



**Частное учреждение профессионального образования
«Высшая школа предпринимательства (колледж)»
(ЧУПО «ВШП»)**

Кафедра информационных технологий

**Инструментальные средства
разработки программного
обеспечения**

О чём этот междисциплинарный курс?

МДК 02.02 "Инструментальные средства разработки программного обеспечения" формирует у будущих специалистов теоретические знания и практические навыки разработки программного обеспечения.

Позволяет овладеть соответствующими общими и профессиональными компетенциями.

В результате освоения профессионального модуля студент должен:

Иметь практический опыт	модели процесса разработки программного обеспечения; основные принципы процесса разработки программного обеспечения; основные подходы к интегрированию программных модулей; основы верификации и аттестации программного обеспечения
уметь	использовать выбранную систему контроля версий; использовать методы для получения кода с заданной функциональностью и степенью качества
знать	модели процесса разработки программного обеспечения; основные принципы процесса разработки программного обеспечения; основные подходы к интегрированию программных модулей; основы верификации и аттестации программного обеспечения

ΛΛ

скучно

[

Что мы изучим (простыми словами)

систему контроля версий

**графический редактор для совместного проектирования
сайтов, приложений и других дизайнерских продуктов**

**основные принципы процесса разработки программного
обеспечения**

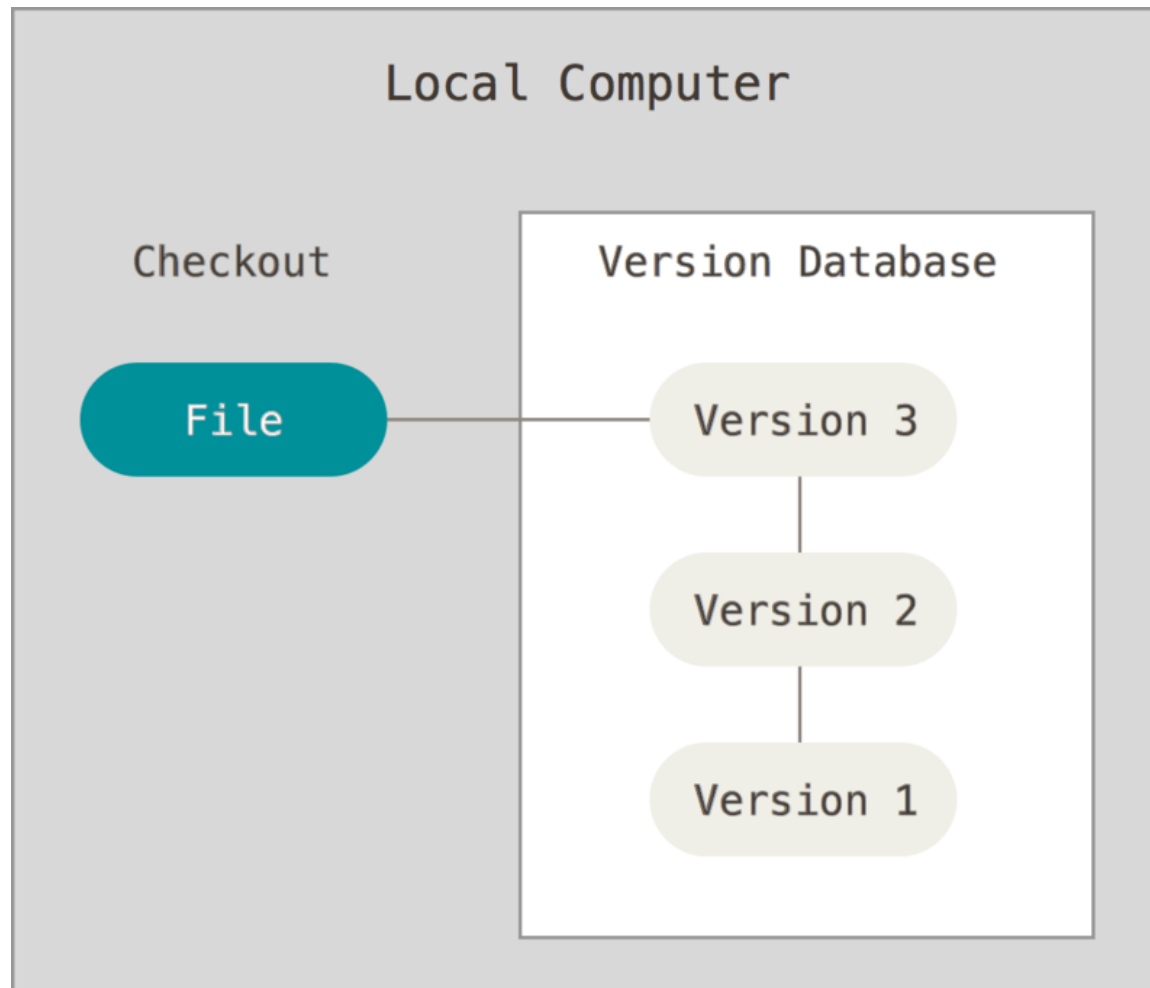
Что такое «система контроля версий» и почему это важно?

Система контроля версий (от англ. Version Control System, VCS) - это система, записывающая изменения в файл или набор файлов в течение времени. Позволяет вернуть файлы к состоянию, в котором они были до изменений, вернуть проект к исходному состоянию, увидеть изменения, увидеть, кто последний менял что-то и вызвал проблему. Использование VCS также значит в целом, что, если вы сломали что-то или потеряли файлы, вы спокойно можете всё исправить.

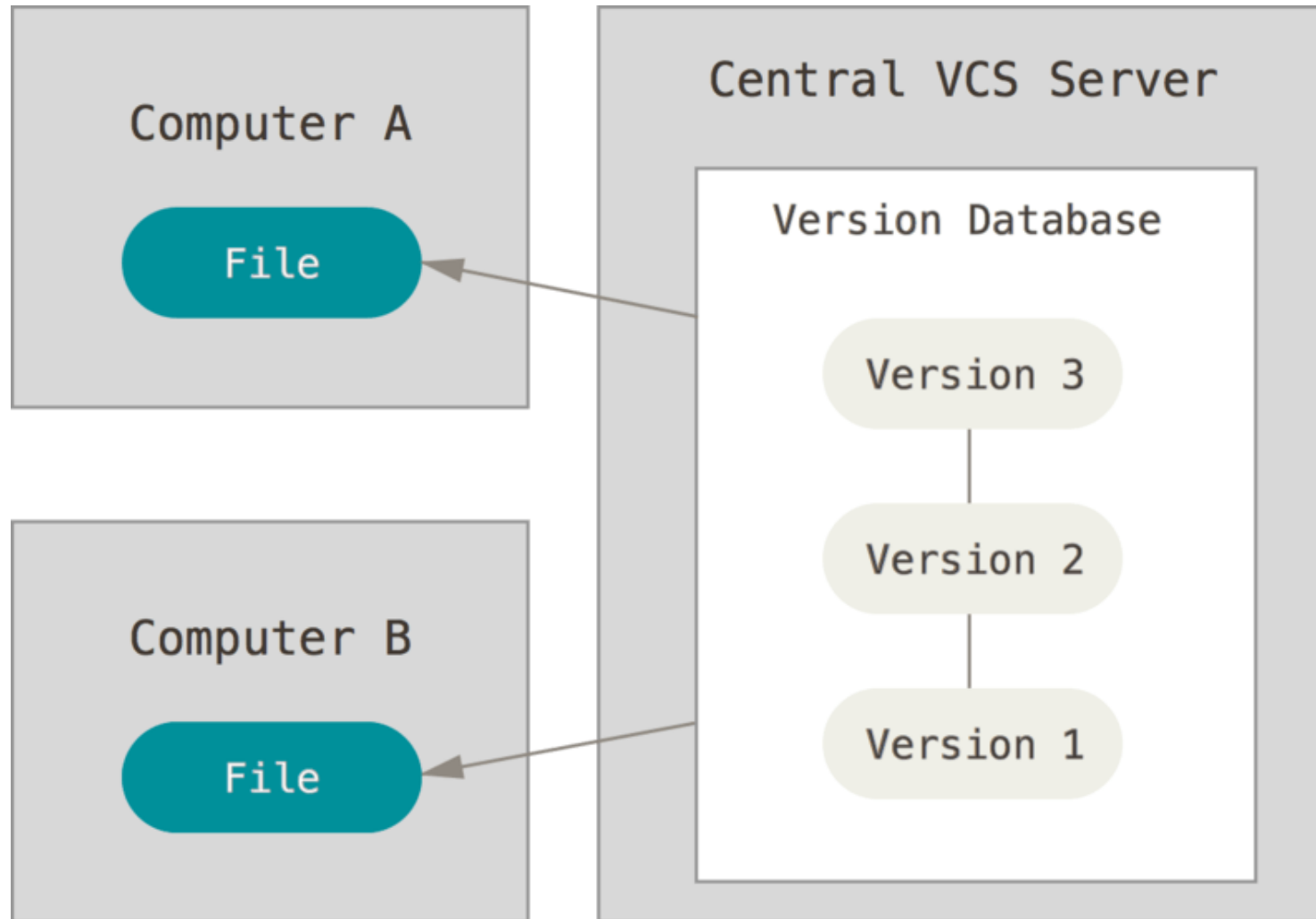
Виды VCS

- Локальные системы контроля версий
- Централизованные системы контроля версий
- Распределённые системы контроля версий

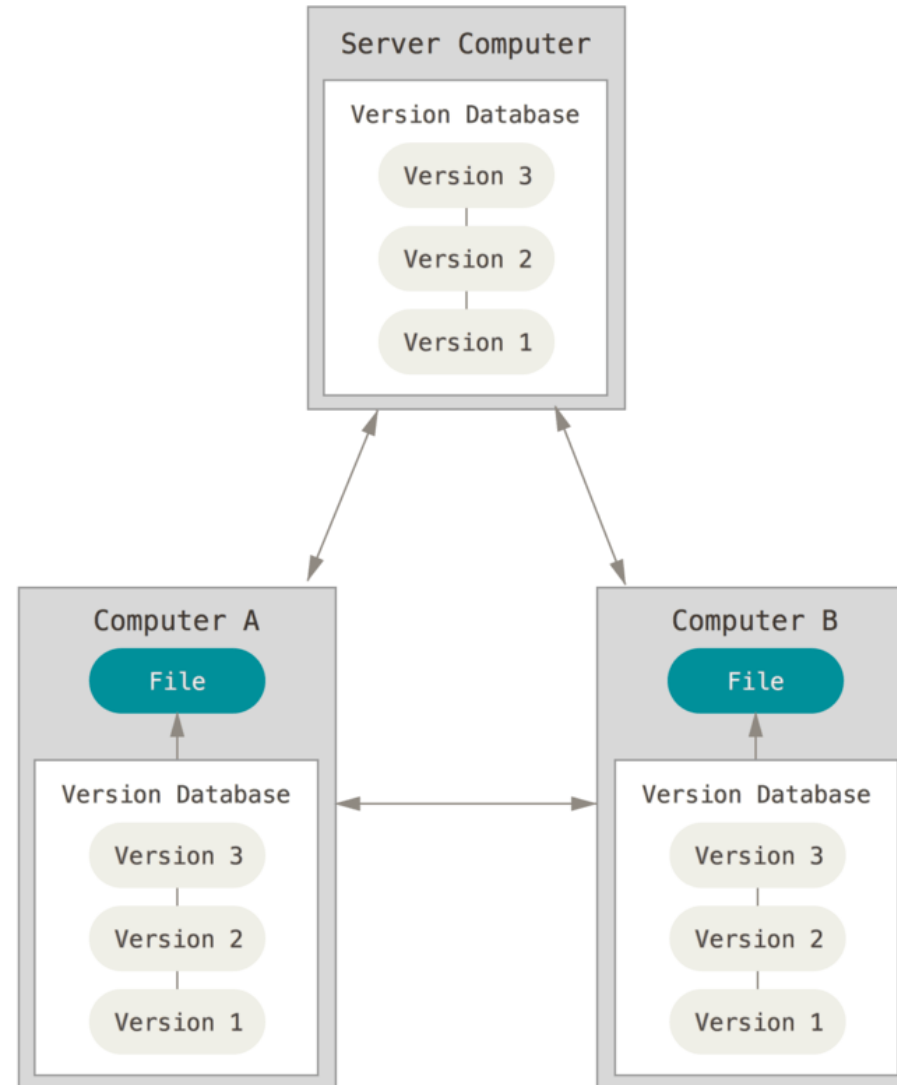
Локальные системы контроля версий



Централизованные системы контроля версий



Распределённые системы контроля версий



Самая популярная VCS

Git — мощная и сложная распределенная система контроля версий. Понимание всех возможностей git открывает для разработчика новые горизонты в управлении исходным кодом.

Термины

Git или Гит — система контроля и управления версиями файлов.

GitHub или Гитхаб — веб-сервис для размещения репозиториев и совместной разработки проектов.

Репозиторий Git — каталог файловой системы, в котором находятся: файлы конфигурации, файлы журналов операций, выполняемых над репозиторием, индекс расположения файлов и хранилище, содержащее сами контролируемые файлы.

Локальный репозиторий — репозиторий, расположенный на локальном компьютере разработчика в каталоге. Именно в нём происходит разработка и фиксация изменений, которые отправляются на удалённый репозиторий.

Удалённый репозиторий — репозиторий, находящийся на удалённом сервере. Это общий репозиторий, в который приходят все изменения и из которого забираются все обновления.

Термины(1)

Форк (Fork) — копия репозитория. Его также можно рассматривать как внешнюю ветку для текущего репозитория. Копия вашего открытого репозитория на Гитхабе может быть сделана любым пользователем, после чего он может прислать изменения в ваш репозиторий через пулреквест.

Клонирование (Clone) — скачивание репозитория с удалённого сервера на локальный компьютер в определённый каталог для дальнейшей работы с этим каталогом как с репозиторием.

Ветка (Branch) — это параллельная версия репозитория. Она включена в этот репозиторий, но не влияет на главную версию, тем самым позволяя свободно работать в параллельной. Когда вы внесли нужные изменения, то вы можете объединить их с главной версией.

Мастер (Master) или Мейн (Main) — главная или основная ветка репозитория.

Термины(2)

Коммит (Commit) — фиксация изменений или запись изменений в репозиторий. Коммит происходит на локальной машине.

Пул (Pull) — получение последних изменений с удалённого сервера репозитория.

Пуш (Push) — отправка всех неотправленных коммитов на удалённый сервер репозитория.

Мёрдж (Merge) — слияние изменений из какой-либо ветки репозитория с любой веткой этого же репозитория. Чаще всего слияние изменений из ветки репозитория с основной веткой репозитория.

Самый верный способ обучиться владению Git — испытать его
своими руками.

learngitbranching.js.org