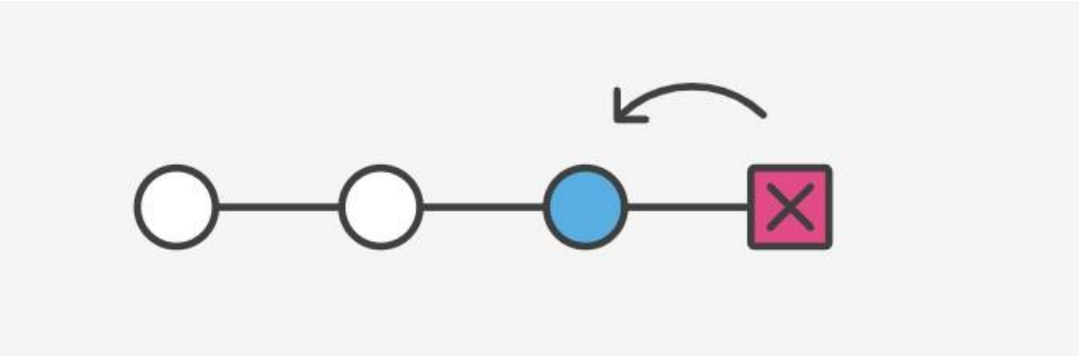
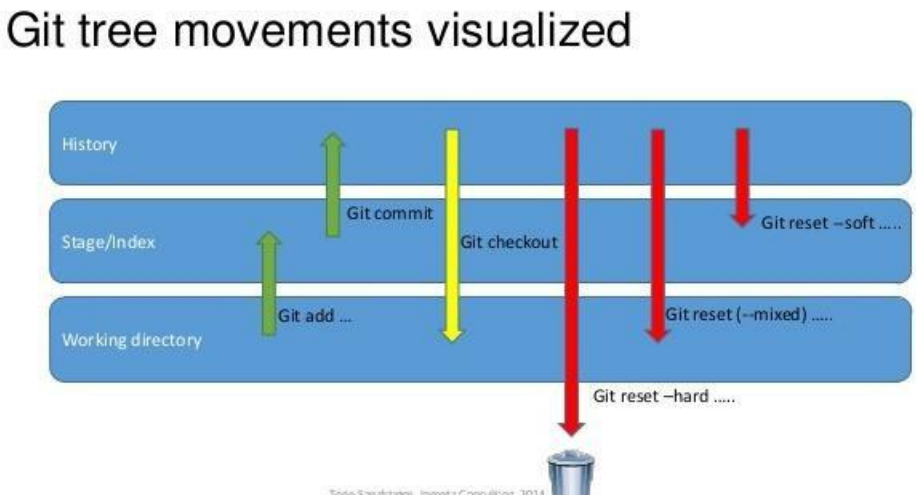
Git Reset

The git reset command is a complex and versatile tool for undoing changes. It has three primary forms of invocation. These forms correspond to command line arguments --soft, --mixed, --hard. The three arguments each correspond to Git's three internal state management mechanism's, The Commit Tree (HEAD), The Staging Index, and The Working Directory.



git reset changes, at minimum, where the current branch (HEAD) is pointing. The difference between --mixed and --soft is whether or not your index is also modified.

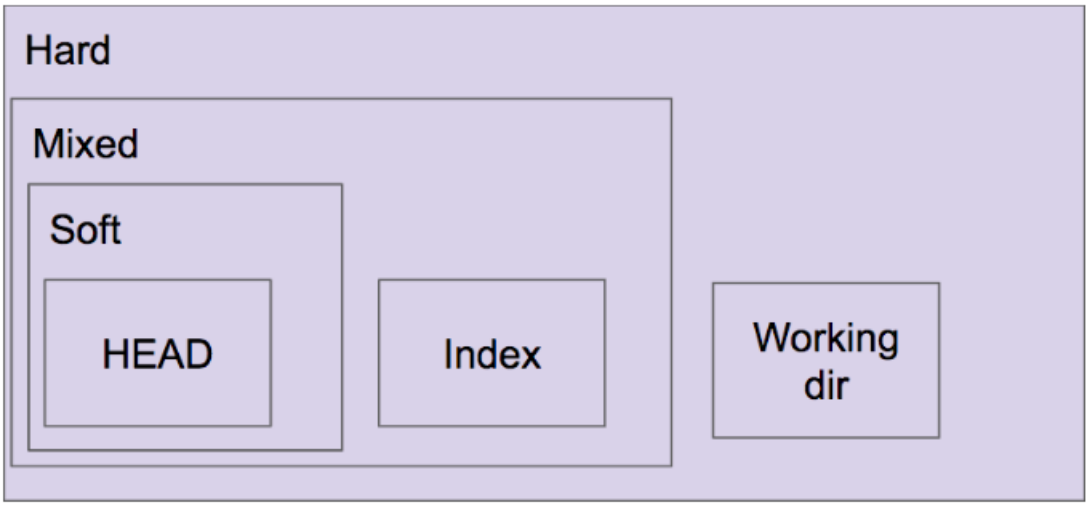


--soft: **uncommit** changes, changes are left staged (index).

--mixed (default): **uncommit + unstage** changes, changes are left in working tree.

--hard: **uncommit + unstage + delete** changes, nothing left.

`git reset --[hard/mixed/soft]` :



So each effect different scopes

1. Hard => WorkingDir + Index + HEAD
2. Mixed => Index + HEAD
3. Soft => HEAD only (index and working dir unchanged).

Discard the files in working area we will use **git checkout <filename>** if you want all **git checkout .**

Discard the staged (after git add .) files using **git reset HEAD *** or **git reset HEAD filename**

Remove the committed messages using **git reset HEAD~1**

git reset --soft HEAD~1 → --soft option will remove commit still files kept in staged area and working area.

git reset HEAD~1 → by default its mixed option it will undo your staging area and keep it working directory

git reset --hard HEAD~1 → --hard option will remove commits a and also files removed in staged area and working area both.

https://davidzych.com/difference-between-git-reset-soft-mixed-and-hard/