

# Git

## Система контроля версий

TeachMeSkills

# Урок

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18

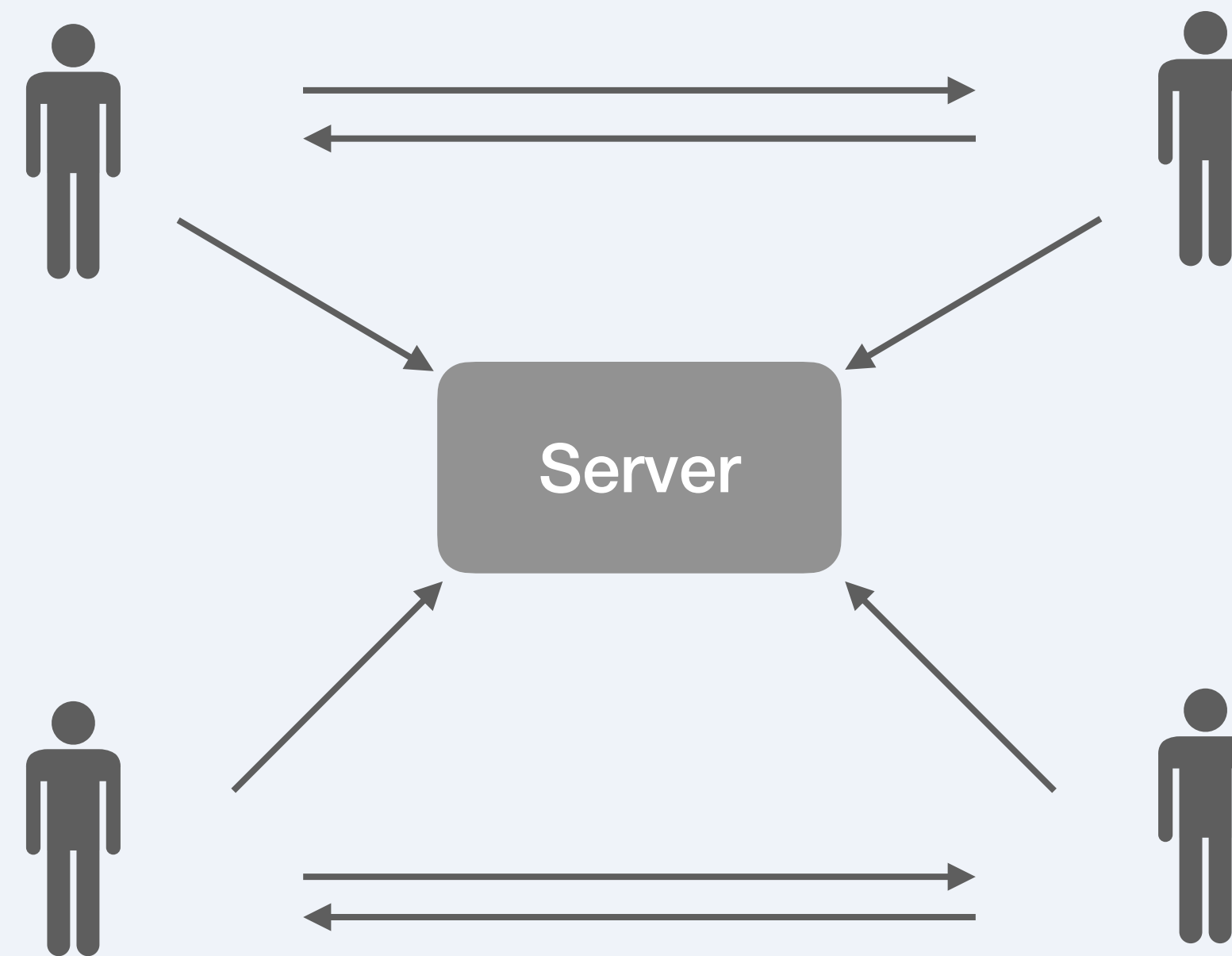
# Git

Что это такое?

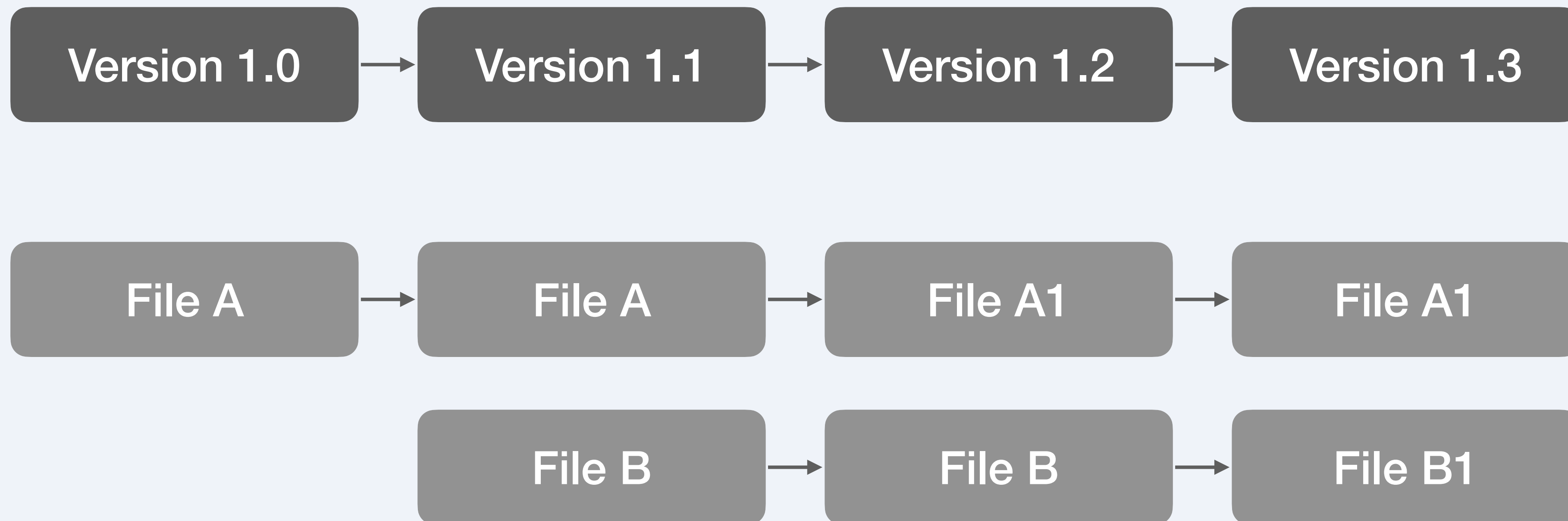
Для чего нужен ?

Как им пользоваться ?

# Git распределенная система контроля версий

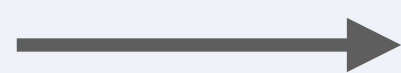


# Git отслеживает изменения в проекте



# Установка Git

Загружаем и устанавливаем git



<https://git-scm.com/>

# Графические интерфейсы git

Github Desktop

Git Kraken

Sourcetree

And other

# Сервисы для хранения проектов

Github

Gitlab

Bitbucket



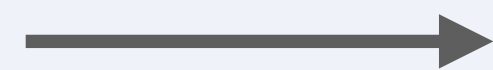
# Что такое репозиторий?

**В репозитории хранятся все документы вместе с историей их изменения и другой служебной информацией.**

**То есть, эта папка вашего проекта**

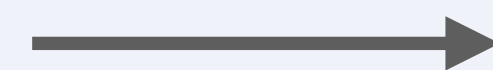
# Основные команды

**git init**



Создает в текущей директории новую поддиректорию `.git`  
Она содержит в себе структуру git-репозитория

**git status**



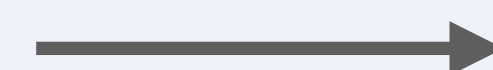
Команда позволяющая отслеживать состояние в  
текущем репозитории

**git add**



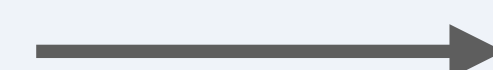
Команда добавляет файл в индекс для фиксации его  
изменений

**git commit**



Описание сохранения подготовленных и измененных  
файлов

**git push**



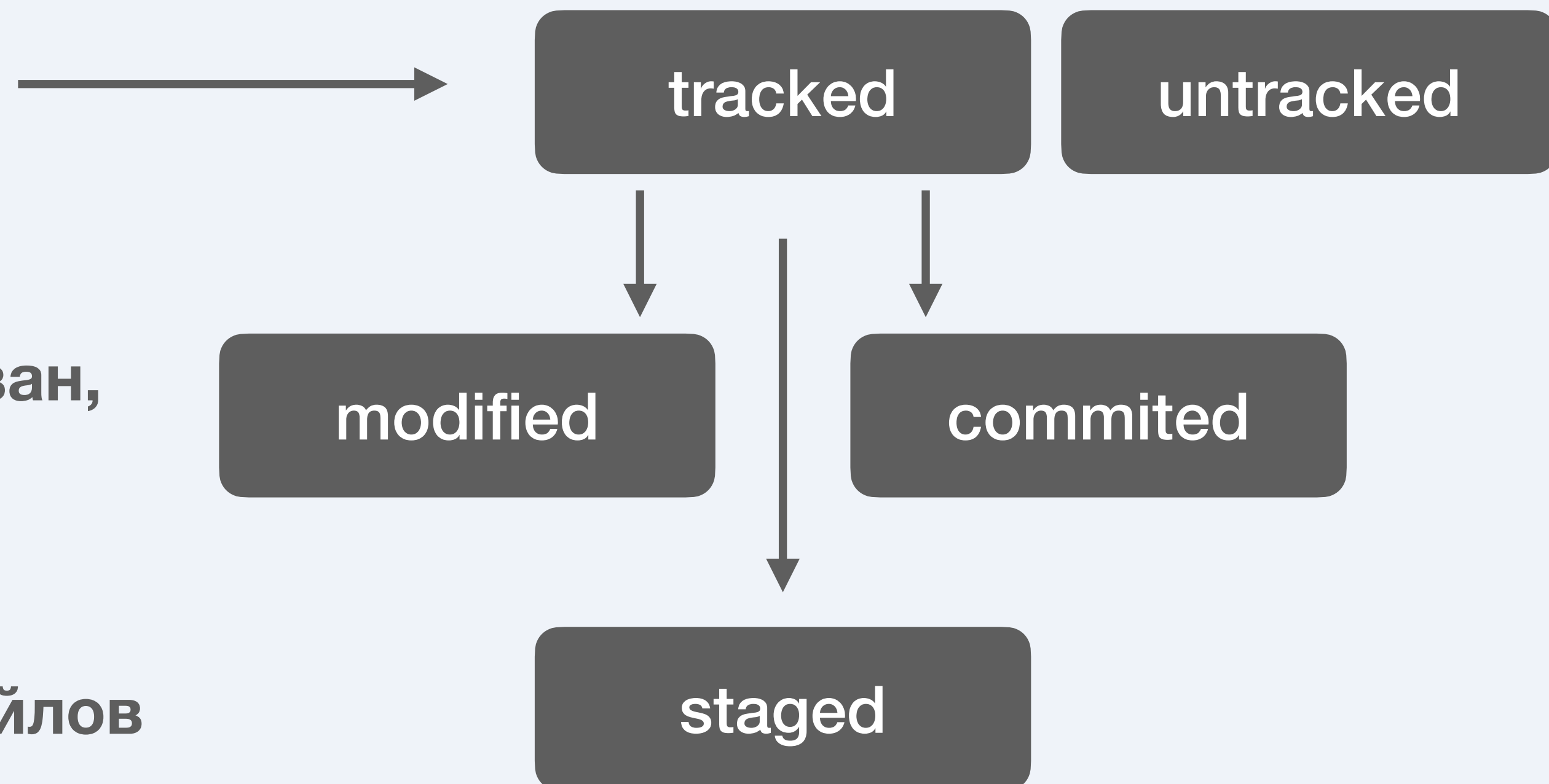
Отправляет изменения в удаленный репозиторий

# Состояние файлов в git

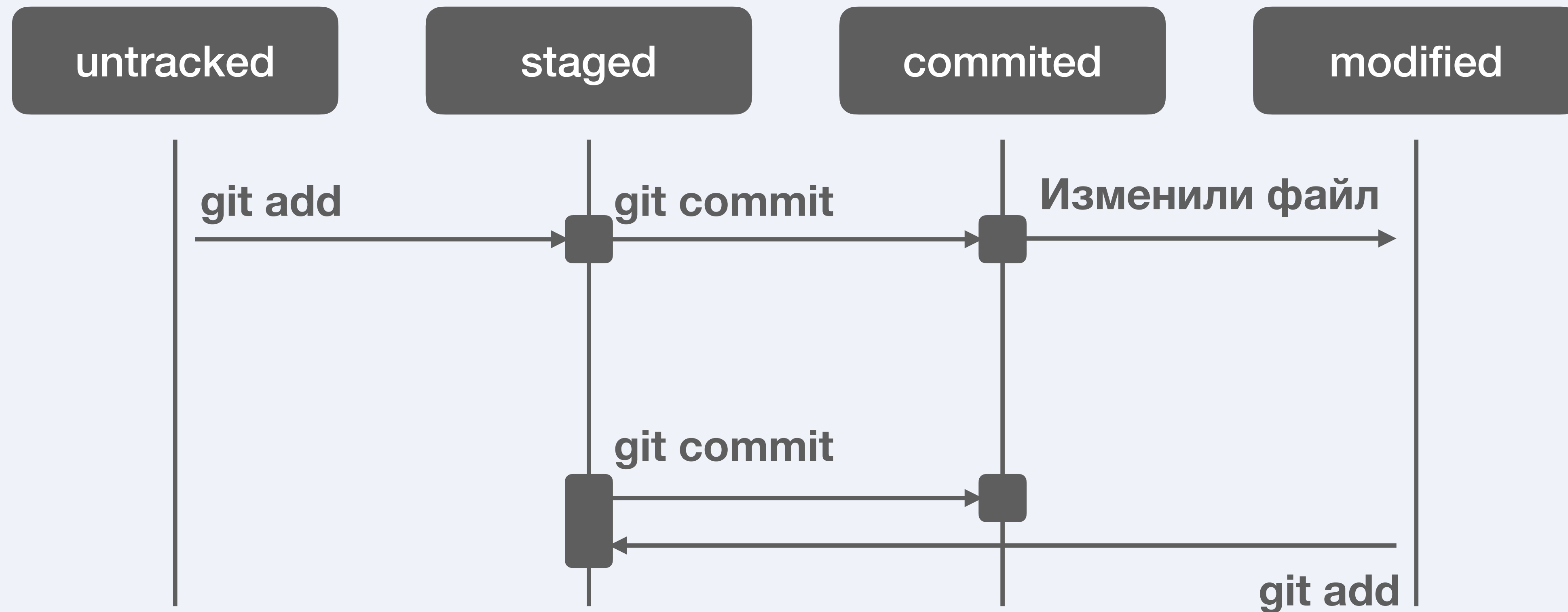
Отслеживает или нет  
изменения файлов

Файл изменен и не фиксирован,  
файл попал в commit

Область подготовленных файлов

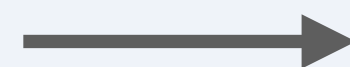


# Состояние файлов в git



# Игнорируем файлы для отслеживания

Создаем файл .gitignore



```
# Comments  
*.js  
/cache  
form.html  
log/  
log/logs.txt
```

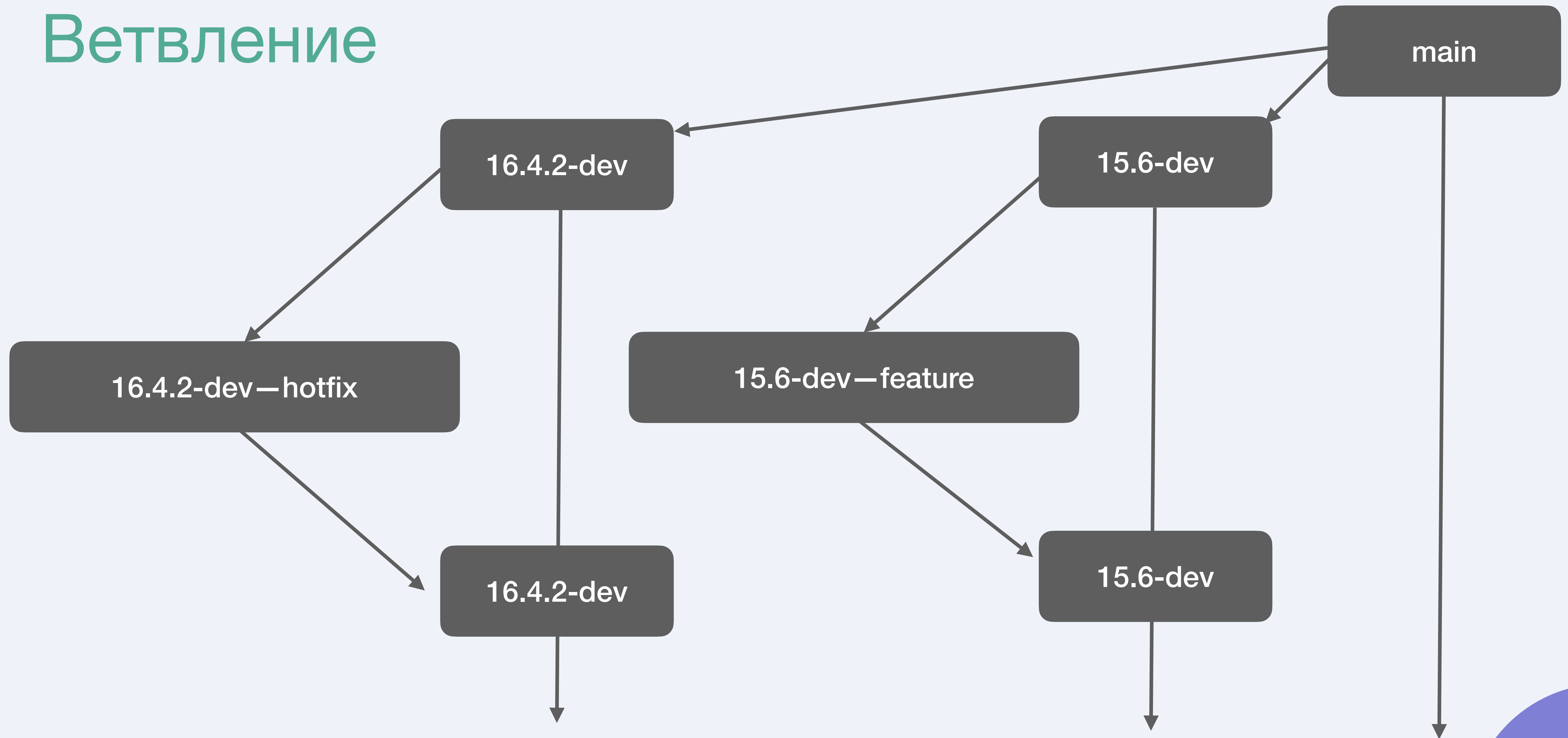
# Откатываемся на предыдущий коммит

```
$ git reset --hard a3775a5485af0af20375cedf46112db5f813322a  
$ git push --force
```

# Ветвление

- Ветка в Git это подвижный указатель на один из коммитов.
- Обычно ветка указывает на последний коммит в цепочке коммитов.
- Ветка берет свое начало от какого-то одного коммита.

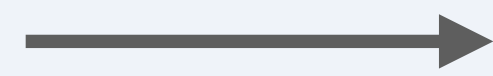
# Ветвление





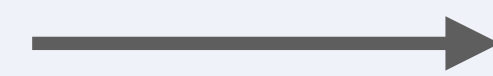
# Основные команды

`git branch`



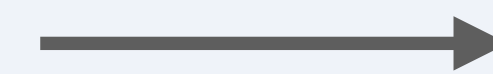
Список, создание или удаление веток

`git checkout`



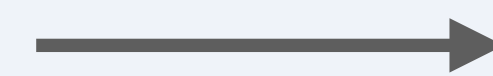
Переключаемся между ветками

`git merge`



Объединяет ветки вместе

`git pull`



Получает и интегрирует данные с другой ветки