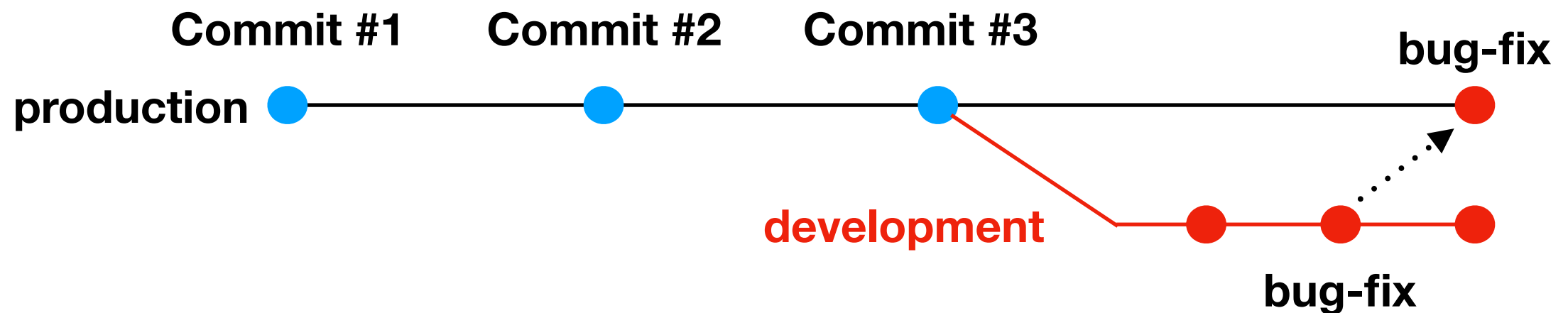


GIT

cherry-pick

git cherry-pick

Используется тогда, когда нам надо "взять" один или несколько коммитов из другой ветки в нашу ветку



Нам нужен единственный коммит **[bug-fix]** из ветки **development**

Метафора



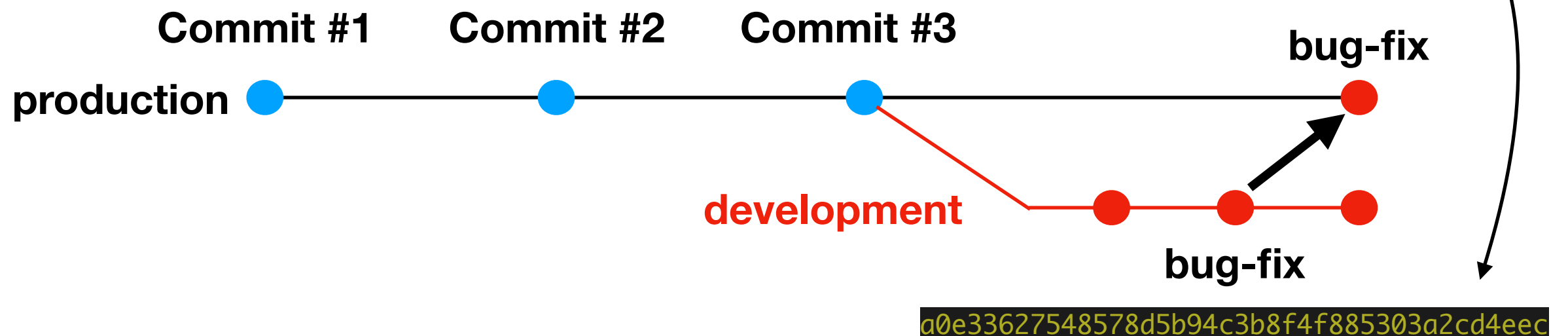
cherry pick - отбирать только хорошие вишенки

git cherry-pick

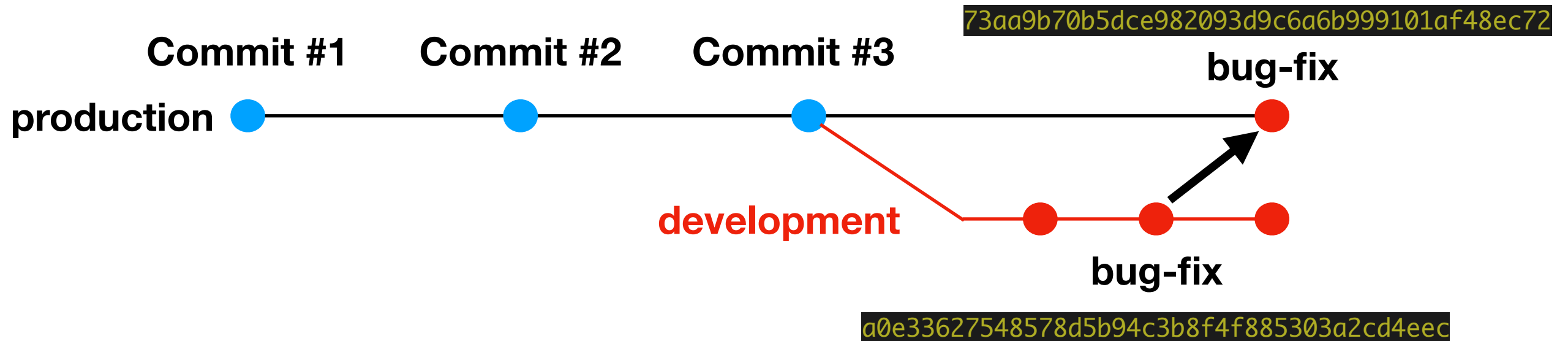
Убеждаемся, что мы в
правильной ветке

git checkout production

git cherry-pick хэш_коммита



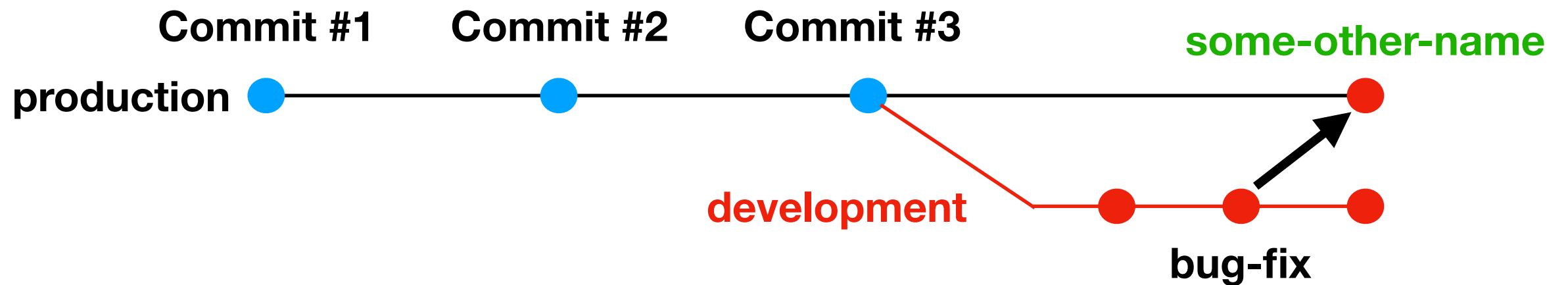
git cherry-pick



- У взятого коммита будет другой хэш, отличный от хэша в оригинальной ветке (то есть формально это **НОВЫЙ КОММИТ**)

*будет разобрано на практике

git cherry-pick --edit

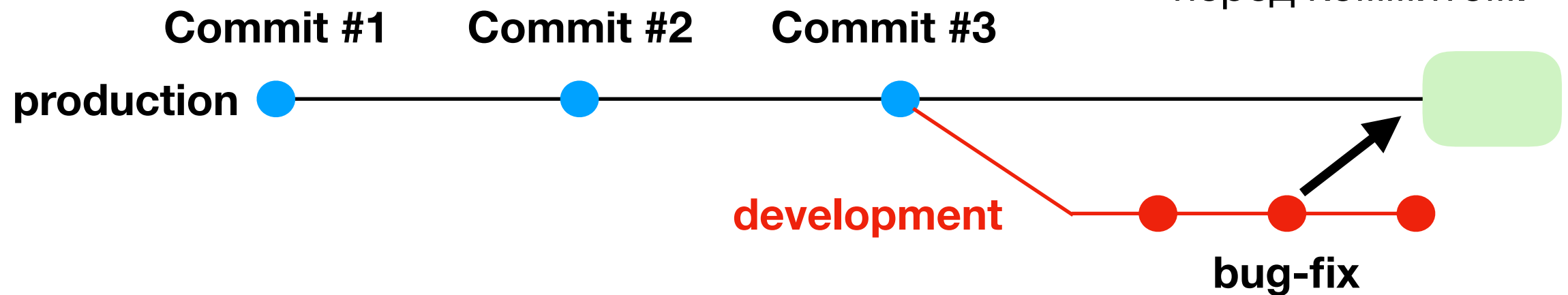


- Хотим перенести коммит из другой ветки, но при этом хотим поменять сообщение коммита

*будет разобрано на практике

git cherry-pick --no-commit

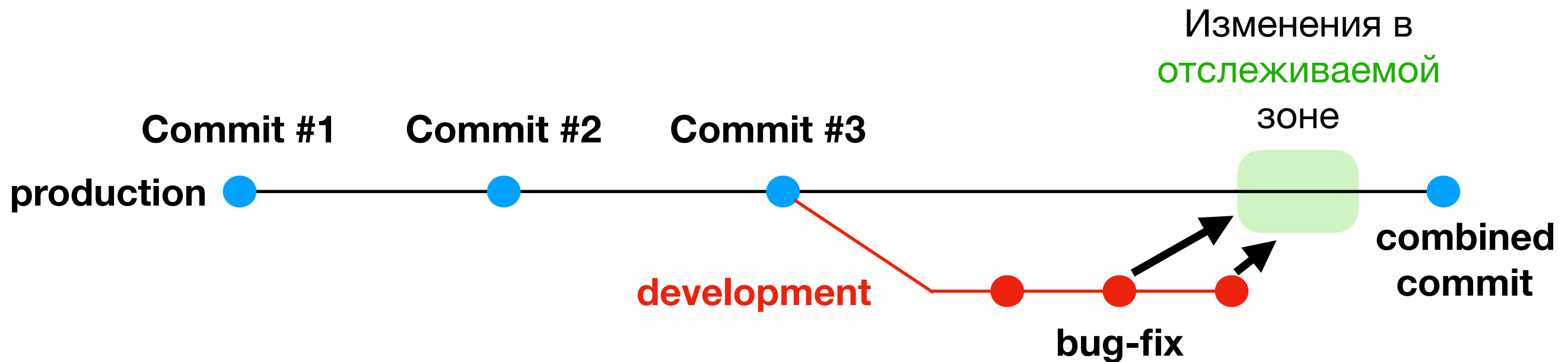
Изменения находятся в
отслеживаемой зоне.
Можем внести правки
перед коммитом.



- Хотим перенести изменения из коммита из другой ветки, но при этом не хотим делать коммит в нашей ветке. Хотим, чтобы изменения просто попали в отслеживаемую зону.
- Это бывает полезно, если мы хотим внести небольшие правки в тот **КОММИТ**, который мы забираем из другой ветки

*будет разобрано на практике

git cherry-pick --no-commit



- Другой сценарий - слияние двух коммитов из другой ветки в один коммит

*будет разобрано на практике

Другие параметры

- **git cherry-pick -x** хэш_коммита
 - Указывает в сообщении коммита хэш того коммита, из которого мы сделали cherry-pick
- **git cherry-pick --signoff** хэш_коммита
 - Указывает в сообщении коммита имя того пользователя, кто совершил cherry-pick

*будет разобрано на практике

Практика