


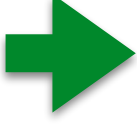



Публикация и полный вперед!



Елена Иванова
@liveldi90
Front-end developer
Artec3D

Содержание

-  Консоль. Основные команды для работы с файловой системой.
-  Системы контроля версий
-  Git, установка и настройка. Github. Публикация.
-  Сборщики. Шаблонизаторы и препроцессоры. Примеры.
-  Что дальше?

Консоль. Основные команды для работы с файловой системой.

Установка



Cygwin
(cygwin.com)



iTerm
(iterm2.com)

Дополнительно mac OS: <https://github.com/robbyrussell/oh-my-zsh>

Основные команды

`pwd`

— Путь, где вы находитесь?

`ls`

— Перечислить содержимое текущего каталога.

`ls -a`

— Содержимое каталога со всеми скрытыми файлами

`cd folder`

— Перейти из текущего каталога в папку folder (данная папка должна находиться в этом каталоге, иначе никуда не перейдет ;)). Регистр букв не имеет значения.*

`cd ~/`

— Перейти в корневую директорию вашего пользователя.

* Регистр букв — заглавные и строчные буквы.

Основные команды

`rm file.txt`

— удалить файл.

`open .`

— Открывает каталог в котором находитесь в стандартном виде.

`mkdir folder`

— Создать папку folder

`touch file.txt`

— Создать файл file.txt

* Регистр букв — заглавные и строчные буквы.

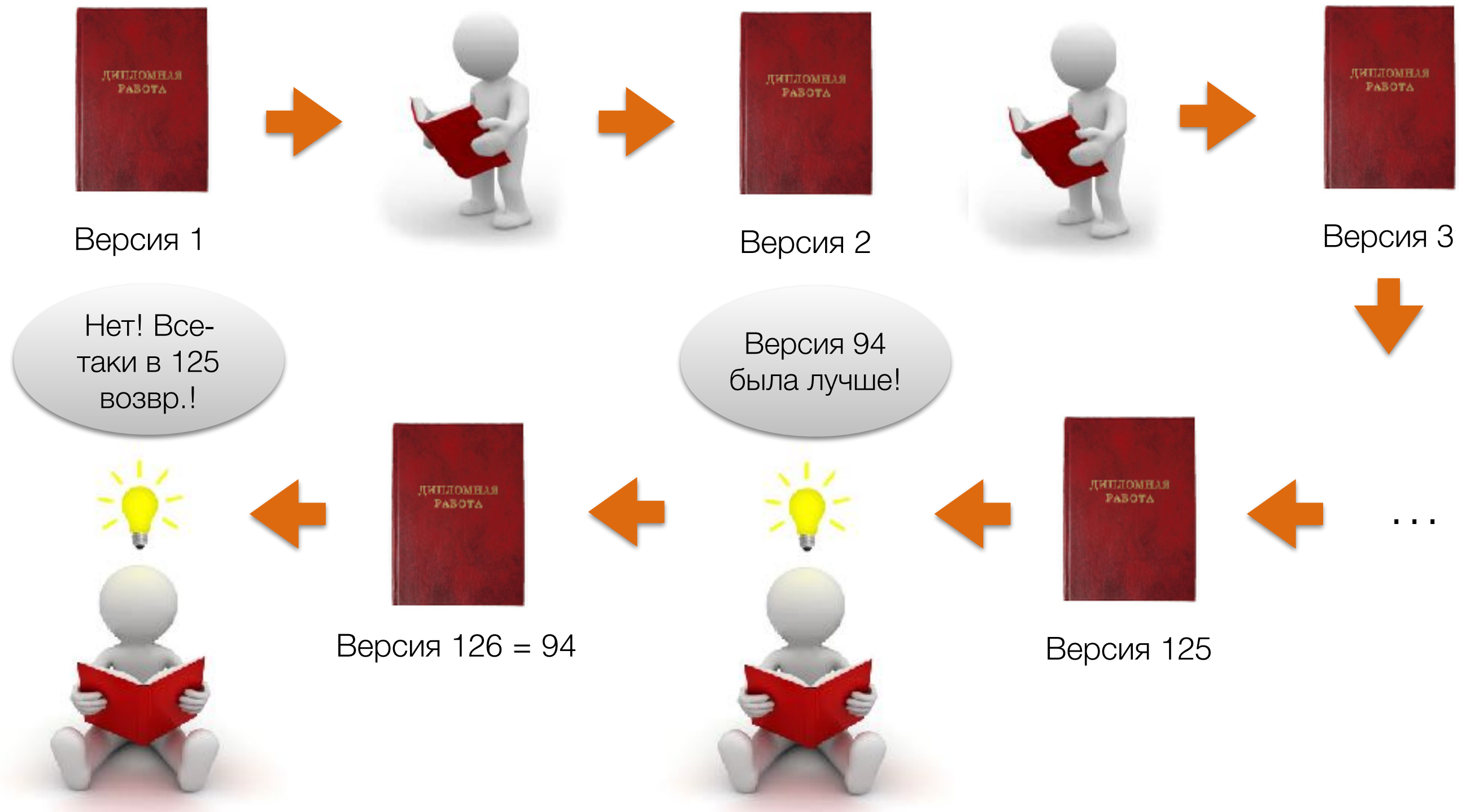
Системы контроля версий

Система управления версиями (СУВ)

— это система, сохраняющая изменения в одном или нескольких файлах так, чтобы потом можно было восстановить определённые старые версии.



Пишем дипломную работу

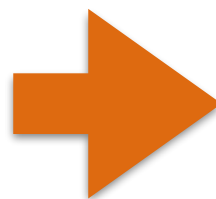


AAAAAA!





Элементарная система



- Версия 1
- Версия 2
- Версия 3
- Версия 4
- Версия 5
- .
- .
- .
- .
- Версия 94
- .
- .
- .
- Версия 125

Плюсы и минусы

+ Можем легко вернуться к любой из версий

- Если нужно объединить версии 94 + 125.
- Преподаватель дает вам поправленную версию, но вы в это время тоже работали.
- А что если у вас командная работа и над дипломом работают 5 человек.
- Высокая вероятность ошибок!



Если бы у вас была система, которая хранила бы все ваши изменения, но при этом вы бы работали всегда только с одной версией...

- ➔ Вы могли бы легко переключаться между версиями
- ➔ Соединять разные версии
- ➔ Система бы сама объединяла работу над проектом всей команды и отдавала общий итог....



Системы контроля версий



SVN



Git



Mercurial

Возможности СУВ

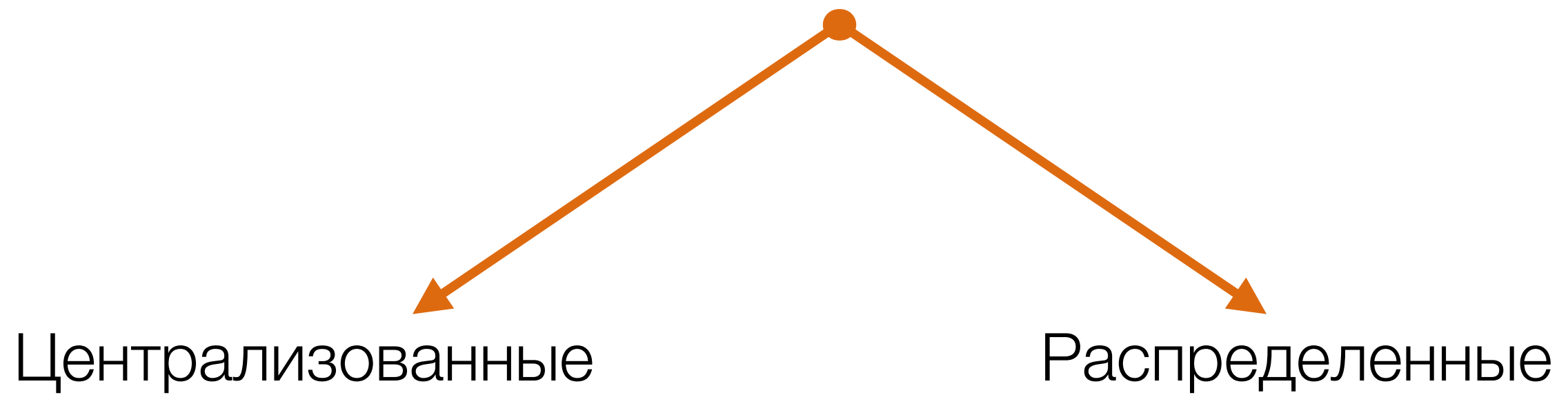
- ➔ Возврат файлов к прежнему виду,
- ➔ Возврат к прежнему состоянию всего проекта,
- ➔ Сравнение изменений с какой-то версией,
- ➔ Информация, о том кто последним изменял модуль, который дал сбой.

Вообще, если, пользуясь СУВ, вы всё испортили или потеряли файлы, всё можно легко восстановить. Кроме того, издержки на всё это будут очень маленькими.

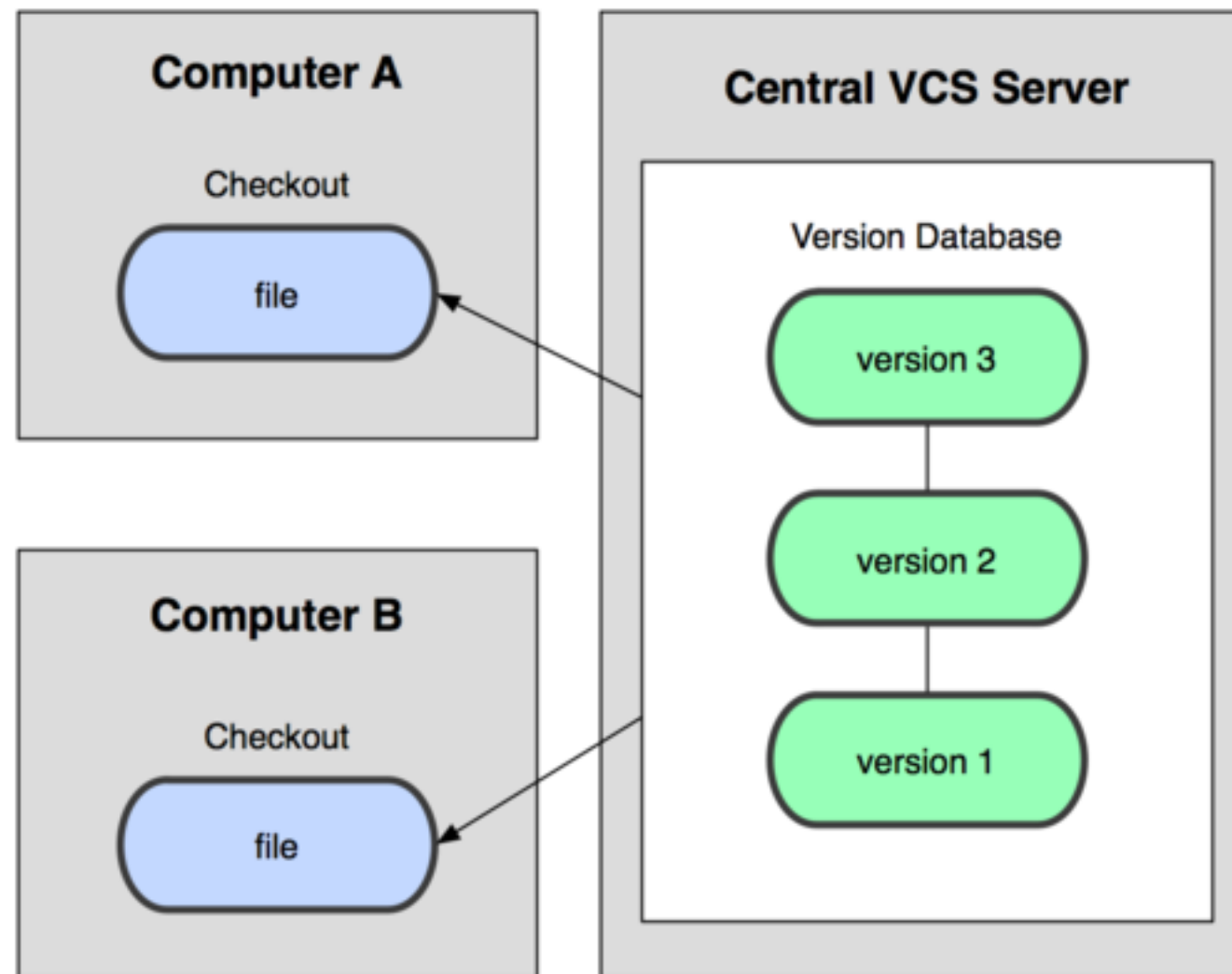
Восторг!



Виды СУВ



Централизованные СУВ



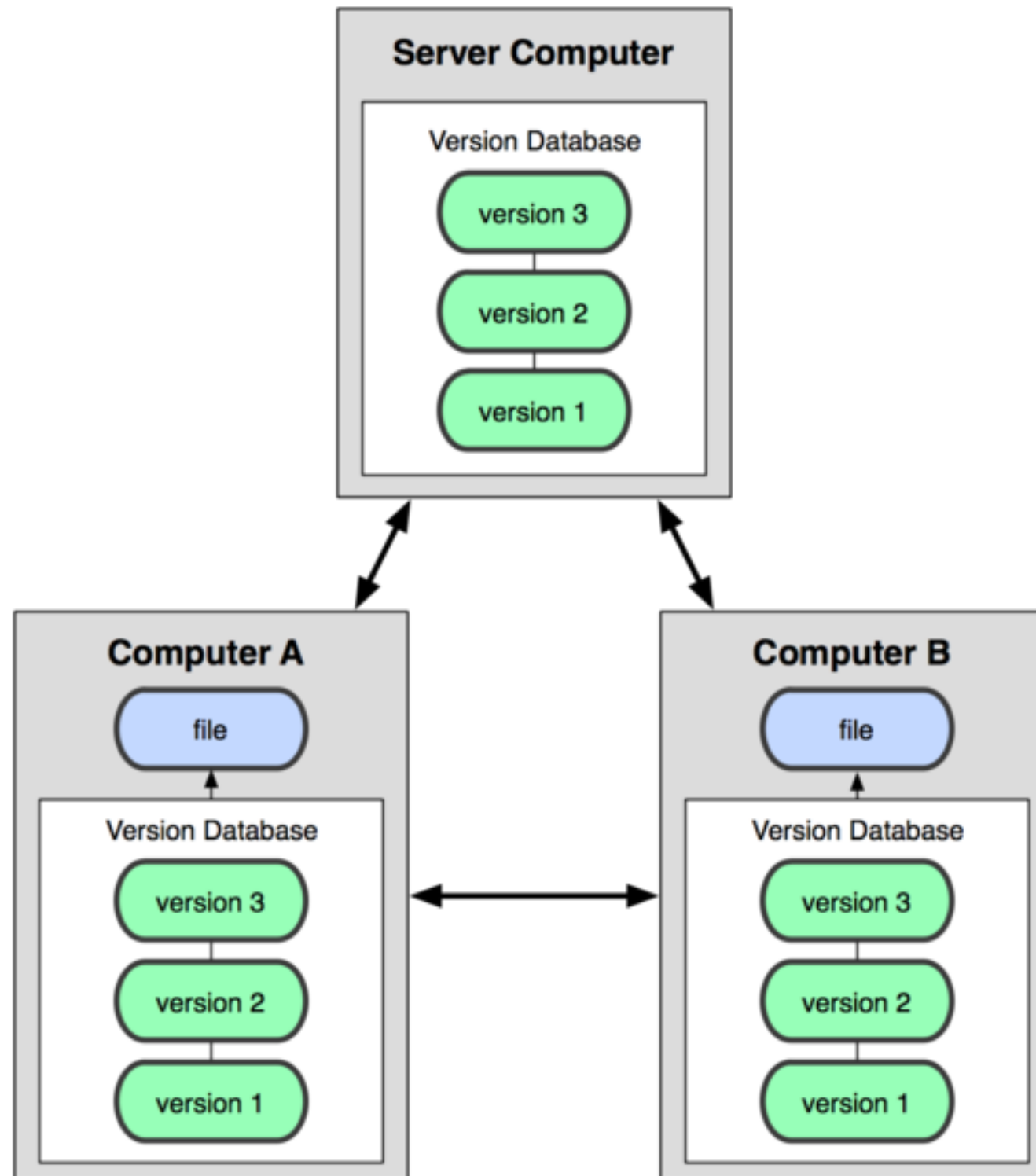
Централизованные СУБ

- + все знают, кто и чем занимается в проекте.
- + у администраторов есть чёткий контроль над тем, кто и что может делать.
- + администрировать ЦСУБ гораздо легче, чем локальные файлы на компьютере.

- централизованный сервер является уязвимым местом всей системы. Выключили сервер, повредился диск!



Распределенные СУБ



Распределенные СУБ

- + Все преимущества централизованных СУБ
- + Каждый раз, когда клиент забирает свежую версию файлов, создаётся полная копия всех данных.

Когда «умирает» сервер, любой клиентский репозиторий может быть скопирован обратно на сервер, чтобы восстановить базу данных.



**Пользуясь СУВ, значительно сократите
время на разработку любого проекта!**



Системы контроля версий



SVN

Централизованные



Git

Распределенные



Mercurial

Системы контроля версий



SVN



Git



Mercurial

Git, установка и
настройка. Github.
Публикация.

Установка



git-scm.com/downloads

git-for-windows.github.io — установщик для windows + графический редактор.

Настройка

Открываем консоль!

```
git config --global user.name "Your Name"  
git config --global user.email "Your Email"  
git config --list
```

Идем в проект и создаем репозиторий.

```
cd path_to_your_folder  
git init
```

(git init создаем .git содержащий все необходимые файлы репозитория — основу Git.)



Основные команды работы с Git

Добавляем файлы в проект.

`git status` — показывает состояние файлов.

Файлы в **красном цвете** не отслеживаемые, git их видит, в репозиторий они не добавятся

`git add .` — добавляет все файлы под git контроль.

Файлы станут **зеленого цвета**, т.е. git их взял под свой контроль

`git commit -m 'First version'`

— Делаем коммит, т.е. сохраняем первую версию!

Основные команды работы с Git

`git log` — Смотрим все наши версии.

`git checkout number`

— Возвращаемся к любому коммиту

`git checkout master`

— Возвращаемся назад



GitHub

— Внешнее хранилище для git репозиториев,

Вероятность потерять ваши файлы стремится к 0!



Опять консоль!

```
ssh-keygen -t rsa -C "your_email@example.com"
```

— Генерируем ssh ключ, для легкой авторизации на gitHub.

```
cat ~/.ssh/id_rsa.pub
```

— Копируем ключ

Идем по адресу:

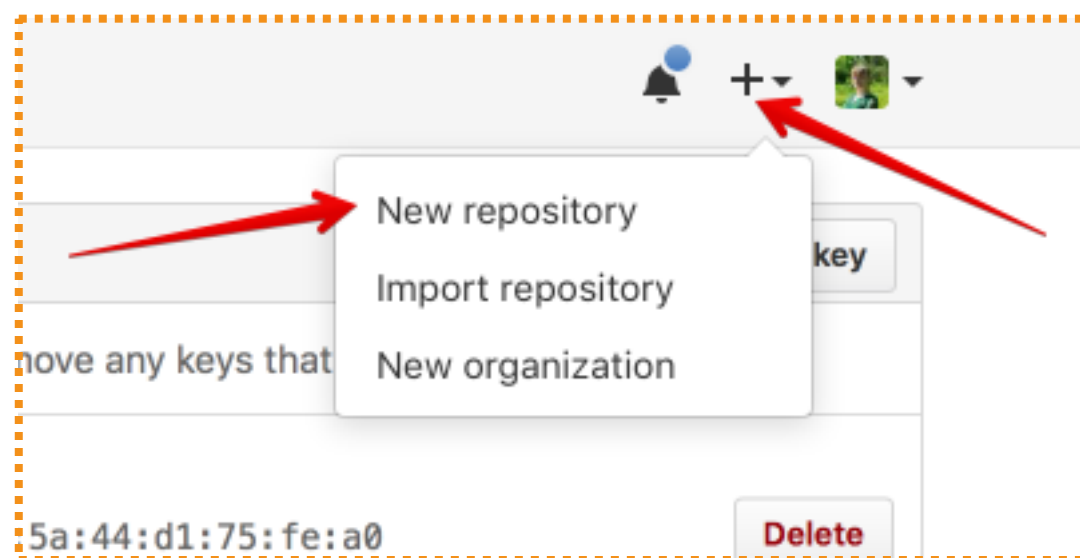
<https://github.com/settings/keys>

SSH and GPG keys -> New SSH key

(title любой, в key вставляем скопированный ключ)



Создаем проект на GitHub




Создаем проект на GitHub

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name


 liveld190 ▾

 /


your_name ✓

Great repository names are short and memorable. Need inspiration? How about **silver-octo-guacamole**.

Description (optional)

☒  **Public**

Anyone can see this repository. You choose who can commit.

☐  **Private**


You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

 |

Add a license: **None** ▾ 

Create repository

Название: **nickname.github.io**

Сохраняем проект на GitHub

The screenshot shows the GitHub 'Quick setup' page. At the top, there is a navigation bar with links: Code, Issues (0), Pull requests (0), Projects (0), Wiki, Pulse, Graphs, and Settings. Below this, the 'Quick setup' section is divided into three main parts. The first part, 'Quick setup — if you've done this kind of thing before', has a red arrow pointing to the word 'done'. It contains buttons for 'Set up in Desktop', 'HTTPS', and 'SSH', followed by the repository URL 'git@github.com:live1di90/test.git'. The second part, '...or create a new repository on the command line', has a red arrow pointing to the section title. It contains a list of Git commands, with the last three lines ('git remote add origin...', 'git push -u origin master') enclosed in a red dashed box. The third part, '...or push an existing repository from the command line', contains two lines of Git commands. The final part, '...or import code from another repository', includes a note about initializing from other VCS systems and an 'Import code' button.

Code Issues 0 Pull requests 0 Projects 0 Wiki Pulse Graphs Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH git@github.com:live1di90/test.git

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# test" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:live1di90/test.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@github.com:live1di90/test.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Резюме :)

git init

Делаем изменения

git add .

git commit -m “First version”

Делаем изменения

git add .

git commit -m “Second version”

Сохраняем в GitHub репозиторий:

git push



Сборщики.
Шаблонизаторы
и препроцессоры.
Примеры.

Сборщики

— это инструмент сборки веб-приложения, позволяющий автоматизировать повторяющиеся задачи.

- Grunt (утаревший)
- Gulp
- Webpack (больше для js)



Шаблонизаторы HTML

— позволяют упростить написание html

Pug — документация.

Пример обычного HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <link href="https://fonts.googleapis.com/
css?family=Open+Sans:300,400,600,700&subset=cyrillic" rel="
stylesheet" />
    <link href='styles/main.css' rel='stylesheet' type='text/
css' />
    <title>MCS – кодинг для девушек</title>
  </head>
  <body id="page">
    <div class="container">
      <p>Некоторый текст.</p>
      <ul>
        <li>пункт 1</li>
        <li>пункт 2</li>
      </ul>
    </div>
  </body>
</html>
```


Пример pug — HTML

```
doctype html
html
  head
    meta(charset="utf-8")
    link(href="https://fonts.googleapis.com/
css?family=Open+Sans:300,400,600,700&subset=cyrillic" rel="
stylesheet")
    link(href='styles/main.css' rel='stylesheet' type='text/css
')

    title= 'MCS — кодинг для девушек'

  body#page
    .container
      p Некоторый текст.

      ul
        li пункт 1
        li пункт 2
```

Препроцессоры CSS

— упрощение написания CSS

- SCSS
- less

Пример обычного CSS

```
.menu {  
  text-align: center;  
}  
  
.menu--hover {  
  position: absolute;  
  top: 0;  
  left: 0;  
}  
  
.menu--title {  
  position: relative;  
  font-size: 1.4em;  
  font-weight: 400;  
  text-align: center;  
  line-height: 1.2em;  
}
```

Пример SCSS

```
.menu {  
  text-align: center;  
  
  .menu--hover {  
    position: absolute;  
    top: 0;  
    left: 0;  
  }  
  
  &--title {  
    position: relative;  
    font-size: 1.4em;  
    font-weight: 400;  
    text-align: center;  
    line-height: 1.2em;  
  }  
}
```

Что дальше?

Ссылки

- [HTML & CSS. Джон Дакетт](#)
- [Язык JavaScript](#)
- [Интерактивные онлайн-курсы html academy](#)
- [Большой справочник по веб-разработке](#)
- [HTML book](#)
- [Записи школы разработки интерфейсов в ya.ru](#)
- [Шпаргалка jquery](#)
- [Уроки и интересные примеры](#)
- [Physics for JavaScript Games, Animation, and Simulations](#)

Вопросы?



Автор курса: Елена Иванова
lessons7.liveldi.ru