

# Управление конфигурацией ПО (часть 1)

Алексей Островский

Физико-технический учебно-научный центр НАН Украины

30 апреля 2015 г.

# Управление конфигурацией ПО

## Определение

**Конфигурация** — совокупность функциональных и/или физических характеристик оборудования (*hardware*), прошивок (*firmware*) и ПО (*software*), определенных в технической документации и реализованных программным продуктом.

## Определение

**Управление конфигурацией** (англ. *software control management*) — процесс идентификации конфигурации программной системы в заданные моменты времени с целью:

- ▶ систематического контроля изменений конфигурации;
- ▶ поддержки целостности (*integrity*) и отслеживаемости (*traceability*) конфигурации на протяжении жизненного цикла ПО.

# Зачем нужно управление конфигурацией?

## **Организация и хранение изменений, вносимых в систему:**

- ▶ причина (исправление ошибок, адаптация, расширение функциональности);
- ▶ локализация изменений (измененные, добавленные, или удаленные файлы);
- ▶ отслеживание авторства;
- ▶ доступность актуальных версий компонентов для всей команды разработки.

## **Контроль целостности системы:**

- ▶ согласование изменений, вносимых различными разработчиками;
- ▶ поддержка совместимых версий отдельных компонентов системы;
- ▶ определение корректной версии компонентов для выпусков.

# Составляющие управления конфигурацией

## **Управление изменениями** (англ. *change management*):

- ▶ отслеживание и сохранение запросов на изменение от пользователей и разработчиков;
- ▶ оценка затрат и объема вносимых изменений;
- ▶ составление плана внесения изменений.

## **Управление версиями** (англ. *version management*):

- ▶ отслеживание различных версий компонент программной системы;
- ▶ предотвращение и разрешение конфликтов при внесении изменений в компоненты.

(Рассматриваются в этой лекции.)

# Составляющие управления конфигурацией (продолжение)

## **Построение программной системы** (англ. *system building*):

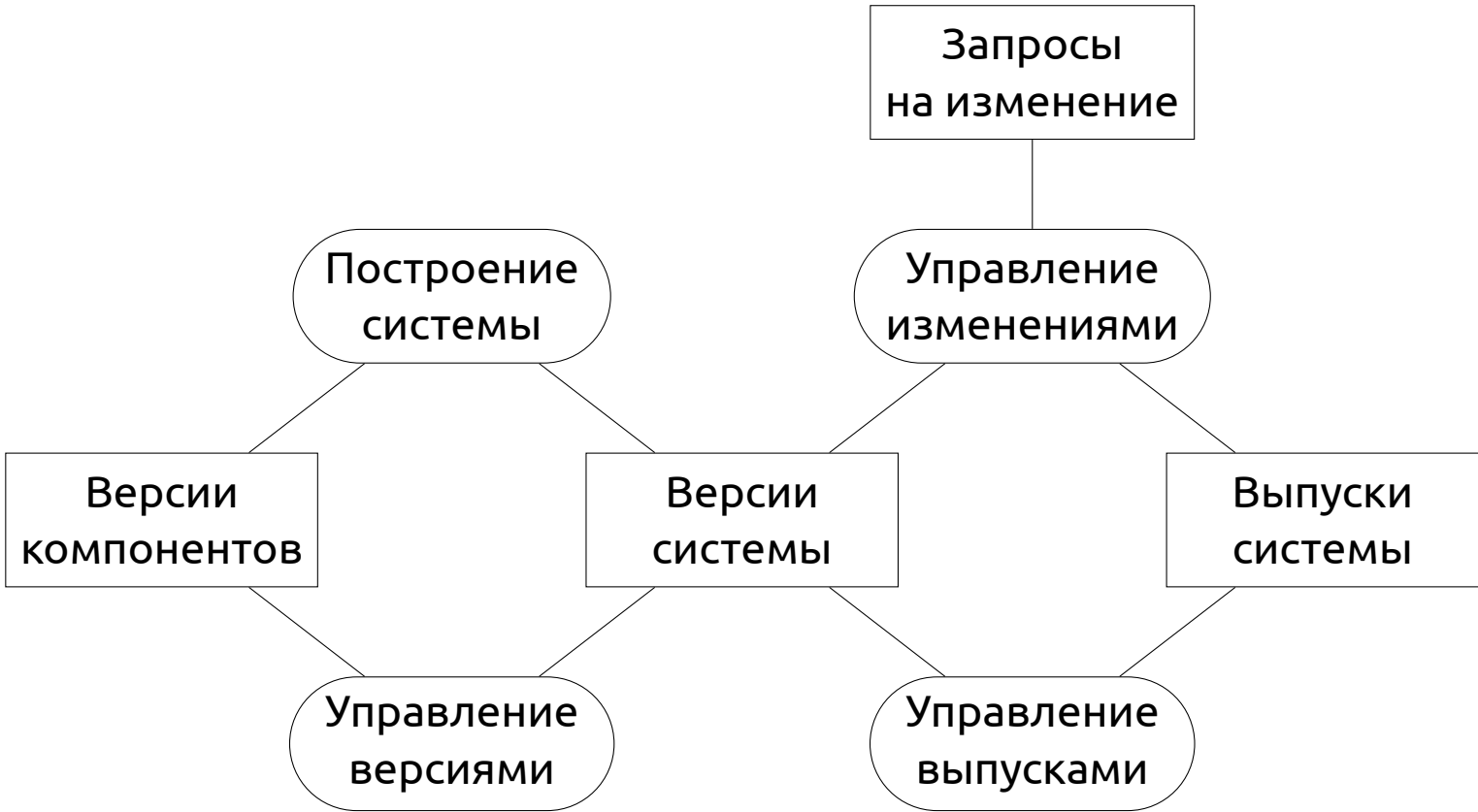
- ▶ организация программных компонент, данных и используемых библиотек для построения;
- ▶ компиляция и сборка (англ. *linking*) для создания выполняемых программ;
- ▶ конфигурация режима построения для создания семейства программных систем.

## **Управление выпусками** (англ. *release management*):

- ▶ подготовка системы для выпуска для использования вне отдела разработки;
- ▶ контроль версий выпущенных компонентов.

(Рассматриваются в следующей лекции.)

# Составляющие управления конфигурацией



Взаимоотношения между аспектами управления конфигурацией

# Идентификация конфигурации ПО

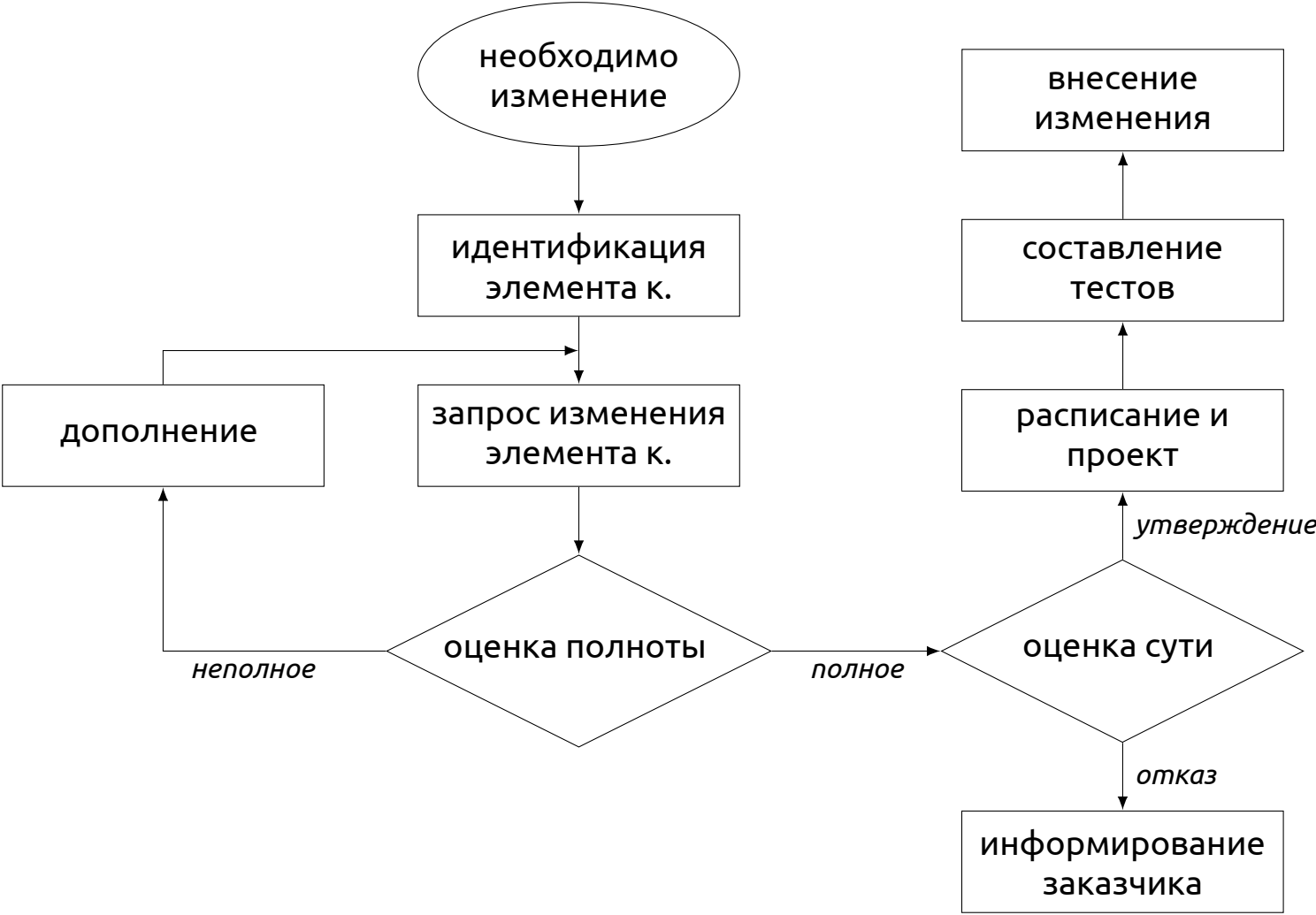
## Этапы идентификации конфигурации:

1. идентификация контролируемых элементов;
2. создание схем идентификации для объектов и их версий;
3. определение инструментов и методов для получения и управления контролируемыми элементами.

## Элементы конфигурации:

- ▶ исполняемый и исходный код;
- ▶ планы;
- ▶ спецификации и проектная документация;
- ▶ система тестирования;
- ▶ программные инструменты;
- ▶ сторонние библиотеки;
- ▶ справочная документация.

# Управление изменениями



**Управление изменениями** — деятельность по координации и оценке реализованных изменений элементов конфигурации, а также их утверждению или отбрасыванию.



# Запросы на изменение

## Содержание запроса (англ. *issue ticket*):

- ▶ дата, номер;
- ▶ отправитель запроса;
- ▶ суть проблемы, возможный набросок ее решения;
- ▶ ответственный(е) за рассмотрение и имплементацию;
- ▶ характеристики (приоритет, затронутые компоненты, теги, ...);
- ▶ комментарии, оценки разработчиков и т. п.

## Организация запросов:

- ▶ изолированная система: клиент / сервер или распределенная (напр., Bugzilla);
- ▶ часть системы управления проектом, напр., с помощью веб-интерфейса (GitHub, Google Code, ...).

# Оценка запросов на изменение

## Валидность изменения:

- ▶ Не является ли изменение результатом недопонимания системы потребителем?
- ▶ Не реализована ли требуемая функциональность?
- ▶ Был ли аналогичный запрос зарегистрирован ранее?

## Полнота запроса:

- ▶ Следует ли объединить запрос с аналогичными или расширить?
- ▶ Следует ли разбить запрос на составляющие?
- ▶ Достаточно ли четка формулировка?

# Оценка запросов на изменение (продолжение)

## Приоритет запроса:

- ▶ важность внесения изменений (критическая ошибка vs мелкое исправление, напр., интерфейса пользователя);
- ▶ круг потребителей, которых затрагивает изменение;
- ▶ затраты на внесение изменения;
- ▶ интеграция изменения в цикл выпусков программной системы.

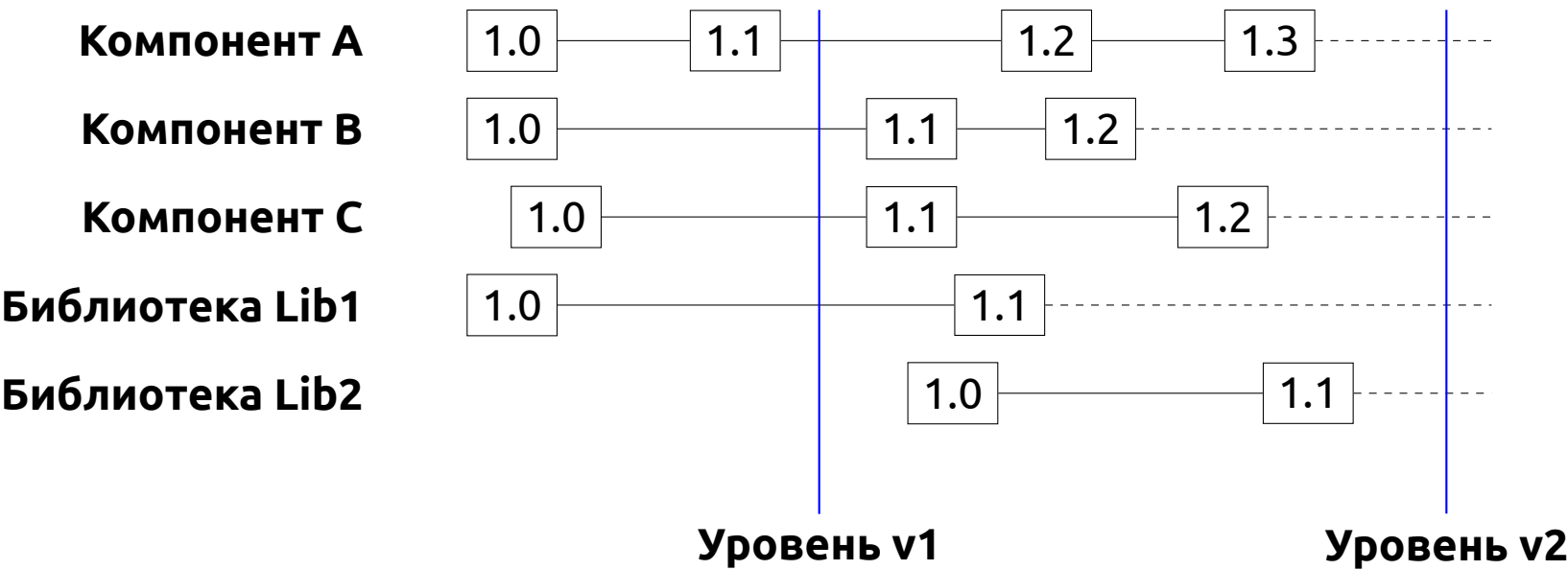
## Реализация запроса:

- ▶ затронутые компоненты системы и план внесения изменений;
- ▶ влияние на функциональность (не нарушает ли изменение требования к системе? не снижает ли оно нефункциональные характеристики ПО — производительность, безопасность, ...);
- ▶ тесты для проверки корректности изменения.

# Управление версиями

## Определение

**Управление версиями** (англ. *revision control*, *version control*) — аспект управления конфигурацией программного проекта, связанный с хранением, организацией и согласованием изменений, вносимых в данные, относящиеся к проекту.



Управление версиями включает управление уровнями (англ. *baseline*) и версиями компонентов (англ. *codeline*).

# Функциональность систем управления версиями

- ▶ **Идентификация версий** для компонентов программной системы; подбор версий для выпусков системы.
- ▶ **Организация хранения версий** компонентов, в частности, минимизация потребляемой памяти при помощи хранения изменений (delta).
- ▶ Хранение **информации о вносимых изменениях**: авторство, описание, теги, обсуждение и т. д.
- ▶ Обеспечение поддержки **параллельной и распределенной разработки**; разрешение конфликтов при внесении изменений.
- ▶ Поддержка **модульности, зависимостей** между репозиториями.

# Понятия управления версиями

- ▶ **Хранилище** (англ. *repository*) — объект (напр., выделенный сервер), хранящий все версии конфигурируемых документов и данные об их изменениях;
- ▶ **рабочая копия** (англ. *working copy*) — локальная копия документов из хранилища, соответствующих определенному моменту времени; песочница для тестирования изменений.
- ▶ **клонирование** (англ. *check-out, cloning*) — создание локальной копии на основе хранилища;
- ▶ **изменение** (англ. *change, diff, delta*) — атомарная операция, влияющая на конкретный элемент конфигурации (правка, создание, удаление).

## Понятия управления версиями (продолжение)

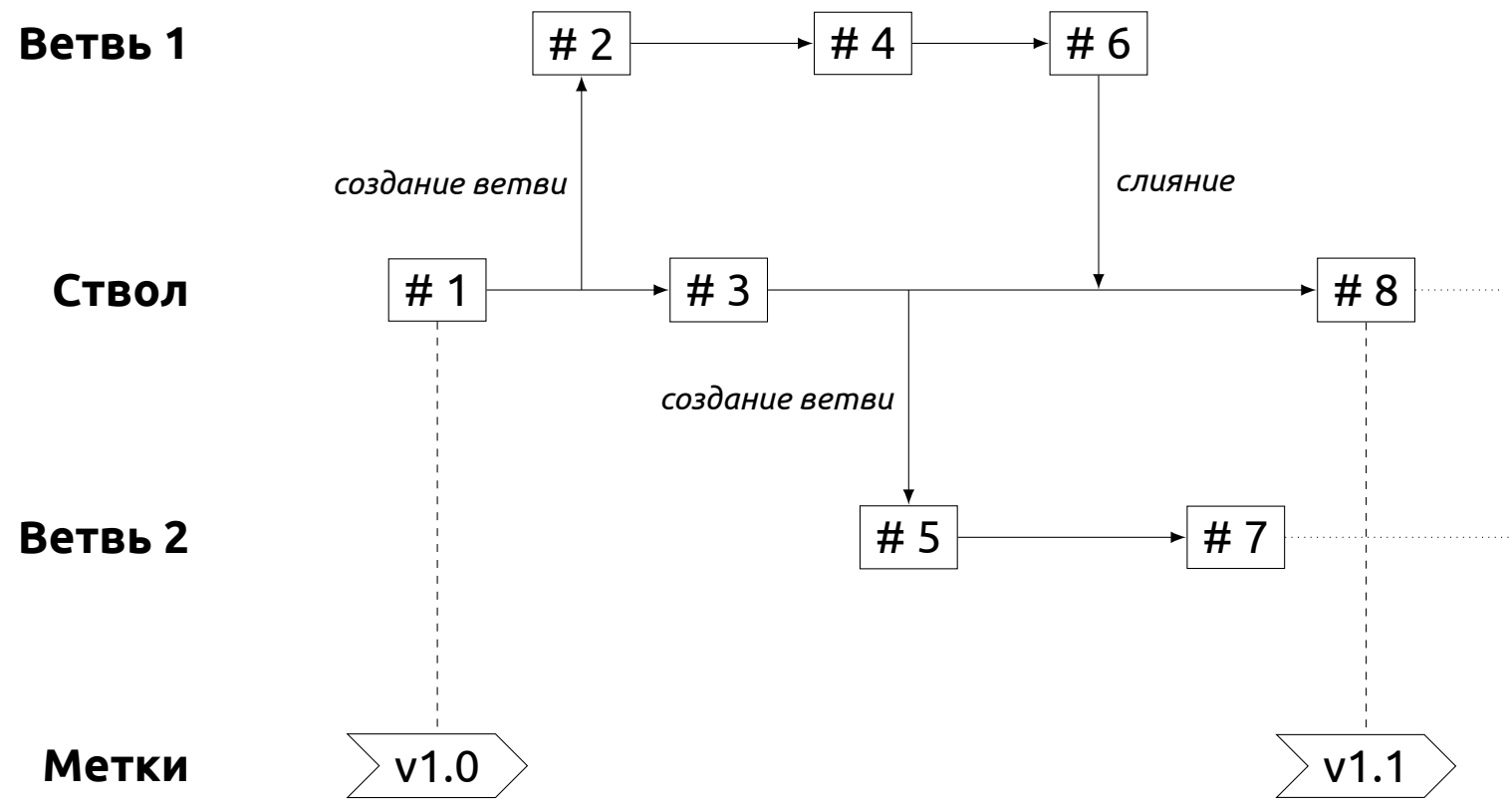
- ▶ **набор изменений** (англ. *change list*) — группа связанных изменений (напр., для решения одной проблемы);
- ▶ **фиксация** (англ. *commit, check-in*) — отправка набора изменений в хранилище для постоянного хранения;
- ▶ **слияние** (англ. *merge*) — согласование двух наборов изменений, касающихся одних и тех же документов;
- ▶ **конфликт** (англ. *conflict*) — противоречие между двумя наборами изменений при слиянии, которое должно решаться вручную разработчиком.

## Понятия управления версиями (продолжение)

- ▶ **ветвь** (англ. *branch*) — копия набора файлов в определенный момент времени, обеспечивающая возможность независимого направления разработки (напр., внедрение нового компонента);
- ▶ **ствол** (англ. *trunk, master*) — основная ветвь, соответствующая главному направлению разработки; другие ветви после подготовки вливаются в ствол.
- ▶ **основная версия** (англ. *head*) — самая новая версия ветви или ствола в хранилище ( $\simeq$  указатель на последнюю операцию фиксации для этой ветви);
- ▶ **метка** (англ. *tag, label*) — обозначение для группы логически связанных версий документов в хранилище (напр., версии файлов, входящих в определенный выпуск).

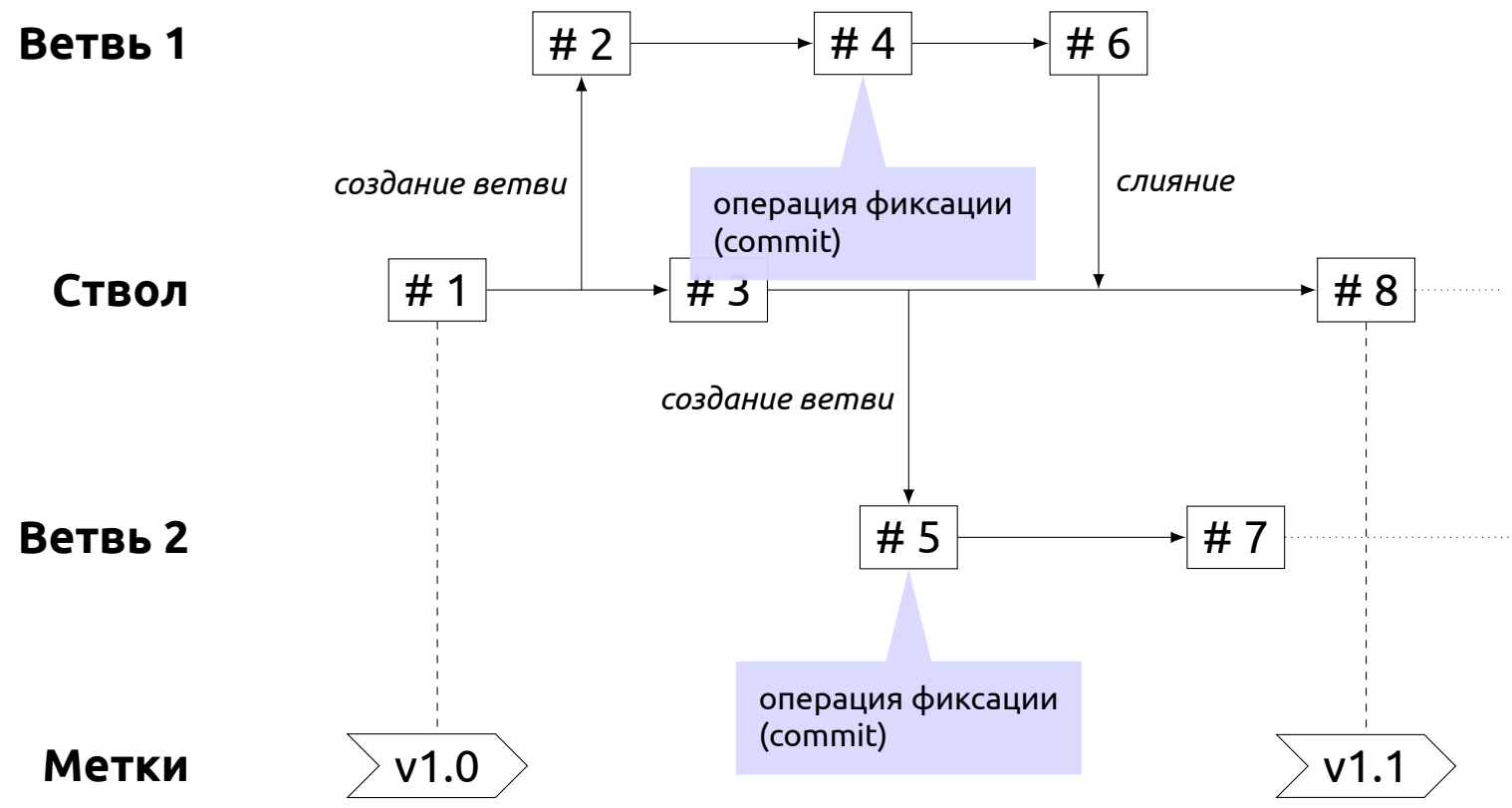


# Образец схемы изменений



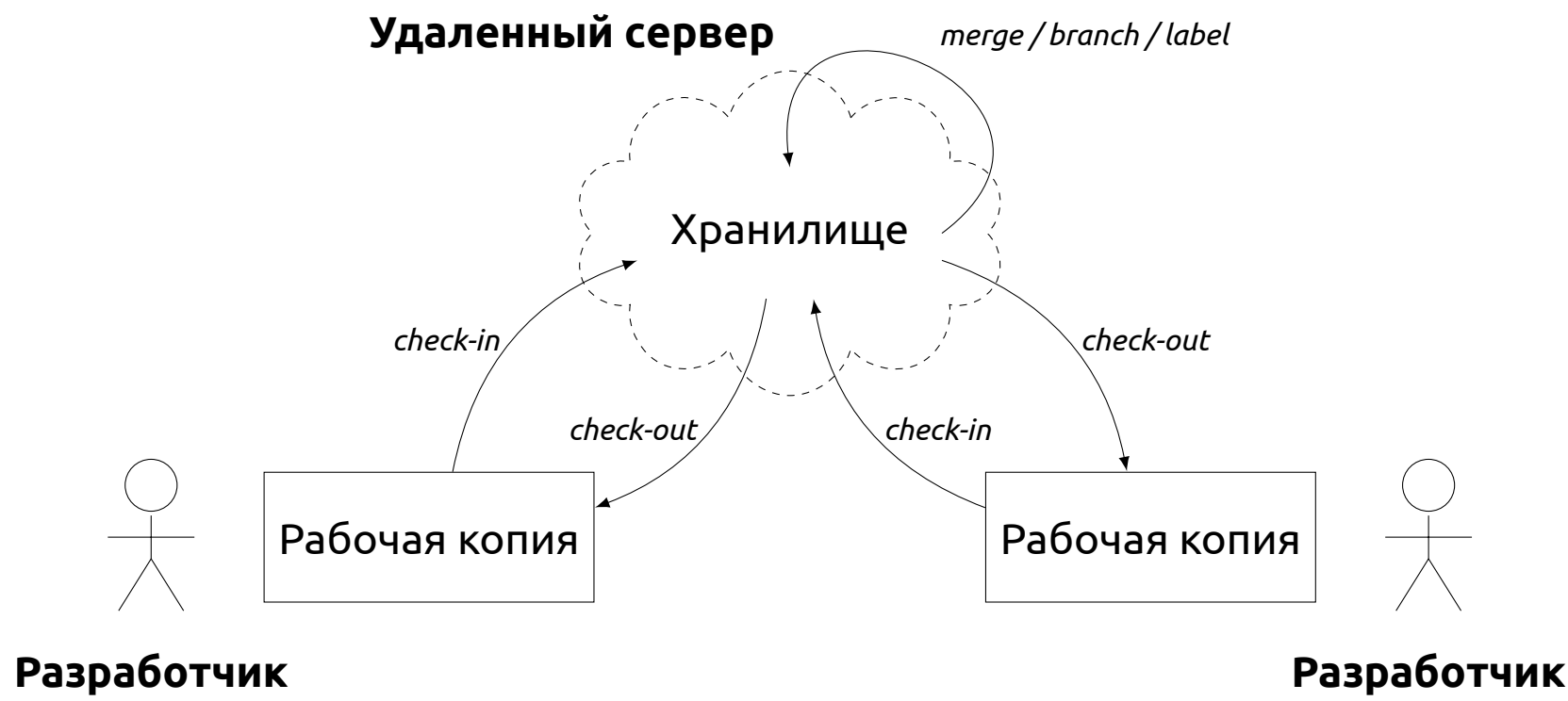
Работа с версиями документов представляется в виде ориентированного ациклического графа

# Образец схемы изменений



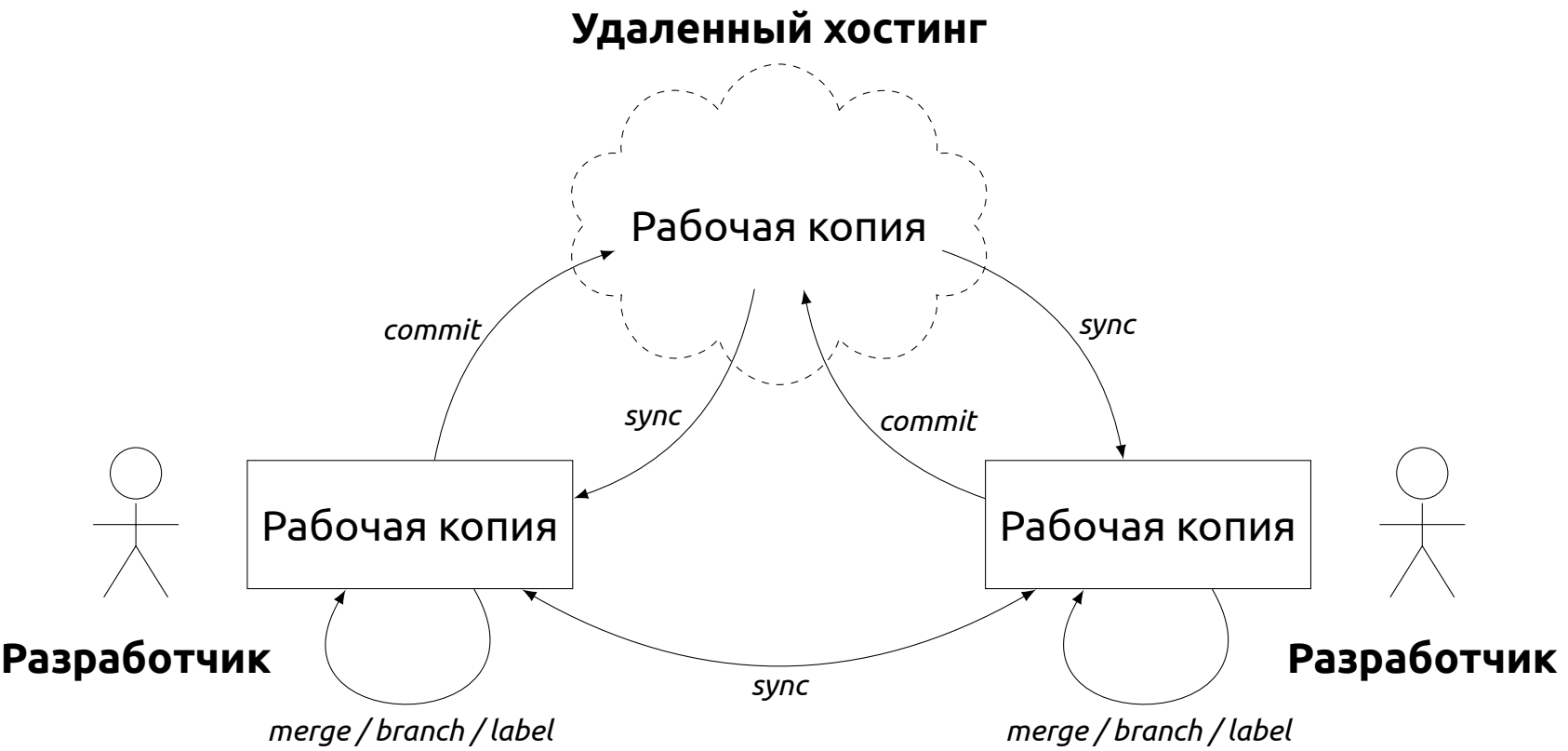
Работа с версиями документов представляется в виде ориентированного ациклического графа

# Организация управления версиями



Организация системы управления версиями с архитектурой «клиент — сервер»

# Организация управления версиями



Организация распределенной системы управления версиями

# Организация управления версиями

## Архитектура «клиент — сервер»:

- ▶ единый сервер, хранящий версии конфигурируемых документов;
- ▶ для операций фиксации и других необходим доступ к серверу через протокол связи (напр., с помощью HTTP).

**Примеры:** CVS, Subversion.

## Распределенное управление версиями:

- ▶ отсутствие выделенного хранилища, равноправность всех рабочих копий;
- ▶ рабочие копии содержат всю информацию о конфигурируемых элементах;
- ▶ операции фиксации, создания ветвей, слияния и т. д. локальны и не требуют соединения с сервером;
- ▶ согласование изменений за счет синхронизации с другими рабочими копиями.

**Примеры:** Git, Mercurial.

# Характеристики СУВ

## Методы разрешения / предотвращения конфликтов:

- ▶ слияние конфликтующих наборов изменений: выбор более подходящей или компромиссной версии каждого файла;
- ▶ замыкание (англ. *locking*) файлов для предотвращения правки несколькими пользователями (невозможно в распределенных системах).

## Метод хранения версий документов:

- ▶ в виде отличий от предыдущей версии (англ. *diff*);
- ▶ в виде снимков (англ. *snapshot*), т. е. файлов целиком;
- ▶ смешанный способ (напр., отличия для текстовых файлов и снимки для двоичных).

## Базовые объекты:

- ▶ отдельные файлы;
- ▶ деревья файлов.

# Характеристики СУБ (продолжение)

## Автоматизация действий:

- ▶ преобразование концов строки (напр.,  $\backslash r \backslash n \rightarrow \backslash n$ );
- ▶ возможность выполнения произвольных операций при фиксации.

**Атомарность операций фиксации** — невозможность частичного применения набора изменений, связанных с операцией.

Поддержка **переименований файлов**; учет переименованных файлов при слиянии.

**Игнорируемые файлы** рабочей копии (определяются по имени).

## Свойства файловой системы:

- ▶ наличие символических ссылок;
- ▶ поддержка имен файлов в кодировке UTF-8.

# Пример СУБ: Git

## Определение

**Git** — система управления версиями, первоначально созданная Л. Торвальдсом для разработки ядра Linux.

### Особенности:

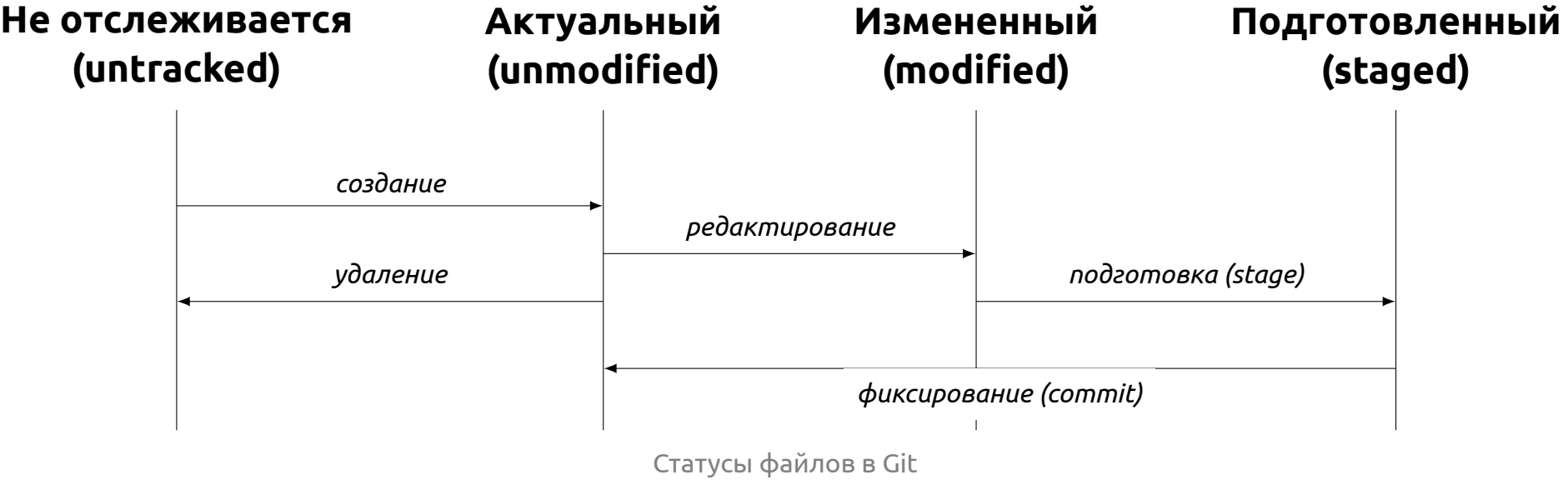
- ▶ распределенная;
- ▶ похожа на файловую систему, в частности, имеет индекс объектов;
- ▶ хранит снимки (т. е. документы полностью) для дерева файлов проекта;
- ▶ ориентирована на использование большого числа ветвей и операций слияния.

### Вспомогательные инструменты:

- ▶ графические оболочки (напр., gitg для GTK+);
- ▶ плагины для сред разработки (напр., EGit для Eclipse);
- ▶ сайты для удаленного хранения репозиториев (напр., GitHub).



# Схема работы с файлами в Git



# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master

software-engineering-lectures / +

Merge pull request #25 from slowli/new-lecture

slowli

 authored 16 hours ago 

latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

<> Code

Issues 1

Pull requests 0

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

You can clone with HTTPS, SSH, or Subversion.

Download ZIP

# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master ▾

software-engineering-lectures / +

Merge pull request #25 from slowli/new-lecture ...

slowli authored 16 hours ago latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

Список файлов  
(для просмотра и редактирования)

Code

Issues 1

Pull requests 0

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

You can clone with [HTTPS](#), [SSH](#), or [Subversion](#).

Download ZIP

# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

операции фиксации в текущей ветви

software-engineering-lectures / +

Merge pull request #25 from slowli/new-lecture

slowli authored 16 hours ago

latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

<> Code

Issues 1

Pull requests 0

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

You can clone with HTTPS, SSH, or Subversion.

Download ZIP

# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master

software-engineering-lectures / +

Merge

slowli/new-lecture

slowli authored 16 hours ago

latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

Code

Issues1

Pull requests0

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

You can clone with HTTPS, SSH, or Subversion.

Download ZIP

выбор ветви

# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master software-engineering-l

Merge pull request #25 from slowli/new-lecture

slowli authored 16 hours ago

latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

Code

Issues1

Pull requests0

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

You can clone with HTTPS, SSH, or Subversion.

Download ZIP

создание и  
скачивание выпусков

# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master software-engineering-lectures / +

Merge pull request #25 from slowli/new-lecture

slowli authored 16 hours ago

latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

<> Code

Issues 1

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

You can clone with HTTPS, SSH, or Subversion.

Download ZIP

работа с запросами  
на изменение

# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master software-engineering-lectures / +

Merge pull request #25 from slowli/new-lecture

slowli authored 16 hours ago latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	6 months ago
08	исправление даты лекции 8	5 months ago
09	мелкие исправления	5 months ago

ссылка для  
клонирования

HTTPS clone URL

https://github.cc

You can clone with HTTPS, SSH, or Subversion.

Download ZIP



# Интерфейс GitHub

Лекции по программной инженерии — Edit

54 commits

1 branch

1 release

1 contributor

branch: master software-engineering-lectures / +

Merge pull request #25 from slowli/new-lecture

slowli authored 16 hours ago

latest commit de8e289667

01	Лекции 1-7	6 months ago
02	изменение размера шрифтов	6 months ago
03	изменение размера шрифтов	6 months ago
04	изменение размера шрифтов	6 months ago
05	укрощение длинной ссылки	5 months ago
06	изменение размера шрифтов	6 months ago
07	изменение размера шрифтов	
08	исправление даты лекции 8	
09	мелкие исправления	5 months ago

<> Code

Issues 1

Pull requests 0

Wiki

Pulse

Graphs

Settings

HTTPS clone URL

https://github.cc

h HTTPS, SSH,


Download ZIP


скачивание основной версии репозитория


# Выводы

1. Управление изменениями и версиями — важная часть управления разработкой ПО, в особенности при коллективной разработке.
2. Движущей силой при модификации ПО являются запросы на изменение (change requests). Инструменты для управления запросами могут быть самостоятельными приложениями либо встраиваться в более общую систему управления разработкой.
3. Системы управления версиями облегчают разработку ПО в нескольких направлениях за счет механизма ветвления / слияния.
4. Одна из наиболее популярных систем управления версиями — Git; для нее существуют веб-хостинги репозиториев, такие как GitHub.

# Материалы

 **Sommerville, Ian**  
Software Engineering.  
  
Pearson, 2011. — 790 p.

 **Лавріщева К. М.**  
Програмна інженерія (підручник).  
  
К., 2008. — 319 с.

 **Eclipse Wiki**  
EGit User Guide.  
[http://wiki.eclipse.org/EGit/User\\_Guide](http://wiki.eclipse.org/EGit/User_Guide)  
  
(руководство по использованию плагина Git для Eclipse)

Спасибо за внимание!