

# Отчет по лабораторной работе 1:

## L<sup>A</sup>T<sub>E</sub>X Git GPG

Семён Мартынов

<semen.martynov@gmail.com>

14 февраля 2015 г.

## Содержание

<b>1</b>	<b>Система верстки T<sub>E</sub>X и расширения L<sup>A</sup>T<sub>E</sub>X</b>	<b>2</b>
1.1	Цель работы . . . . .	2
1.2	Ход работы . . . . .	2
1.2.1	Компиляция в командной строке . . . . .	2
1.2.2	Оболочка TexMaker . . . . .	2
1.2.3	Классы документов . . . . .	4
1.2.4	Подключаемые пакеты . . . . .	4
1.2.5	Вёрстка формул . . . . .	5
1.3	Выводы . . . . .	5
<b>2</b>	<b>Система контроля версий Git</b>	<b>6</b>
2.1	Цель работы . . . . .	6
2.2	Ход работы . . . . .	6
2.3	Выводы . . . . .	7
<b>3</b>	<b>Создание электронных цифровых подписей с PGP</b>	<b>8</b>
3.1	Цель работы . . . . .	8
3.2	Ход работы . . . . .	8
3.3	Выводы . . . . .	8

# 1 Система верстки $\text{\TeX}$ и расширения $\text{\LaTeX}$

## 1.1 Цель работы

Изучение принципов верстки  $\text{\TeX}$ , создание первого отчёта.

## 1.2 Ход работы

Файл `.tex` представляет из себя обычный текстовый файл содержащий макрокоманды текстовой разметки.

### 1.2.1 Компиляция в командной строке

- `latex` генерирует файл в формате DVI (**DeV**ice **I**ndependent — аппаратно независимый) не предназначенный для чтения человеком, но содержит двоичные данные, описывающие визуальное представление документа способом, не ориентированным на какой-либо формат изображения, монитор или принтер. Файлы DVI обычно подаются на вход другой программы (называемой DVI-драйвером), которая преобразует их в графические данные.

```
latex report.tex
```

- `xdvi` одна из программ DVI-драйверов, позволяющих отображать данные в формате DVI в X Window системах

```
xdvi report.dvi
```

Результат показан на рисунке 1.

- `pdflatex` позволяет сразу сгенерировать pdf файл. Главное различие между  $\text{\TeX}$  и `pdfLaTeX` состоит в том, что  $\text{\TeX}$  после трансляции выдаёт DVI-файлы, а `pdfTeX` — PDF-файлы, минуя цепочку преобразований DVI -> PS -> PDF.

```
pdflatex report.tex
```

### 1.2.2 Оболочка `TexMaker`

`Texmaker` является мощным редактором текста и исходного кода, работающий с языком разметки  $\text{\LaTeX}$ . Он позволяет форматировать текст и готовить многостраничные документы к печати. Редактор предоставляет возможность работы с библиографическими списками, оглавлением и другими атрибутами профессионального оформления. В `Texmaker` есть так

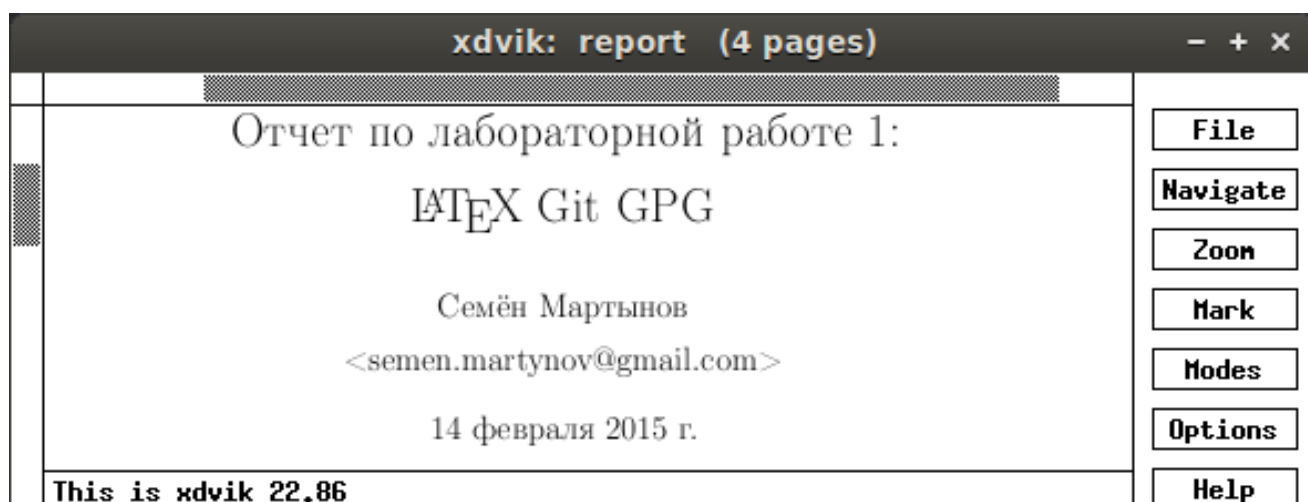


Рис. 1: Запуск xdvik

же возможность конвертирования документов в различные форматы, функции сворачивания блоков кода и автозавершения кода, встроенный просмотрщик PDF документов и многое другое. Внешний вид редактора представлен на рисунке 2.

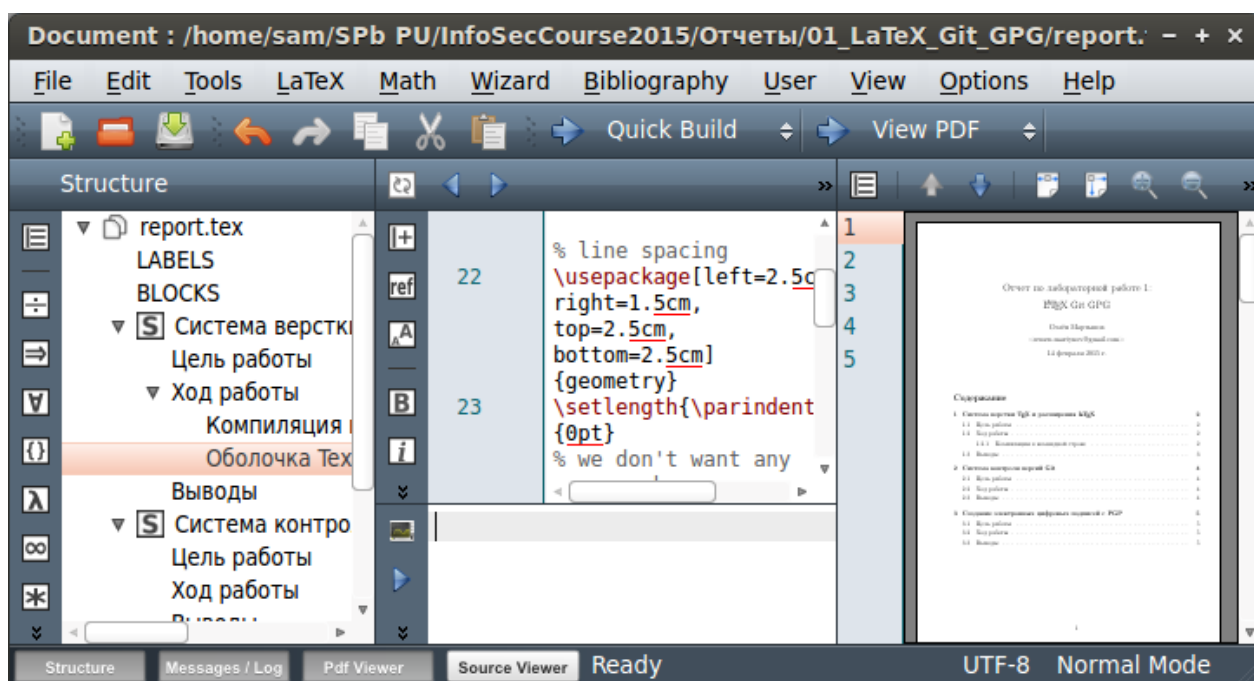


Рис. 2: Редактор TexMaker

Texmaker обладает двумя интересными возможностями: быстрый старт и быстрая сборка. Быстрый старт (рисунок 2) позволяет задать преамбулу (главные особенности - класс, размер бумаги, кодировку...) документа. Имеется возможность создать собственную модель преамбулы в редакторе.

Самый простой способ скомпилировать документ это использовать команду "Быстрая

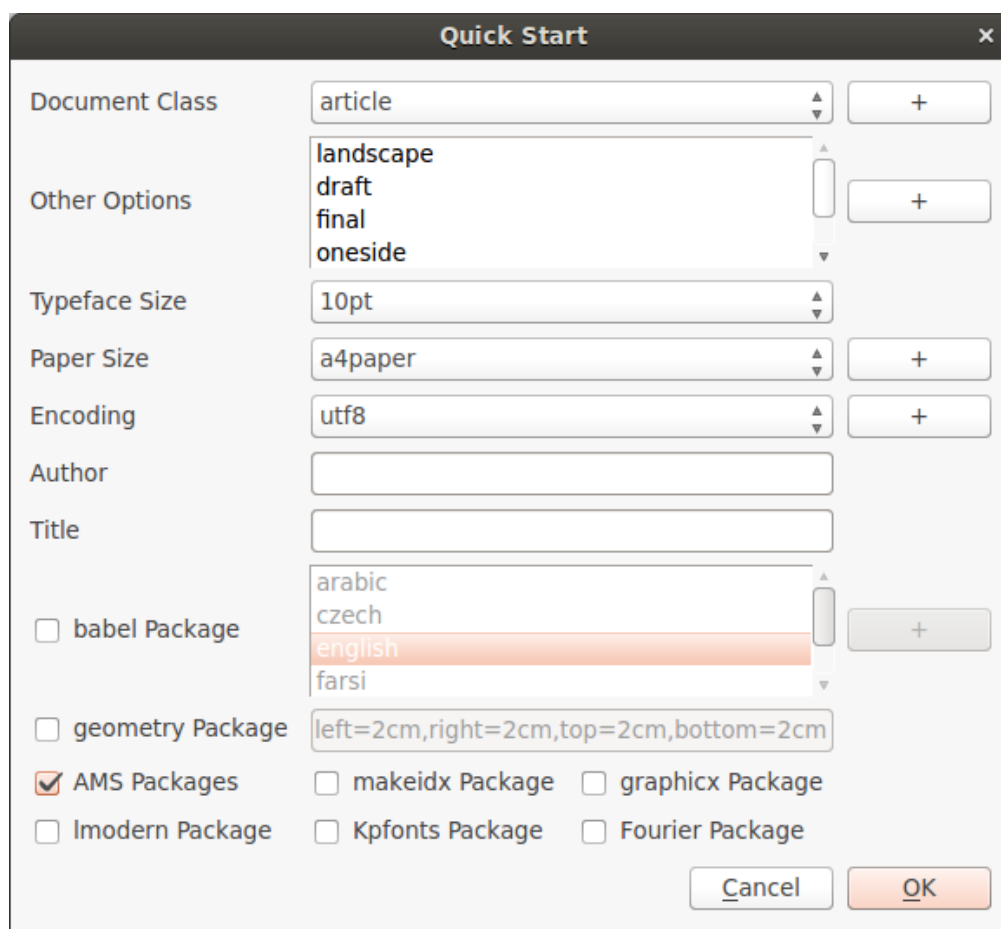


Рис. 3: Редактор TexMaker

сборка". Задать последовательность команд используемых быстрой сборкой можно в диалоге "Настроить Texmaker". Если в коде документа содержатся ошибки, Texmaker напишет об этом в окне сообщений.

### 1.2.3 Классы документов

Каждый созданный файл в  $\text{\LaTeX}$  начинается с команды `\documentclass[...]{...}`, в фигурных скобках которой задаются параметры оформления стиля документа, а в квадратных — список классовых опций.

Всего же в  $\text{\LaTeX}$  5 основных классов документов: `article` (для статей), `report` (для верстки небольших книг, статей, разбитых на главы), `book` (для верстки книг), `procs` (возможно использовать для докладов) и `letter` (для оформления деловых писем). Помимо этих основных, есть ещё множество дополнительных классов, таких как `beamer`.

### 1.2.4 Подключаемые пакеты

В  $\text{\LaTeX}$  можно применять специфические, отличные от изначальных, настройки (поля, списки и таблицы, библиографические ссылки и прочее). Для этого используются пакеты

расширений, подключаемые в "шапке" документа.

Пример:

```
\usepackage[russian]{babel} % Пакет поддержки русского языка
```

### 1.2.5 Вёрстка формул

Вёрстка формул не представляет никакой сложности.

Система дифференциальных уравнений Рёсслера

$$\begin{cases} \frac{dx}{dt} = -y - z \\ \frac{dy}{dt} = x + ay \\ \frac{dz}{dt} = b + z(x - c) \end{cases};$$

Массив связанных осцилляторов Рёсслера:

$$\begin{aligned} \dot{x}_i &= -\omega_i y_i - z_i + k(2x_i - x_{i-1} - x_{i+1}), \\ \dot{y}_i &= \omega_i x_i - a y_i, \\ \dot{z}_i &= b + z_i(x_i - c), \end{aligned}$$

## 1.3 Выводы

Л<sup>A</sup>T<sub>E</sub>X наиболее популярный набор макрорасширений (или макропакет) системы компьютерной вёрстки T<sub>E</sub>X, который облегчает набор сложных документов.

Пакет позволяет автоматизировать многие задачи набора текста и подготовки статей, включая набор текста на нескольких языках, нумерацию разделов и формул, перекрёстные ссылки, размещение иллюстраций и таблиц на странице, ведение библиографии и др. Кроме базового набора существует множество пакетов расширения Л<sup>A</sup>T<sub>E</sub>X.

Термин Л<sup>A</sup>T<sub>E</sub>X относится только к языку разметки, он не является текстовым редактором. Для того, чтобы создать документ с его помощью, надо набрать .tex-файл с помощью какого-нибудь текстового редактора. В принципе, подойдёт любой редактор, но большая часть людей предпочитает использовать специализированные, которые так или иначе облегчают работу по набору текста Л<sup>A</sup>T<sub>E</sub>X-разметки.

Будучи распространяемым под лицензией LaTeX Project Public License, Л<sup>A</sup>T<sub>E</sub>X относится к свободному программному обеспечению.

## 2 Система контроля версий Git

### 2.1 Цель работы

Изучить систему контроля версий Git, освоить основные приёмы работы с ней.

### 2.2 Ход работы

- Получить содержимое репозитория

```
git clone git@github.com:SemenMartynov/InfoSecCourse2015.git
```

- Добавить новую папку и первого файла под контроль версий

```
cd InfoSecCourse2015/  
mkdir tmp  
cd tmp  
echo 1 >> file  
git add --all
```

- Зафиксировать изменения в локальном репозитории

```
git commit -a -m "file added"
```

- Внести изменения в файл и просмотреть различия

```
echo 2 >> file  
git diff master:./file ./file
```

- Отменить локальные изменения

```
git reset HEAD ./file  
git checkout ./file
```

- Внести изменения в файл и просмотреть различия

```
echo 3 >> file  
git diff master:./file ./file
```

- Зафиксировать изменения в локальном репозитории, зафиксировать изменения в центральном репозитории

```
git commit -a -m "file changed"
git push
```

- Получить изменения из центрального репозитория

```
git pull
```

- Поэкспериментировать с ветками

```
git branch -v
git checkout -b temp
git checkout master
git merge temp
git branch
git branch -D temp
git branch
```

## 2.3 Выводы

Git распределённая система управления версиями файлов. Git используется многими продуктами с открытым исходным кодом, такими как ядро Linux, Android, GNU Core Utilities, Mesa, Wine, Chromium и т.д. Программа является свободной и выпущена под лицензией GNU GPL версии 2.

Преимущества и недостатки git по сравнению с централизованными системами управления версиями (такими как, например, Subversion) типичны для любой распределённой системы. Если же сравнивать git с «родственными» ей распределёнными системами, можно отметить, что git изначально идеологически ориентирован на работу с изменениями, а не с файлами, «единицей обработки» для него является набор изменений, или патч. Эта особенность прослеживается как в структуре самой системы (в частности — в структуре репозитория), так и в принципах построения команд; она отражается на производительности системы в различных вариантах её использования и на достоинствах и недостатках git по сравнению с другими DVCS.

### **3 Создание электронных цифровых подписей с PGP**

#### **3.1 Цель работы**

aaa

#### **3.2 Ход работы**

bbb

#### **3.3 Выводы**

ccc