



# git

---

*GIT & GITHub: Revert in GIT*

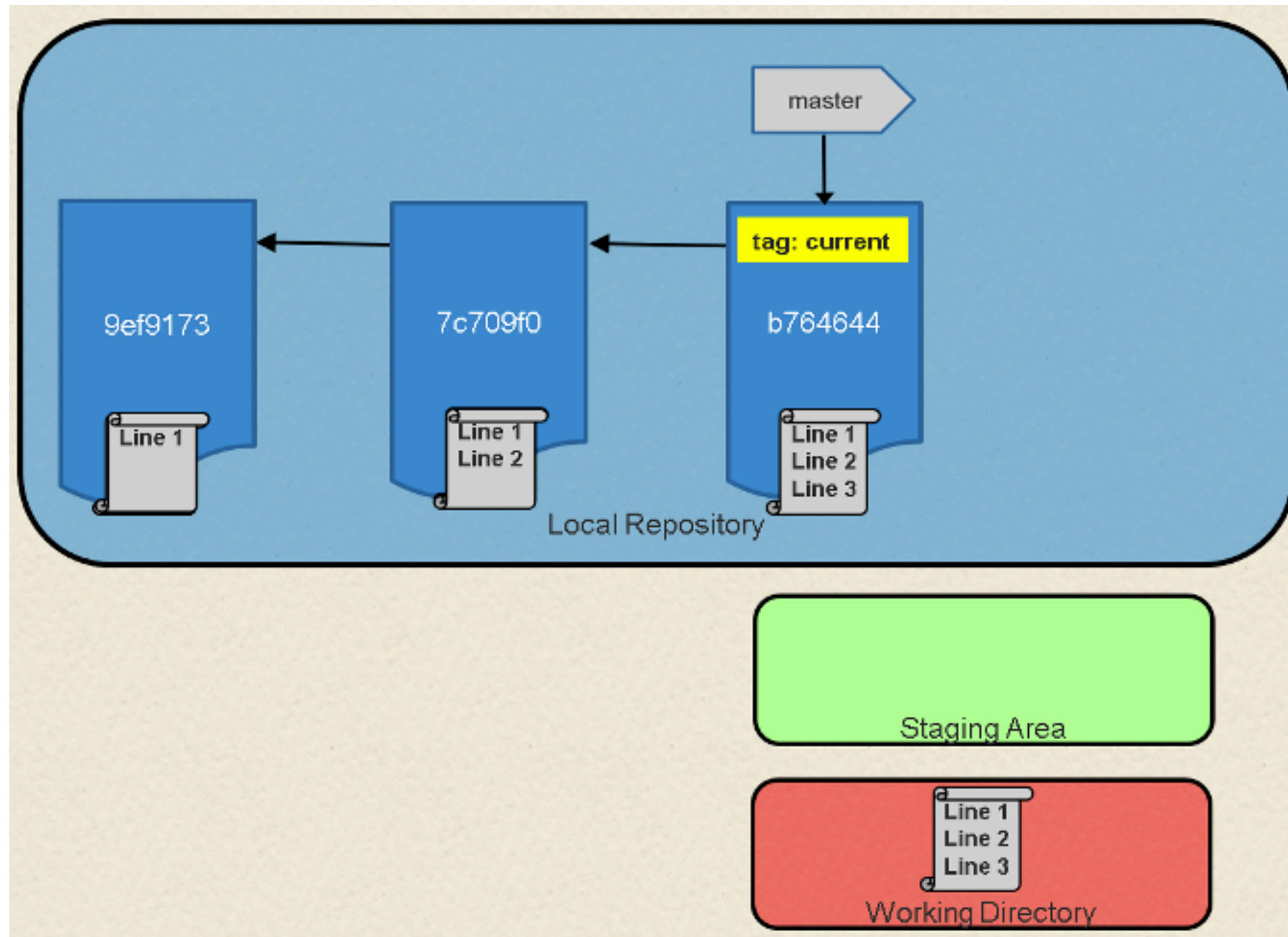
## ***GIT & GITHub : Version Control System***

---

- Revert is all about to Undo the changes, you did in repo.
- In GIT this can be done via RESET and REVERT.
- **RESET** - Practically, user can think of it as a “rollback”.
- Reset points local environment back to a previous commit.

# ***GIT & GITHub : Version Control System***

.....



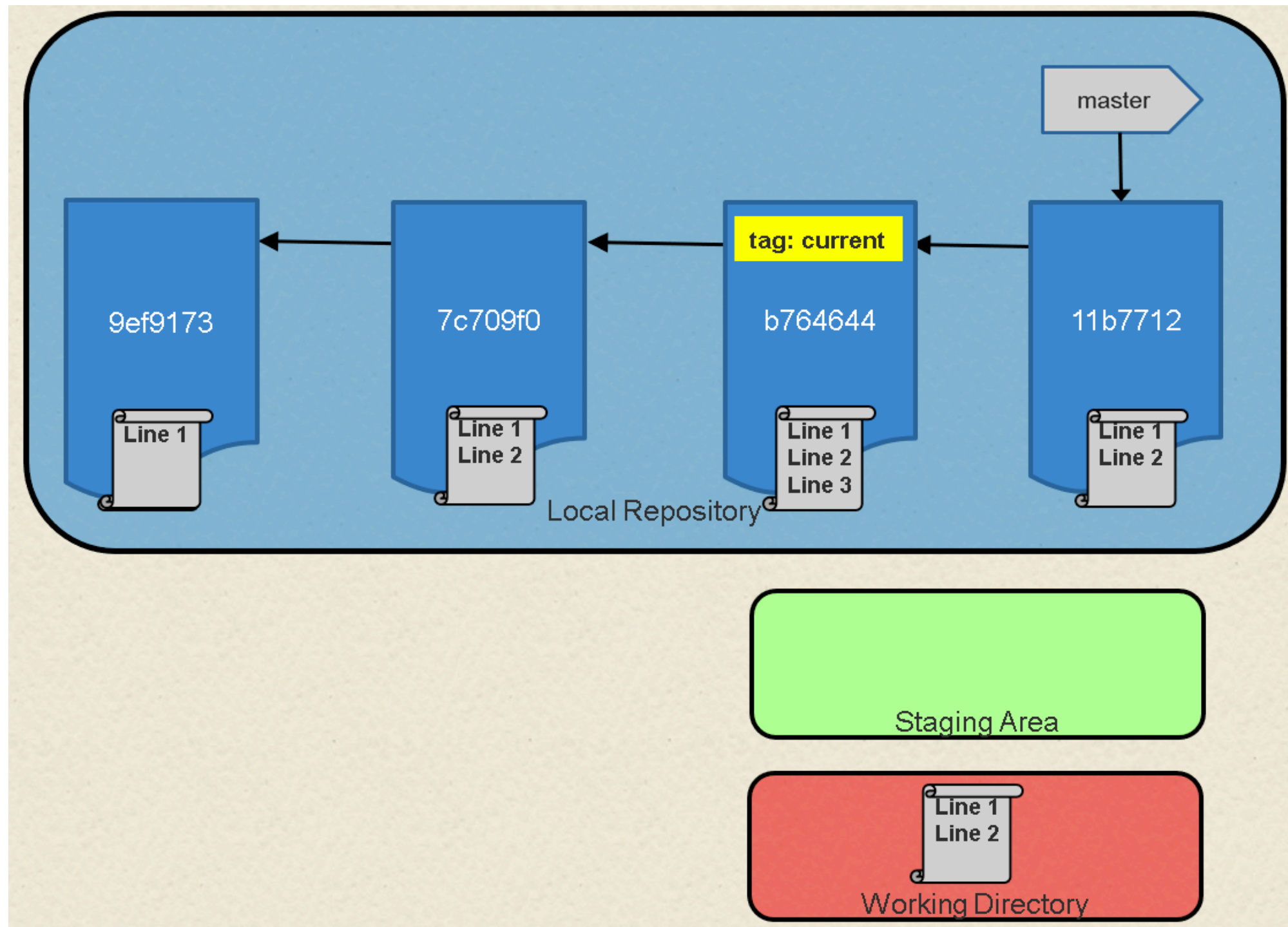
## ***GIT & GITHub : Version Control System***

---

- **REVERT** - Net effect of the *git revert* command is similar to reset, but its approach is different.
- Revert adds a new commit at the end of the chain to "**cancel**" changes.
- **Revert or Reset ?**
- If user have already pushed commits to the remote repo, a **revert** is a nicer way to cancel out changes.
- Git workflow works well for picking up additional **commits** at the end of a branch, but it can be challenging if a set of commits is no longer seen in the chain when someone resets the branch pointer back.
- If Commit in local then Reset is good, If commit is pushed then revert is good option.

# *GIT & GITHub : Version Control System*

.....



*Will see you in Next Lecture...*

---

*Thank you!*

A close-up photograph of a hand holding a black marker, completing the cursive word 'Thank you!' on a white surface. The marker is positioned at the end of the exclamation mark, and the hand is visible on the right side of the frame.

*See you in next lecture ...*