

# Система контроля версий (Version Control System)

# Понятие и задачи

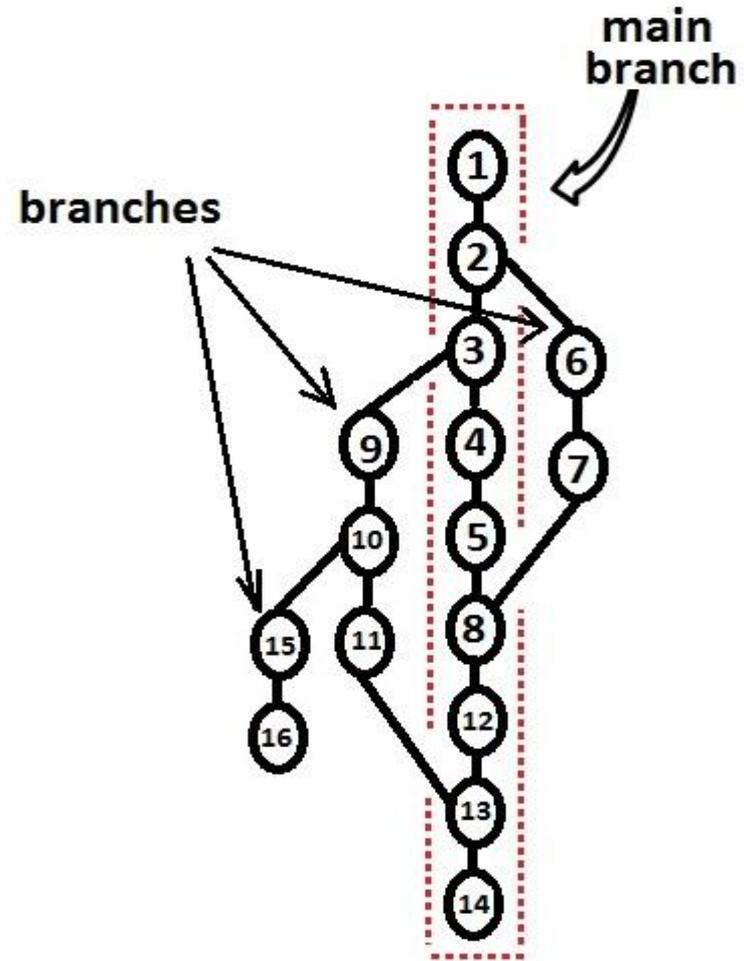
- ❑ хранение всех выполненных изменений, возможность посмотреть кем и когда они были выполнены, возможность «откатить» неудачные изменения
- ❑ возможность параллельной и независимой работы над данными
- ❑ возможность использовать изменения выполненные другими членами команды

Основное применение систем контроля версий:  
организация работы группы программистов над проектом



# Основные термины

- Версия (version)
- Метка (label, tag)
- Дерево версий (tree)
- Ветка (branch)
- Слияние (merge)

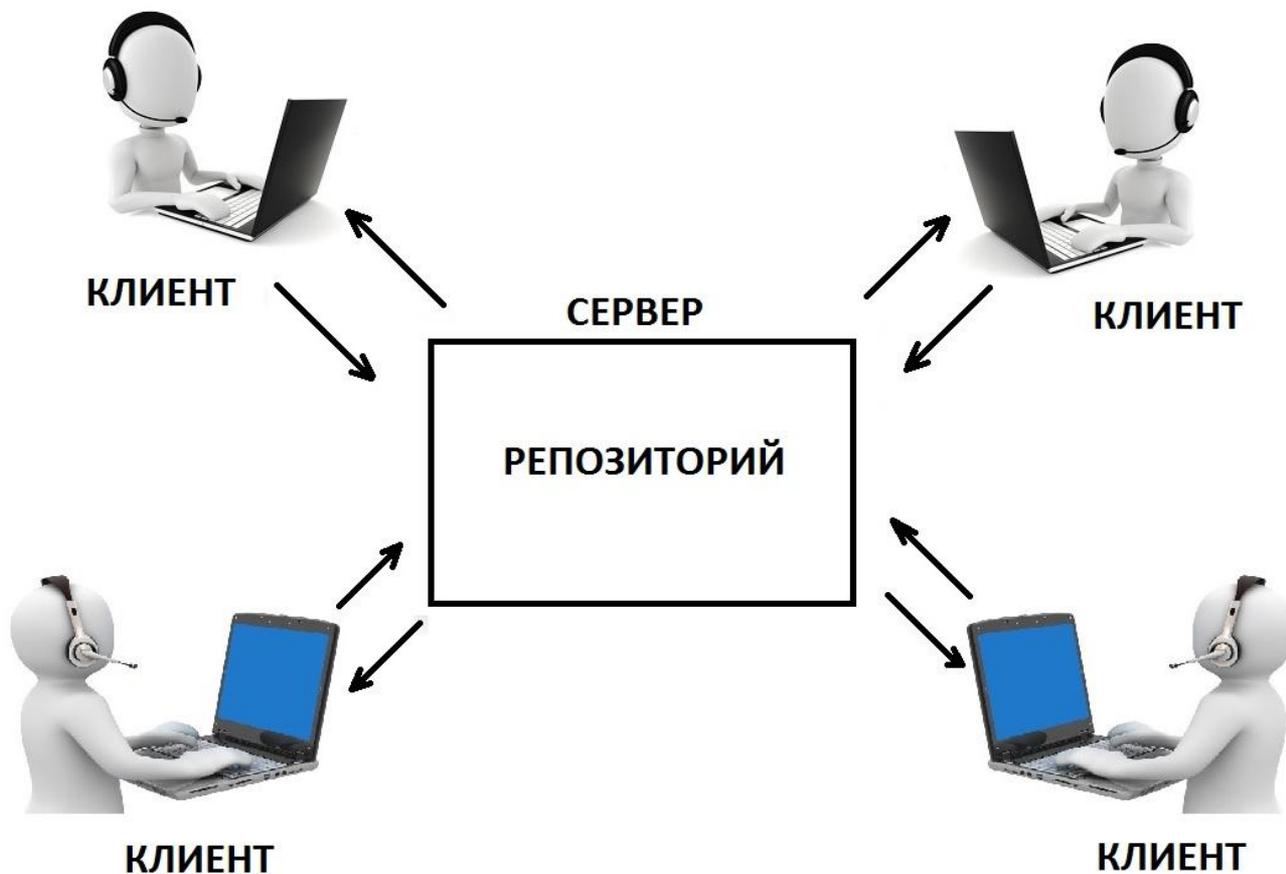


# Первая система контроля версий

SCCS (Source Code Control System, Bell Labs, 1972)

- ❑ позволяет осуществить работу только с одним файлом
- ❑ основан на хранении разницы между двумя соседними версиями файлов
- ❑ возможности сохранения, поиска, редактирования и отката изменений

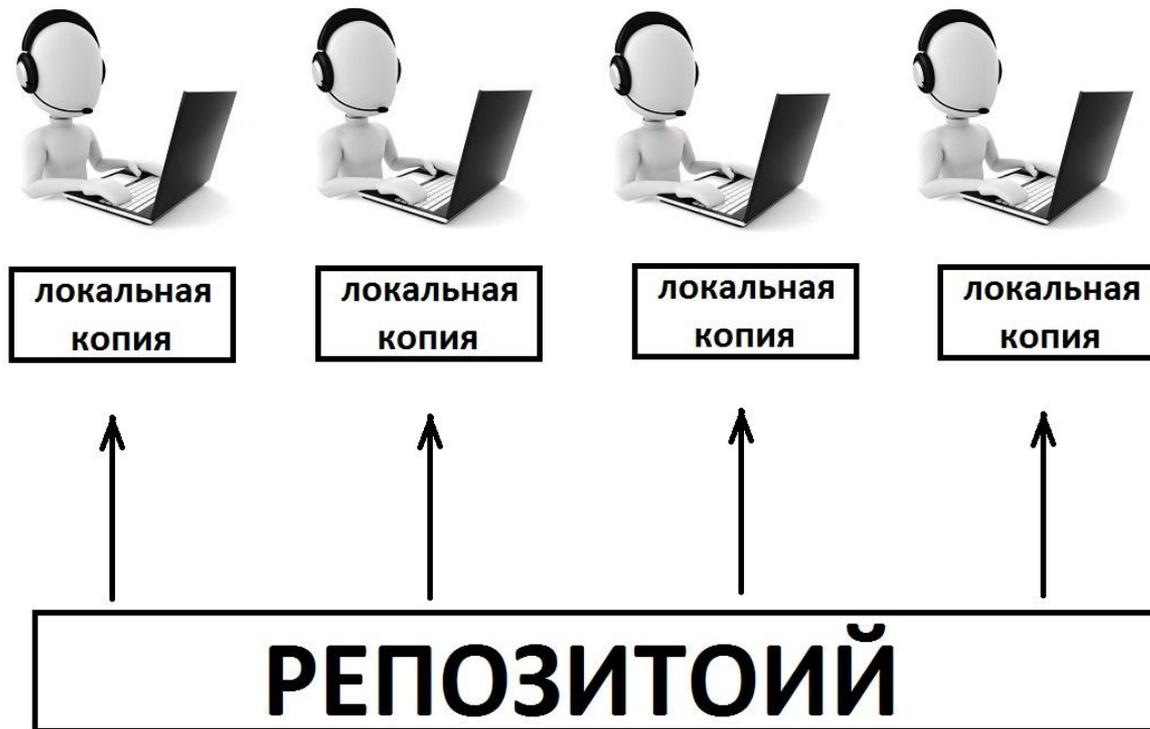
# Клиент-серверные системы контроля версий



# Популярные клиент-серверные СКВ

- ❑ CVS (Concurrent Version System, CVS team) – свободное ПО, разработана в 80-х годах 20 века
- ❑ Rational ClearCase (IBM) – коммерческий продукт, 90-е года 20 века.
- ❑ SVN (subversion) – свободное ПО, 2004г.

# Распределенные системы контроля версий



# Популярные распределенные СКВ

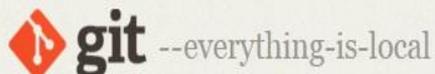
- ❑ Git (*Linus Torvalds*) – свободное ПО, 2005г.
- ❑ Mercurial (Matt Mackall) – свободное ПО, 2005г.
- ❑ Bazaar (Canonical Ltd.) - свободное ПО, 2007г.

# Лабораторная работа

- установить СКВ
- создать репозиторий, выполнить операции: добавление различных файлов и папок в репозиторий, создание версий файлов, создание веток, слияние веток.

# Установка

Официальный сайт: <http://git-scm.com/>



Search entire site...

## About

### Documentation

Reference  
Book  
Videos  
External Links

## Blog

## Downloads

## Community

This book is translated into [Deutsch](#), [简体中文](#), [繁體中文](#), [Français](#), [日本語](#), [Nederlands](#), [Русский](#), [한국어](#), [Portugués \(Brasil\)](#) and [Čeština](#).

Partial translations available in [Arabic](#), [Español](#), [Indonesian](#), [Italiano](#), [Suomi](#), [Македонски](#), [Polski](#) and [Türkçe](#).

Translations started for [Azərbaycan dili](#), [Беларуская](#), [Català](#), [Esperanto](#), [Español \(Nicaragua\)](#), [فارسی](#), [हिन्दी](#), [Magyar](#), [Norwegian Bokmål](#), [Română](#), [Српски](#), [ภาษาไทย](#), [Tiếng](#)

## Chapters • 1st Edition

# 1.4 Введение - Установка Git

## Установка Git

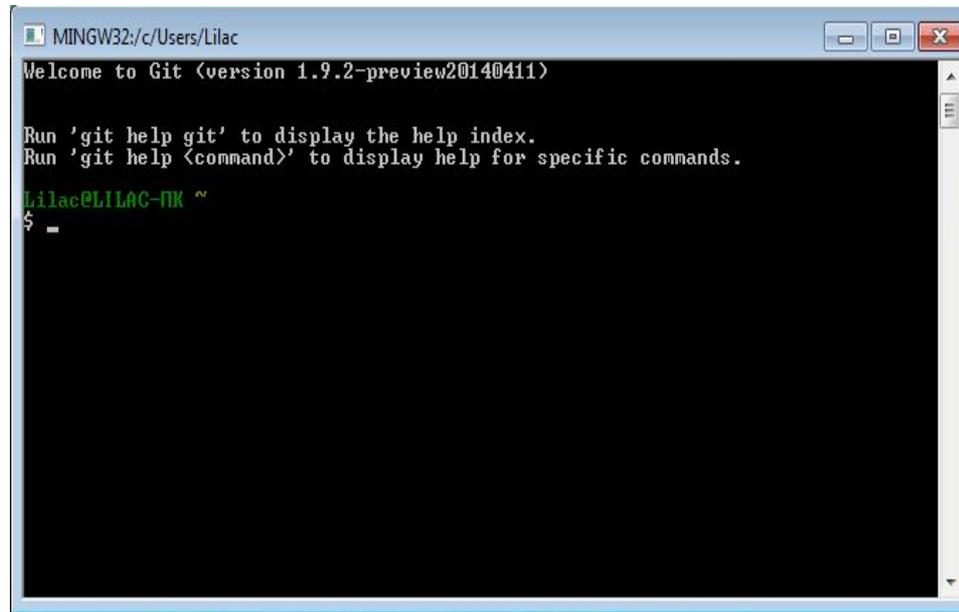
Настало время немного ознакомиться с использованием Git'a. Первое, что вам необходимо сделать, — установить его. Есть несколько способов сделать это; два основных — установка из исходников и установка собранного пакета для вашей платформы.

### Установка из исходников

Если есть возможность, то, как правило, лучше установить Git из исходных кодов, поскольку так вы получите самую свежую версию. Каждая новая версия Git'a обычно включает полезные улучшения пользовательского интерфейса, поэтому получение последней версии — часто лучший путь, если, конечно, вас не затрудняет установка программ из исходников. К тому же, многие дистрибутивы Linux содержат очень старые пакеты. Поэтому, если только вы не на очень свежем дистрибутиве или используете пакеты из экспериментальной ветки, установка из исходников может быть самым выигрышным решением.

Для установки Git'a вам понадобятся библиотеки, от которых он зависит: curl, zlib, openssl, expat и libiconv. Например, если в вашей системе менеджер пакетов — yum (Fedora), или apt-get (Debian, Ubuntu), можно воспользоваться следующими командами, чтобы разрешить все зависимости:

# Командная строка

A screenshot of a Windows command prompt window. The title bar shows the path 'MINGW32/c/Users/Lilac'. The window content displays the following text: 'Welcome to Git (version 1.9.2-preview2@140411)', 'Run 'git help git' to display the help index.', 'Run 'git help <command>' to display help for specific commands.', and the prompt 'Lilac@LILAC-ПК ~' followed by '\$ \_'.

```
MINGW32/c/Users/Lilac
Welcome to Git (version 1.9.2-preview2@140411)
Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.
Lilac@LILAC-ПК ~
$ _
```

Навигация по файловой системе: команда `cd`(использование: `cd<путь к папке>`)

Посмотреть содержимое папки: команда `ls`

Создать папку: команда `mkdir` (использование: `mkdir<имя папки>`)

Создать файл: команда `touch` (использование: `touch<имя файла>`)

# Команды

## Git

- git add, git commit
- git tag
- git branch
- gitk
- git merge

## ClearCase

- ct check-out, ct check-in
- ct mklbtype, ct mklabel
- ct mkbranch
- ct lsvtree
- ct merge