

Федеральное государственное автономное
образовательное учреждение высшего
образования
«Национальный исследовательский университет
ИТМО»

Факультет Информационных технологий и программирования

Работа: Описание системы управления версиями Git

Выполнила студентка группы №М3103
Кравченко Елизавета Александровна

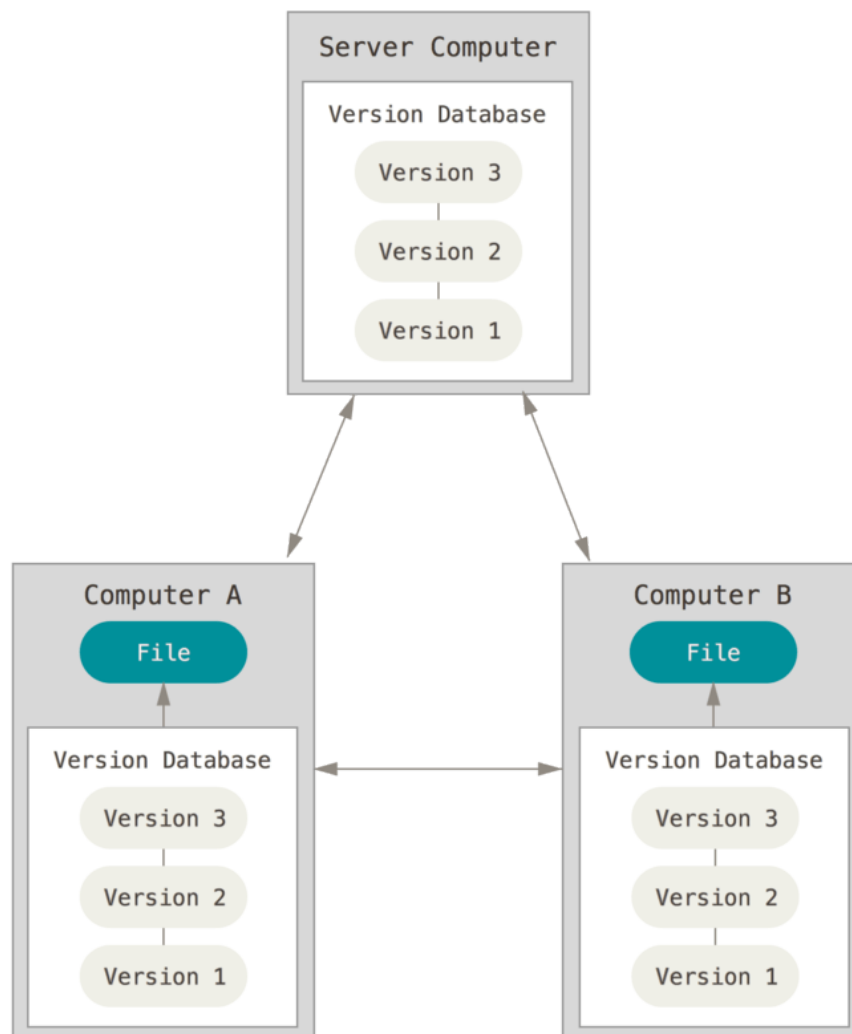
Проверил:
Повышев Владислав Вячеславович

Санкт-Петербург
2022 г.

Что такое система контроля версий?

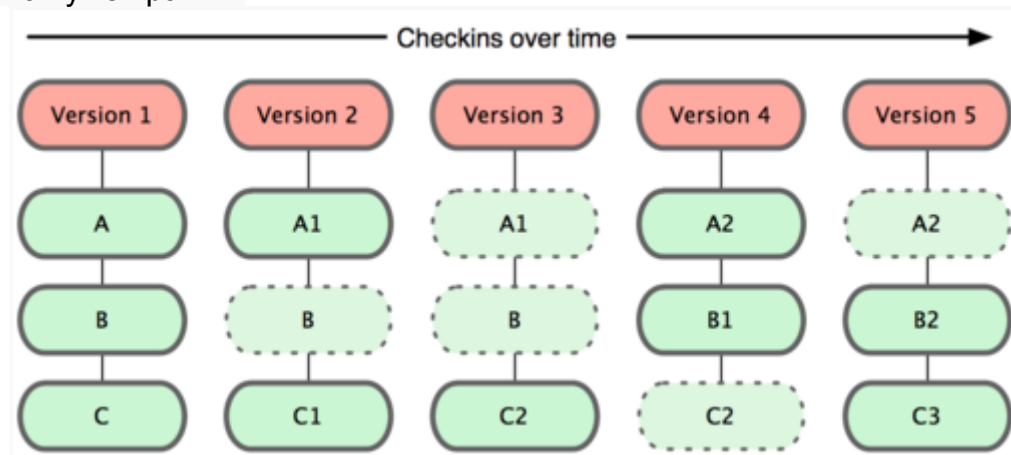
Система контроля версий (*Version Control System, VCS*) представляет собой программное обеспечение, которое позволяет отслеживать изменения в документах, при необходимости производить их откат, определять, кто и когда внес исправления и т.п.

Git — распределённая система контроля версиями. Распределённая-значит позволяет хранить репозиторий (его копию) у каждого разработчика, работающего с данной системой



Особенности работы git

Чтобы быть эффективным, если файлы не изменились, Git не хранит копию старого файла--он записывает просто ссылку на предыдущий идентичный файл, который он уже хранил.



Локальность разработки

Для работы большинства операций в Git достаточно локальных файлов и ресурсов — в основном, системе не нужна никакая информация с других компьютеров в сети. Вся история проекта хранится прямо на локальном диске. Git не нужно соединяться с сервером для её получения и отображения — система просто считывает данные напрямую из локальной базы данных. Следовательно разработка может вестись локально без огромных затрат на время запросов к серверу, мобильно. Подключившись к сети, можно запросить изменения или синхронизировать глобальную версию проекта и локальную (возможны решения конфликтов). Добиться такого же поведения во многих других системах либо очень сложно, либо вовсе невозможно.

Целостность Git

В Git для всего вычисляется хеш-сумма, и только потом происходит сохранение. В дальнейшем обращение к сохраненным объектам происходит по этой хеш-сумме. Это значит, что невозможно изменение содержимого уже сохраненных в системе файлов или каталогов так, чтобы Git не узнал об этом.

Механизм, которым пользуется Git при вычислении хеш-сумм, называется SHA-1 хеш.

хеш выглядит примерно так:

```
987ab52f24a493b9da037365522528696cd6d3b0
```

Git сохраняет все объекты в свою базу данных по хеш-сумме содержимого объекта.

Git обычно только добавляет данные

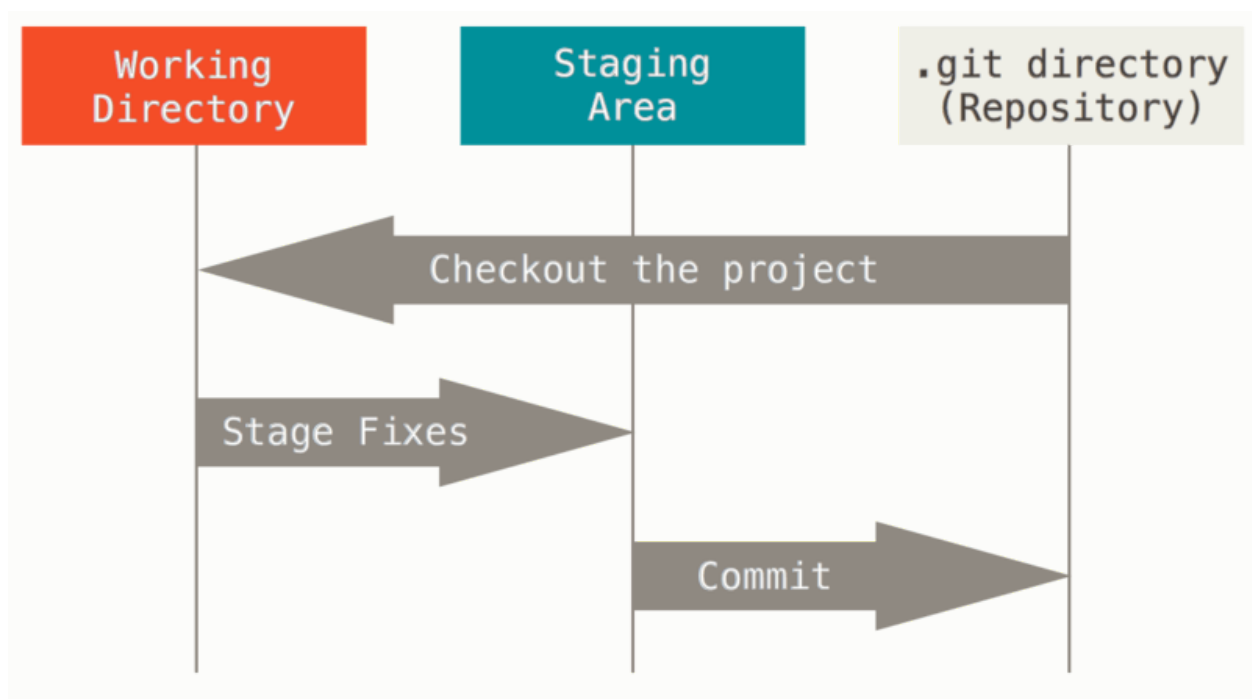
Любые действия в Git (фиксируются) добавляются, поэтому нельзя удалить из системы какие-либо данные так, чтобы нельзя было отменить эти действия. Как и в любой другой СКВ, вы можете потерять или испортить свои изменения, пока они не зафиксированы, но после того, как вы зафиксируете снимок в Git, будет очень сложно что-либо потерять, особенно при синхронизации локальной и глобальной версии.

Три состояния файла

- Если определённая версия файла есть в каталоге Git, эта версия считается **зафиксированной** (committed).
- Если файл был изменён и добавлен в индекс, значит, он **индексирован** (staged).
- И если файл был изменён с момента последнего распаковывания из репозитория, но не был добавлен в индекс, он считается **изменённым** (modified).
- фиксированный значит, что файл уже сохранён в вашей локальной базе

Три основные секции проекта Git

рабочая копия (working tree), область индексирования (staging area) и каталог Git (Git directory).



Рабочая копия является снимком одной версии проекта. Эти файлы извлекаются из сжатой базы данных в каталоге Git и помещаются на диск, для того чтобы их можно было использовать или редактировать.

Область индексирования — это файл, обычно находящийся в каталоге Git, в нём содержится информация о том, что попадёт в следующий коммит. Её техническое название на языке Git — «индекс», но фраза «область индексирования» также работает.

Каталог Git — это то место, где Git хранит метаданные и базу объектов вашего проекта. Это самая важная часть Git и это та часть, которая копируется при **клонировании** репозитория с другого компьютера.

Базовый подход в работе с Git выглядит так:

1. Изменяете файлы вашей рабочей копии.
2. Выборочно добавляете в индекс только те изменения, которые должны попасть в следующий коммит, добавляя тем самым снимки **только** этих изменений в индекс.
3. Когда вы делаете коммит, используются файлы из индекса как есть, и этот снимок сохраняется в ваш каталог Git.

Командная строка

Есть много различных способов использования Git. Помимо оригинального клиента, имеющего интерфейс командной строки, существует множество клиентов с графическим пользовательским интерфейсом, в той или иной степени реализующих функциональность Git. В рамках данной книги мы будем использовать Git в командной строке. С одной стороны, командная строка — это единственное место, где вы можете запустить все команды Git, так как большинство клиентов с графическим интерфейсом реализуют для простоты только некоторую часть функциональности Git. Если вы знаете, как выполнить какое-либо действие в командной строке, вы, вероятно, сможете выяснить, как то же самое сделать и в GUI-версии, а вот обратное не всегда верно. Кроме того, в то время, как выбор графического клиента — это дело личного вкуса, инструменты командной строки доступны всем пользователям сразу после установки Git.