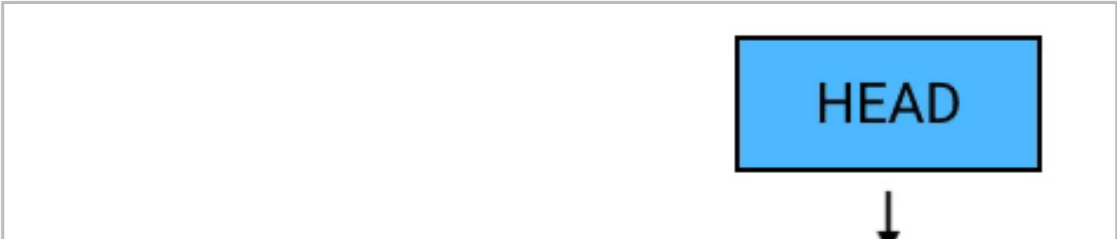


# Git 原理详解及实用指南 - 扔物线 - 掘金小册

## 高级 8: checkout 的本质

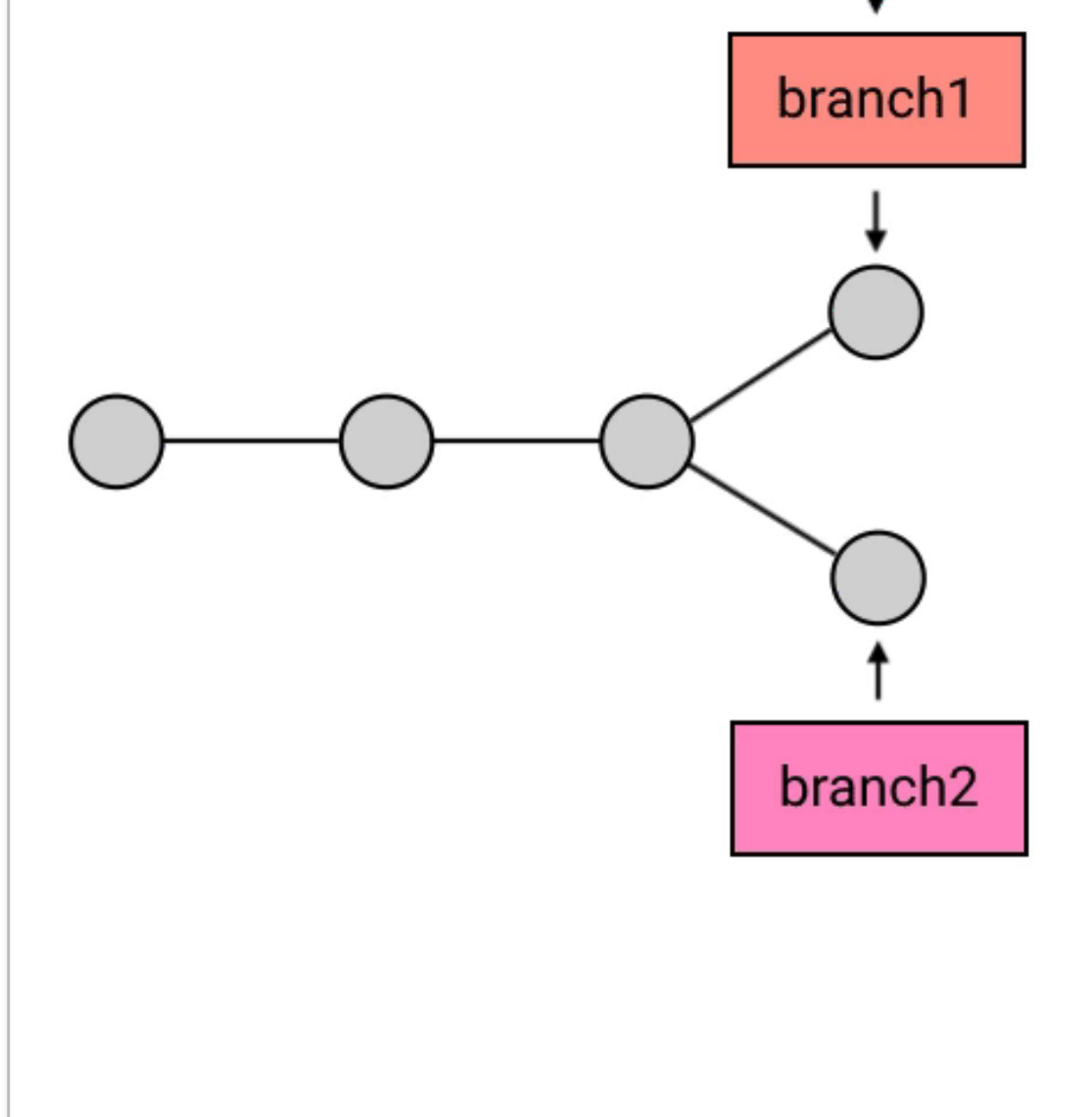
在前面的 `branch` 的部分, 我说到 `checkout` 可以用来切换 `branch` :

```
git checkout branch2
```



A diagram illustrating the state of the HEAD pointer. It consists of a blue rectangular box with the text "HEAD" inside. Below the box is a black downward-pointing arrow. The entire diagram is enclosed in a thin black rectangular border.

HEAD



checkout

不过实质上，`checkout` 并不止可以切换 `branch` 。`checkout` 本质上的功能其实是：签出（`checkout`）指定的 `commit` 。

`git checkout branch名` 的本质，其实是把 `HEAD` 指向指定的 `branch` ，然后签出这个 `branch` 所对应的 `commit` 的工作目录。所以同

样的，`checkout` 的目标也可以不是 `branch` ，而直接指定某个 `commit` ：

```
git checkout HEAD^^
```

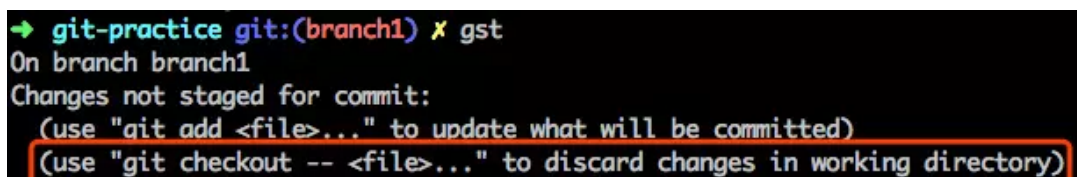
```
git checkout master~5
```

```
git checkout 78a4bc
```

```
git checkout 78a4bc^
```

这些都是可以的。

另外，如果你留心的话可能会发现，在 `git status` 的提示语中，Git 会告诉你可以用 `checkout -- 文件名` 的格式，通过「签出」的方式来撤销指定文件的修改：



```
➔ git-practice git:(branch1) ✕ gst
On branch branch1
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

modified:   game.txt
```

## 小结

---

这节的内容是对 `checkout` 的本质进行简述：

`checkout` 的本质是签出指定的 `commit`，所以你不止可以切换 `branch`，也可以直接指定 `commit` 作为参数，来把 `HEAD` 移动到指定的 `commit`。

## checkout 和 reset 的不同

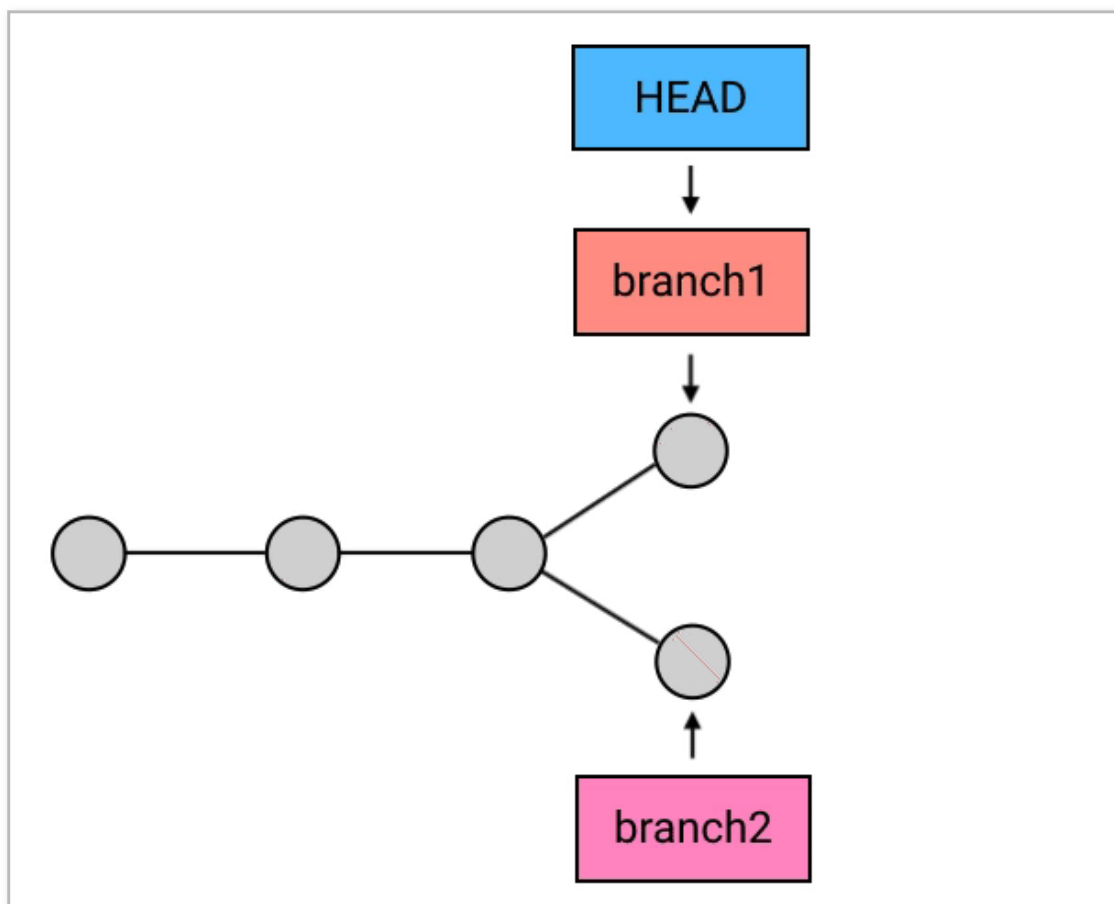
---

`checkout` 和 `reset` 都可以切换 `HEAD` 的位置，它们除了有许多细节的差异外，最大的区别在于：`reset` 在移动 `HEAD` 时会带着它所指向的 `branch` 一起移动，而 `checkout` 不会。当你用 `checkout` 指向其他地方的时候，`HEAD` 和它所指向的 `branch` 就自动脱离了。

事实上，`checkout` 有一个专门用来只让 `HEAD` 和 `branch` 脱离而不移动 `HEAD` 的用法：

```
git checkout --detach
```

执行这行代码，Git 就会把 **HEAD** 和 **branch** 脱离，直接指向当前 **commit**：



`git checkout --detach`

留言

评论将在后台进行审核，审核通过后对所有人可见

sailingfaraway java web

checkout 是带着 head 走，reset 是带着两个走

▲0 评论 1 月前



sailingfaraway java web

让分支移动的方法有 commit (merge、rebase 类似)、reset，噢，好像没了，最主要的就是 reset

▲0 评论 1 月前



卢炜杰

最后一点，脱离的，有什么实际效果？在什么场景下可以用到？

▲0 收起评论 2 月前



订阅公众号

看下面的评论，大佬有说

1 月前

评论审核通过后显示

评论

九天揽月在掘金

关于“签出”一词的含义，能进一步说明一下吗？似乎不是个日常会用到的词汇

▲0 评论 2 月前

NiuAnqi Android

可以有一些应用场景吗？detach 之后有啥用处呢？

▲0 收起评论 3 月前

• 订阅公众号  
下面的评论有说  
1 月前

评论审核通过后显示

评论

有看没懂这些都是干啥用的

▲0 评论 5 月前

---



**blankul** web 前端开发 @ dtdream

和楼上一样 checkout 和 reset 的不同 这部分还能详细些吗

▲0 评论 6 月前

---



**ClimberJing** 后台开发 @ 携程

checkout 和 reset 的不同 这部分还能详细些吗

▲0 评论 8 月前

---



**pangrongxians** Android 开发 @ 广州创必承

试试评论看看

▲0 评论 8 月前

---



**钢铁大白菜** Android 开发 @ Chinapex



请问 git checkout --detach 的应用场景可能会是什么？

▲1

收起评论 9 月前



扔物线

Android 布道师

(假装) @

HenCoder

怎么舒服怎么用，但不常用。举例的话，比如你想在某个 commit 出写一些临时的新的 commits，但不确定是否保留这些内容（或者倾向于不保留），那么你就：1. 不希望把当前的 branch 带跑；2. 不想新建一个别的 branch。那么你就 detach 一下，这样就可以在后面的 commits 中只有 HEAD 在跟着 commits 跑，而 branch 一直留在原地。

9 月前

---

- Loren1994  
androider @ 青  
岛

回复 扔物线：请  
问如果是你说的第二  
种场景, 那 push 之  
后 HEAD 会重新指  
向这个分支吗?

8 月前

---

- 钢铁大白菜  
Android 开发 @  
Chinapex

回复 扔物线：三  
个月前认认真真看了  
凯哥的回答，当时感  
觉懂了，又好像哪里  
不太懂，现在回头整  
个小册再看一遍，更  
明白了些，感谢凯哥  
的小册，小册读一遍

和读两遍，是不一样的收获~~~

5 月前

- 毛毛大婶

回复 [扔物线](#)：这个应用场景举例真棒，赞一个

2 月前

- 豆芽酱 [Android](#)

git checkout --  
detach 分离后，可以 git checkout  
[branch\_name] 重新绑定

28 天前

评论审核通过后显示

评论

- 

**微风 161027** 安卓开发工程师 @ 苏宁  
没看明白.....

▲0

评论 9 月前



小葱  前端

留名

▲0

评论 9 月前



noone9

感谢

▲0

评论 9 月前

---

全文完

---

本文由 简悦 SimpRead 优化，用以提升阅读体验。