

Version Control System

Иван Домашних

Сложности разработки

Сложности разработки

Совместная разработка

- Передача изменений другим
- Объединение изменений

Куча разнообразного функционала

- Что и зачем было сделано
- Параллельная разработка функционала
- Исправление багов в релизе

Ненужные изменения

- Риск неудачного эксперимента
- Риск оставить «хаки» и «хлам»
- Случайные баги

Совместная разработка

Передача изменений другим

ZIP-архив на флешке или по email

Объединение изменений

Устное описание изменений и ручное объединение

Куча разнообразного функционала

Что и зачем было сделано

Хорошая память, комментирование кода

Параллельная разработка функционала

Последующее мучительное объединение изменений

Исправление багов в релизе

Исправление багов в релизе и актуальной версии

Ненужные изменения

Риск неудачного эксперимента

Ручной бэкап

Риск оставить «хаки» и «хлам»

Помнить о всех «хаках» и «хламе»

Случайные баги

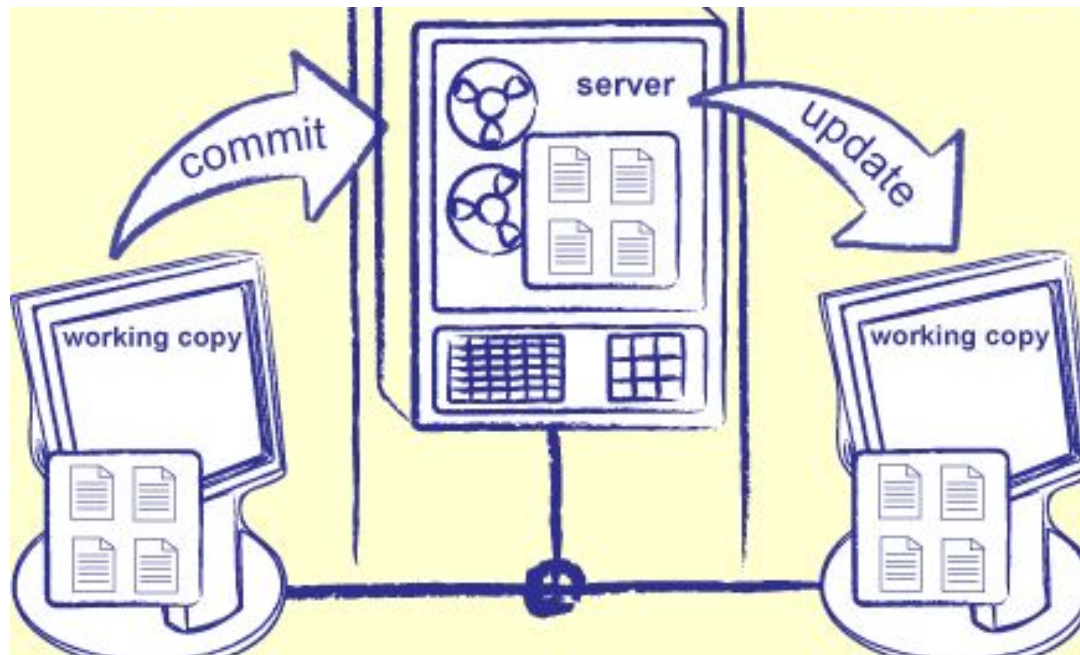
Аккуратность и внимание

Система контроля версий

Система контроля версий

VCS – Version Control System

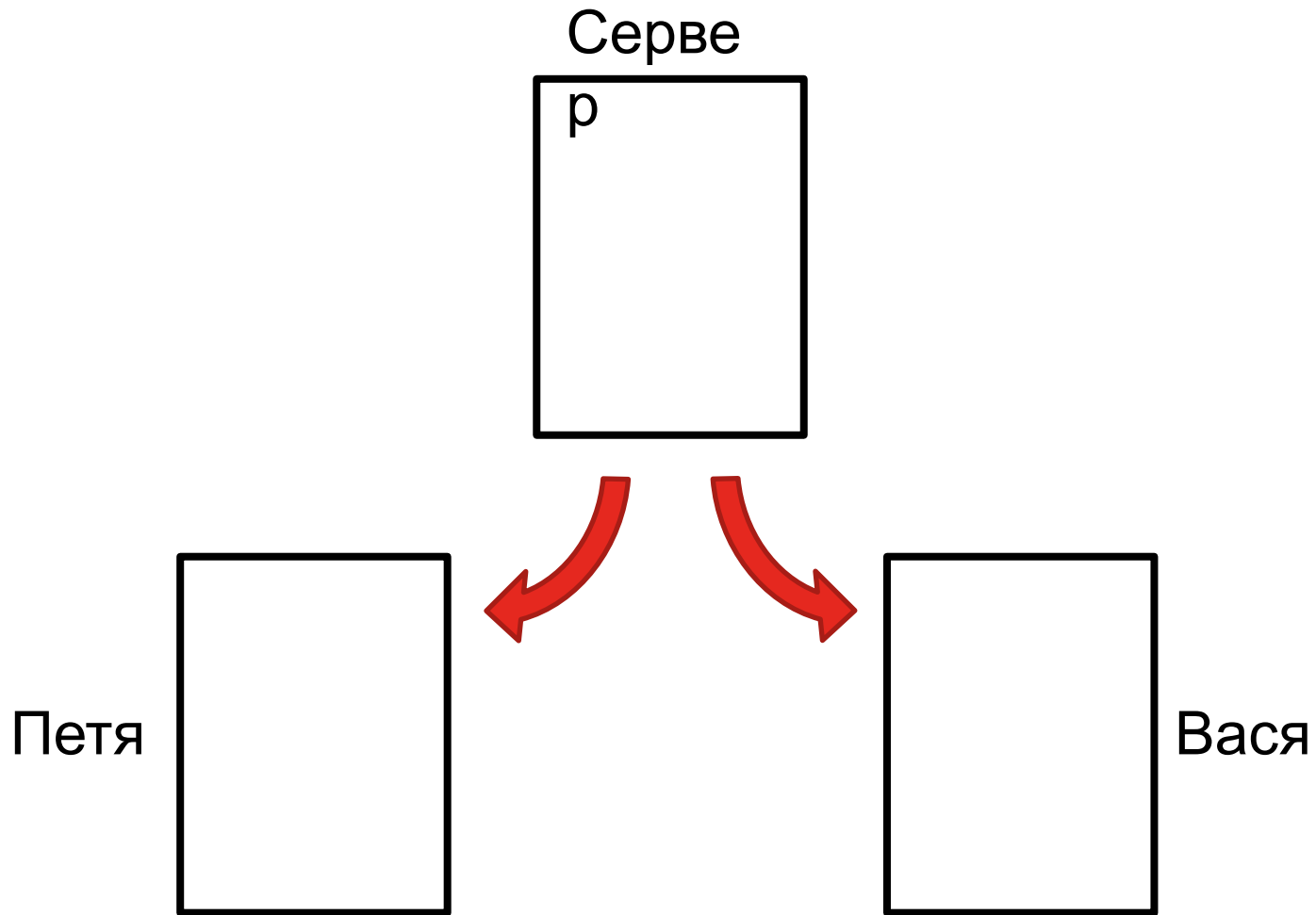
SCM – Source Code Management



Главная идея

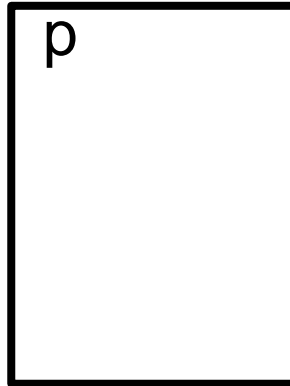
Храним не файлы,
а изменения

Совместное редактирование

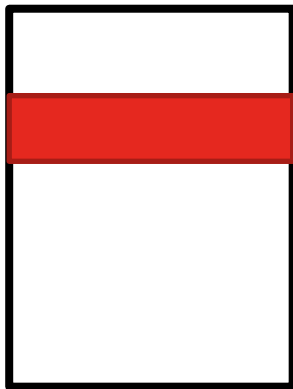


Совместное редактирование

Серве



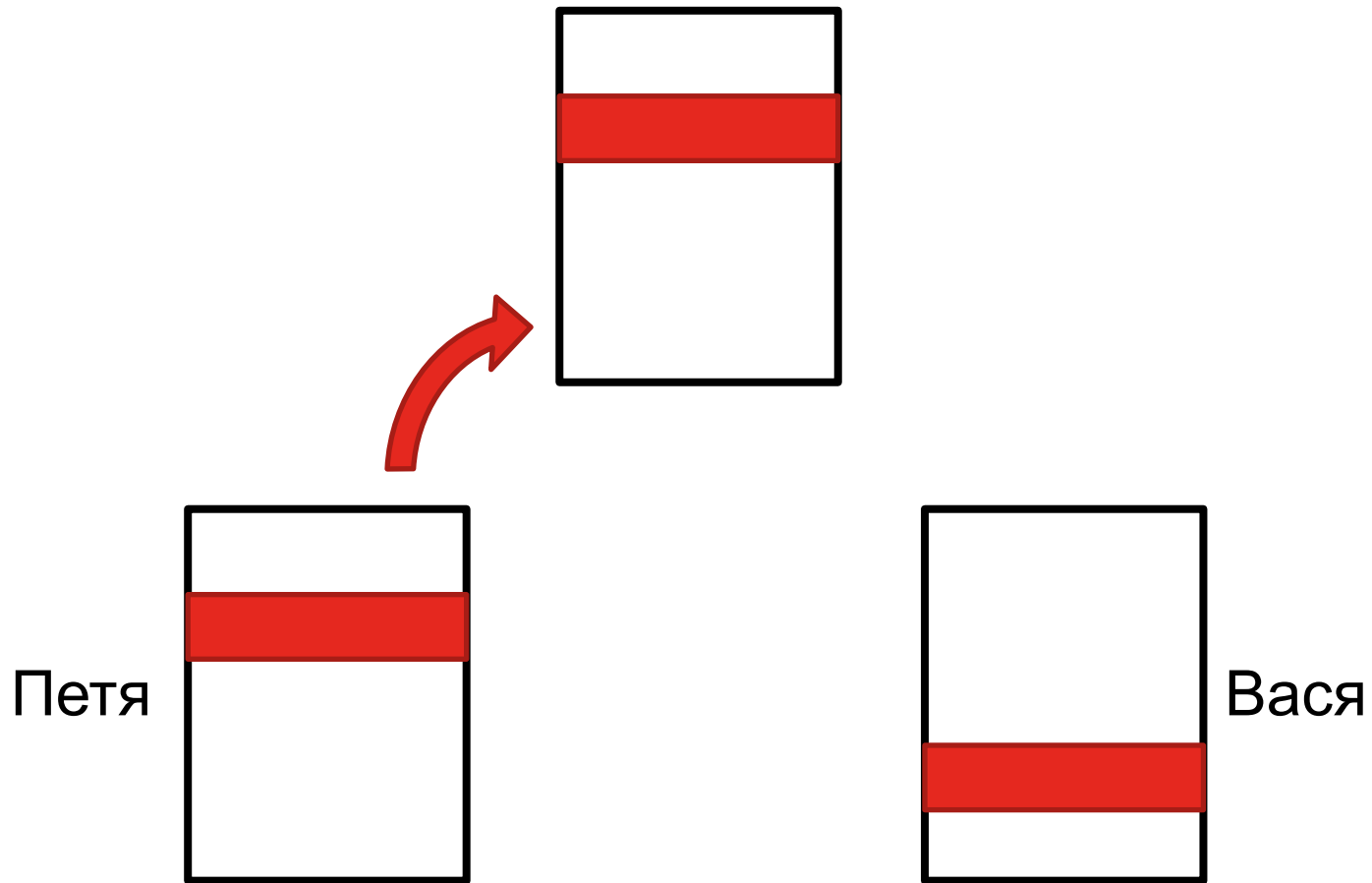
Петя



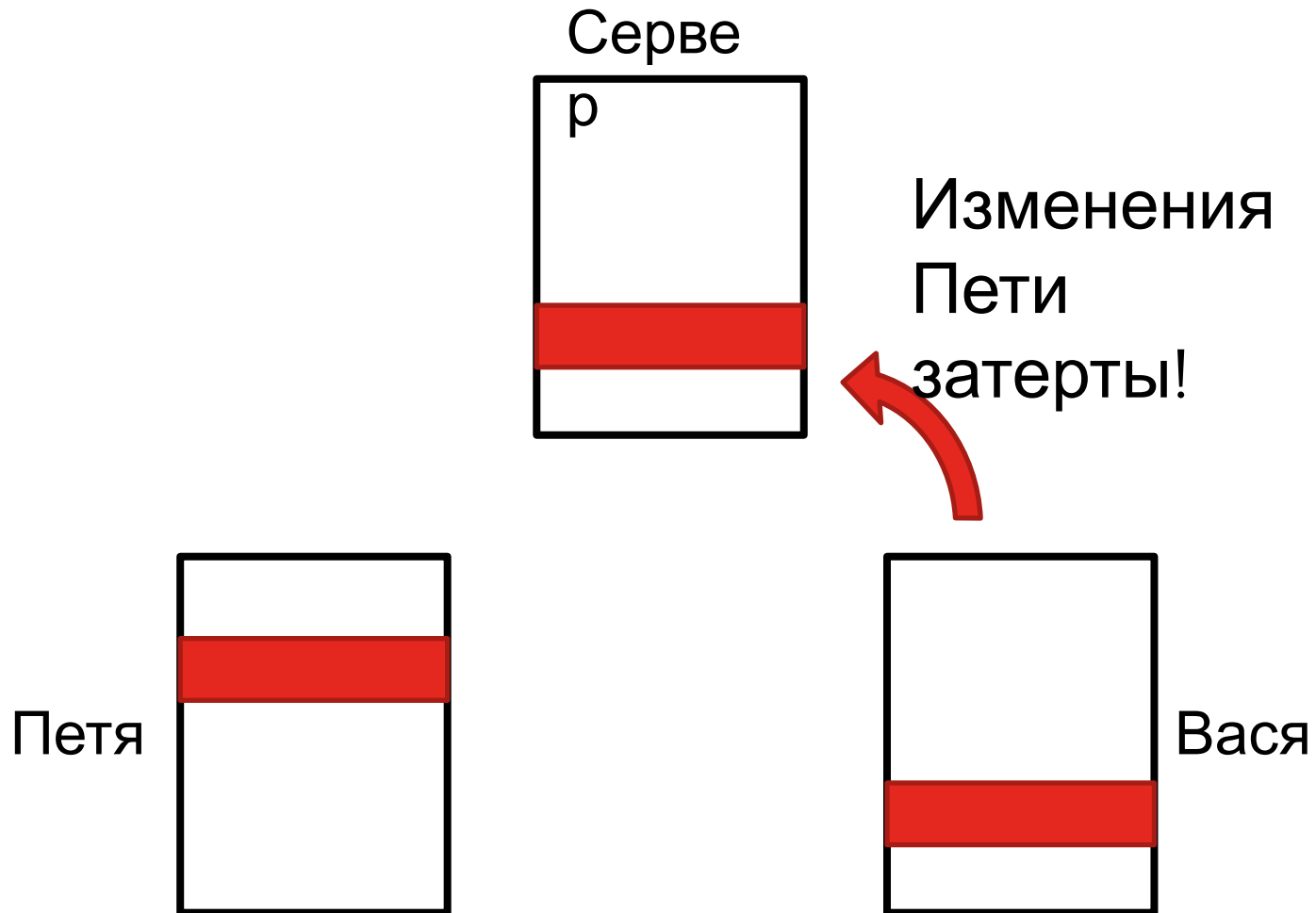
Вася



Совместное редактирование



Совместное редактирование



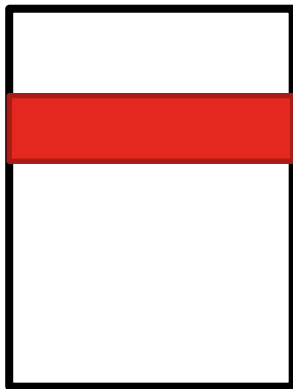
Совместное редактирование

Серве



Должно было
быть так!

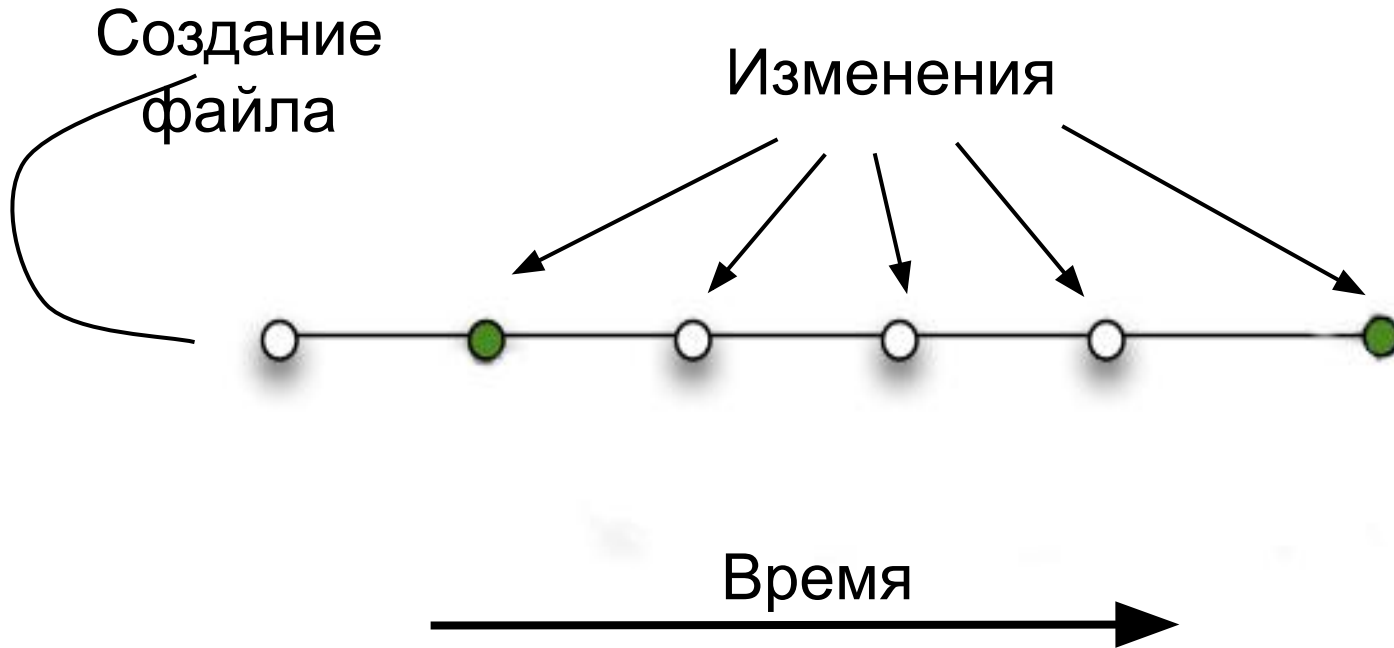
Петя



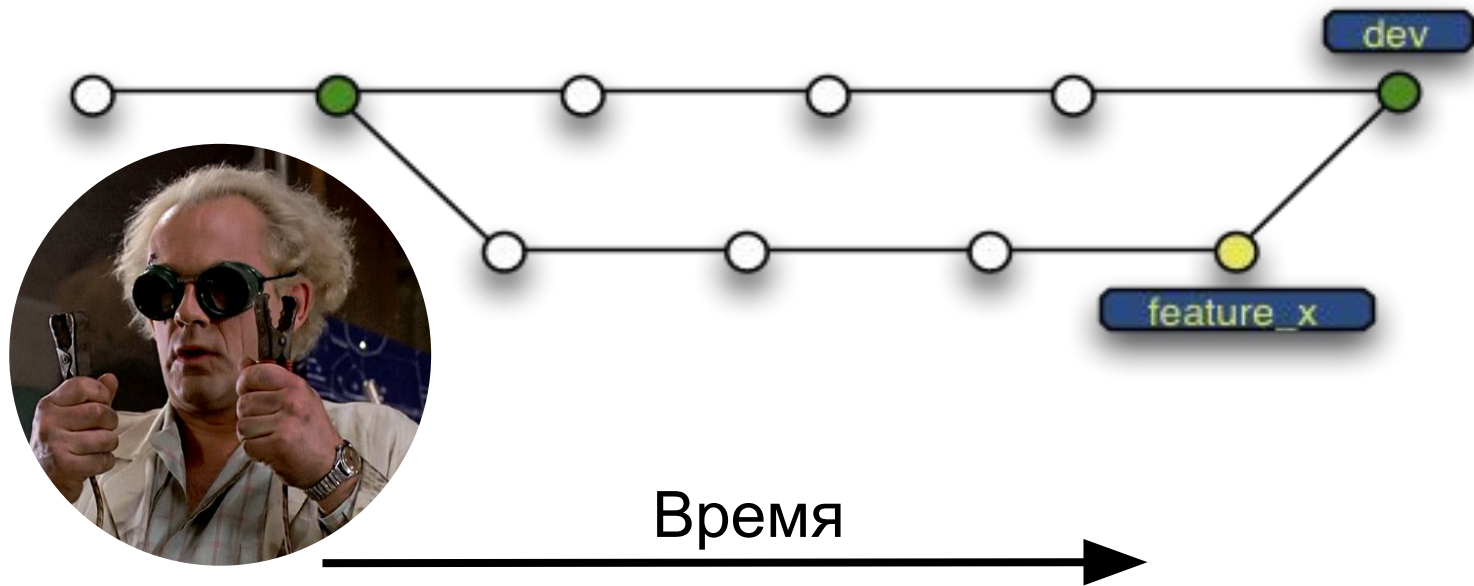
Вася



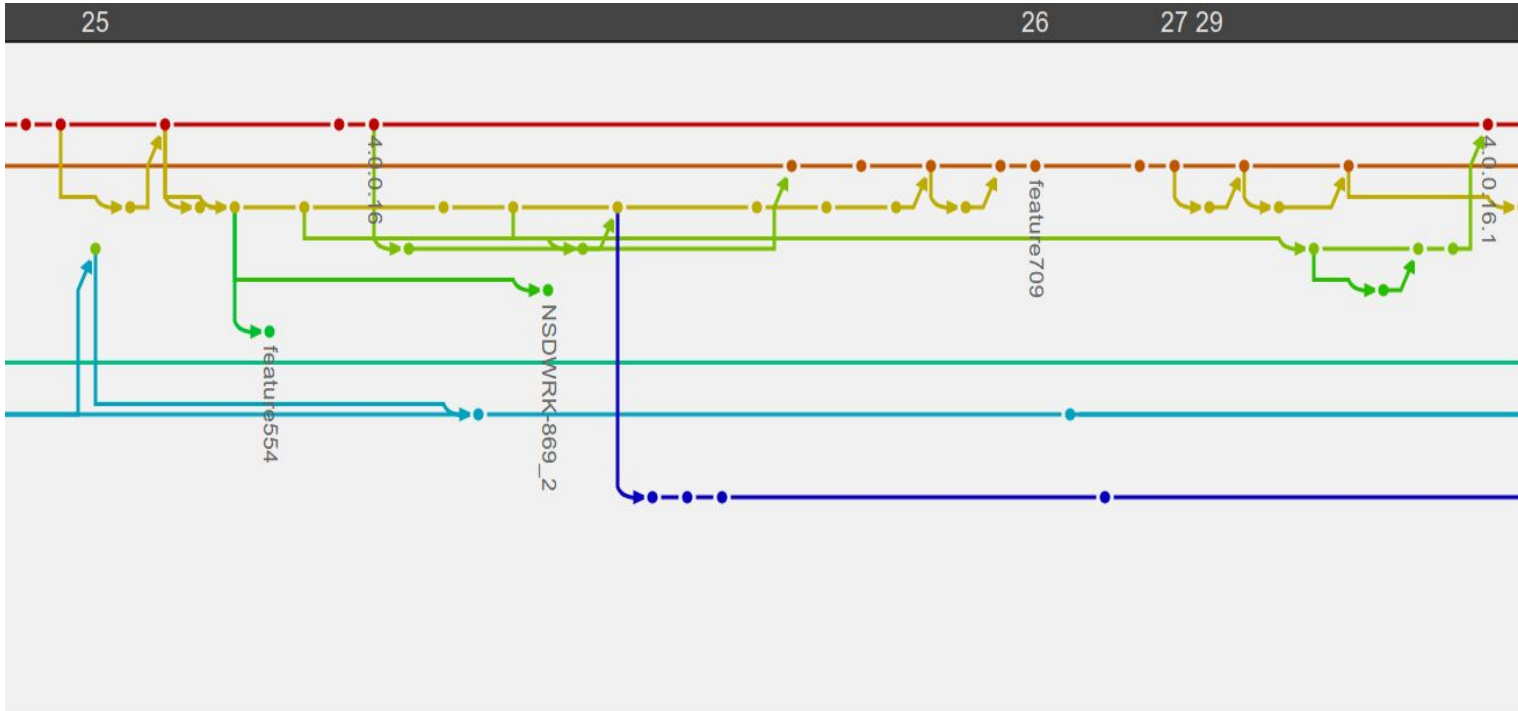
Последовательность изменений



Ответвления и слияния



История изменений



Составные части

Repository – репозиторий хранит историю всех изменений

Working directory – можно получить рабочую копию на любой момент

Класификация VCS

CVS

SVN

Fossil

TFS

Bazaar

Git

Perforce

Veracity

Mercurial

Классификация

Локальные

- Тулзы для сравнения файлов

Централизованные

- Visual SourceSafe
- Subversion – SVN

Распределенные

- Git
- Mercurial – Hg

Наш выбор – Git

Распределенный

- Каждому по репозиторию

Поддерживается

- Есть в популярных IDE
e.g. Visual Studio, WebStorm
- Есть online-репозитории
e.g. GitHub, GitLab, BitBucket

Дает понимание

- Hg аналогичен
- SVN после Git тривиален



ИТОГИ

Совместная разработка

Передача изменений другим

ZIP-архив на флешке или по email

Единое место хранения всех изменений

Объединение изменений

Устное описание изменений и ручное объединение

Автоматическое объединение в большинстве случаев

Куча разнообразного функционала

Что и зачем было сделано

Хорошая память, комментирование кода

История всех изменений с описанием и авторством

Параллельная разработка функционала

Последующее мучительное объединение изменений

Переключение на нужную ветку в истории

Исправление багов в релизе

Исправление багов в релизе и актуальной версии

Возможность слияния/копирования изменений

Ненужные изменения

Риск неудачного эксперимента

Ручной бэкап

Возможность отката изменений

Риск оставить «хаки» и «хлам»

Помнить о всех «хаках» и «хламе»

Просмотр изменений перед их публикацией

Случайные баги

Аккуратность и внимание

Просмотр изменений в любой момент

VCS – комфорт от которого не отказаться



Вопросы?