

Git: основы работы

Павел Селиванов
инженер Southbridge

Для чего вы используете Git?





Для чего вы используете Git?

Хранение информации

С историей





Для чего вы используете Git?

- Хранение информации
- С историей
- Распространение информации





Для чего вы используете Git?

- Хранение информации
- С историей
- Распространение информации
- Командная работа с информацией





Для чего вы используете Git?

- Хранение информации
- С историей
- Распространение информации
- Командная работа с информацией
- Автоматизации
 - Деплой
 - ChatOps
 - Конфигурирование





Для чего вы используете Git?

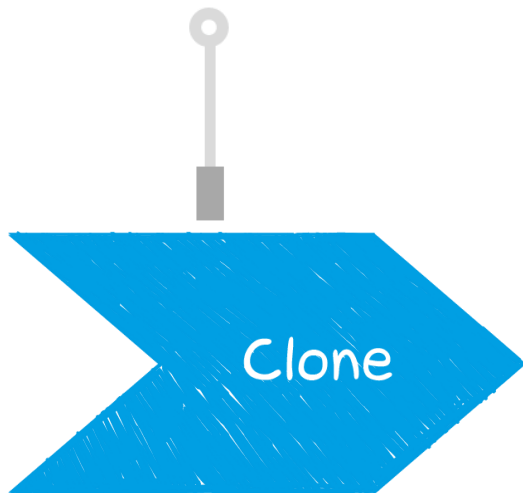
- Хранение информации
- С историей
- Распространение информации
- Командная работа с информацией
- Автоматизации
 - Деплой
 - ChatOps
 - Конфигурирование
- Управление инфраструктурой





Как вы используете Git?

```
git clone  
git@my.company.com:repo.git
```





Как вы используете Git?

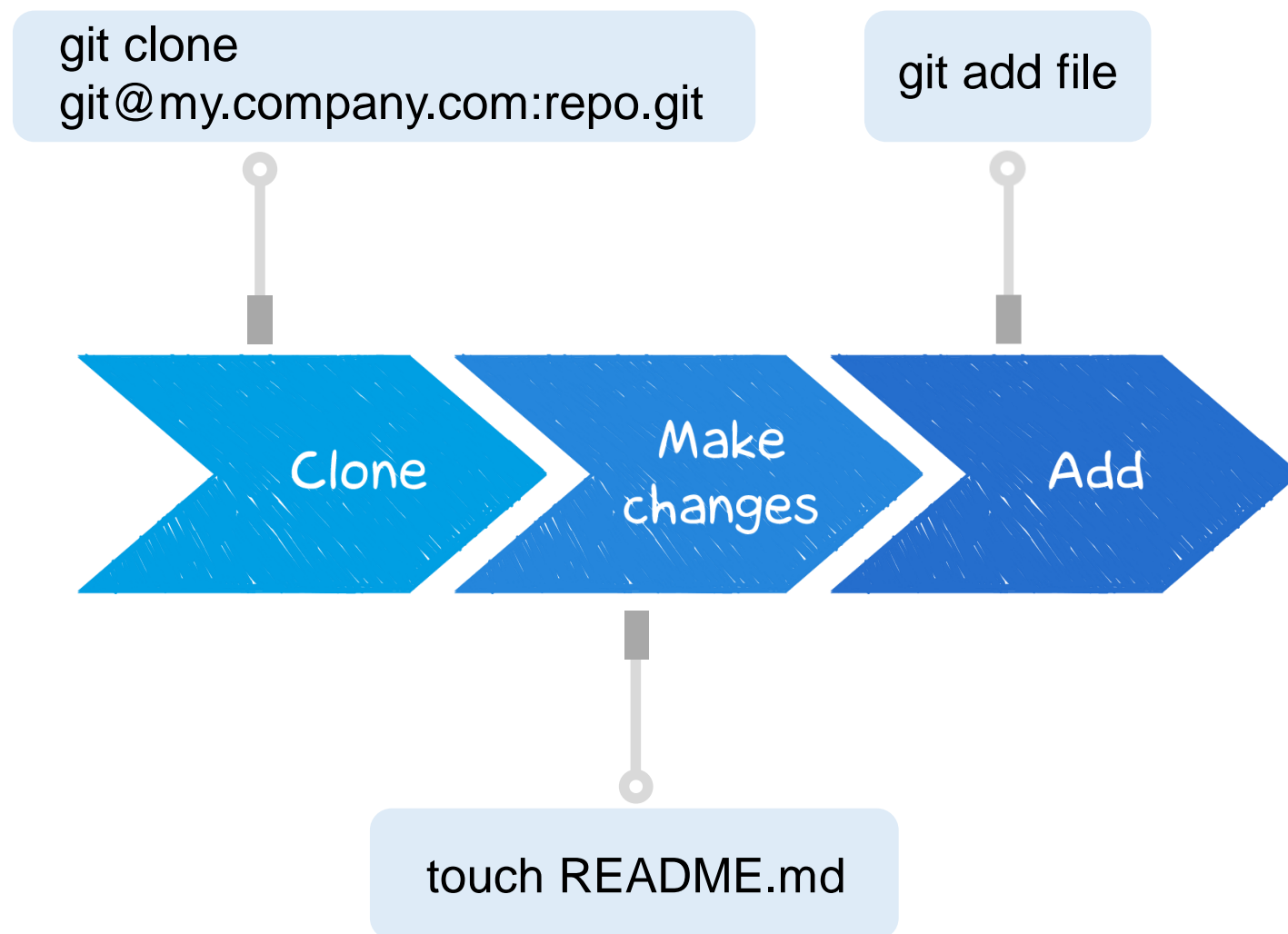
```
git clone  
git@my.company.com:repo.git
```



```
touch README.md
```



Как вы используете Git?





Как вы используете Git?

```
git clone  
git@my.company.com:repo.git
```

```
git add file
```

Clone

Make
changes

Add

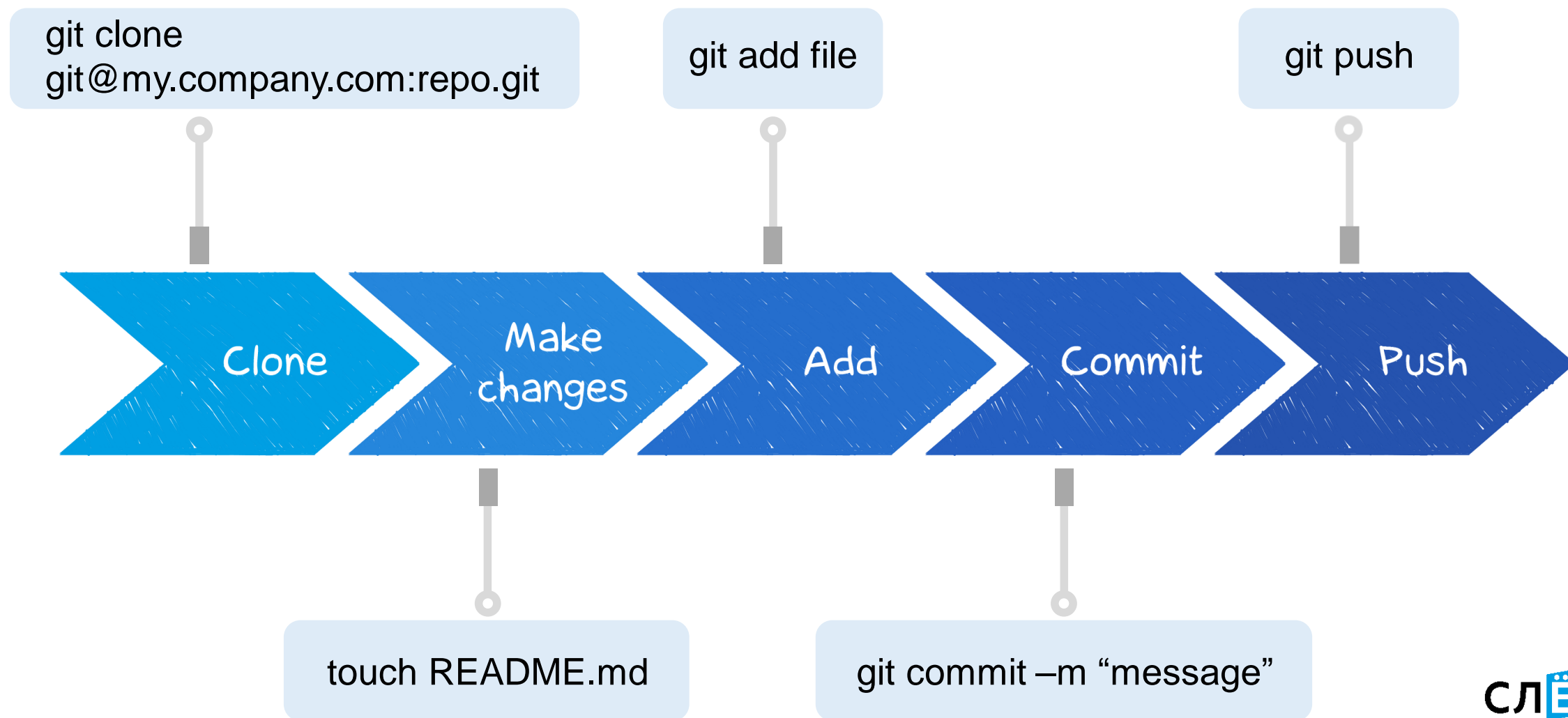
Commit

```
touch README.md
```

```
git commit -m "message"
```



Как вы используете Git?





Commit



Комитить сразу и часто



Sausage making





Sausage making





Commit

- Комитить сразу и часто
- Если нужно делать пост продакшен
- **Сделанная** работа **лучше** идеальной **не сделанной**



Commit



Следить за сообщением комита



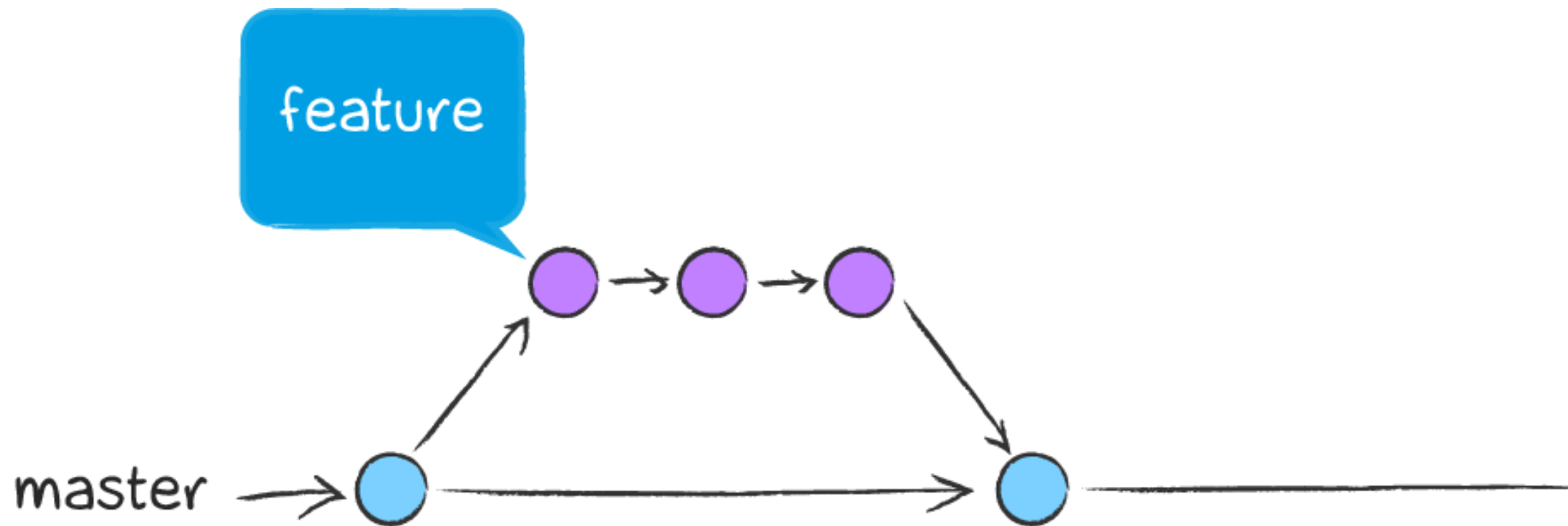
Commit

- Следить за сообщением комита
- Короткий сабж + пустая строка + развернутое пояснение
- Использовать повелительное наклонение
- Избегать общих сообщений
- Использовать английский (???)

Ветки



Feature branch





Review

Почему?

- Комитеры заинтересованы
- Снижение количества ошибок
- Улучшение качества
- Knowledge sharing
- Соблюдение стандартов
- Читаемость
- Автоматизируемость



Review

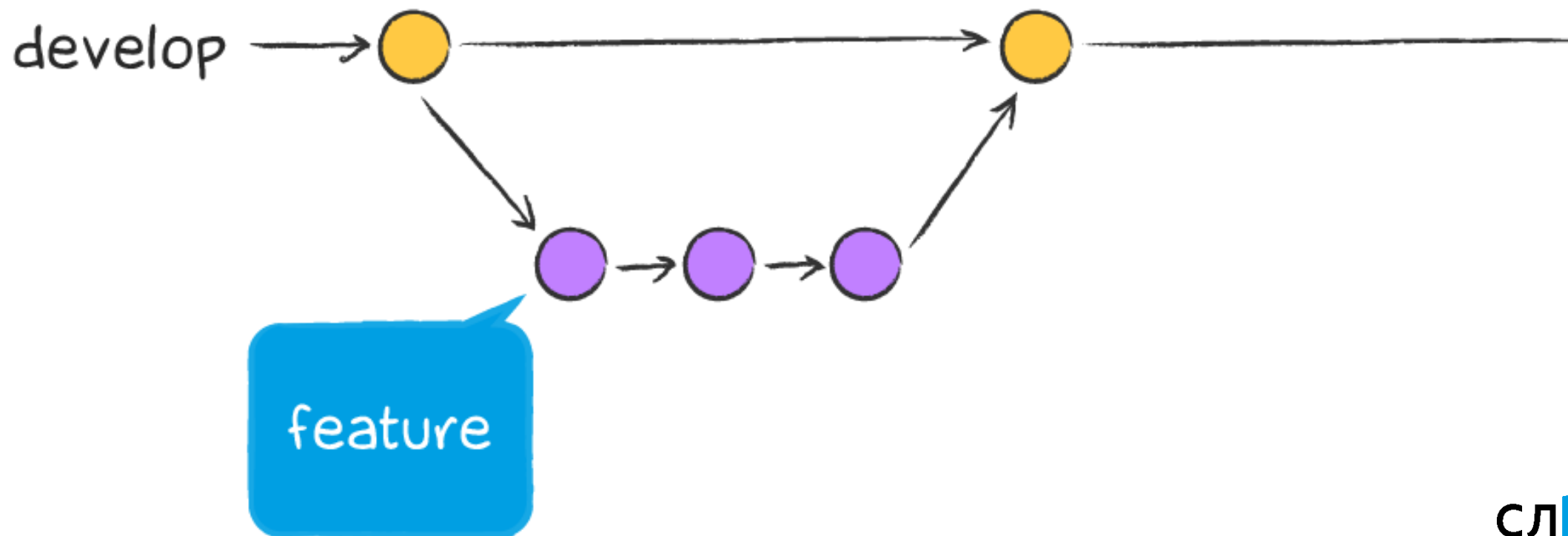
Как?

- Минимум ошибок доходит до Review
- Автотесты
- Статический анализ
- Небольшой объем
- Весь код ревьюится
- Синьор-мастер-тим-лид не исключение!
- В идеале без ревью нет мерджа



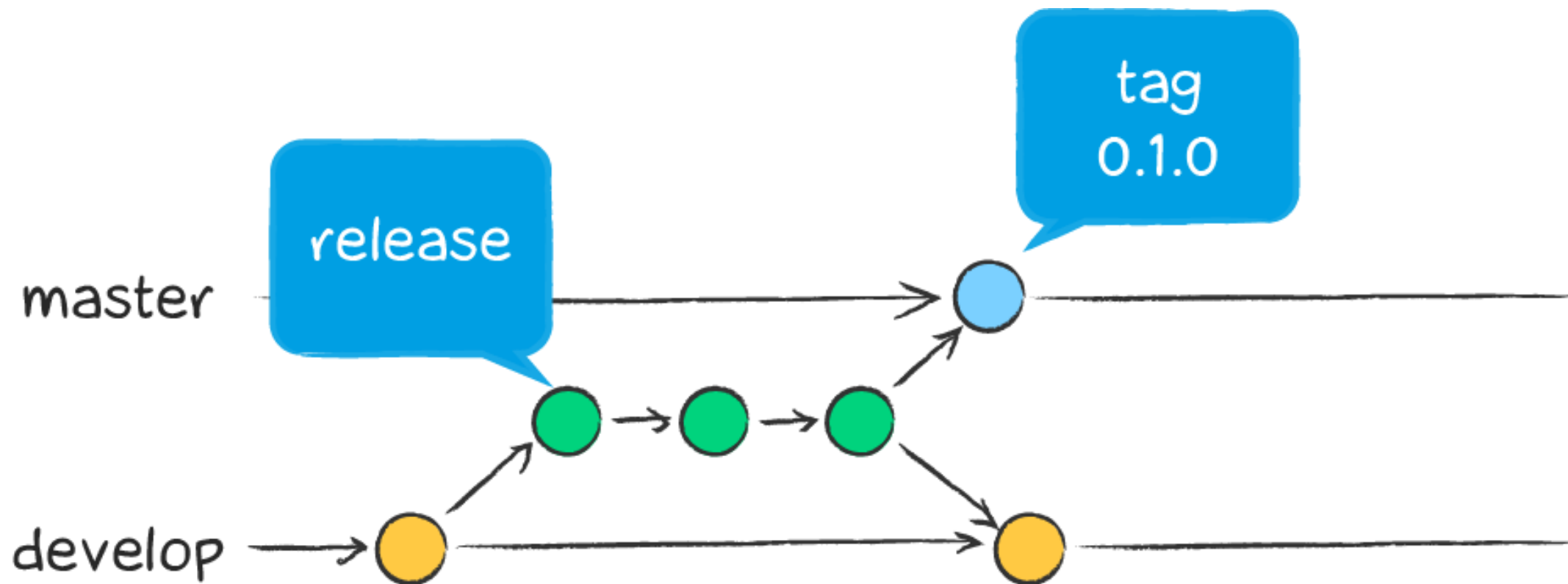
Develop branch

master



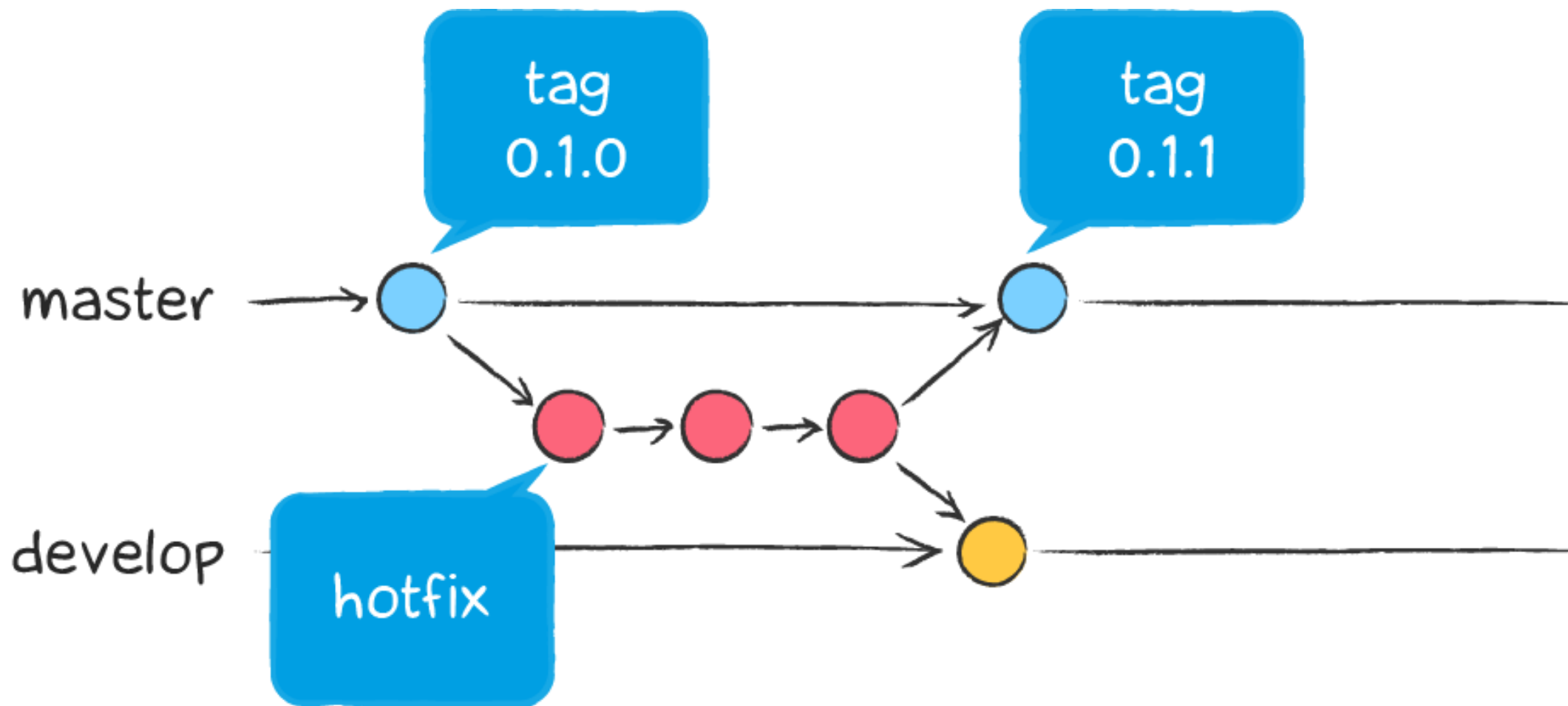


Release branch



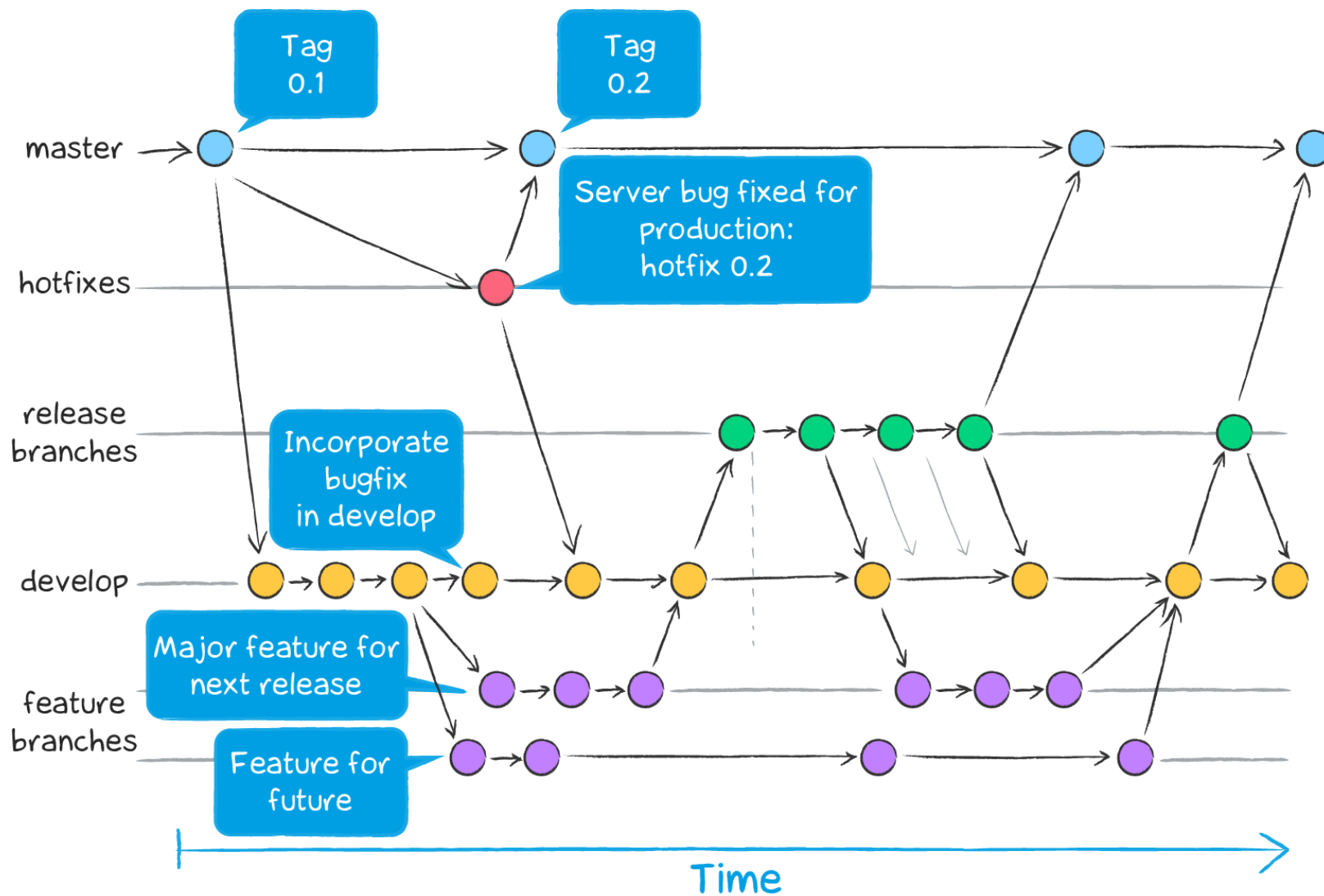


Hotfix branch



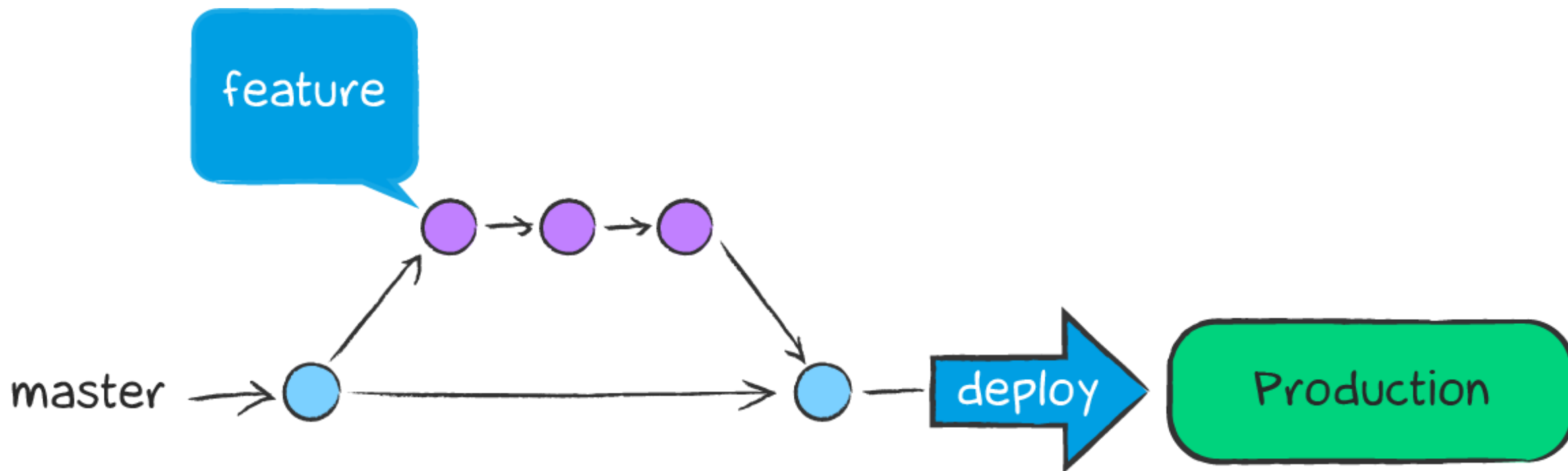


Gitflow



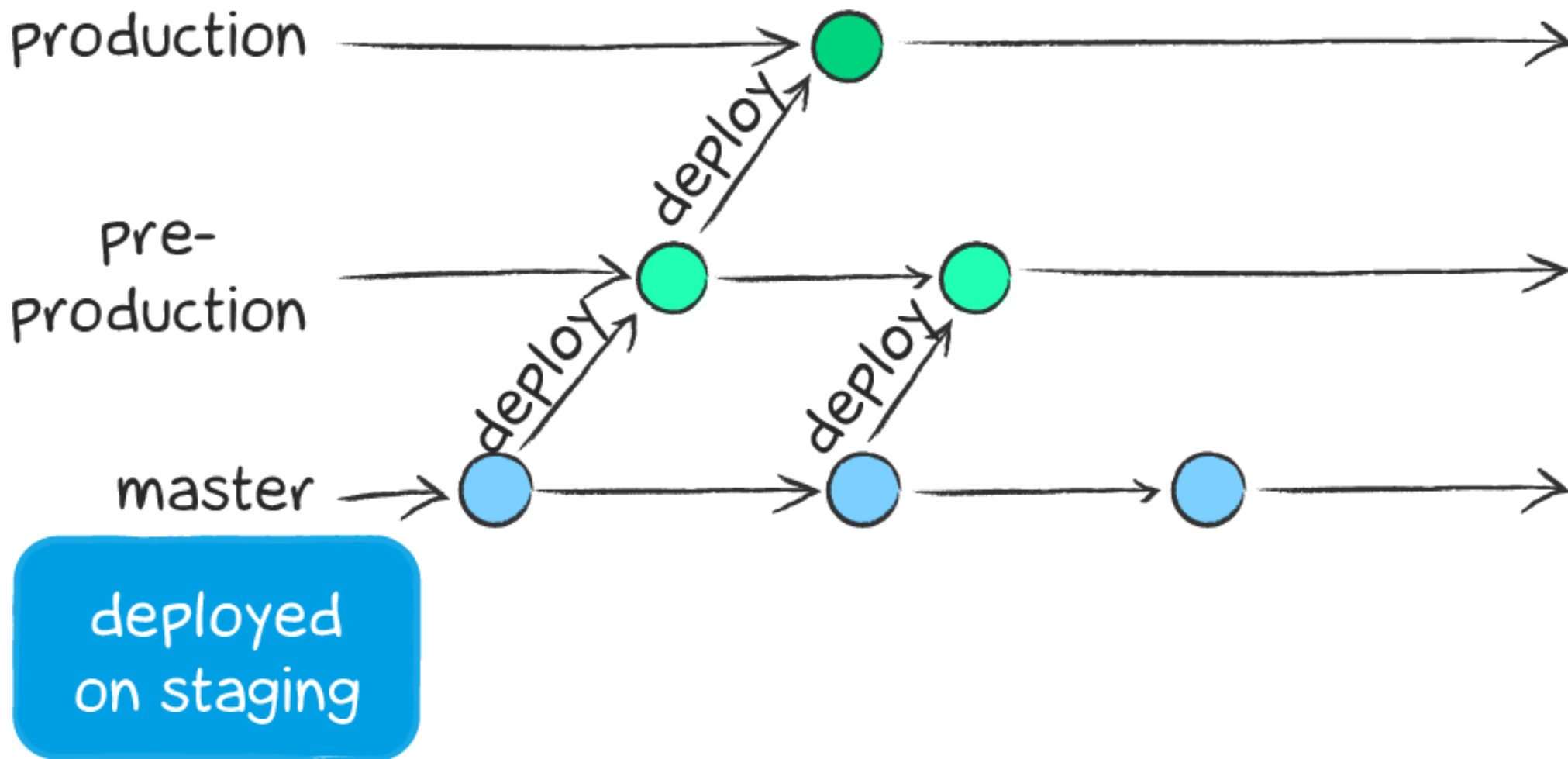


GitHub flow





Gitlab flow





slurm.io