав графической оболочки Linux KDE

более известной реализацией TeX/LaTeX является система MikTeX, написанная немецким программистом Кристианом Шенком. MikTeX доступна по адресу www.miktex.org.

Что касается систем верстки для Linux, то, на мой взгляд, в этой ОС TeX/LaTeX получила большую известность, чем в любой другой системе. Во многом это связано с тем, что наиболее популярная реализация LaTeX для Linux — teTeX — доступна практически в любом дистрибутиве Linux. Текущая версия LaTeX называется LaTeX2ε, причем «ε» в оригинале означает греческое «эпсилон».

Основные принципы работы TeX/LaTeX

Работа с логической структурой документа в LaTeX облегчается за счет использования стилевых файлов, определяющих способы отображения каждого элемента документа определенного класса. Принцип разделения логичес-

кой структуры и оформления очень похож на идеологию, заложенную в XML: основной файл содержит данные и разметку, определяющую отношения элементов, а к нему может применяться файл с описанием правил отображения элементов — стилевая таблица. Разница лишь в том, что XML-документы интерпретируются соответствующими процессорами, а документы TeX транслируются в двоичные файлы.

Исходный файл документа в LaTeX — это текст с разметкой, описывающей структуру документа. Набирать такой файл можно в любом текстовом редакторе. Если уж мы говорим о Linux, то это может быть KWrite, vim, Emacs — все что угодно. Я, например, пользуюсь редактором Kate из среды KDE. Он обеспечивает подсветку синтаксиса многих текстов, в том числе и документов TeX. Расширение у файлов LaTeX, как правило, TEX.

После того как текст набран, в нем нужно правильно расставить все нужные команды разметки. LaTeX не принадлежит к числу WYSIWYG-приложений. А потому вся обработка текста в нем осуществляется пакетно и в консольном режиме. После транслирования исходного файла мы получаем двоичный файл с расширением .DVI, что означает «device independent» — независимый от платформы. Файлы эти можно распечатывать, просматривать, переносить с платформы на платформу, а также экспортировать в популярные форматы PostScript (PS) и Portable Document Format (PDF).

Но давайте обо всем по порядку. Итак, сначала наберем в любом удобном редакторе текст и попробуем его разметить. Но прежде чем размечать, проверим два основных момента:

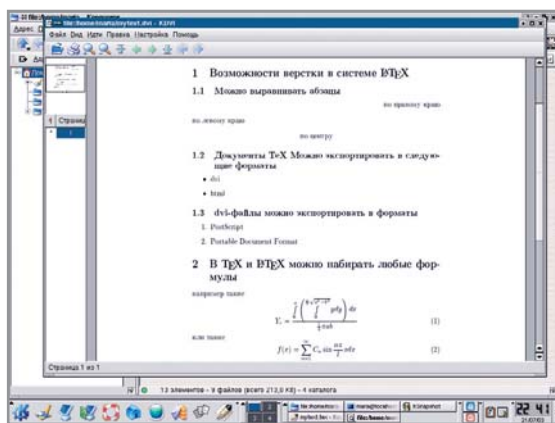
Дополнительная информация

TeX/LaTeX во Всемирной сети

Для тех, кто заинтересовался TeX/LaTeX и хочет узнать о принципах работы в них более подробно, приведем небольшой список наиболее информативных и полезных интернет-ресурсов. Во-первых, вы можете обратиться на сервер ассоциации пользователей кириллического TeX — CyrTUG (www.cemi.rssi.ru/cyrtug). Здесь представлен обширный список ссылок на русскоязычные и иностранные источники по TeX и русифицированным дистрибутивам TeX/LaTeX. Также много материалов по TeX/LaTeX находится на сервере МГУ — <http://tex.msu.ru>.

Огромный список ресурсов, включающий в себя всевозможные реализации TeX для различных операционных систем, утилиты, шрифты, учебные пособия и многое другое, находится на сайте TeX Users Group по адресу www.tug.org/interest.html. Ну и, наконец, вы всегда можете обратиться к первоисточнику — к книге Дональда Кнута «The TeXbook». Оригинальную версию этого издания можно найти по адресу www.ctan.org/tex-archive/systems/knuth/tex. Печатное издание в русском переводе называется «Все про TeX».

Так выглядит скомпилированный DVI-файл в популярном DVI-вьювере Kdvi, который также входит в состав многих дистрибутивов Linux



- » в тексте не должно быть переносов;
- » абзацы должны отделяться друг от друга пустой строкой (одной или несколькими — это не принципиально).

Теперь можно расставлять в тексте команды. Команды разметки в TeX начинаются с символа «\», например \TeX генерирует в тексте эмблему TeX, а \LaTeX, соответственно, эмблему LaTeX. При этом следует учитывать, что регистр символов в командах различается.

Команды TeX делятся на две группы. К первой группе относятся те, которые начинаются с символа «\» и продолжают до следующего небуквенного символа, то есть команды, состоящие из обратного слэша и слова. К командам второго типа относятся те инструкции, в которых после слэша следует единственный небуквенный символ. К специальным символам, используемым в командах второго типа, относятся, например, символы {, }, \$, &, _, %, #. Кстати, символ % без предваряющего обратного слэша означает комментарий, то есть все следующие за ним символы до конца строки игнорируются системой при компиляции текста.

Команды могут иметь фиксированное количество аргументов, каждый из которых задается в фигурных скобках. Некоторые команды могут иметь опции, задаваемые в квадратных скобках.

Подробнее о командах и их назначении можно узнать в любом пособии по TeX, например в книге С. М. Львовского «Набор и верстка в системе LaTeX». Электронный вариант этой книги доступен в Интернете по адресу: http://linux.yaroslavl.ru/docs/conf/format_docs/lang2.pdf.

А теперь немного о логической структуре текста. Прежде всего, необходимо

сказать, что сам документ состоит из двух частей — преамбулы (preamble) и непосредственно самого тела документа (body).

Начинается любой документ TeX с команды `/documentclass{класс_документа}`. Здесь же начинается и преамбула. В LaTeX определены следующие классы документов:

- » `book` — книга, документ, в котором предусмотрена печать на двух сторонах листа, колонтитулы (правые и левые) и т. д.;
- » `article` — класс, предназначенный для оформления небольших документов, статей;
- » `report` — предназначен для больших статей, разделенных на главы и подглавы;
- » `procs` — текст печатается в две колонки с маленькими полями страницы;
- » `letter` — этот класс используется для оформления деловых писем в соответствии с американскими стандартами.

После того как определен класс документа, можно указать дополнительные параметры его обработки, например использование при обработке специальных пакетов. LaTeX не поддерживает русский язык, и для того чтобы его включить, нужно добавить в преамбулу три команды:

```
\usepackage{cp1251}{inputenc} % включение пакета с кодировкой cp1251
\usepackage[T2A]{fontenc} % включение внутренней кодировки документа
\usepackage[russian]{babel} % поддержка русского языка
```

Теперь можно перейти непосредственно к телу документа. Начинается оно с команды `/begin{document}`, а заканчива-

ется, как и сам документ, командой `/end{document}`.

Вообще, команды `/begin{имя}` и `/end{имя}` задают в документе так называемое окружение (environment). Окружения предназначены для управления внешним видом содержащегося в них текста. Они определяют такие параметры, как выравнивание, табуляция текста, а также задают оформление текста в виде списка, таблицы и т. д.

Окружения могут быть вложенными друг в друга; содержащийся в них текст может включать в себя другие команды и комментарии.

В качестве примера разметим несколько абзацев текста, например, так:

```
\documentclass{article}
\usepackage{koi8-r}{inputenc}
\usepackage[T2A]{fontenc}
\usepackage[russian]{babel}
\begin{document}

\section{Возможности верстки в системе LaTeX}
\subsection{Можно выравнивать абзацы}
\begin{flushright} по правому краю
\end{flushright}
\begin{flushleft} по левому краю
\end{flushleft}
\begin{center}по центру\end{center}
\subsection{Документы TeX можно экспортировать в следующие форматы}
\begin{itemize}%нумерованный список
\item dvi
\item html
\end{itemize}

\subsection{DVI-файлы можно экспортировать в форматы}
\begin{enumerate} % нумерованный список
\item PostScript
\item Portable Document Format
\end{enumerate}
```

Указывая в тексте параметры выравнивания строк, нужно иметь в виду, что TeX, компилируя исходные файлы, ориентируется не на то, как символ должен выглядеть, а на то, сколько места он занимает в строке. В TeX существует понятие «клея», заполняющего промежутки между символами. «Клей» может растя-

» гиваться и сжиматься, позволяя расставлять символы в строке так, чтобы создавалось впечатление хорошо набранной страницы. При этом, если существуют некие ограничения на параметры растяжения и сжатия «клея», при которых невозможно растянуть строку и сделать ровный правый край, то TeX этот край выравнивать не будет. Но в это же время пробелы между символами останутся пропорциональными, то есть строка не будет растягиваться за счет огромных пропусков между словами, как это бывает, скажем, в Microsoft Word.

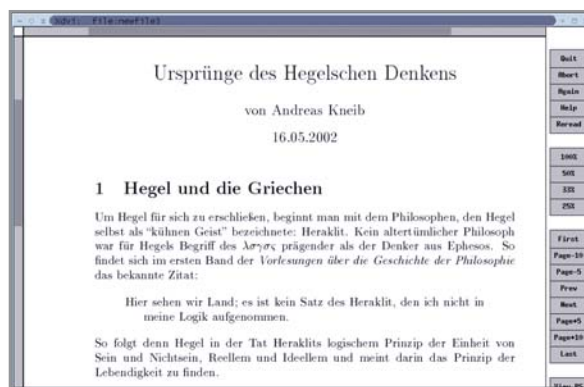
Теперь обратимся к тому, ради чего, собственно, TeX и был задуман — к формулам. Для задания в тексте формул существует специальное окружение — `equation`. Задается оно, соответственно, командами `/begin{equation}` и `/end{equation}`. Однако можно набирать формулы и вне окружения `equation`. В этом случае формула должна отделяться от текста одним значком `$`, если она включена в строку текста (формулой в данном случае считается любой математический символ); или двумя символами `$$`, если формула должны быть вынесена за текст. Попробуем добавить в документ пару формул:

```
\begin{equation}
Y_c = \frac{\int \lim_{a \rightarrow 0} \left( \int \lim_{b \rightarrow 0} \left\{ \frac{a}{b} \right\} \sqrt{a^2 - b^2} \right) dy}{dx} \frac{1}{4} \pi a b
\end{equation}

\begin{equation}
f(x) = \sum_{n=1}^{\infty} C_n \sin \left\{ \frac{n}{\pi} \right\} x dx
\end{equation}
```

Что делать с исходным файлом дальше? Теперь его нужно скомпилировать, используя непосредственно ту систему на базе LaTeX, в которой вы работаете. В Linux достаточно просто набрать в консоли команду `latex`, а дальше система сама попросит вас ввести имя исходного файла, который вы собираетесь верстать.

После того как указано имя файла, LaTeX попытается его скомпилировать. Если ошибок не обнаружено, то в каталоге с исходным файлом TeX будет создан файл с тем же именем, но с расширением DVI. Файлы DVI — это, как уже говорилось, платформонезависимые двоичные фай-



◀ А так отображается DVI-файл в программе Xdvi — еще одном популярном просмотрщике для Linux

лы, в которых хранятся уже сверстанные документы. Просмотреть их можно, используя различные программы, например Xdvi, если вы работаете под X Window. Кстати, в KDE входит своя утилита для работы с DVI-файлами — Kdvi, обладающая аналогичными возможностями. С помощью этих приложений можно просмотреть и распечатать созданный DVI-файл.

Что же касается формул, то выглядят они вполне достойно. Важно то, что они по умолчанию расположены по центру — так же, как вы можете видеть их в печатных изданиях. Вспомните, сколько усилий нужно было приложить, чтобы добиться от средств Microsoft нужного положения объекта MS Equation на странице документа!

Теперь нужно решить, что делать с текстом дальше. Если вы собираетесь, например, передать его издательству для публикации, то отправлять нужно, естественно, исходный файл. Если вы будете его распечатывать, то распечатать можно и dvi-файл. Если же вы хотите, например, опубликовать свое творение в электронном виде в Интернете, то имеет смысл экспортировать текст в форматы PostScript или PDF. Можно, конечно, оставить и DVI-файл, однако сейчас этот формат практически не используется в силу того, что при создании в нем не были предусмотрены некоторые возможности. Кроме того, TeX, генерируя DVI-файл, использует растровые шрифты, создавая их из векторных (для этого применяется утилита METAFONT, входящая практически в любой дистрибутив TeX/LaTeX). А экспорт полученного DVI-файла в формат PDF или PostScript позволяет вставлять в документ шрифты TrueType.

Преобразовать полученный документ в PostScript можно утилитой `dvips`, вхо-

дящей в дистрибутив TeX/LaTeX:

```
dvips mytext.dvi
```

Для преобразования в PDF можно использовать утилиту `dvipdf` или `pdflatex`.

Кроме того, можно перевести исходный файл TeX в формат HTML. Но для этого потребуется самостоятельная утилита `latex2html`, не входящая в пакет TeX/LaTeX. Если ее нет в вашем дистрибутиве Linux, то можно ее скачать с сайта www.latex2html.org.

Чтобы TEX-файл мог быть преобразован в HTML-документ, в преамбуле нужно указать команду использования дополнительных пакетов HTML и `makeidx`:

```
\usepackage{html,makeidx}
```

Однако при преобразовании в HTML могут возникнуть разного рода проблемы со ссылками внутри документа и гиперссылками на внешние источники, что может потребовать дополнительных изменений исходного файла, которые уже придется производить вручную.

Подводя итог, упомяну об одном недостатке TeX. И TeX, и LaTeX практически не поддерживают работу с графикой. При подключении дополнительных пакетов возможно вставлять в документ файлы PostScript или создавать графические примитивы, но этим, к сожалению, все и ограничивается. Также отсутствуют возможности использования различных эффектов — разворот текста или блока, эффекты оформления текста наподобие используемых в WordArt и т. д. Но такая ситуация вполне объяснима изначальным предназначением TeX — научные статьи и книги вряд ли нуждаются в вычурном оформлении.

■ ■ ■ Мария Сысойкина