

Яндекс

**Я**ндекс

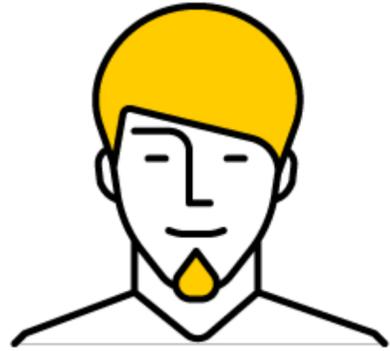
# Про git

Рашид Ксиров, разработчик инфраструктуры

**Зачем нужен git?**



# Дано



Разработчик

или

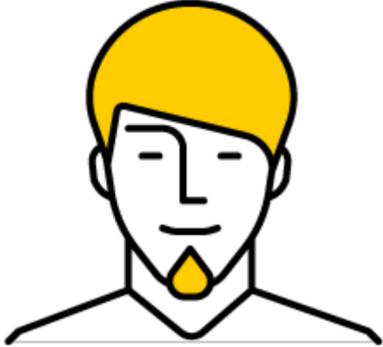


Заказчик



Команда

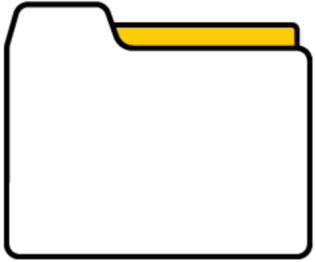
# Разрабатываем и публикуем проект



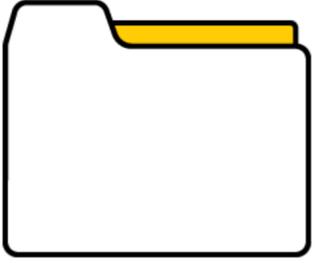
Разработчик



Заказчик



в проде

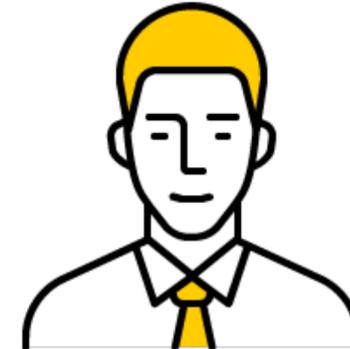
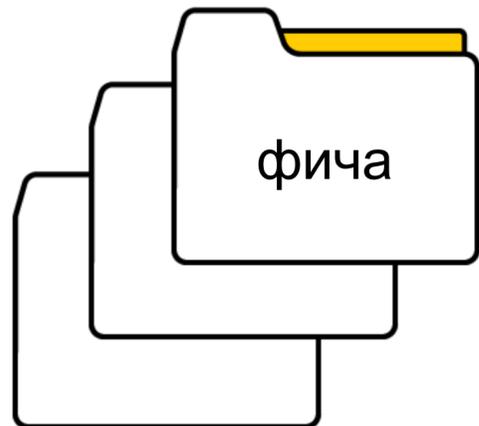


проект

# Добавляем функциональность

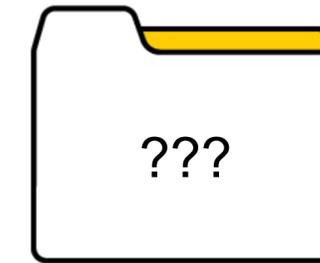


Разработчик



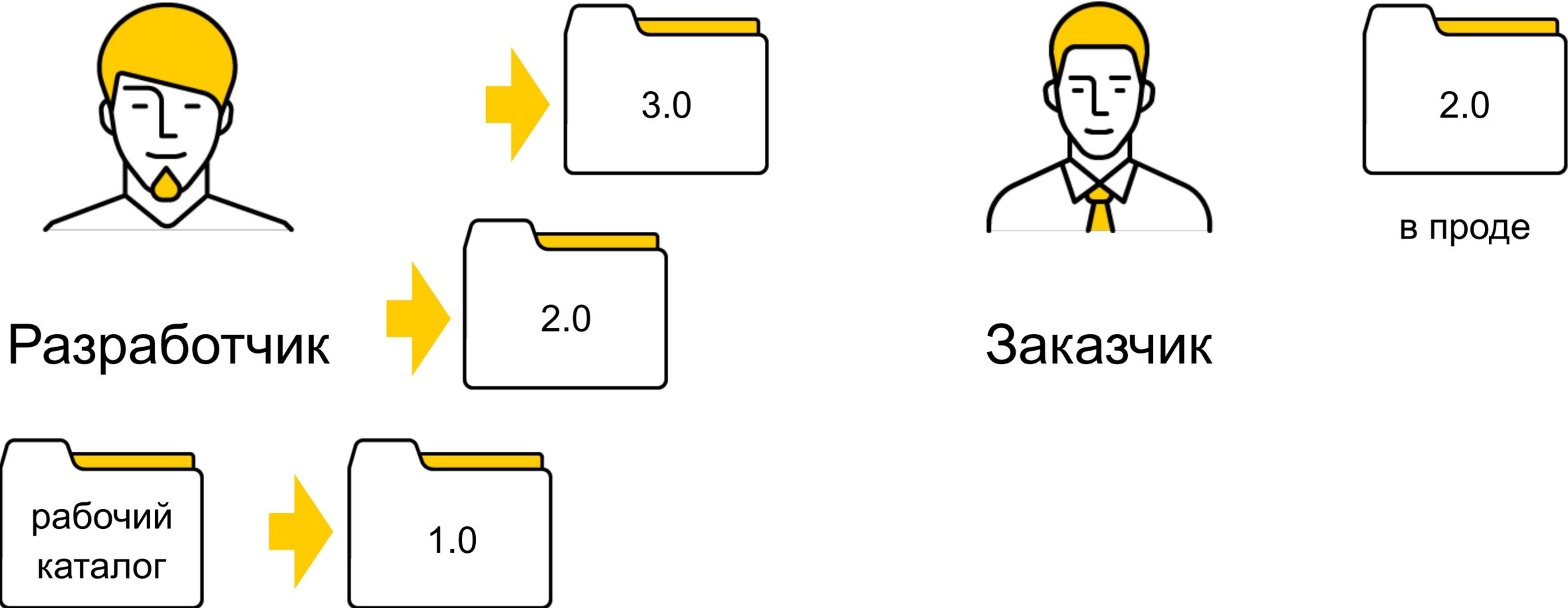
Заказчик

Где моя фича?

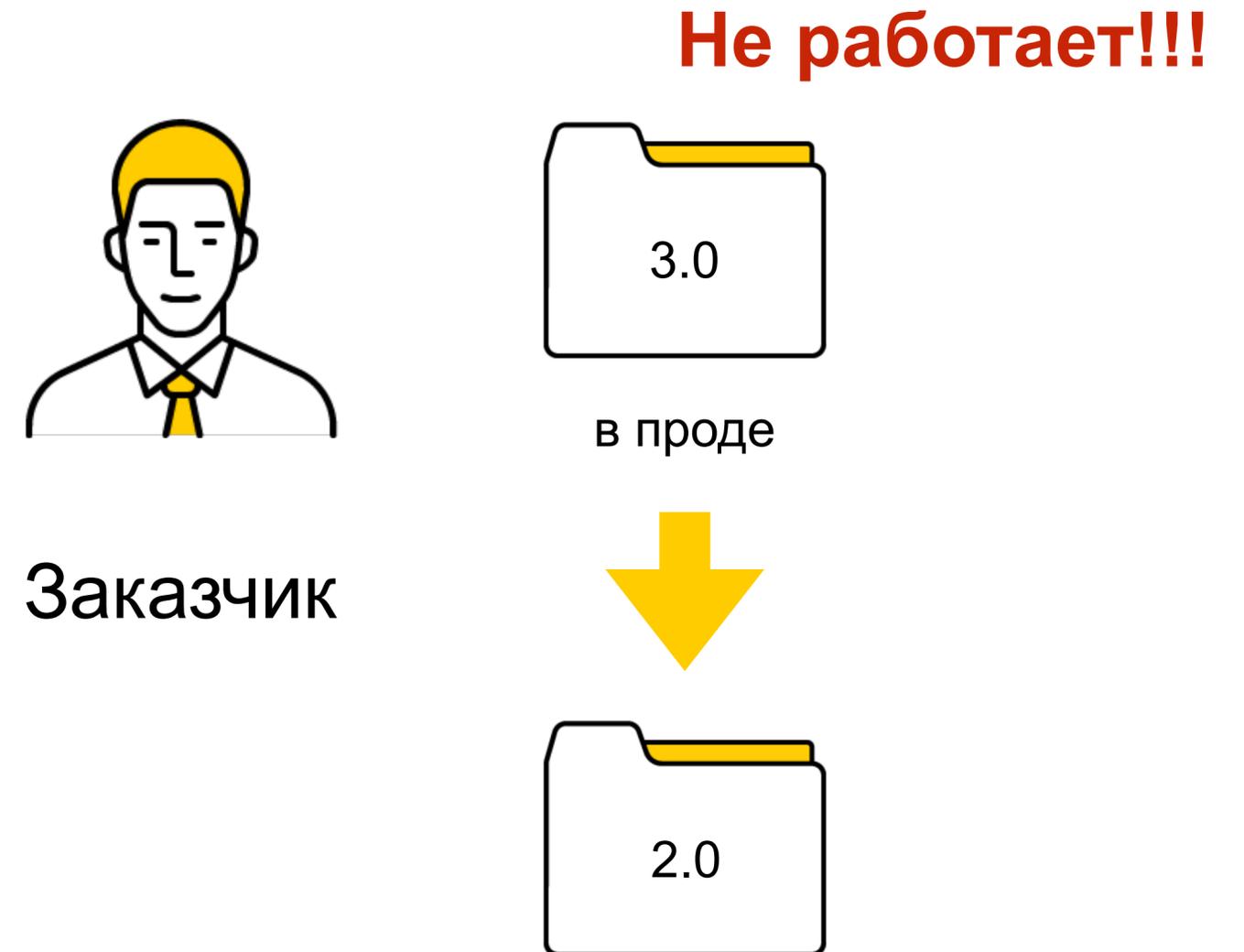
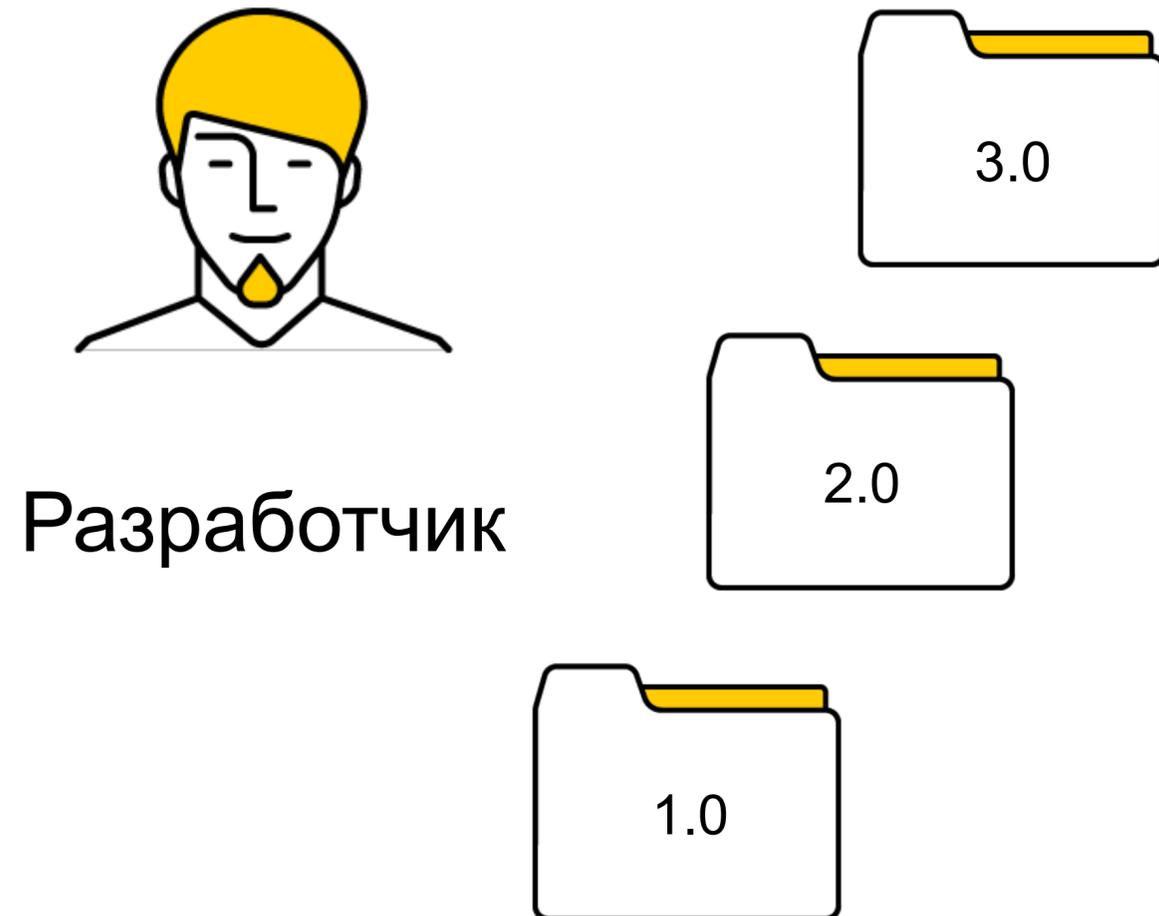


в проде

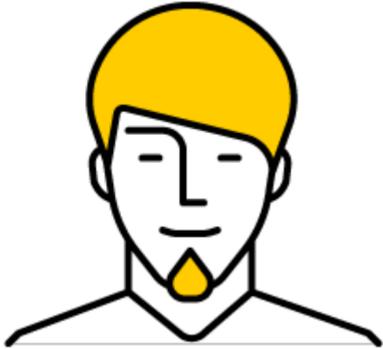
# Фиксируем (нумеруем) состояния проекта



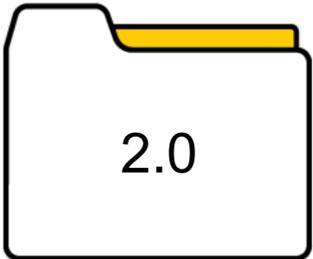
# Если что-то пошло не так: откатываемся



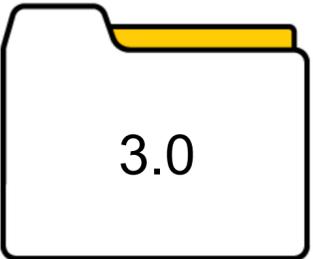
# Разбираемся: что изменилось в проекте?



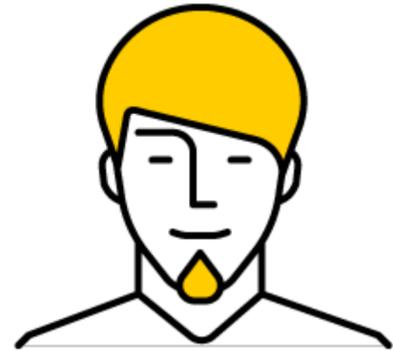
Разработчик



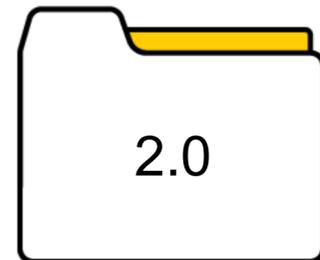
??? diff ???



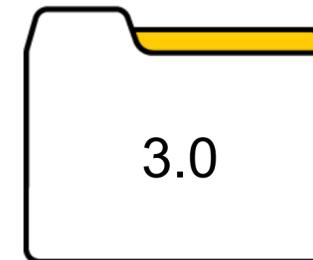
# Добавим историю изменений?



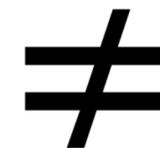
Разработчик



обновляем  
changelog



с точки зрения пользователя



изменения  
в коде проекта

с точки зрения разработчика

# Нужен инструмент:

- 01 | **Архив состояний проекта – КОММИТОВ**
- 02 | **Возможность отката – переключение между состояниями**
- 03 | **Просмотр изменений – вывод разницы между состояниями**





**Как работать с git?**



**[githowto.com/ru](https://githowto.com/ru)**

THE EXPERT'S VOICE®

SECOND EDITION

# Pro Git

EVERYTHING YOU NEED TO  
KNOW ABOUT GIT

Scott Chacon and Ben Straub

Apress®

[git-scm.com/book/ru/v2](https://git-scm.com/book/ru/v2)

# Устанавливаем и настраиваем git

**Mac OS:** `brew install git`

**Windows:** <http://git-for-windows.github.io/>

```
git config --global user.name "Your Name"
```

```
git config --global user.email "your_email@whatever.com"
```

# git init – создаем репозиторий

```
$mkdir work && cd work
```

```
$mkdir hello && cd hello
```

```
$git init
```

Инициализирован пустой репозиторий Git в ~/work/hello/.git/

# git status – смотрим статус репозитория

```
$echo 'Hello, World!' > hello.html
```

```
$git status
```

```
На ветке master
```

```
Еще нет коммитов
```

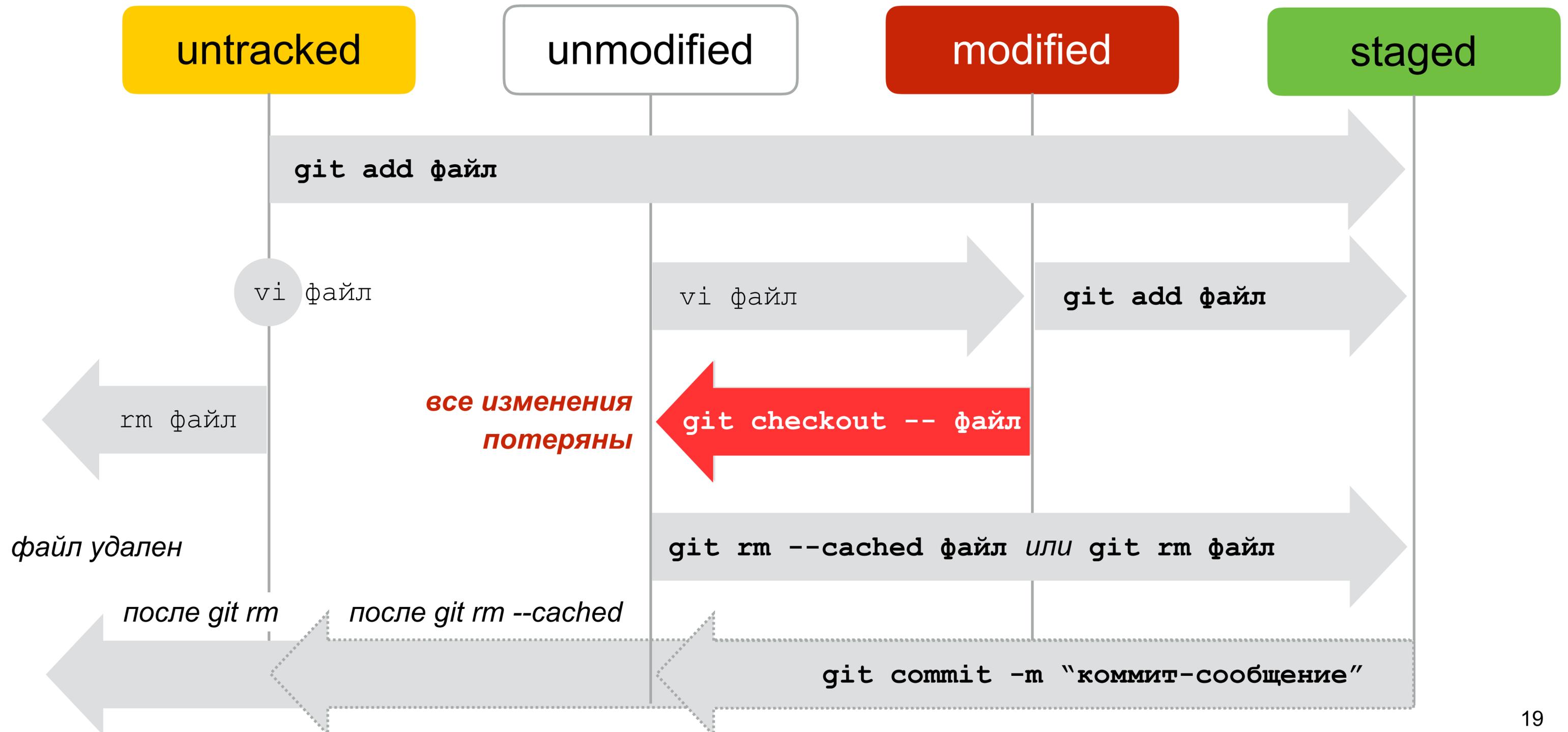
Неотслеживаемые файлы:

(используйте «git add <файл>...», чтобы добавить то, что будет включено в коммит)

**hello.html**

ничего не добавлено в коммит, но есть неотслеживаемые файлы (используйте «git add», чтобы отслеживать их)

# Про состояния файлов в рабочей папке



# git add – добавляем файл в индекс

```
$git add hello.html
```

```
$git status
```

На ветке master

Еще нет коммитов

Изменения, которые будут включены в коммит:

(используйте «`git rm --cached <файл>...`», чтобы убрать из индекса)

**новый файл:** `hello.html`

# git commit – фиксируем состояние файла (делаем коммит)

```
$git commit -m "Add hello message"
```

```
$git status
```

```
[master (корневой коммит) 130830f] Add hello message
```

```
1 file changed, 1 insertion(+)
```

```
create mode 100644 hello.html
```

130830f – это хэш коммита – уникальный код, вычисленный на основе содержимого файлов

# git log – СМОТРИМ ИСТОРИЮ КОММИТОВ

```
$git log
```

```
commit 130830fed52695af2c348f2a3f6a63b6788d606c (HEAD -> master)
```

```
Author: Rashid Ksirov <rashid@yandex-team.ru>
```

```
Date: Tue May 14 10:53:39 2019 +0300
```

```
Add hello message
```

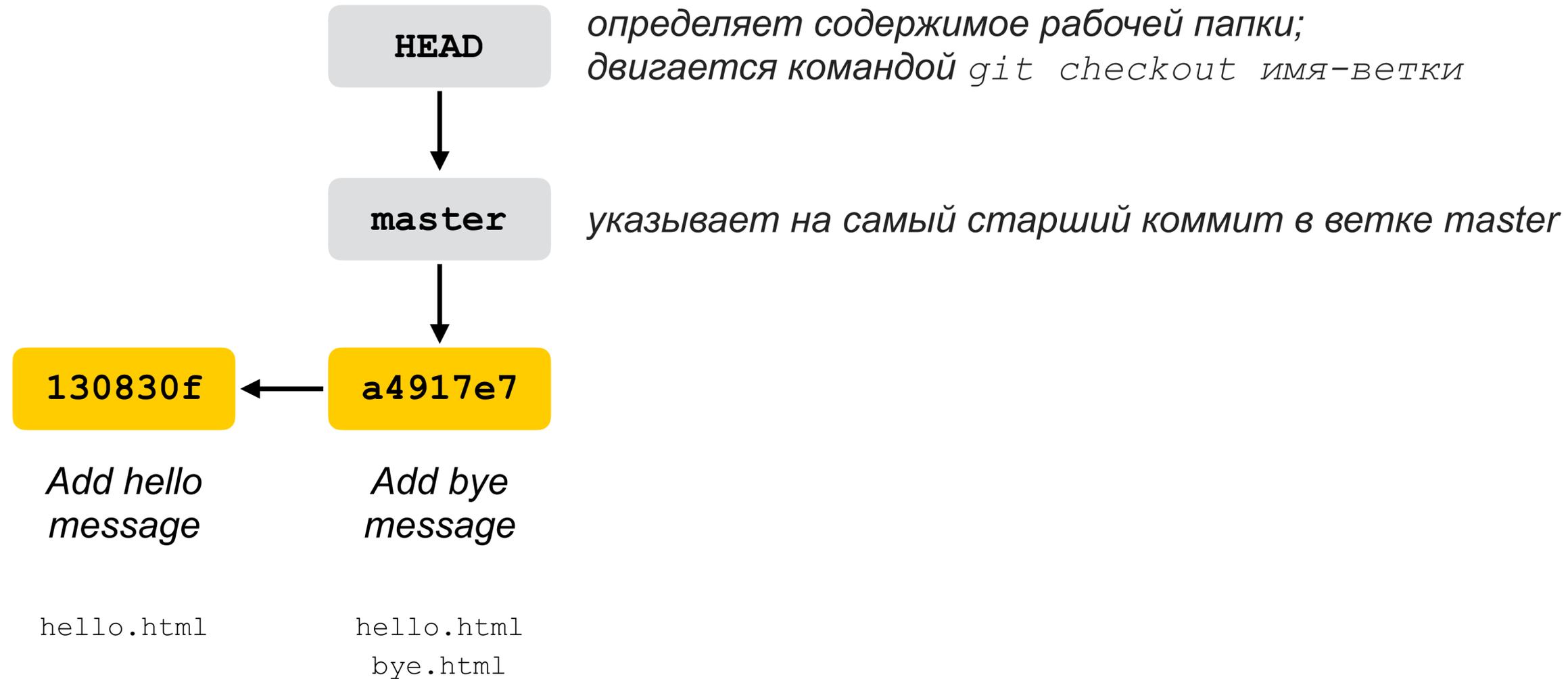
```
$git log --oneline
```

```
130830f (HEAD -> master) Add hello message
```

# Добавляем еще один коммит

```
$echo `Good bye!` > bye.html  
$git status  
$git add bye.html  
$git commit -m "Add bye message"  
[master a4917e7] Add bye message  
1 file changed, 1 insertion(+)  
create mode 100644 bye.html  
$git log --oneline  
a4917e7 (HEAD -> master) Add bye message  
130830f Add hello message
```

# Как выглядят наши коммиты



# git checkout – меняем состояние рабочей папки

```
$git checkout 130830f
```

Примечание: переход на «130830f».

Вы сейчас в состоянии «отделённого HEAD». Вы можете осмотреться, сделать экспериментальные изменения и закоммитить их, также вы можете отменить изменения любых коммитов в этом состоянии не затрагивая любые ветки и не переходя на них.

Если вы хотите создать новую ветку и сохранить свои коммиты, то вы можете сделать это (сейчас или позже) вызвав команду checkout снова, но с параметром -b. Например:

```
git checkout -b <имя-новой-ветки>
```

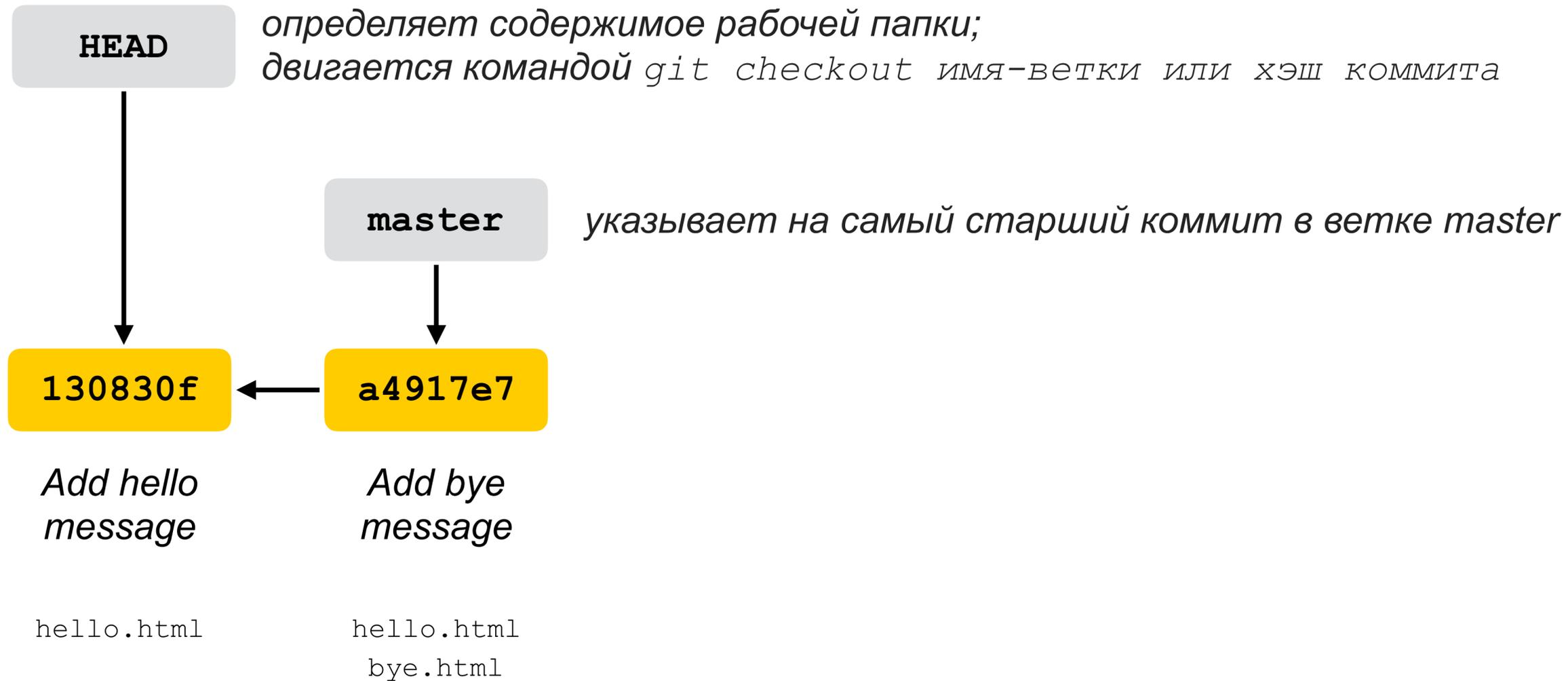
# Посмотрим на содержимое рабочей папки

```
$ls -al
```

```
total 8
```

```
drwxr-xr-x  4 rashid LD\Domain Users 128 14 май 16:37 .
drwxr-xr-x 26 rashid LD\Domain Users 832 14 май 16:29 ..
drwxr-xr-x 13 rashid LD\Domain Users 416 14 май 16:37 .git
-rw-r--r--  1 rashid LD\Domain Users  14 14 май 16:28 hello.html
```

# Как выглядят наши коммиты



# git checkout – возвращаем все назад

```
$git checkout a4917e7
```

```
Предыдущая позиция HEAD была 130830f Add hello message
```

```
HEAD сейчас на a4917e7 Add bye message
```

```
$git status
```

```
HEAD отделён на a4917e7
```

```
ничего коммитить, нет изменений в рабочем каталоге
```

```
$git log --oneline
```

```
a4917e7 (HEAD, origin/master, master) Add bye message
```

```
130830f Add hello message
```

```
$git checkout master
```

```
Переключено на ветку «master»
```

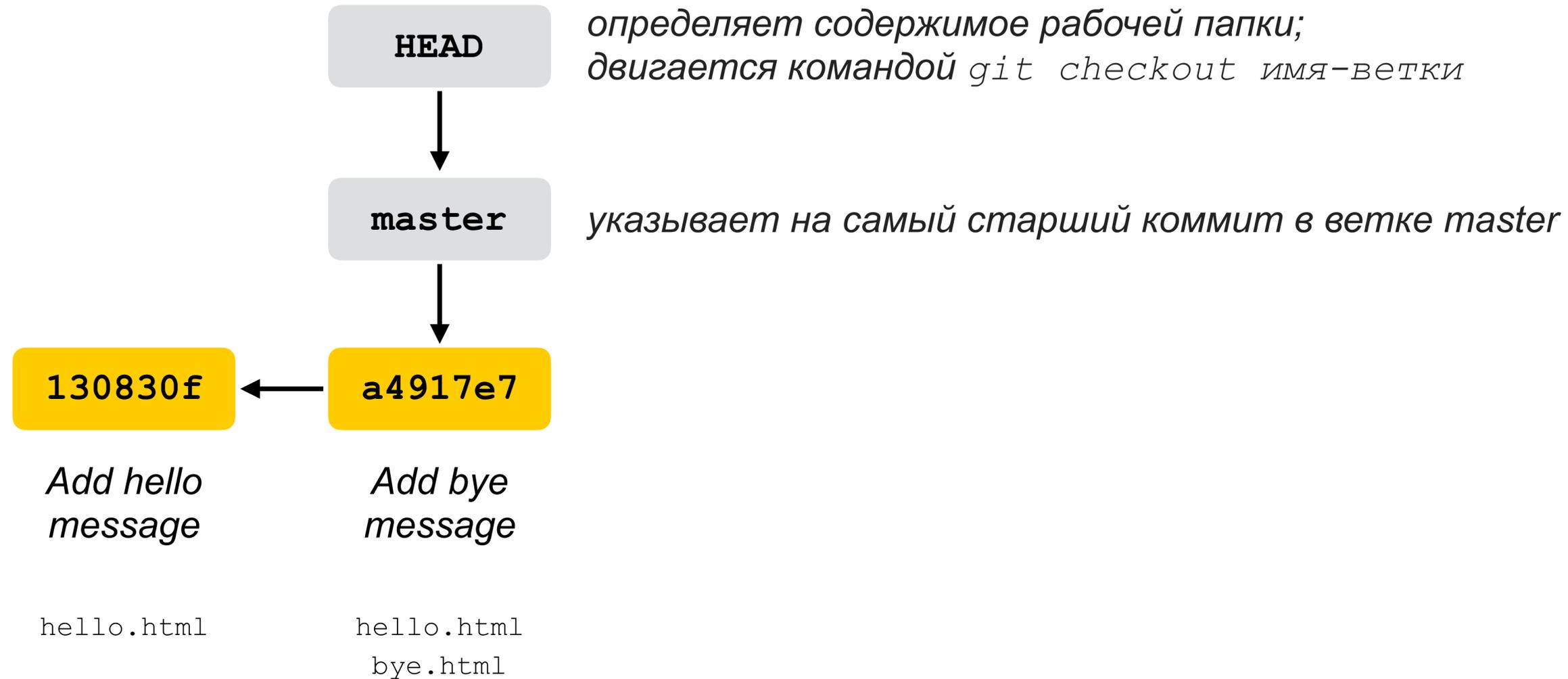
# Посмотрим на содержимое рабочей папки

```
$ls -al
```

```
total 16
```

```
drwxr-xr-x  5 rashid LD\Domain Users 160 14 май 16:37 .
drwxr-xr-x 26 rashid LD\Domain Users 832 14 май 16:29 ..
drwxr-xr-x 13 rashid LD\Domain Users 416 14 май 16:38 .git
-rw-r--r--  1 rashid LD\Domain Users  10 14 май 16:37 bye.html
-rw-r--r--  1 rashid LD\Domain Users  14 14 май 16:28 hello.html
```

# Вернули HEAD на ветку master



# git branch – создаем ветку

```
$git branch
```

```
* master
```

```
$git branch feat/good
```

```
feat/good
```

```
* master
```

```
$git checkout feat/good
```

```
Переключено на ветку «feat/good»
```

**или можно было создать ветку и переключиться на нее одной командой:**

```
$git checkout -b feat/good
```