

Git. Просто git.

undev.coursify.ru



Лекция 2.

- Создание Git-репозитория
- Запись изменений в репозиторий
- Просмотр истории коммитов
- Отмена изменений
- Работа с удалёнными реп.
- Работа с метками
- Полезные советы
- Что такое ветка?
- Основы ветвления и слияния
- Управление ветками
- Приёмы работы с ветками
- Удалённые ветки
- Перемещение
- Итоги

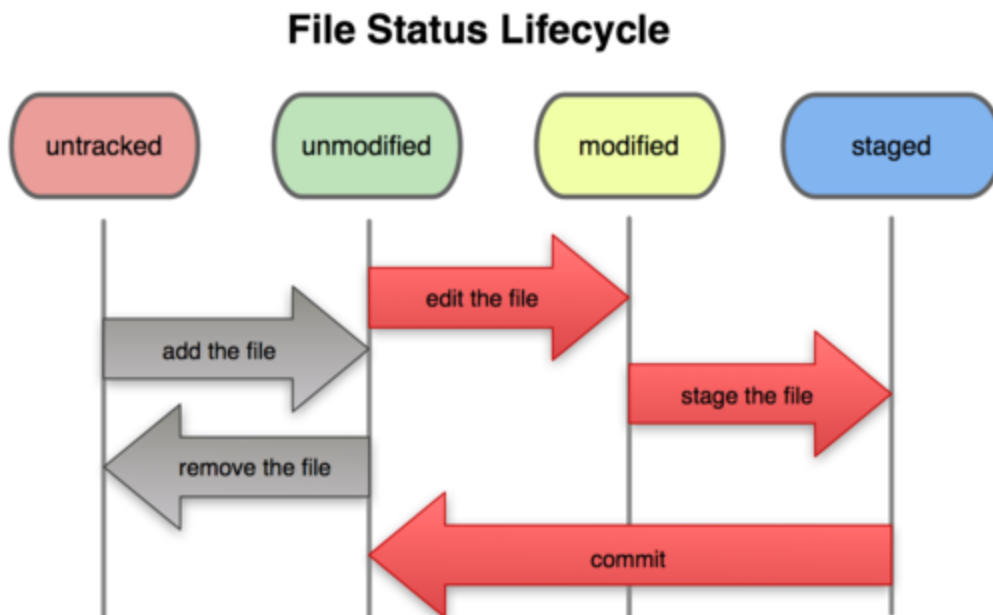
Создание git репозитория

- \$ cd some-project-path
- \$ git init
- \$ git add *.c
- \$ git add README
- \$ git add .
- \$ git commit

Клонирование существующего репозитория

1. \$ `git clone [options] [--] repo [dir]`
2. \$ `git clone git://github.com/libgit2/rugged.git`
3. \$ `git clone git://github.com/libgit2/rugged.git myrugged`

Запись изменений в репозиторий



Определение состояния файлов

\$ `git status`

- Untracked
- Modified
- Added to commit

Отслеживание и индексация файлов

- `git add`
- `git add -a`

Игнорирование файлов

- .gitignore

Просмотр индексированных и неиндексированных изменений

1. `git status`
2. `git diff`
3. `git diff --staged`
4. `git diff file`
5. `git diff revision`

Фиксация изменений

1. `$ git commit`
2. `$ git commit -m "Commit message"`

1. `$EDITOR`
2. `$ git config --global core.editor`

Удаление файлов

- `rm file.txt`
- `git rm file.txt`
- `git rm -f file.txt`
- `git rm --cached file.txt`

Перемещение файлов

1. `git mv file_from file_to`

1. `mv README.txt README`

2. `git rm README.txt`

3. `git add README`

Просмотр истории коммитов

1. `git log`
2. `git log -p -2`
3. `git log -p --word-diff`
4. `git log --stat`
5. `git log --pretty=oneline`
6. `git log --pretty=format:"%h - %an, %ar : %s"`
7. `git log --graph`
8. `git log --since=2.weeks`

Изменение последнего коммита

`git commit --amend`

1. `git commit -m 'initial commit'`
2. `git add forgotten_file`
3. `git commit --amend`

Отмена индексации файла

`git reset`

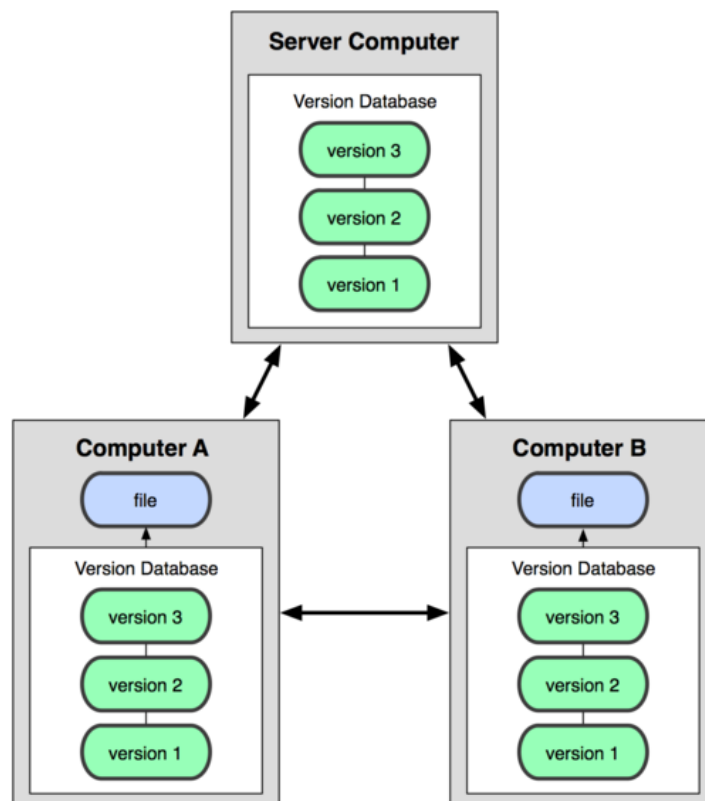
1. `git add .`
2. `git status`
3. `git reset HEAD benchmarks.rb`
4. `git status`

Отмена изменений файла

`git checkout`

1. `git add .`
2. `git status`
3. `git checkout -- benchmarks.rb`
4. `git status`

Работа с удаленными репозиториями



Отображение удалённых репозиториев

1. `git clone git://github.com/schacon/ticgit.git`
2. `cd ticgit`
3. `git remote`
4. `git remote -v`

Добавление удалённых репозиториев

1. `git clone git://github.com/schacon/ticgit.git`
2. `cd ticgit`
3. `git remote`
4. `git remote add pb git://github.com/paulboone/ticgit.git`
5. `git remote -v`

Fetch и Pull

git fetch [имя удал. сервера]

1. git fetch origin
2. git pull origin
3. git pull --rebase origin

Push

```
git push [удал. сервер] [ветка]
```

Инспекция удалённого репозитория

- `git remote show origin`

Удаление и переименование удалённых репозиториев

1. `git remote rename pb paul`
2. `git remote`
3. `git remote rm paul`
4. `git remote`

Просмотр меток

1. `git tag`
2. `git tag -l 'v1.4.*'`

Создание меток

Аннотированные метки

1. `git tag -a v1.4 -m 'my version 1.4'`
2. `git tag`

`git show`

Создание меток

Подписанные метки

1. `git tag -s v1.4 -m 'my version 1.4'`
2. `git tag`

`git show`

Создание меток

Легковесные метки

1. `git tag v1.4`
2. `git tag`

`git show v1.4-lw`

Верификация меток

1. `git tag -v v1.4.2.1`

Выставление меток позже

```
git tag -a v1.2 -m 'version 1.2' 9fceb02
```

Обмен метками

1. `git push origin v1.5`
2. `git push origin --tags`

Полезные советы

Автоматическое дополнение

- `echo "source ~/.git-completion.bash" >> ~/.bashrc`

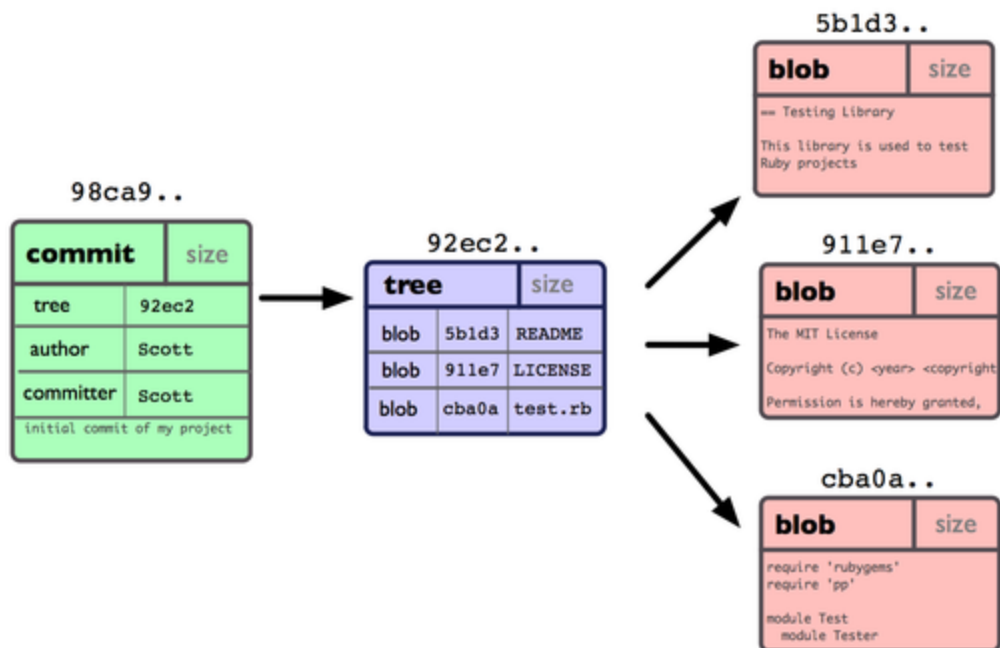
Полезные советы

Псевдонимы

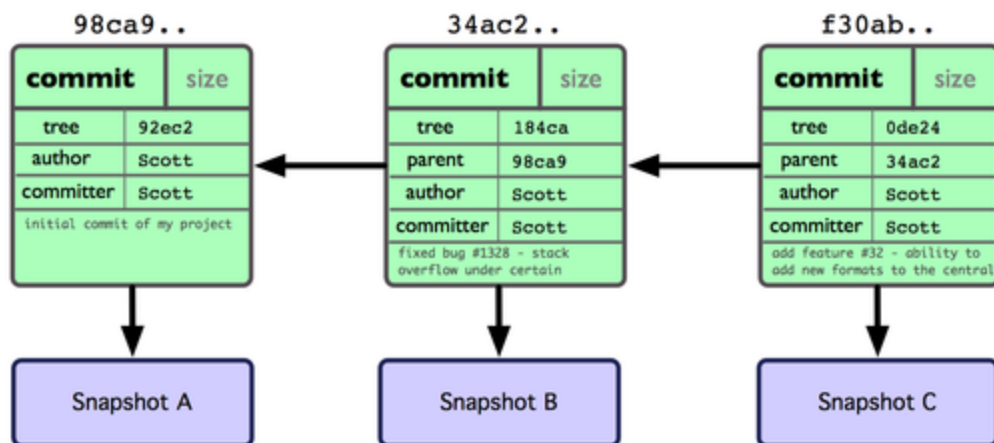
- `$ git config --global alias.co checkout`
- `$ git config --global alias.br branch`
- `$ git config --global alias.ci commit`
- `$ git config --global alias.st status`

Ветвление в Git

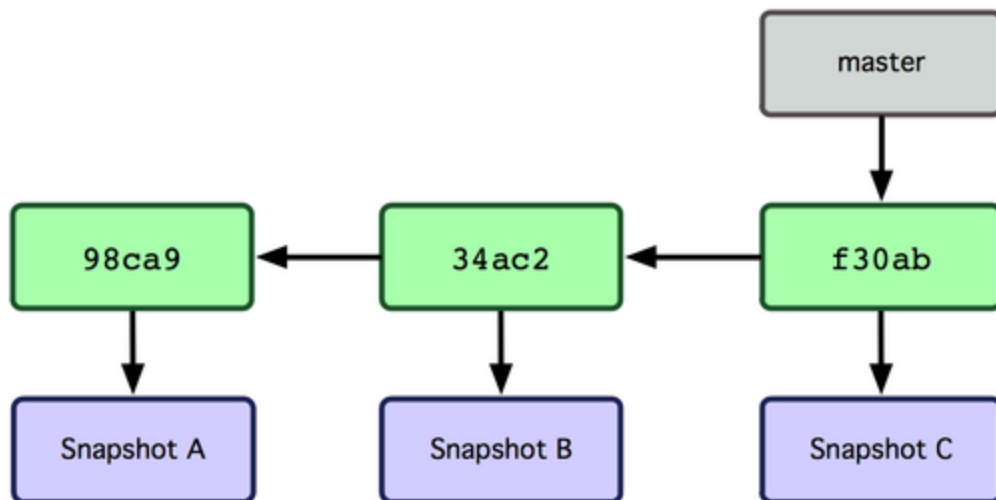
Что такое ветка?



Что такое ветка?

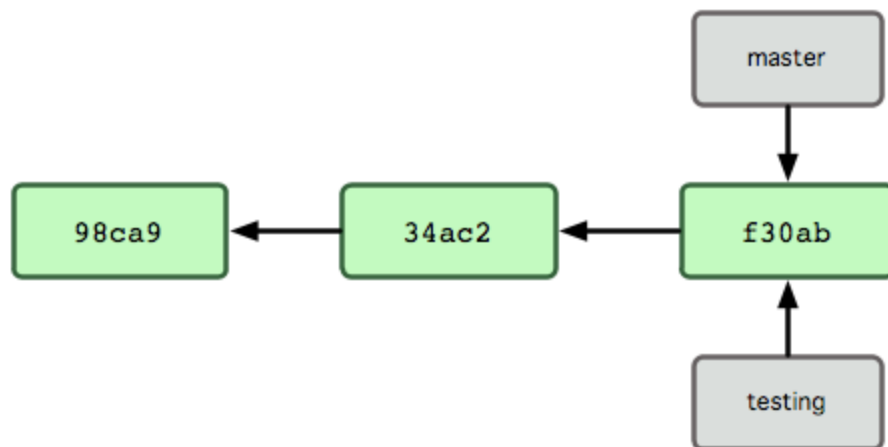


Что такое ветка?

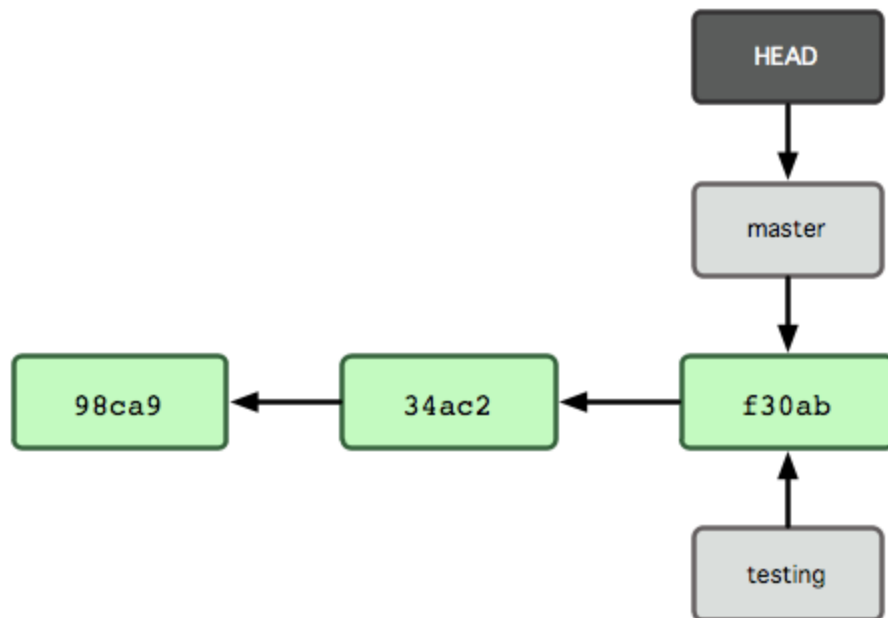


Что такое ветка?

\$ git branch testing

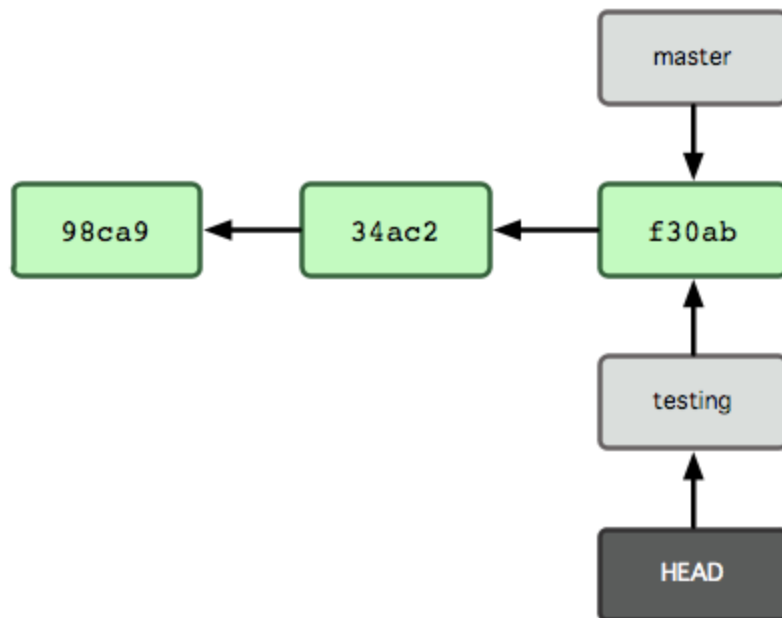


Что такое ветка?

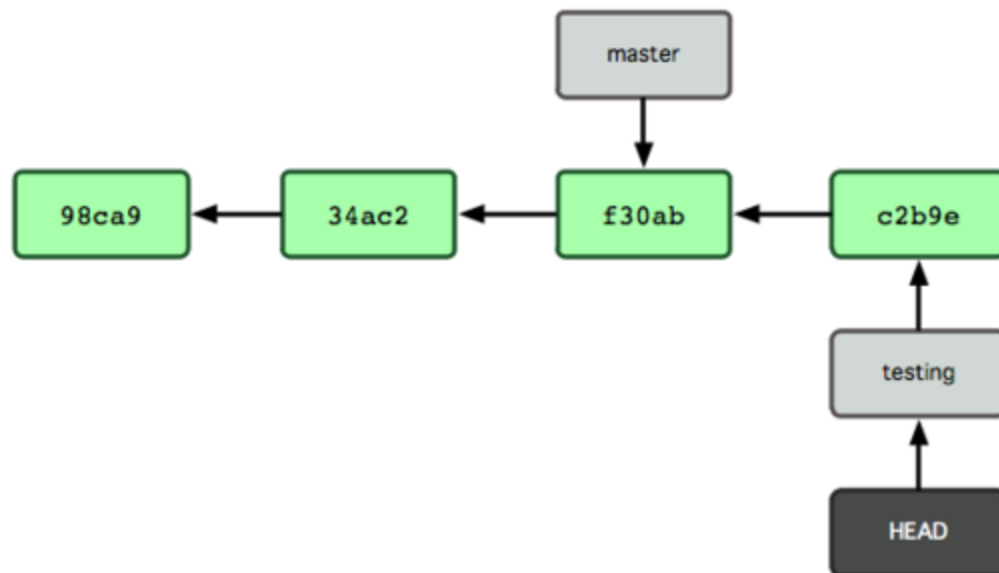


\$ git checkout testing

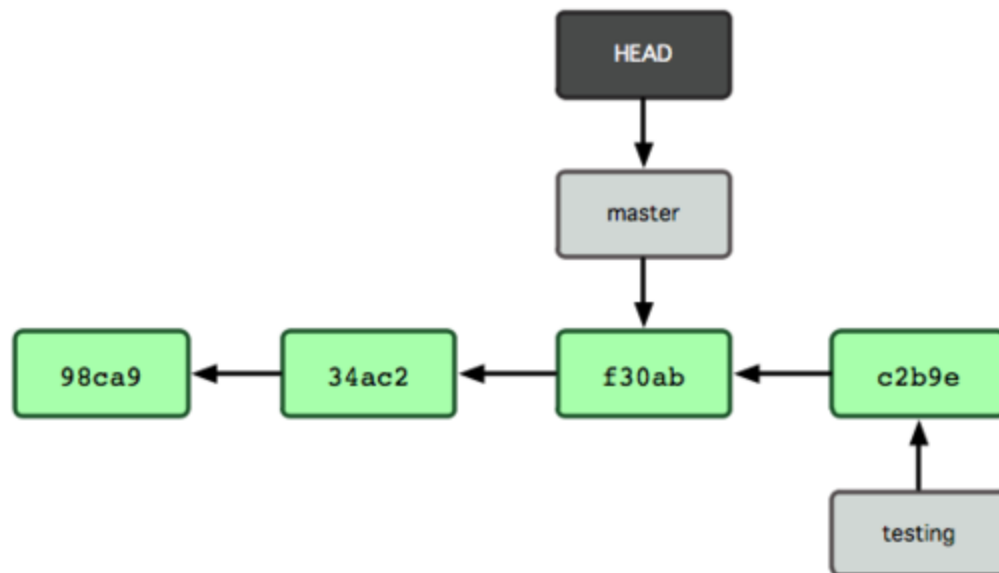
Что такое ветка?



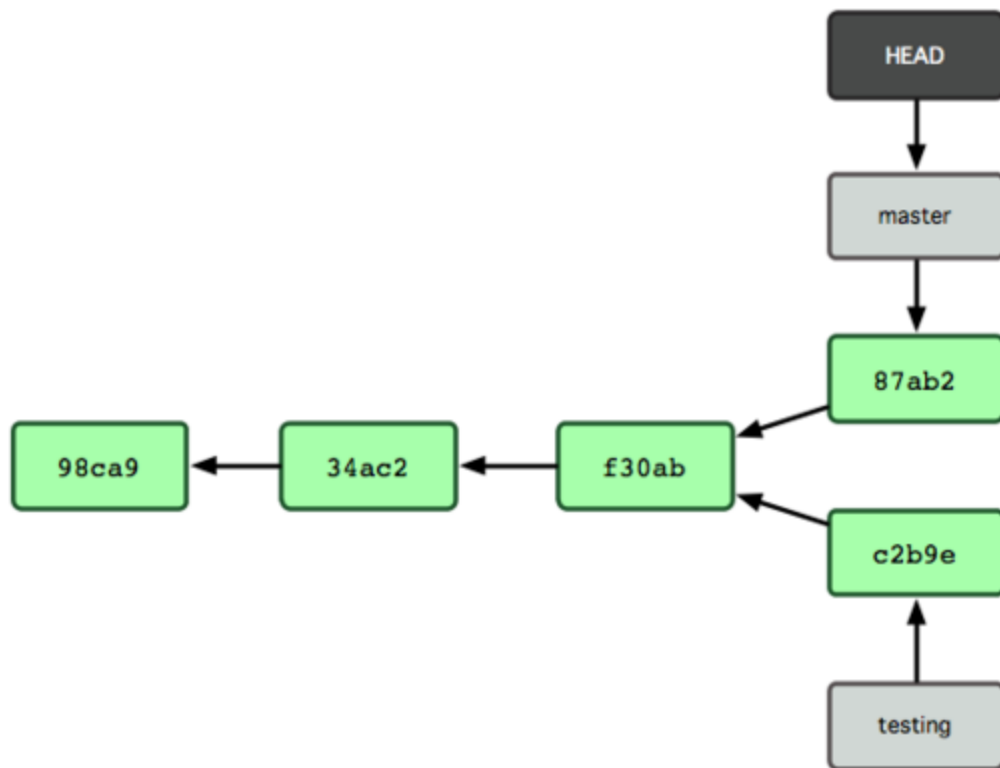
Что такое ветка?



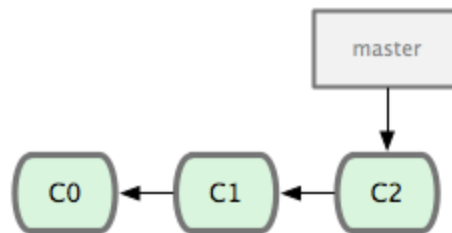
Что такое ветка?



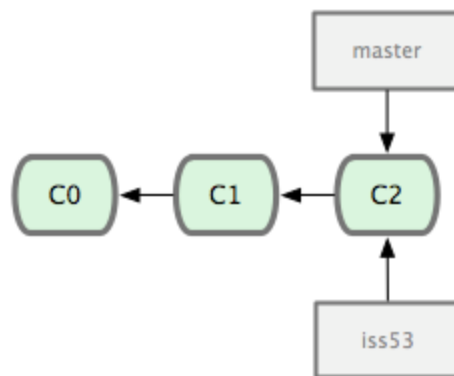
Что такое ветка?



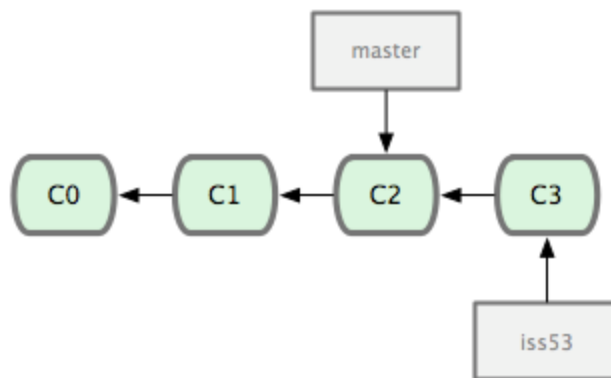
Основы ветвления



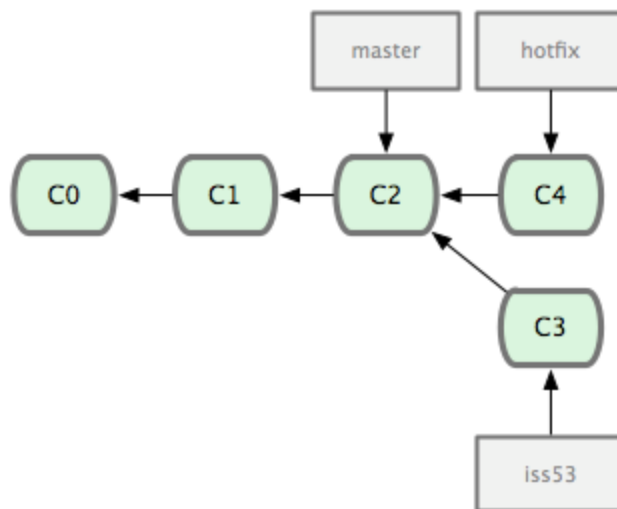
Основы ветвления



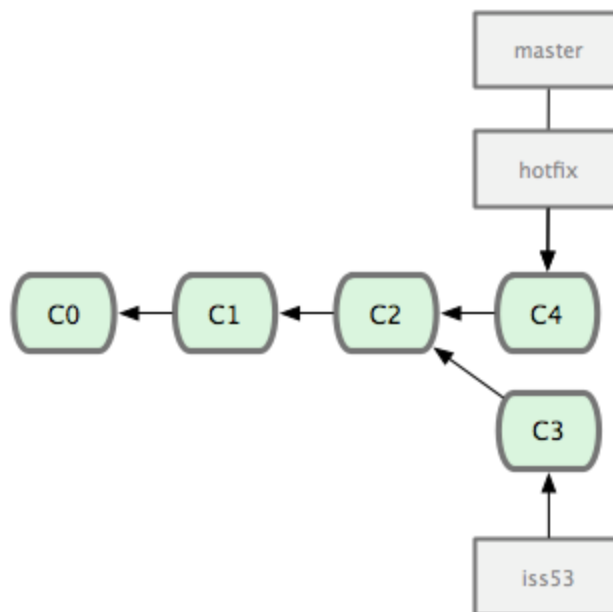
Основы ветвления



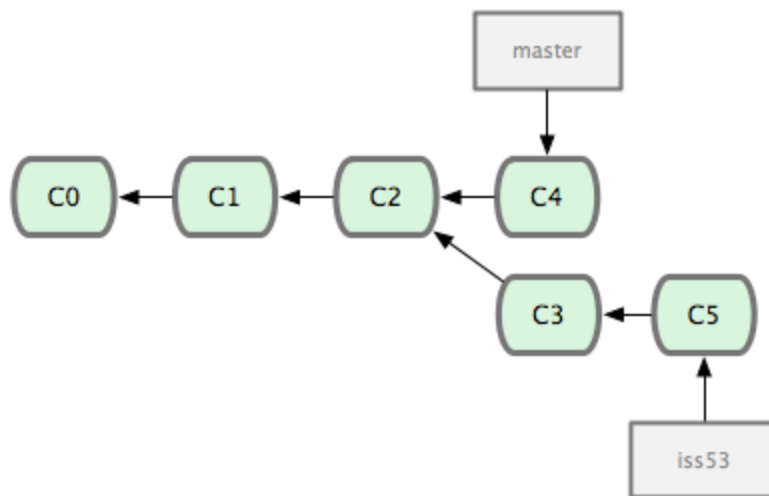
Основы ветвления



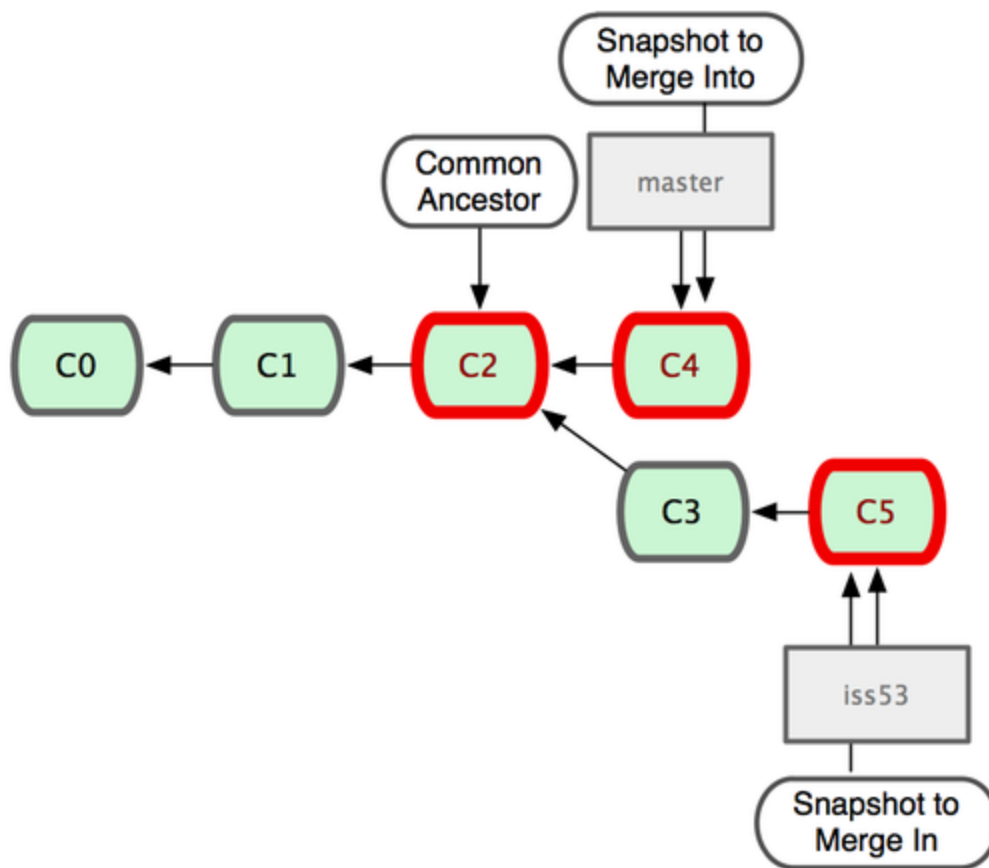
Основы ветвления



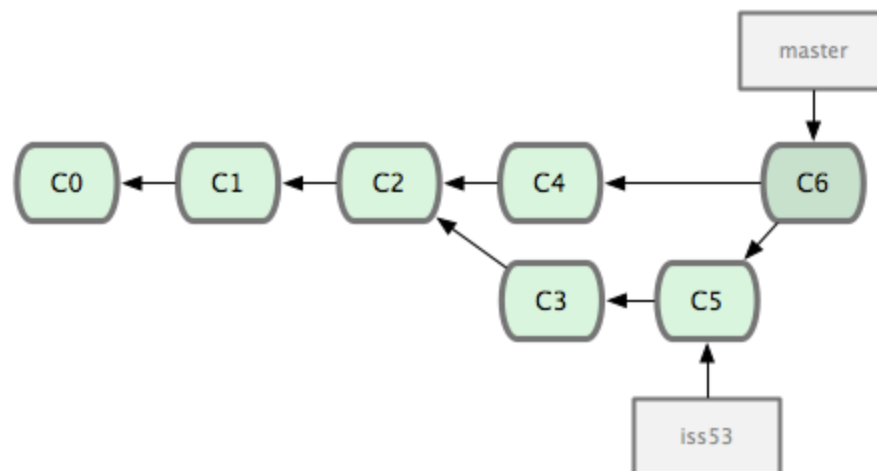
Основы ветвления



Основы слияния



Основы слияния



BOOM

Конфликты

```
$ git merge iss53
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
```

Конфликты

```
[master*]$ git status
index.html: needs merge
# On branch master
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#   unmerged:   index.html
#
```

Конфликты

```
<<<<<<< HEAD:index.html
<div id="footer">contact : email.support@github.com</div>
=====
<div id="footer">
  please contact us at support@github.com
</div>
>>>>>>> iss53:index.html
```

Конфликты

```
<div id="footer">  
  please contact us at email.support@github.com  
</div>
```

Конфликты

```
$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#   modified:   index.html
#
```


Конфликты

```
Merge branch 'iss53'

Conflicts:
  index.html
#
# It looks like you may be committing a MERGE.
# If this is not correct, please remove the file
# .git/MERGE_HEAD
# and try again.
#
```

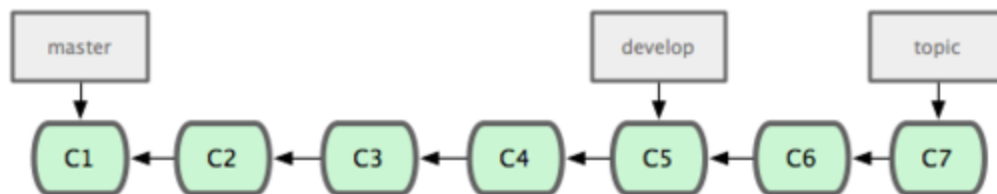
Управление ветками

1. `git branch`
2. `git branch -v`
3. `git branch --merged` (`--no-merged`)
4. `git branch -d branch_name`

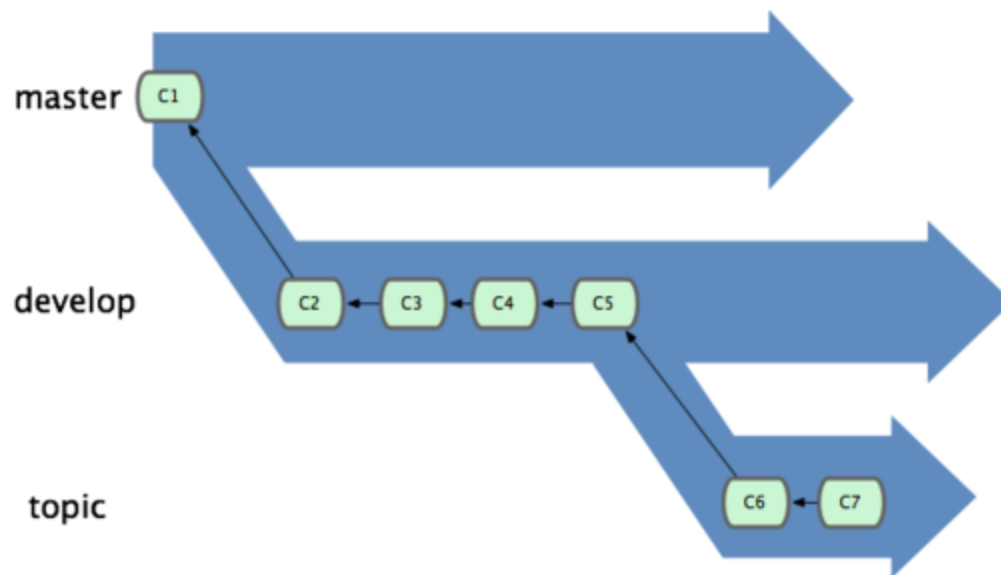
Приёмы работы с ветками

1. Долгоживущие ветки
2. Тематические ветки

Долгоживущие ветки



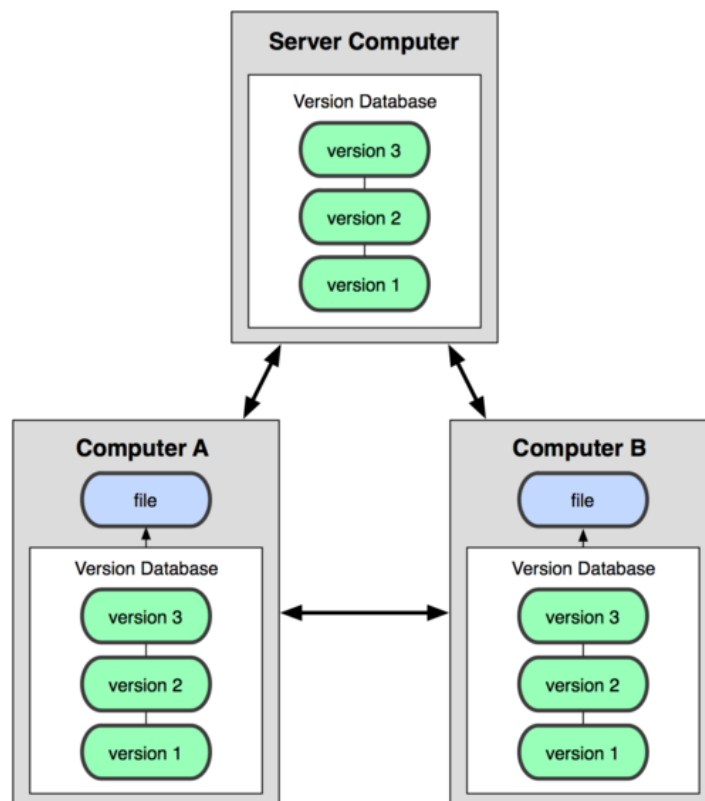
Долгоживущие ветки



Тематические ветки

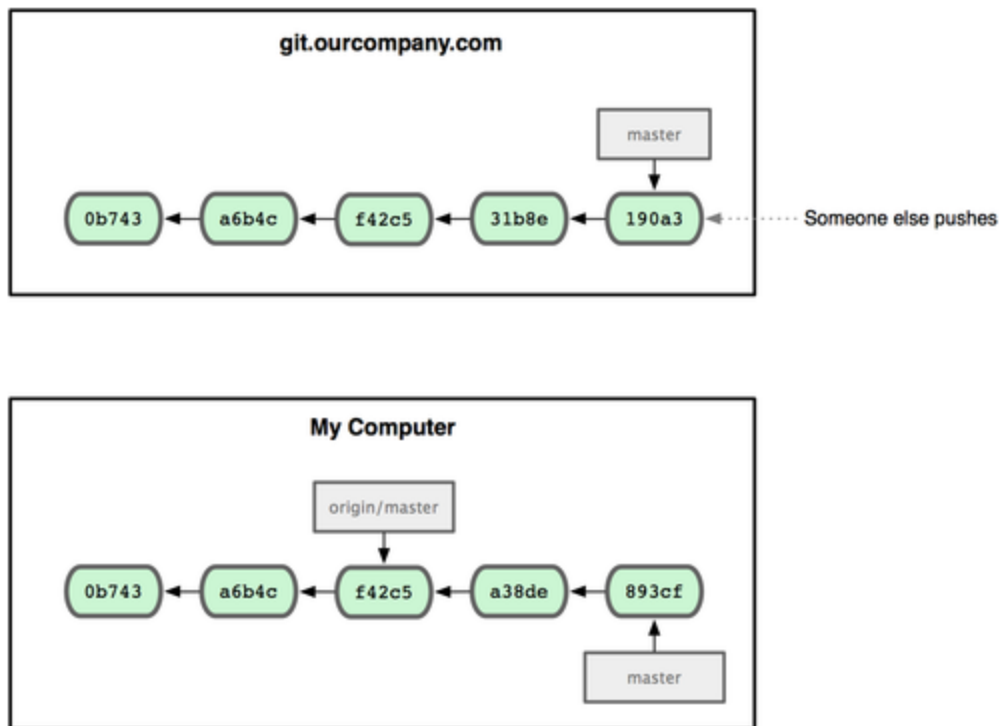
Тематические ветки

Удаленные ветки



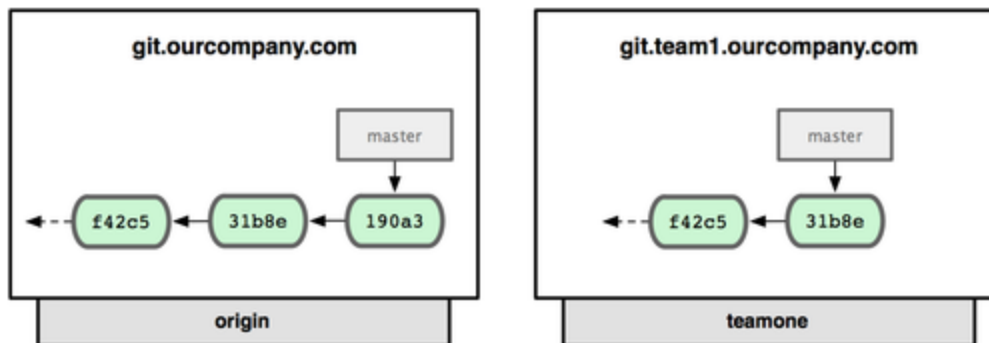
Удаленные ветки

Удаленные ветки

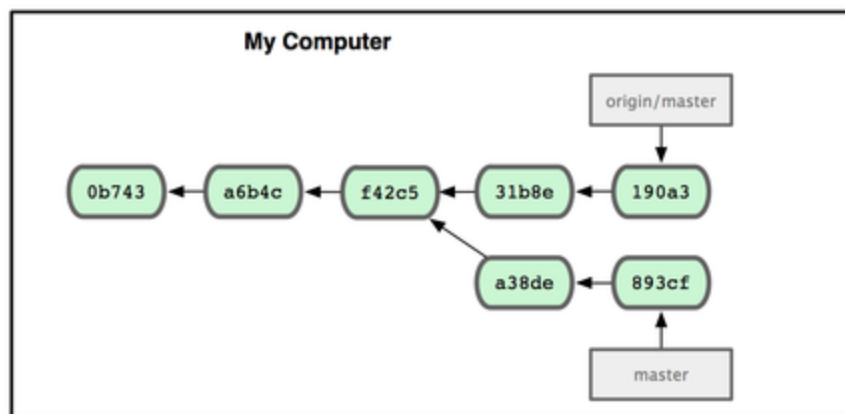


Удаленные ветки

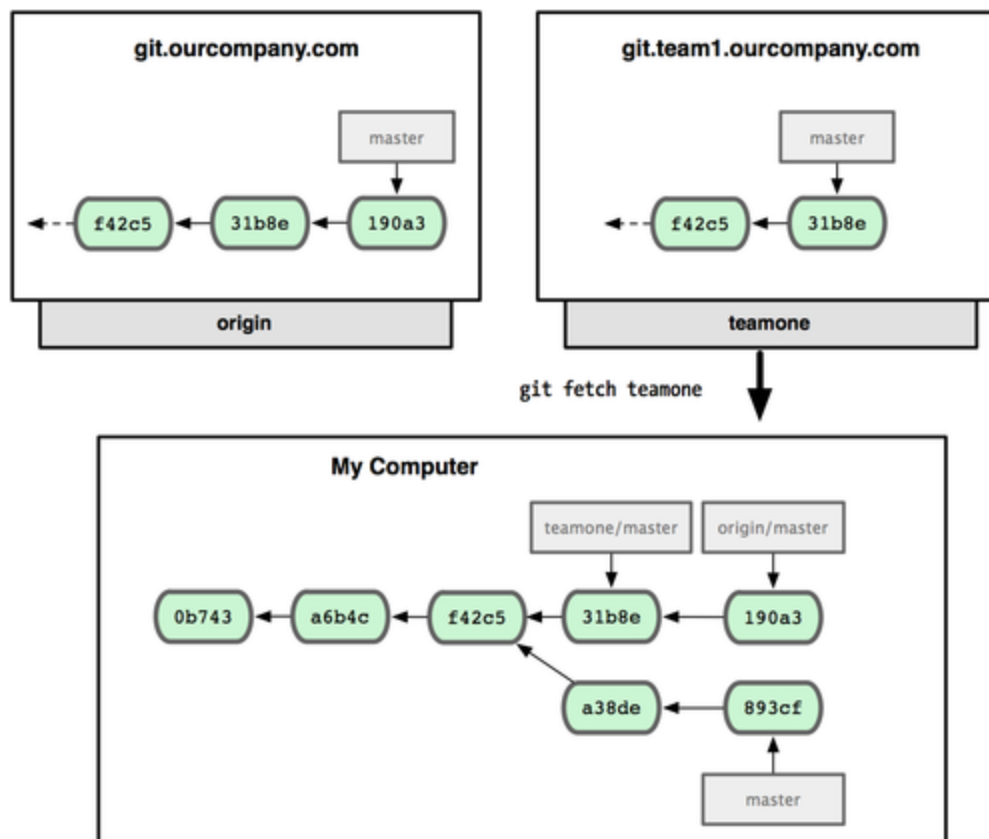
Удаленные ветки



```
git remote add teamone git://git.team1.ourcompany.com
```



Удаленные ветки



Отправка изменений

1. `$ git push origin serverfix`
2. `git fetch origin`
3. `git checkout -b serverfix origin/serverfix`

Отслеживание веток

1. `$ git checkout -b [ветка] [удал. сервер]/[ветка]`
2. `$ git checkout --track origin/serverfix`
3. `$ git checkout -b sf origin/serverfix`

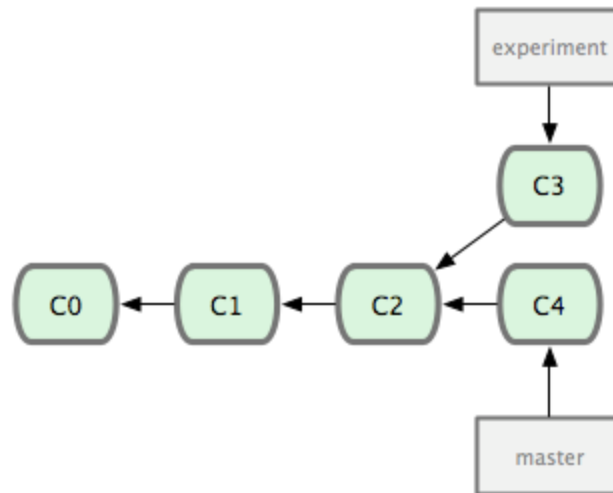
Удаление веток на удаленном сервере

1. `$ git push origin :serverfix`
2. `$ git push [удал. сервер] [лок. ветка]:[удал. ветка]`
3. `$ git push [удал. сервер] :[удал. ветка]`

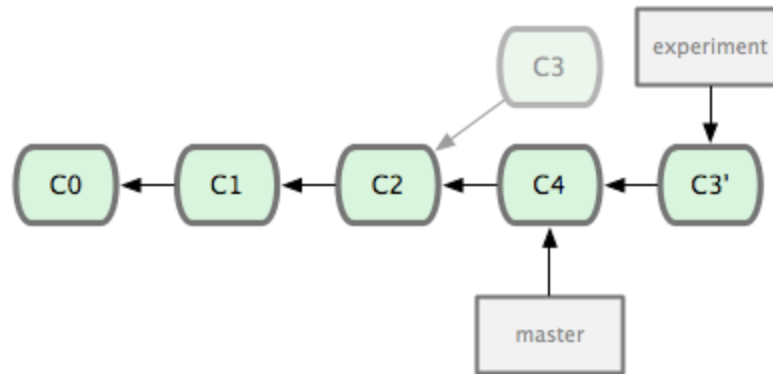
Перемещение

1. merge
2. rebase

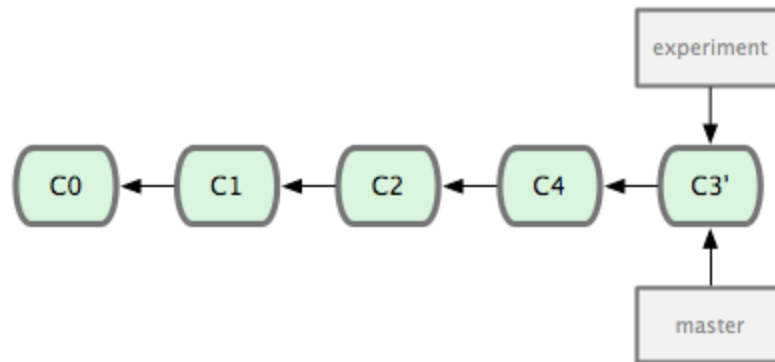
Rebase



Rebase

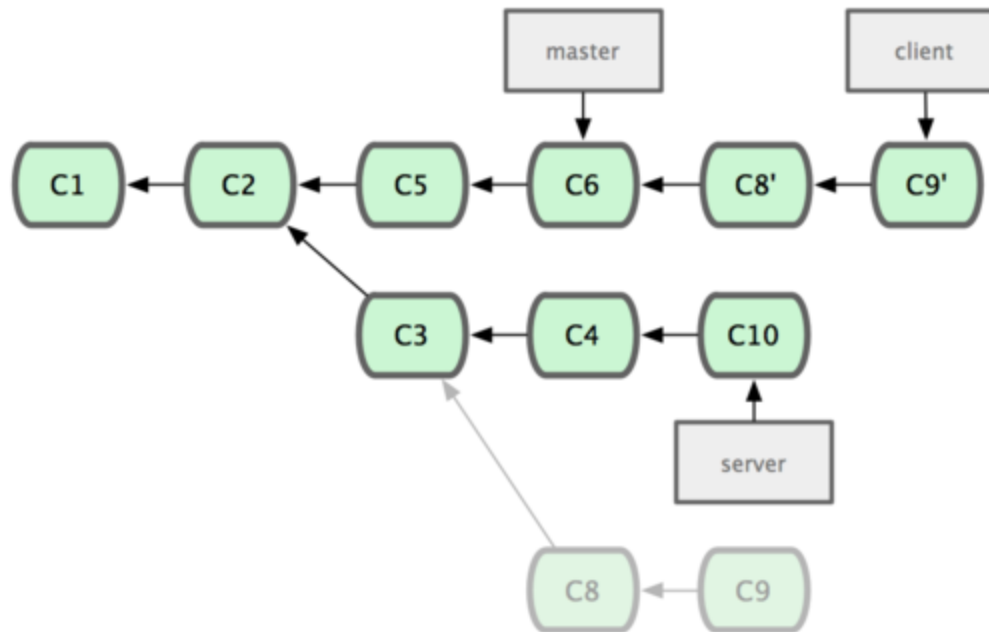


Rebase

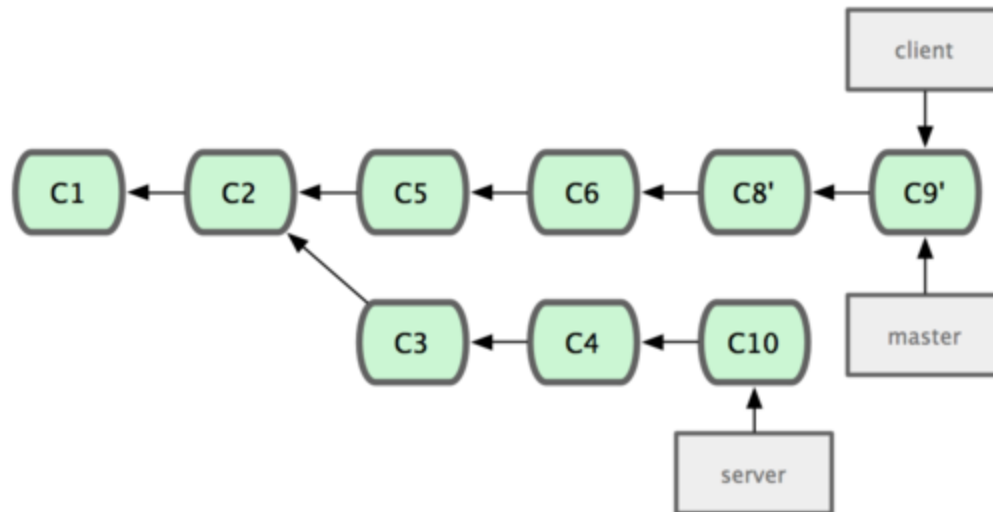


Rebase

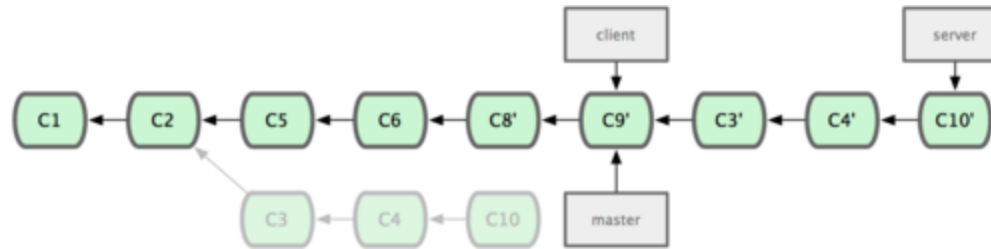
Rebase



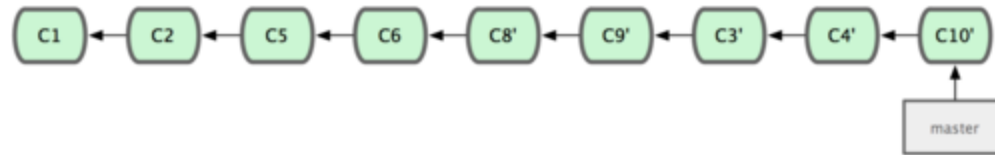
Rebase



Rebase



Rebase

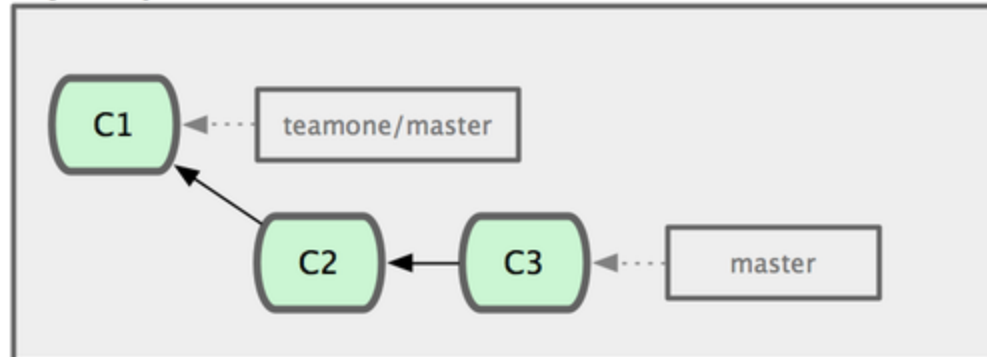


Rebase Risks (или как подложить свинью)

git.team1.ourcompany.com

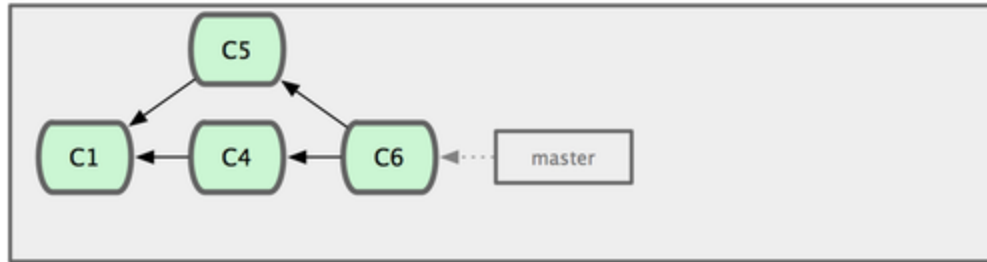


My Computer

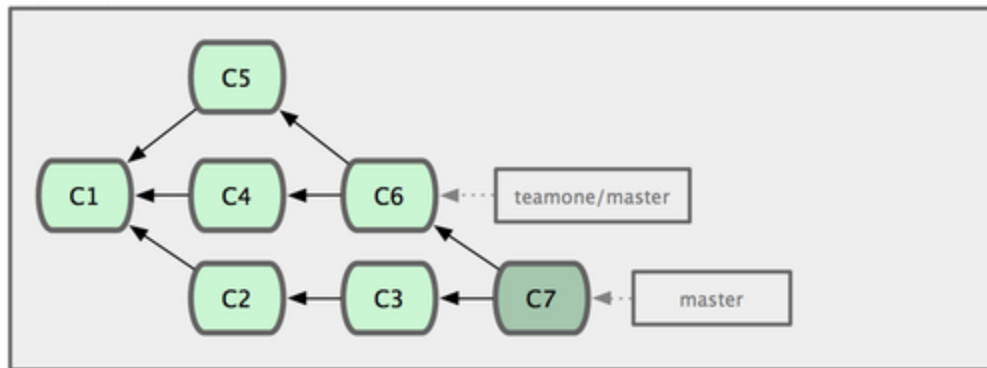


Rebase Risks (или как подложить свинью)

git.team1.ourcompany.com

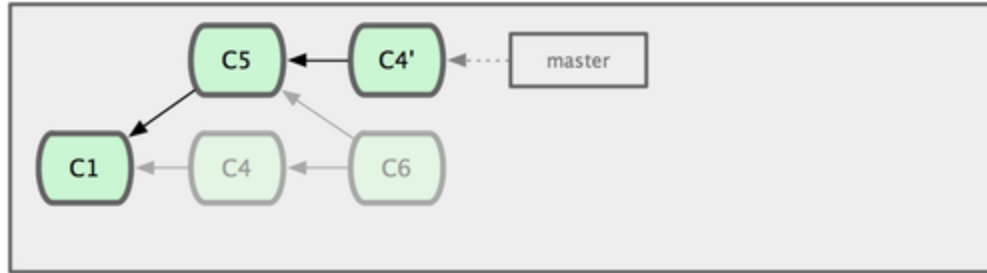


My Computer

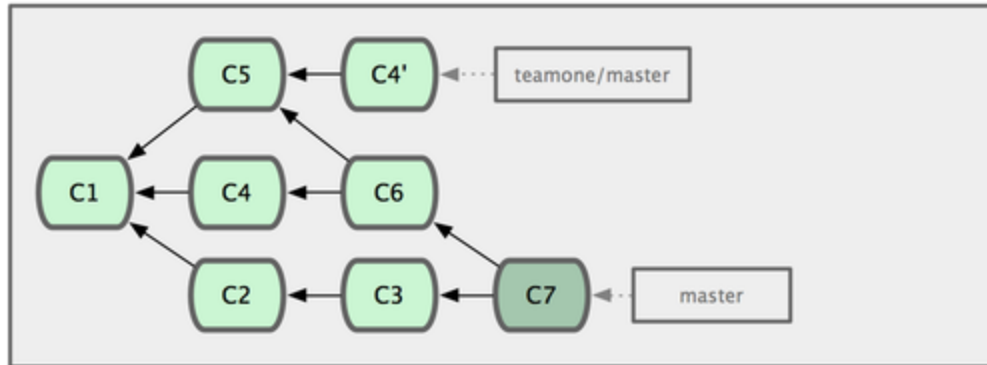


Rebase Risks (или как подложить свинью)

git.team1.ourcompany.com

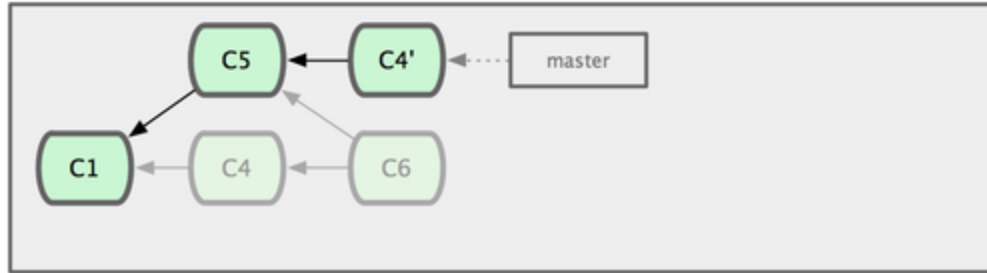


My Computer

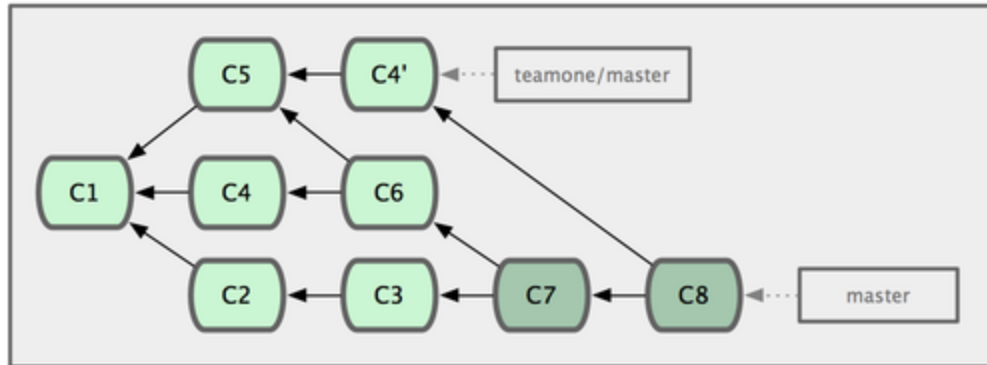


Rebase Risks (или как подложить свинью)

git.team1.ourcompany.com



My Computer



На сегодня ХВАТИТ

:)

И напоследок...

Оставляем фидбек - <http://undev.coursify.ru/courses/18>

Лекция оффлайн - [Тут](#)

Презентация оффлайн - http://zzet.org/learn_git/lection-2.html

Все лекции оффлайн - <http://zzet.org/learn-git.html>