УДК 004.4
ББК 32.973.2-018.2
С28

Книгу «Git tips and tricks» на английском языке можно найти здесь: https://opensource.com/downloads/git-tricks-tips.
Она распространяется по лицензии
Creative Commons Attribution-ShareAlike 4.0.

В книге представлены статьи, написанные Брентом Ластером, Раджживом Берой, Сетом Кенлоном, Олафом Альдерсом, Рамакришной Паттнаиком, Агилом Антонием и Ноа Барки.

Секреты Git. Никогда не теряй свой код! — Москва : Издательство АСТ, 2024. — 96 с. : ил. — (Учимся программировать).
С28 ISBN 978-5-17-160269-7.

Данное издание может стать незаменимым помощником для любого пользователя системы контроля версий Git, в наши дни широко распространенной в сфере разработки программного обеспечения благодаря своей «продвинутости» и универсальности. Авторы разделов этой книги — опытные разработчики, постоянно и активно использующие Git в решении повседневных задач при написании и систематизации программного кода. В частности, здесь рассмотрены восстановление предыдущих состояний системы в случае неудачных действий с Git, возможность отправки в удаленный репозиторий выборочных коммитов, особенности быстрого переключения с одной задачи на другую без потери прогресса в работе, использование рабочих деревьев, восстановление удаленного репозитория и многое другое.

УДК 004.4
ББК 32.973.2-018.2

ISBN 978-5-17-160269-7

Перевод на русский язык: ООО «Интеджер».
Издание на русском языке: ООО «Издательство АСТ».

Содержание

Как сбрасывать, отменять и “откатываться” к предыдущим состояниям в Git

Брент Ластер

Один из наименее понятных (и наиболее важных) аспектов работы с Git — это умение легко “откатиться” к тому состоянию системы контроля версий, на котором вы остановились перед внесением изменений, то есть аспект, связанный с тем, как можно легко отменить даже значительные изменения, внесенные в репозиторий. В этой главе будет рассказано о том, как можно просто и элегантно сбросить, отменить и полностью “откатиться” к предыдущим состояниям с помощью отдельных команд Git.

Как сбросить фиксацию в Git

Начнем с команды Git `reset`. Практически вы можете считать ее “откатом” — она “откатывает” указатель

так, что он указывает на ваше локальное окружение для предыдущего коммита. Под локальным окружением подразумеваются ваш локальный репозиторий, область подготовленных для коммита файлов и рабочий каталог.

Посмотрите на рисунок 1. Здесь изображена серия коммитов в Git. Ветка в Git — это просто поименованный перемещаемый указатель на конкретный коммит. В данном случае ветка *master* — это указатель на последний коммит в цепочке.

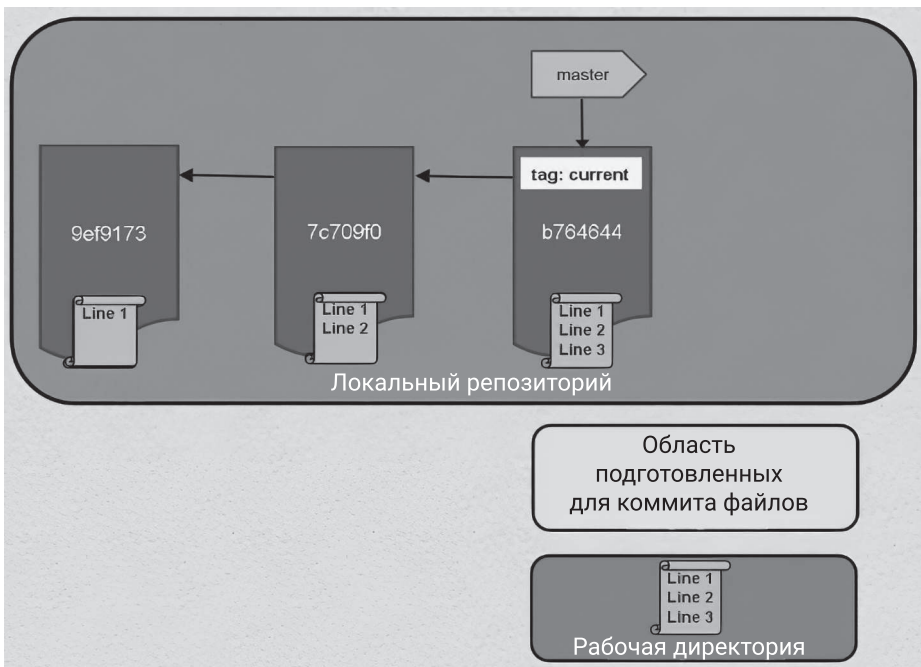

Рис. 1
(Brent Laster, CC BY-SA 4.0)

Если вы посмотрите на то, что сейчас находится в вашей мастер-ветке, вы увидите следующую цепочку коммитов, сделанных на данный момент:

```
$ git log --oneline

b764644 File with three lines // Файл с тремя
строками
7c709f0 File with two lines // Файл с двумя строками
9ef9173 File with one line // Файл с одной строкой
```

Что делать, если вы хотите “откатиться” к предыдущему коммиту? Все просто — вы можете переместить указатель ветки. Git предоставляет команду `reset`, чтобы сделать это за вас. Например, если вы хотите сбросить указатель *master* так, чтобы он указывал на коммит, находящийся на два шага назад от текущего коммита, вы можете воспользоваться одним из следующих способов:

`$ git reset 9ef9173` (используя абсолютное значение хеша SHA1 коммита 9ef9173)

или

`$ git reset current~2` (используя относительное значение -2 перед тегом “current”)

На рисунке 2 показаны результаты этой операции. После этого, если выполнить команду `git log` в текущей ветке (*master*), мы увидим только один коммит:

```
$ git log --oneline
9ef9173 File with one line // Файл с одной строкой
```

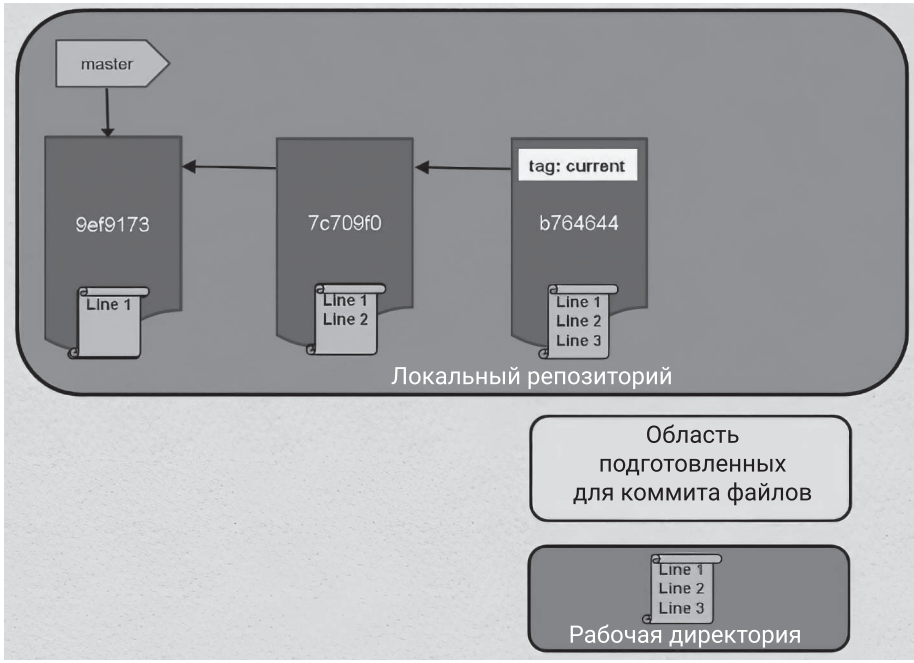

Рис. 2
(Brent Laster, CC BY-SA 4.0)

Команда `git reset` также содержит параметры для обновления других частей вашего локального окружения (область подготовленных для коммита файлов и рабочей директории) содержимым коммита, на котором вы сейчас остановились. Это такие параметры, как: `hard` — жесткий сброс коммита, на который указывал указатель до сброса, с физическим удалением информации из рабочего каталога и области индексирования; `soft` — сброс только указателя в репозитории (аналог команды `git commit --amend`); `mixed` (аналогичное поведение предлагается по умолчанию, то есть в ситуации, когда параметры отсутствуют) — сброс указателя и сброс области индексирования (удаление изменений в индексе для самого