

Using git without alternate branches

- `git add .` (or `git add filename`)
- `git commit -m "One line description of what was changed"`
- `git push`
- `git fetch`
- `git pull`
- `git status`
- `git log`

GitHub Repository

MAIN (branch)

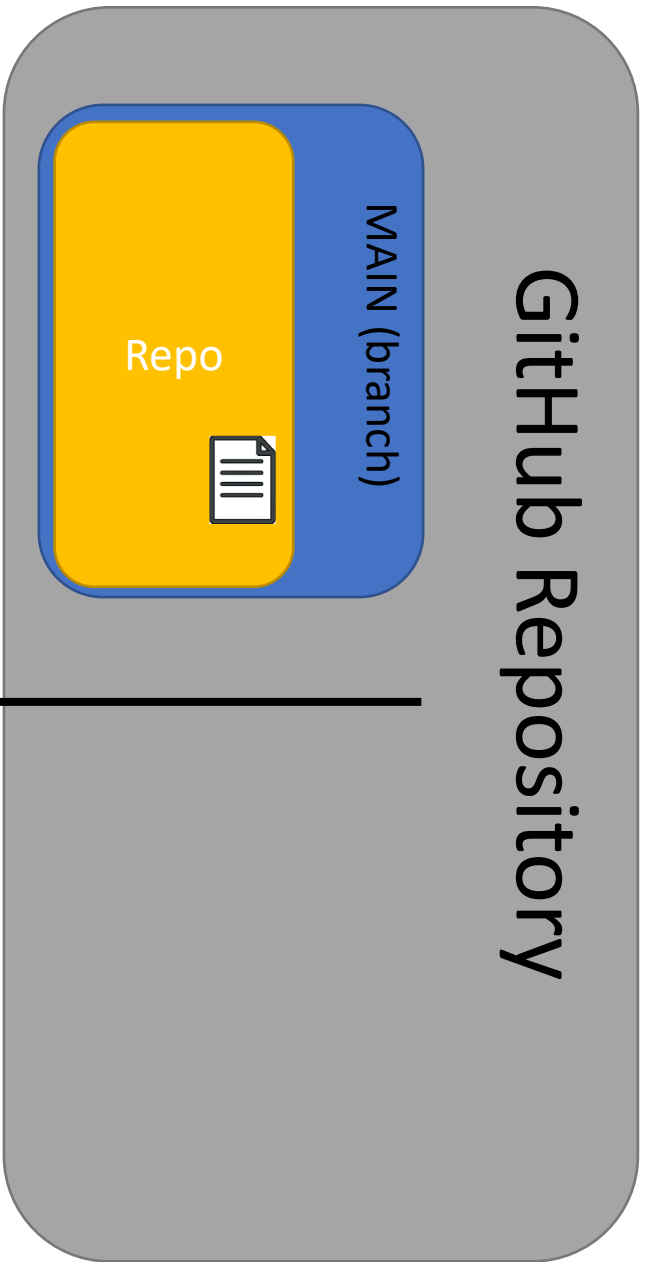
Repo



Local Computer



git clone URL



GitHub Repository

MAIN (branch)

Repo

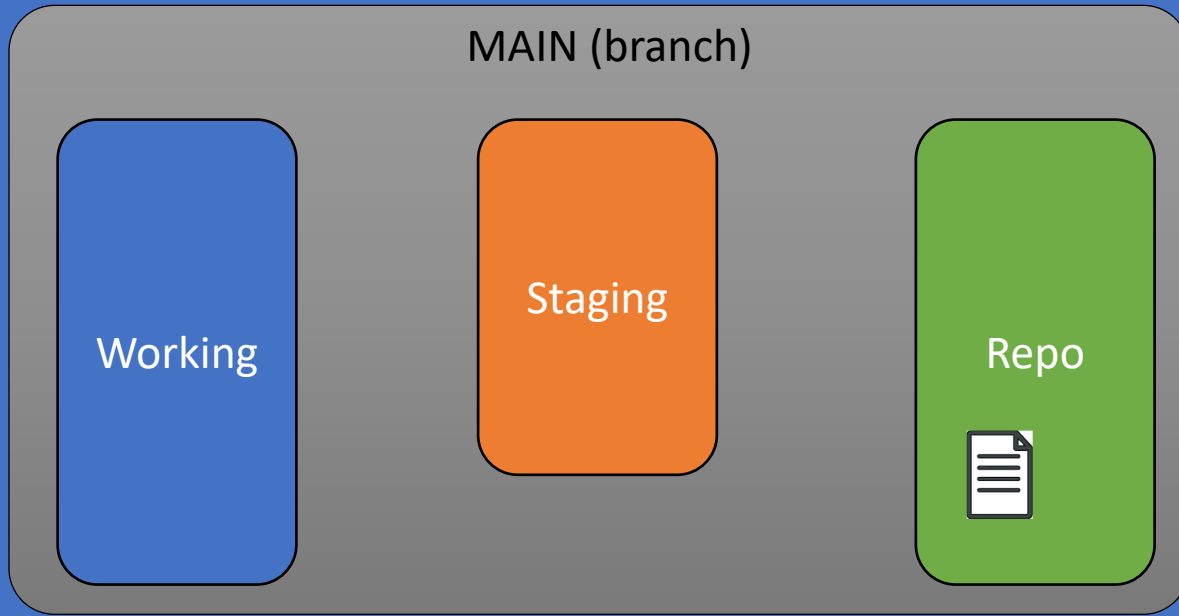


Git clone

Local Repository



Local Repository



git status:
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean



GitHub Repository



Local Repository

MAIN (branch)

Working

Staging

Repo



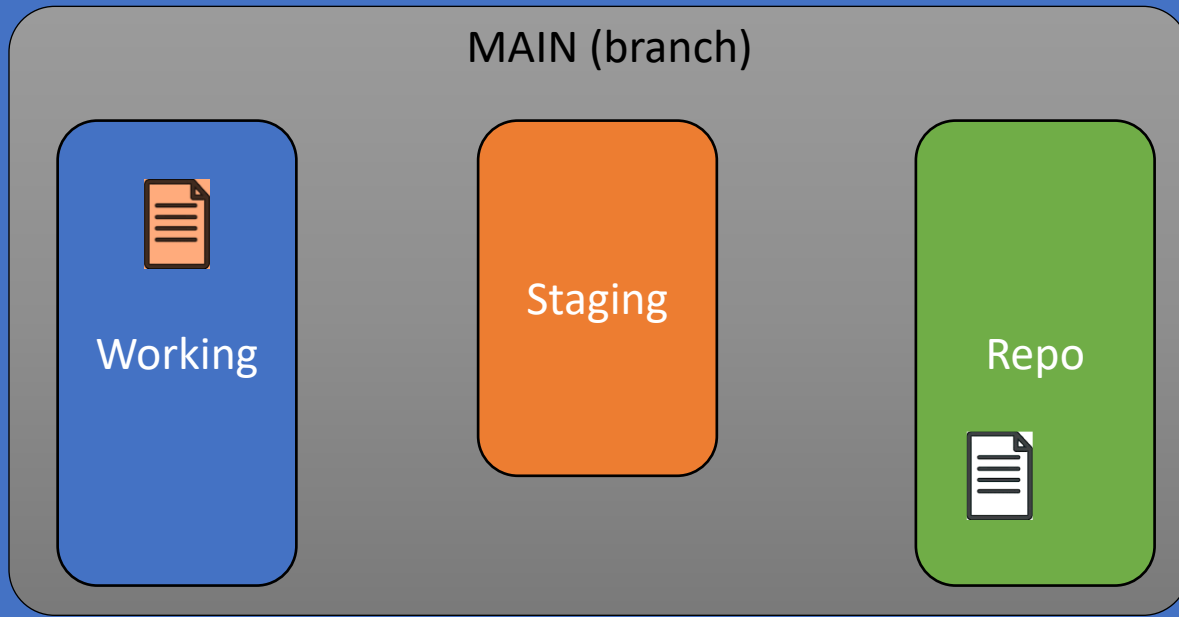
GitHub Repository

MAIN (branch)

Repo



Local Repository



git status:

On branch main

Your branch is up to date with 'origin/main'.

Untracked files:

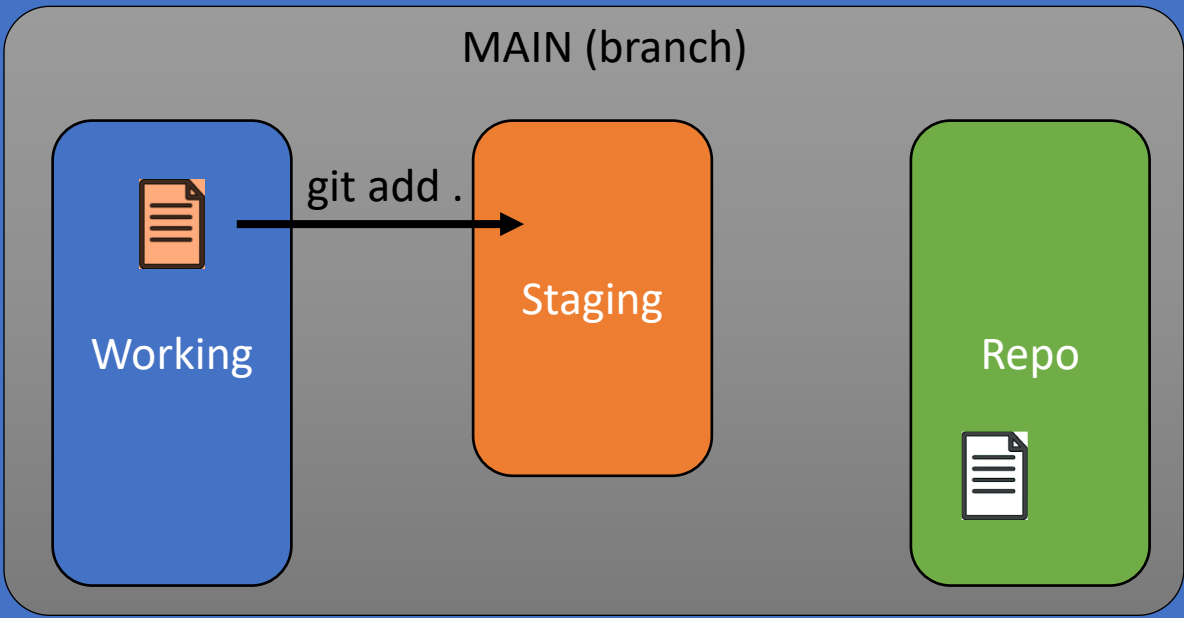
(use "git add <file>..." to include in what will be committed)

NewFile.txt

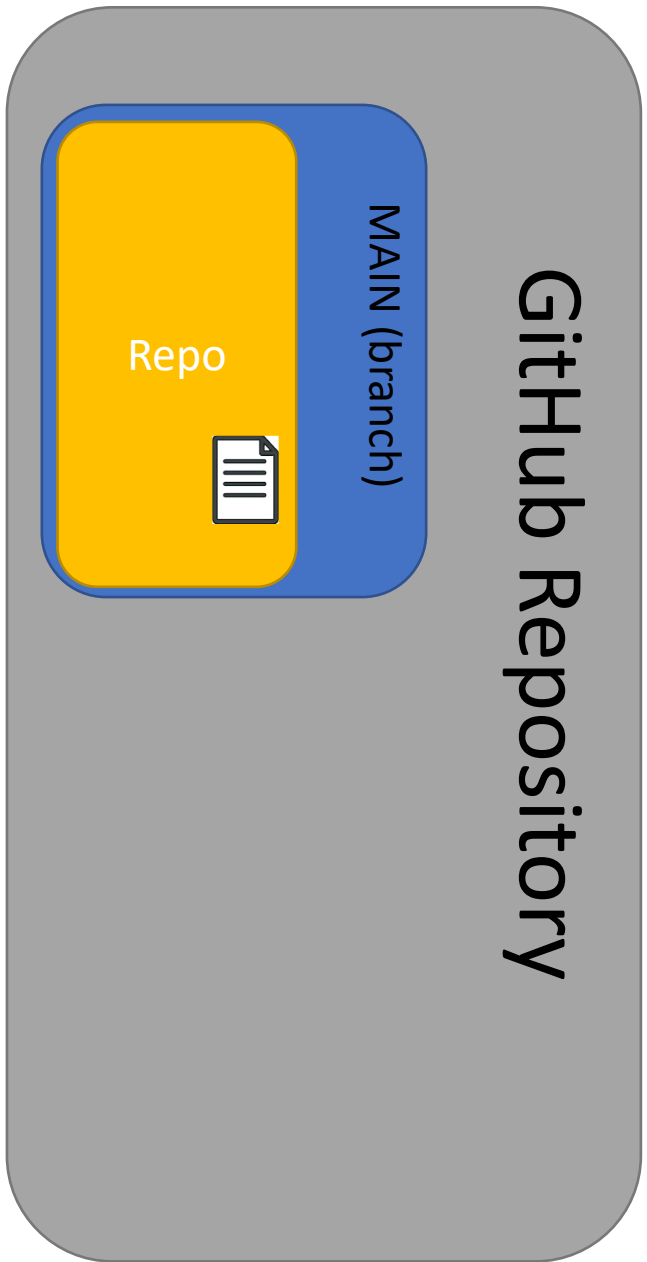
nothing added to commit but untracked files present (use "git add" to track)



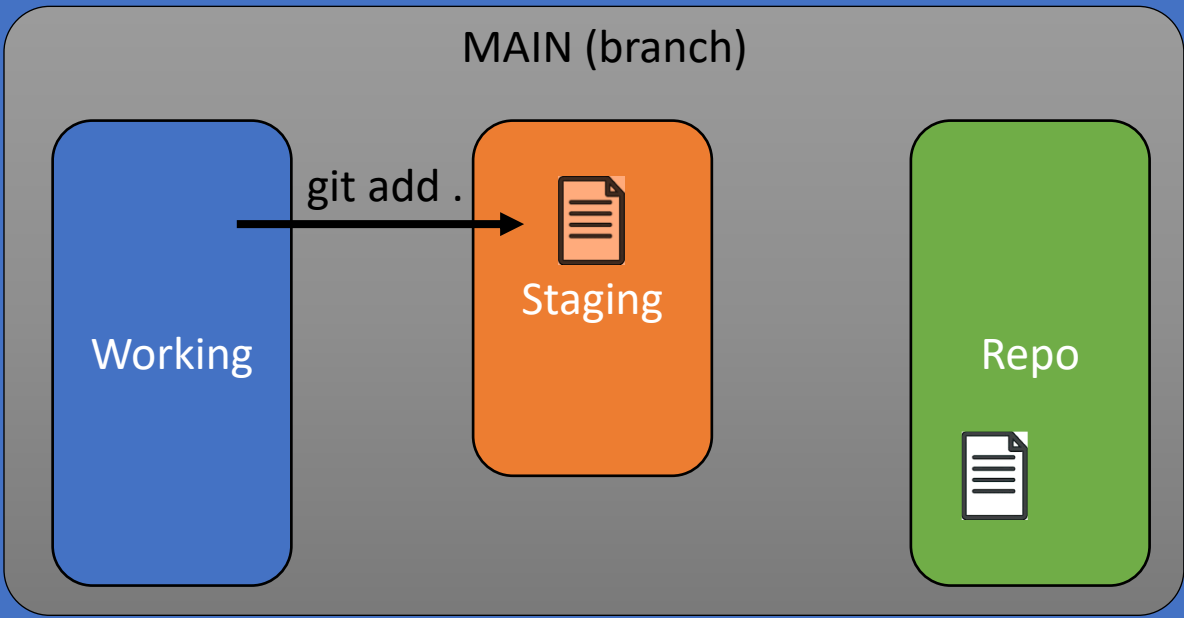
Local Repository



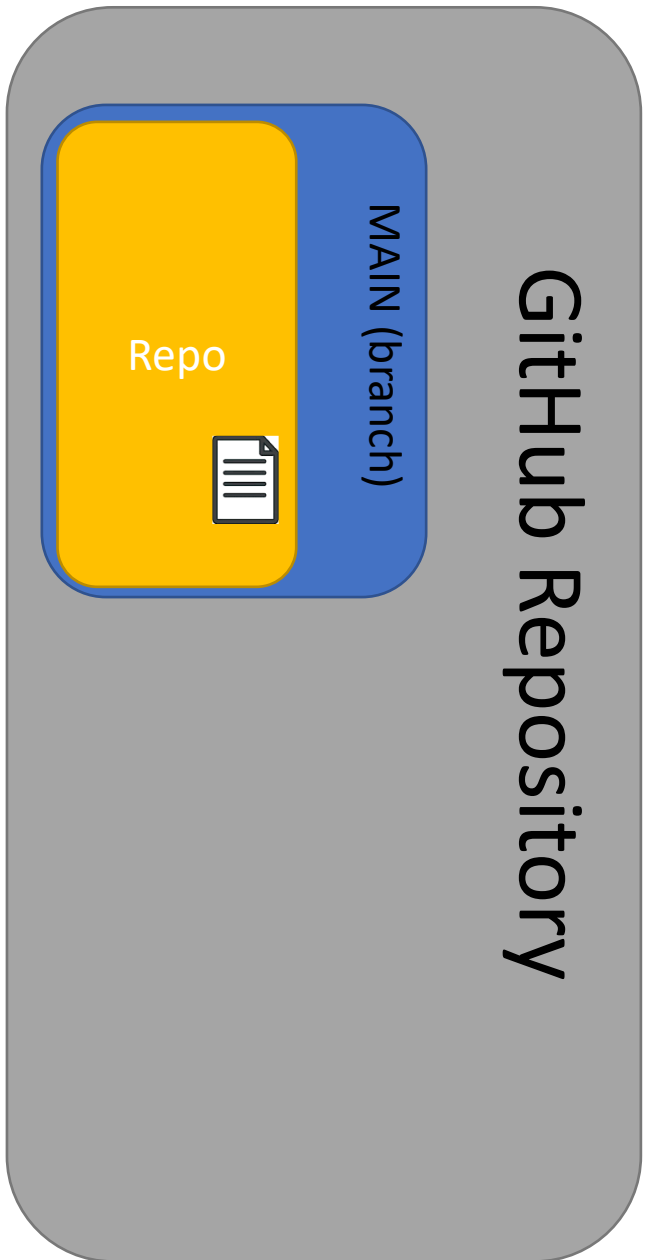
GitHub Repository



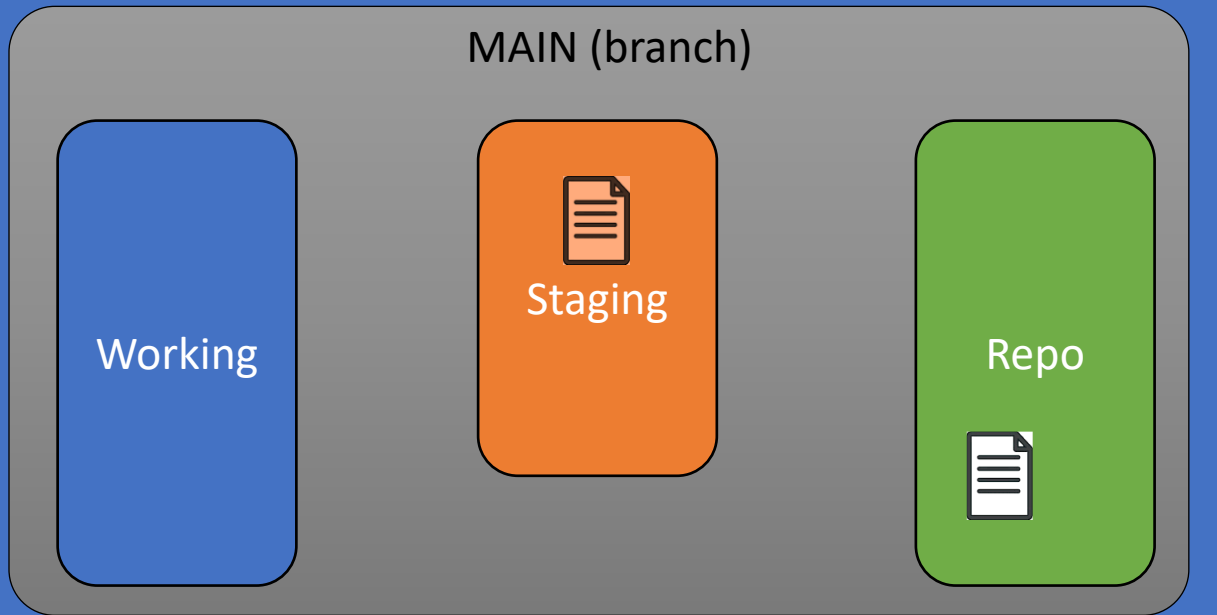
Local Repository



GitHub Repository

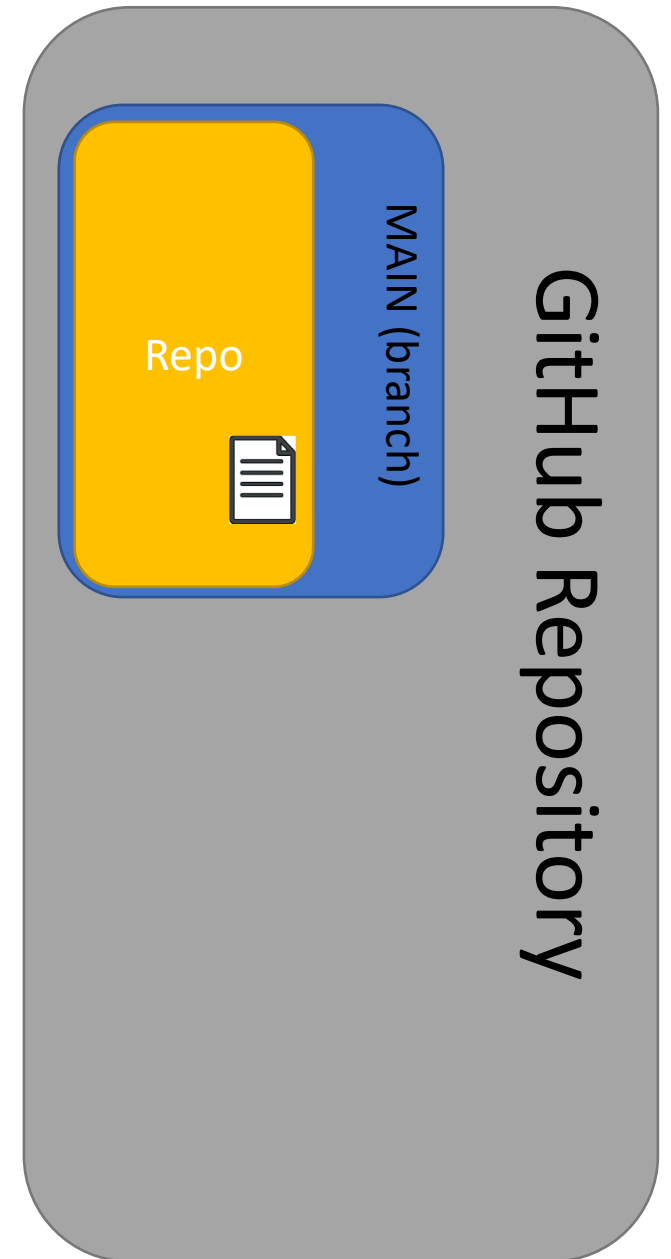
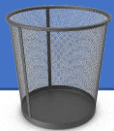


Local Repository



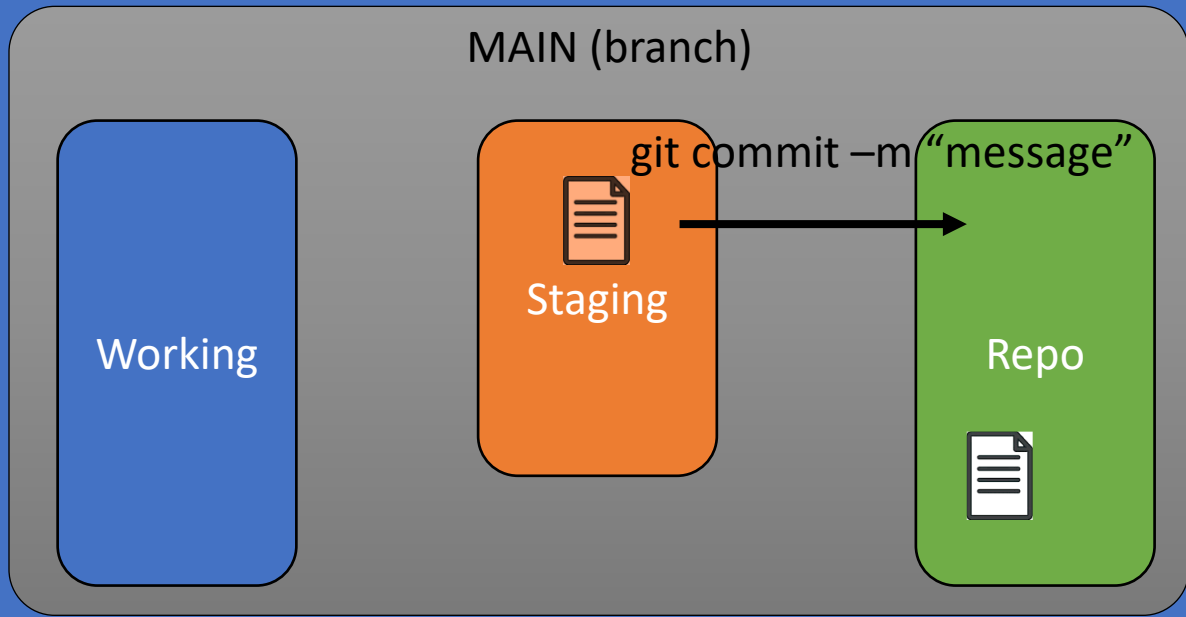
git status:
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file: NewFile.txt



GitHub Repository

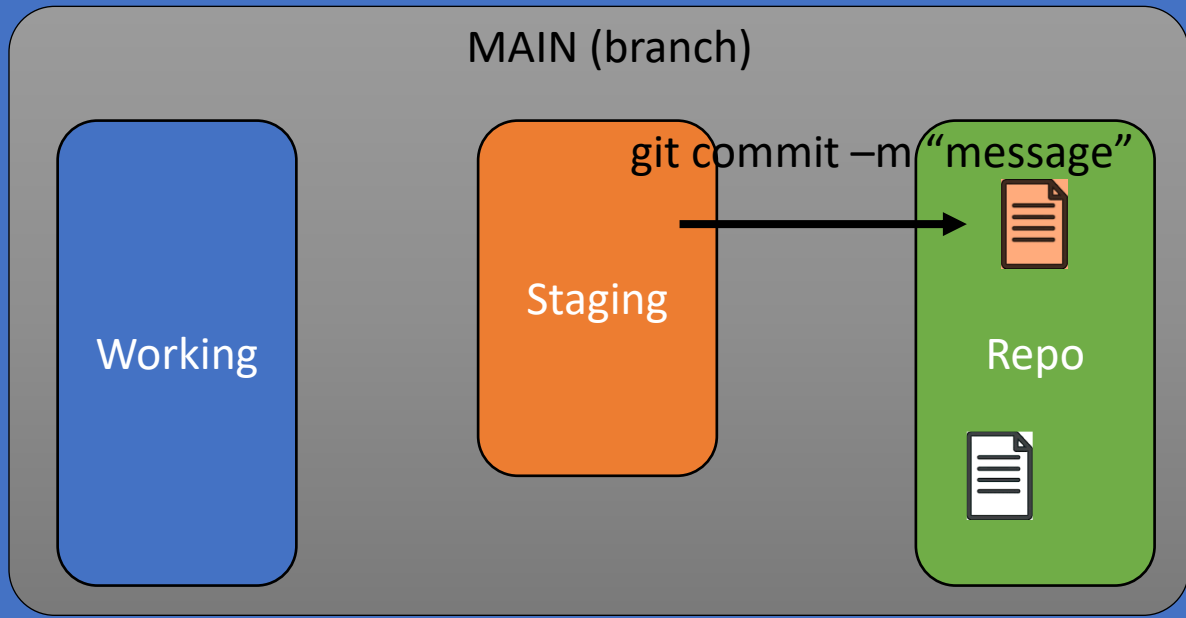
Local Repository



GitHub Repository



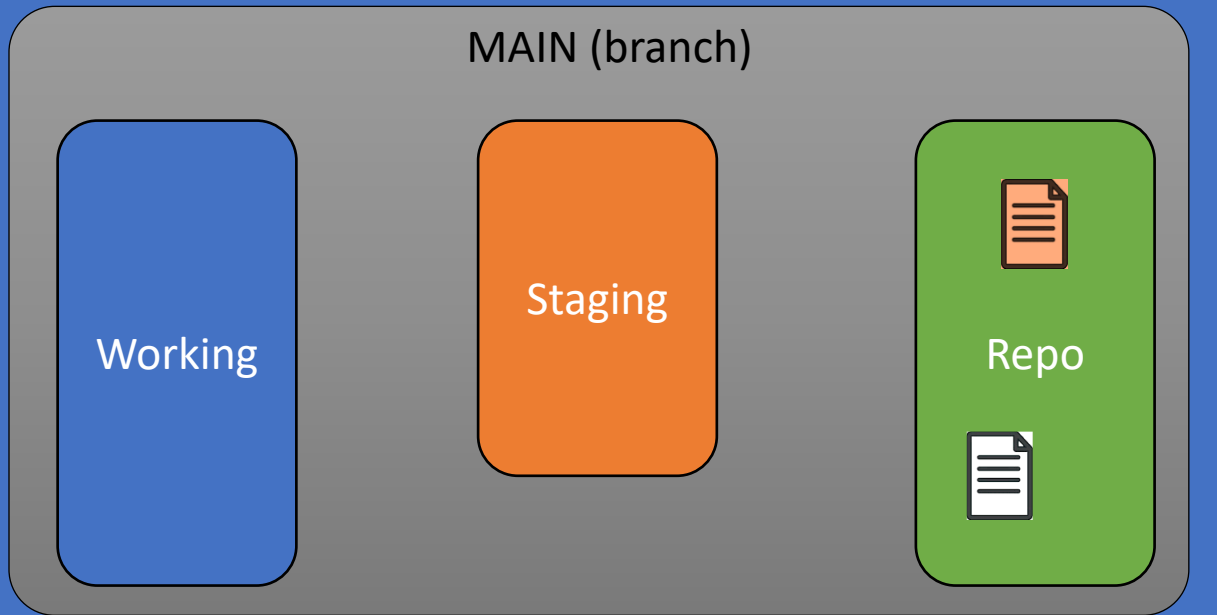
Local Repository



GitHub Repository



Local Repository



git status:

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

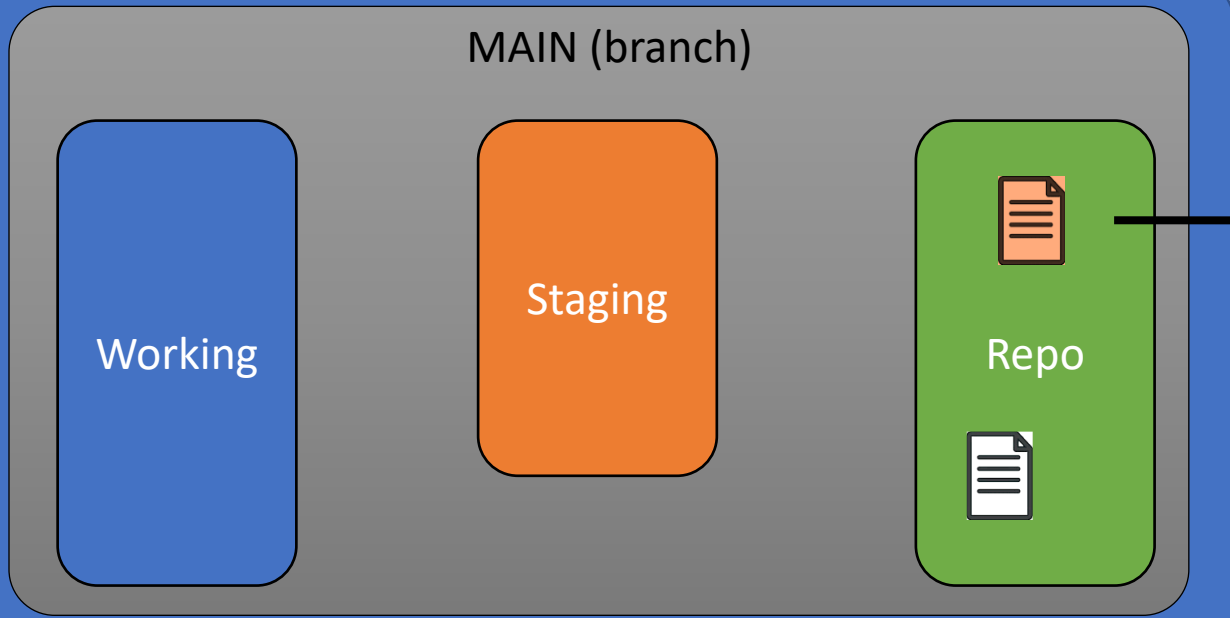
nothing to commit, working tree clean



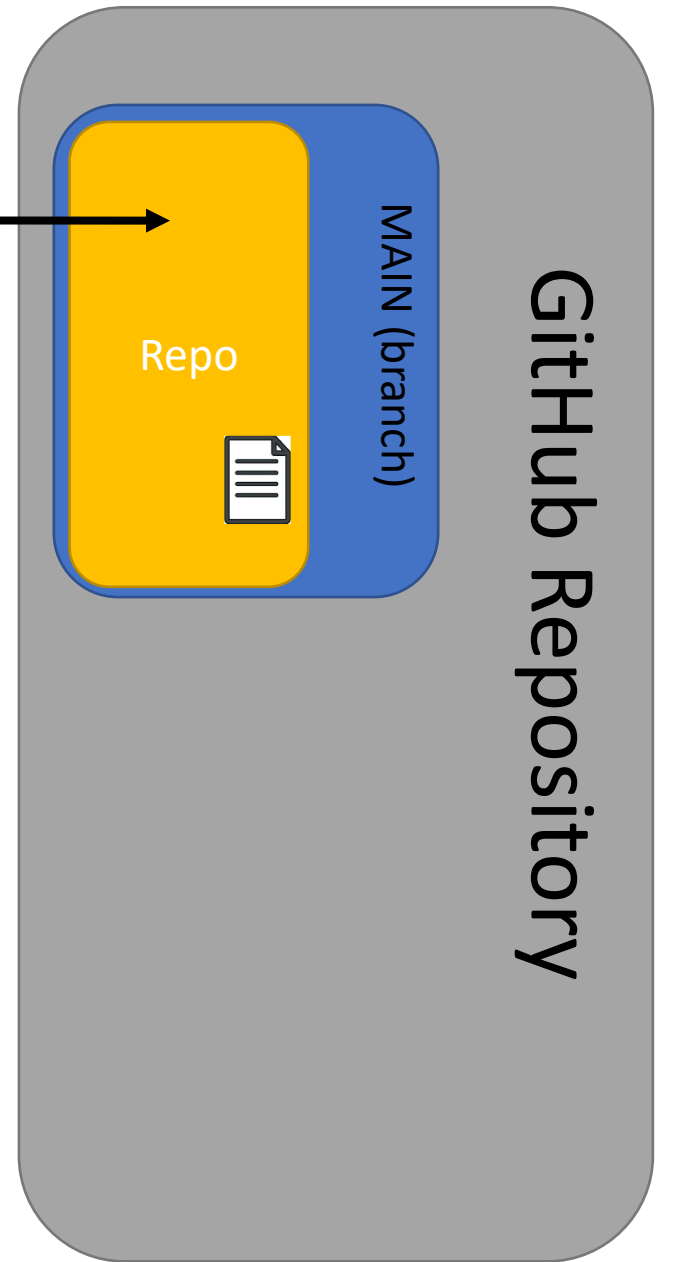
GitHub Repository



Local Repository



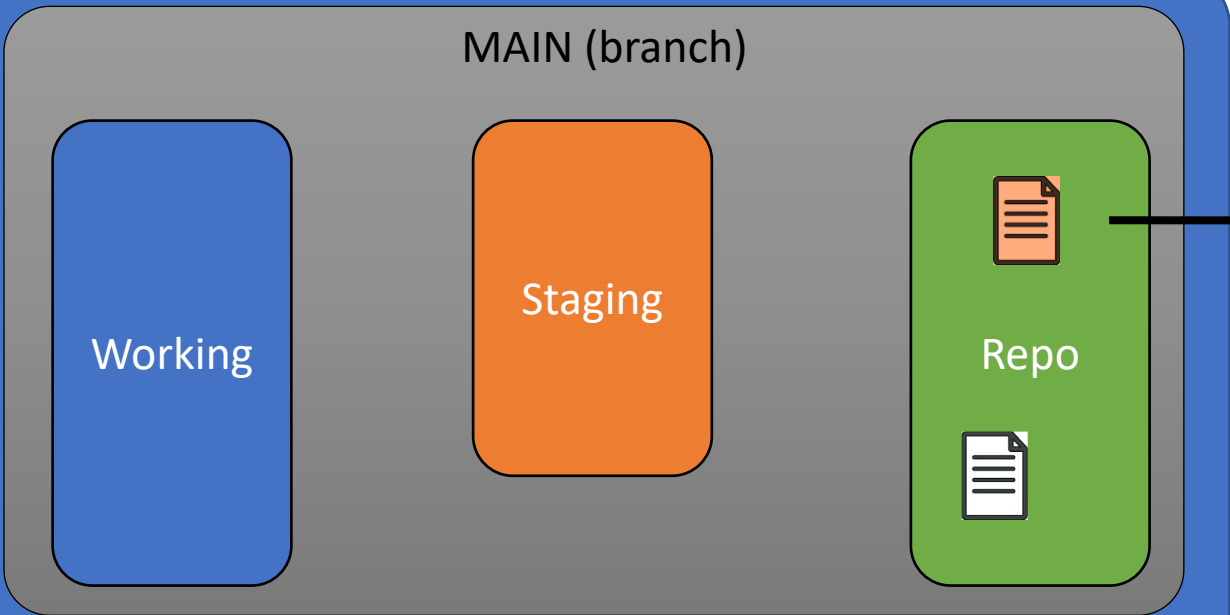
git push



GitHub Repository



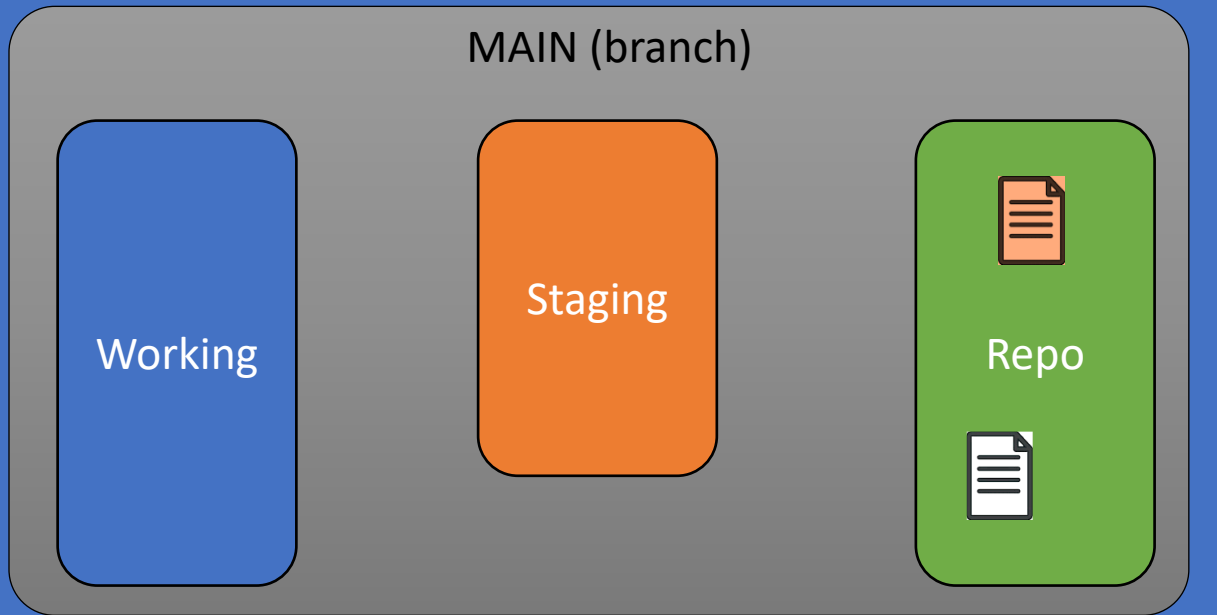
Local Repository



git push

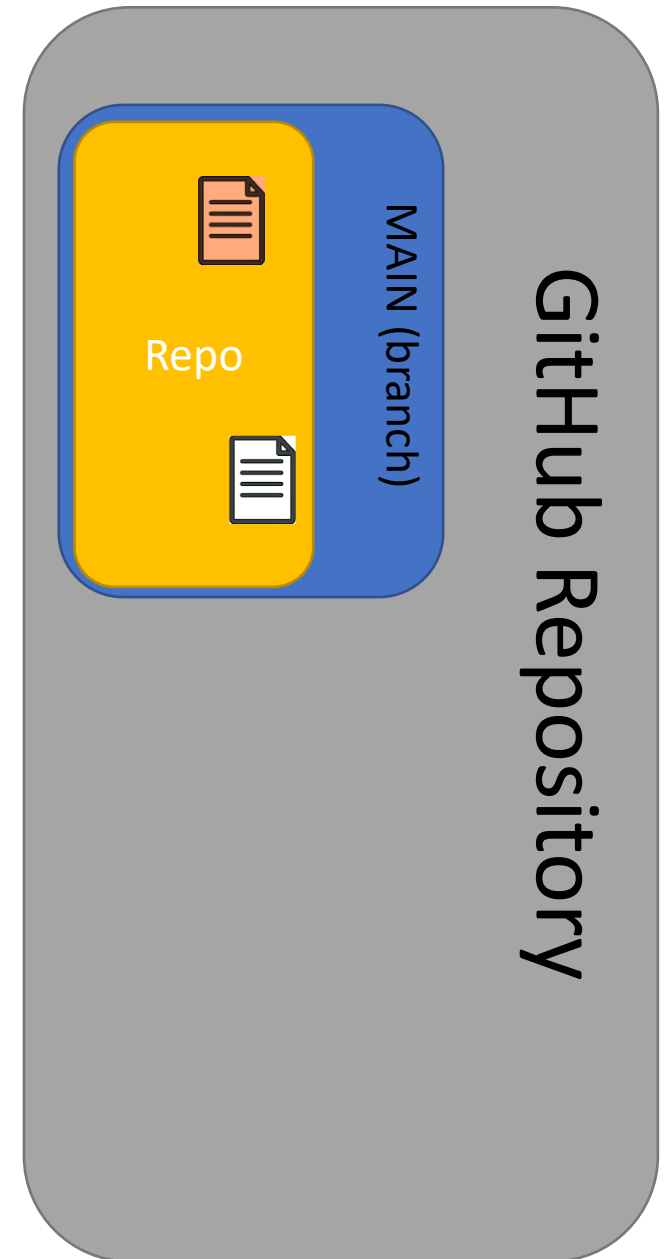


Local Repository



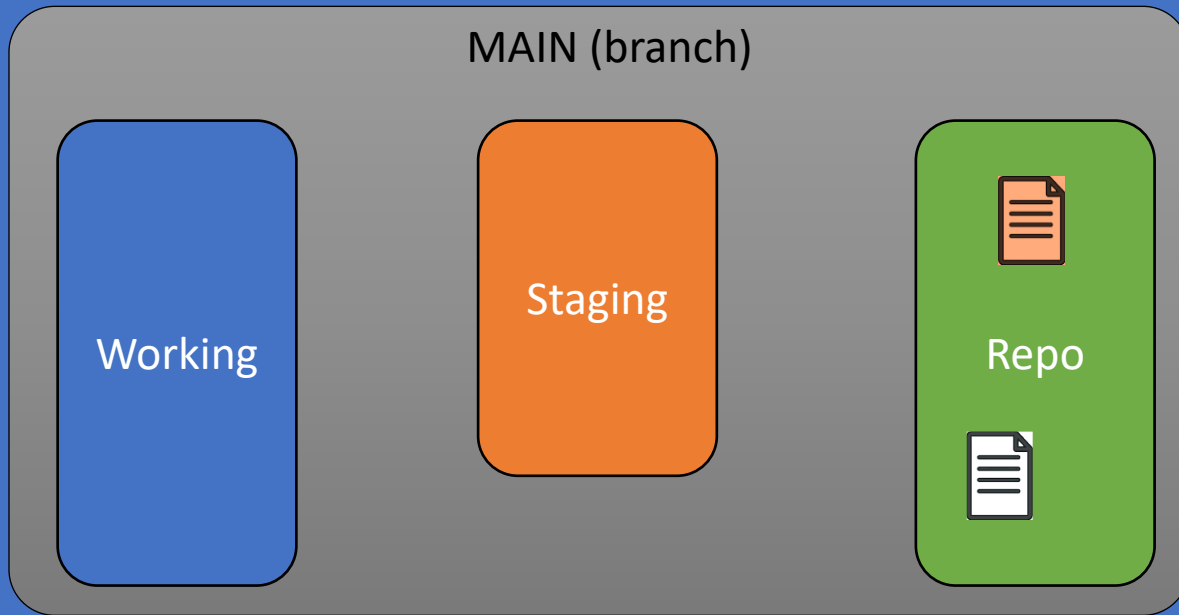
git status:
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean



GitHub Repository

Local Repository



git log

commit 8252d2894a99cd7128be02c5d807f75026e18e64
Author: Brent Denny <brent.denny@ddls.com.au>
Date: Fri Nov 26 10:52:33 2021 +1000

new file

commit 54e36e93531323b245c6ce494521f8c678ec761a
Author: BrentD09 <brent.denny@ddls.com.au>
Date: Fri Nov 26 10:48:10 2021 +1000

Create NewFile2.txt

commit 77bc09560d0b899c0c6356976625c6b919258d11
Author: Brent Denny <brent.denny@ddls.com.au>
Date: Fri Nov 26 10:44:04 2021 +1000

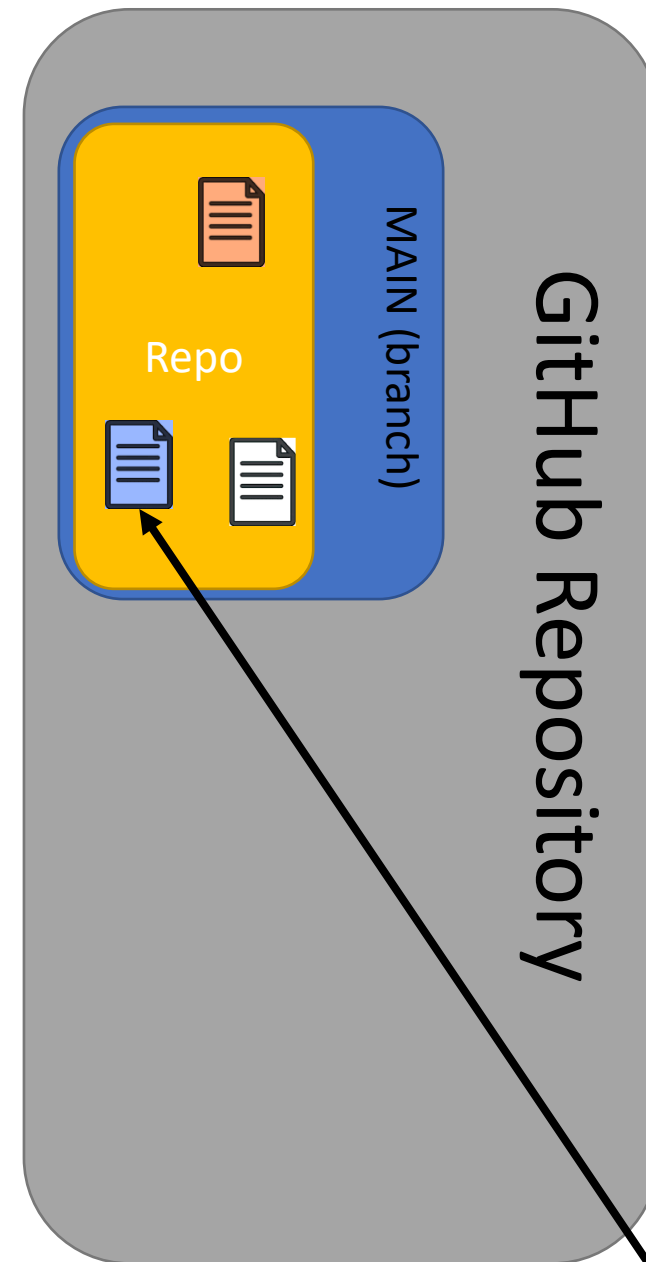
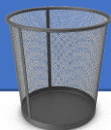
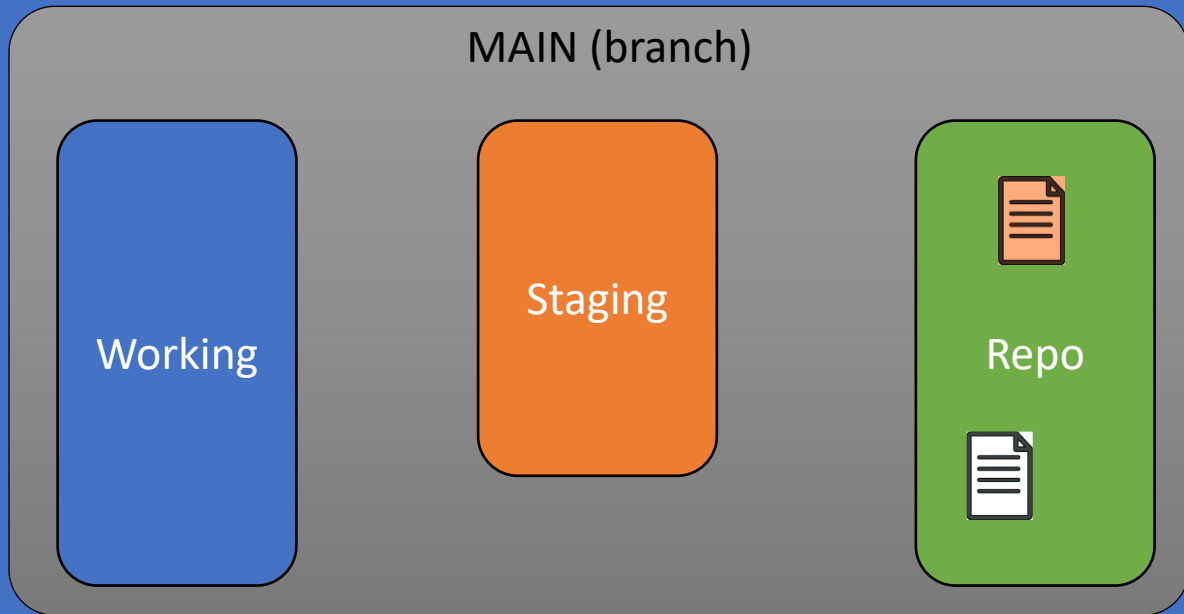
Added NewFile.txt



GitHub Repository

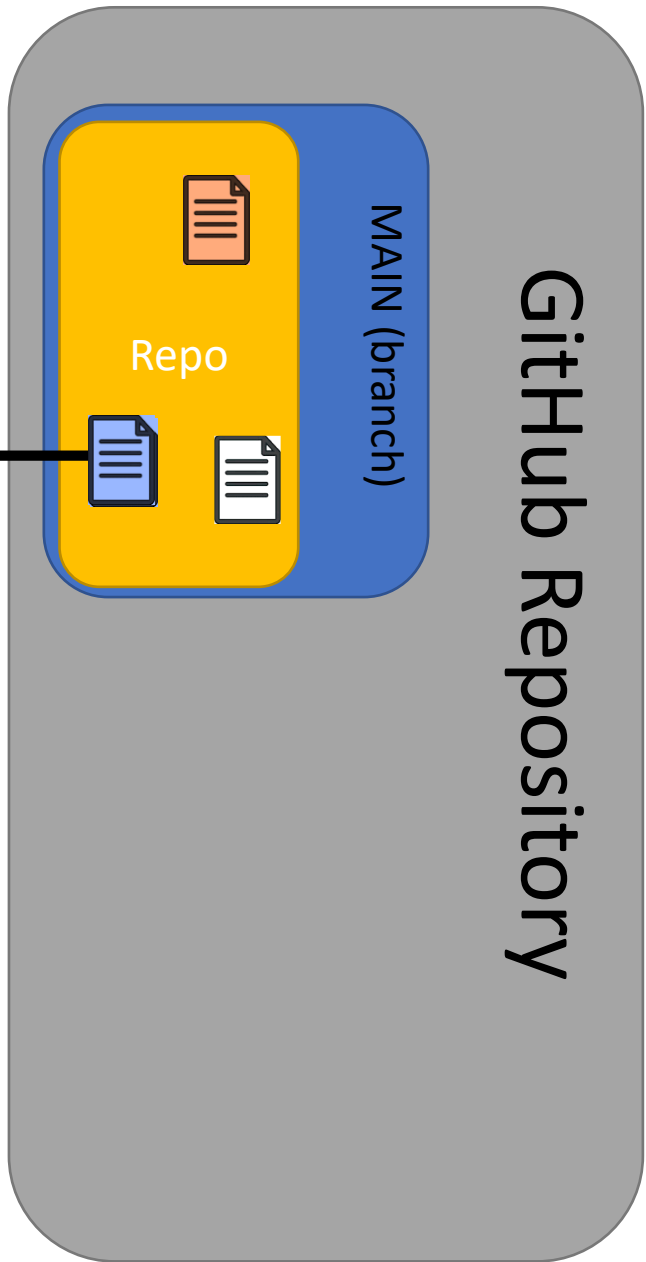
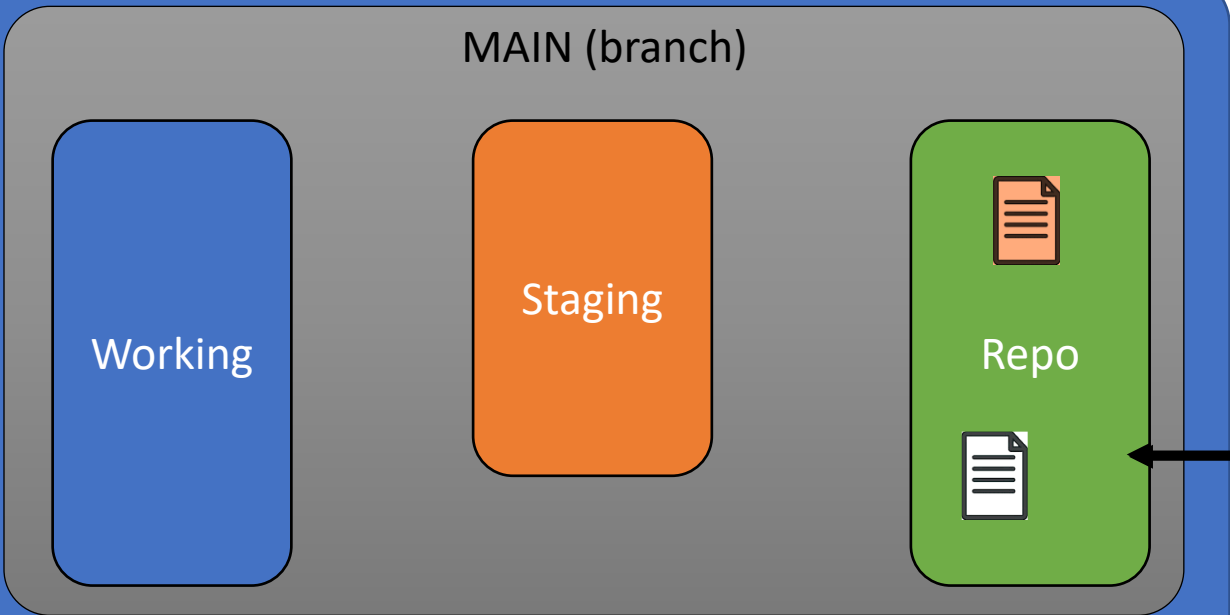


Local Repository



File added by someone else

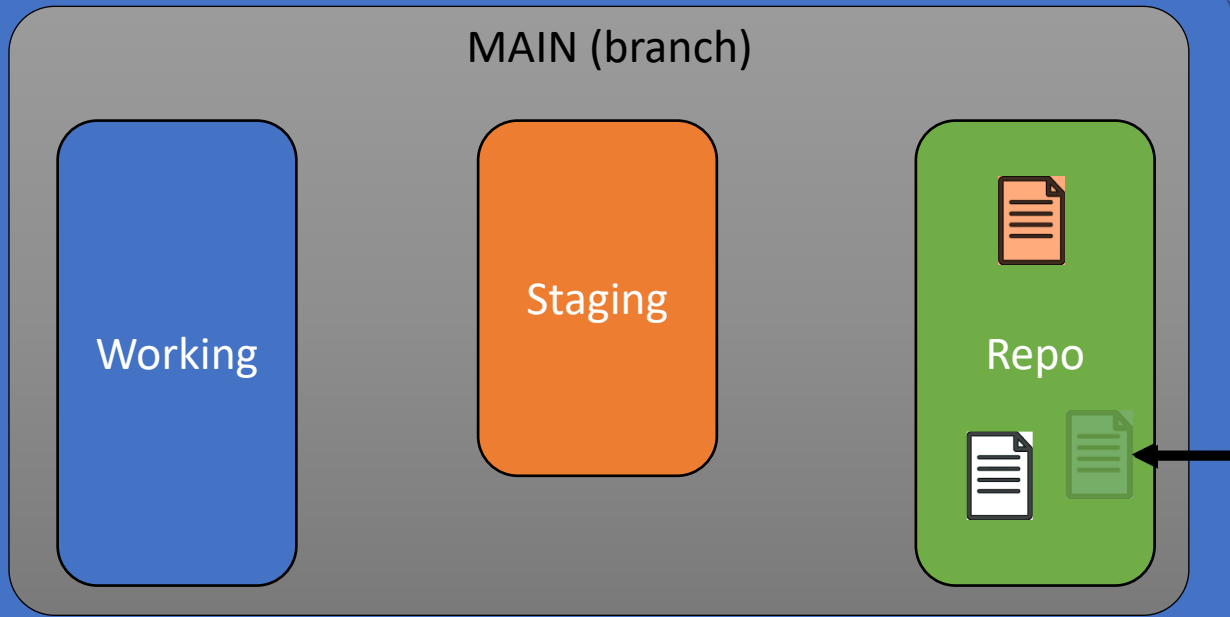
Local Repository



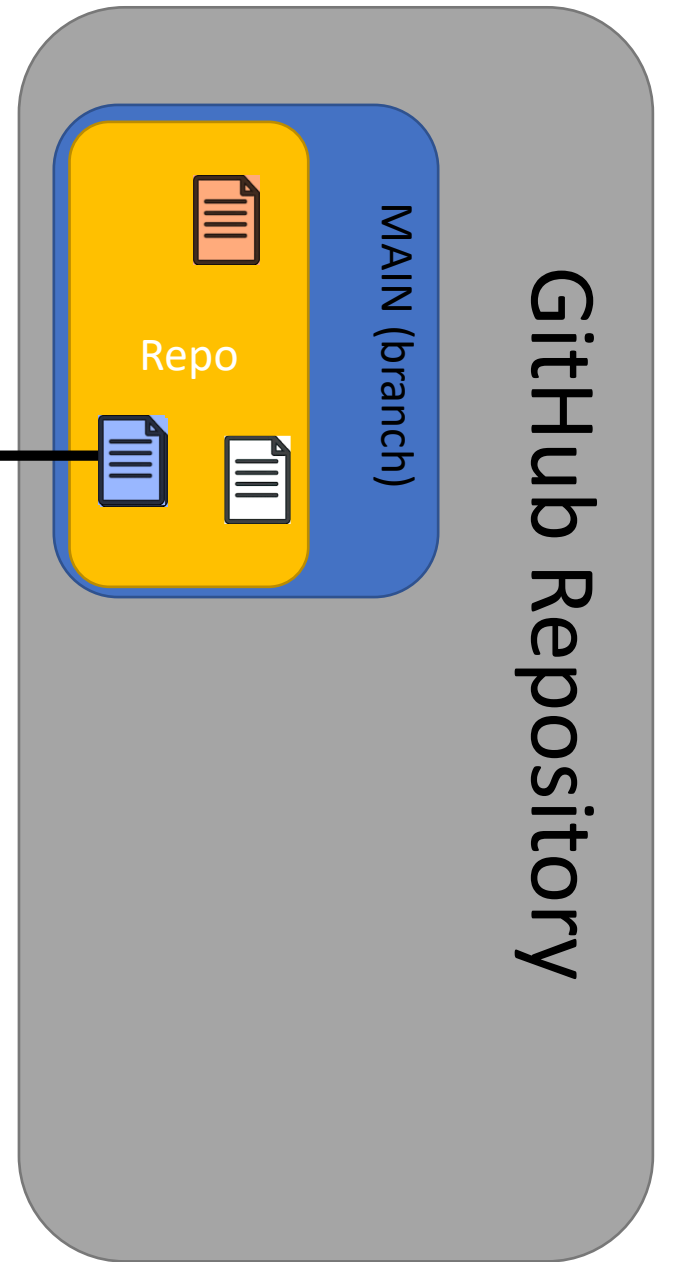
git fetch



Local Repository



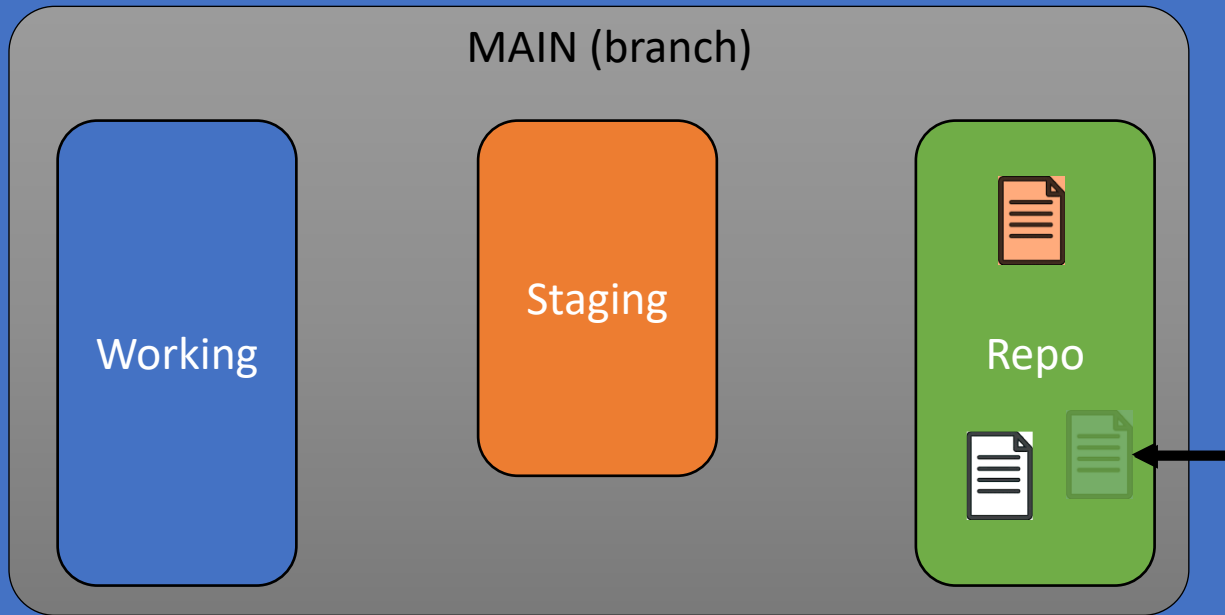
git fetch



GitHub Repository



Local Repository



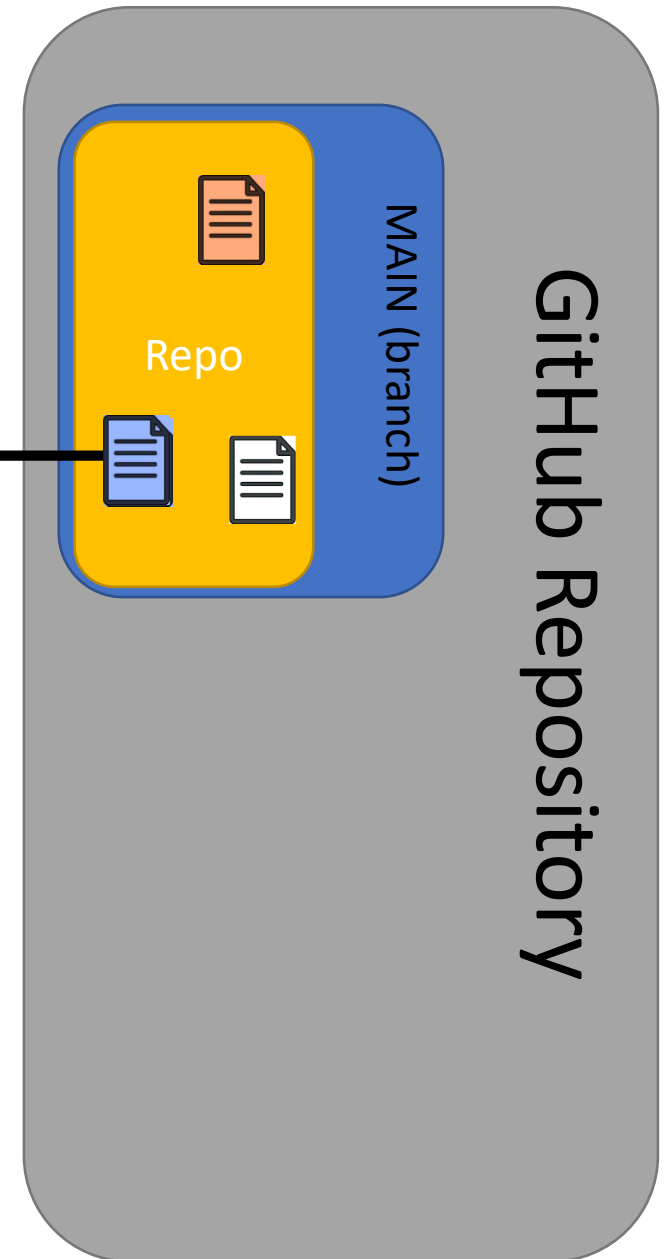
git status:

On branch main

Your branch is behind 'origin/main' by 1 commit, and
can be fast-forwarded.

(use "git pull" to update your local branch)

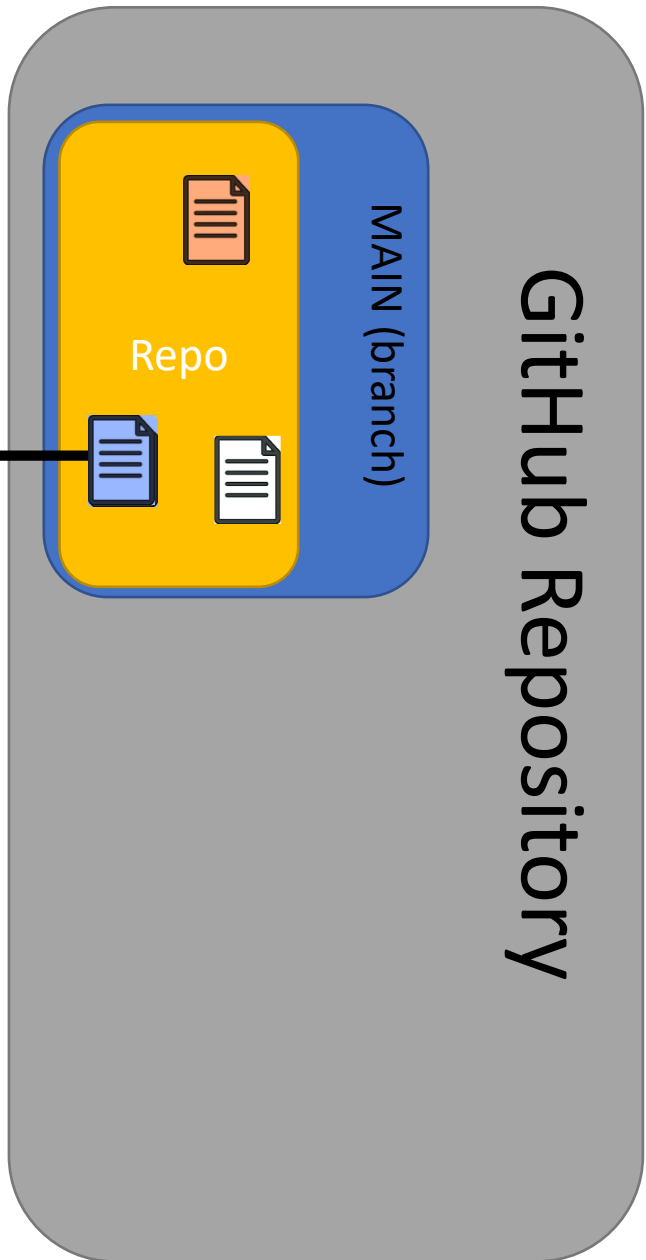
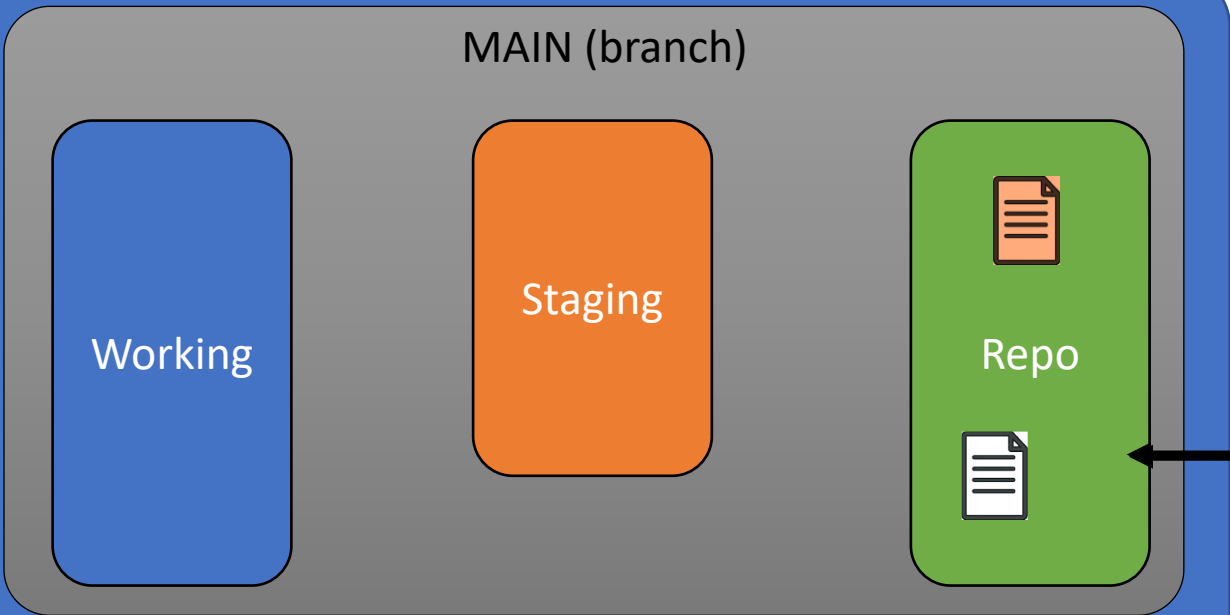
nothing to commit, working tree clean



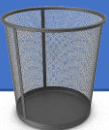
GitHub Repository



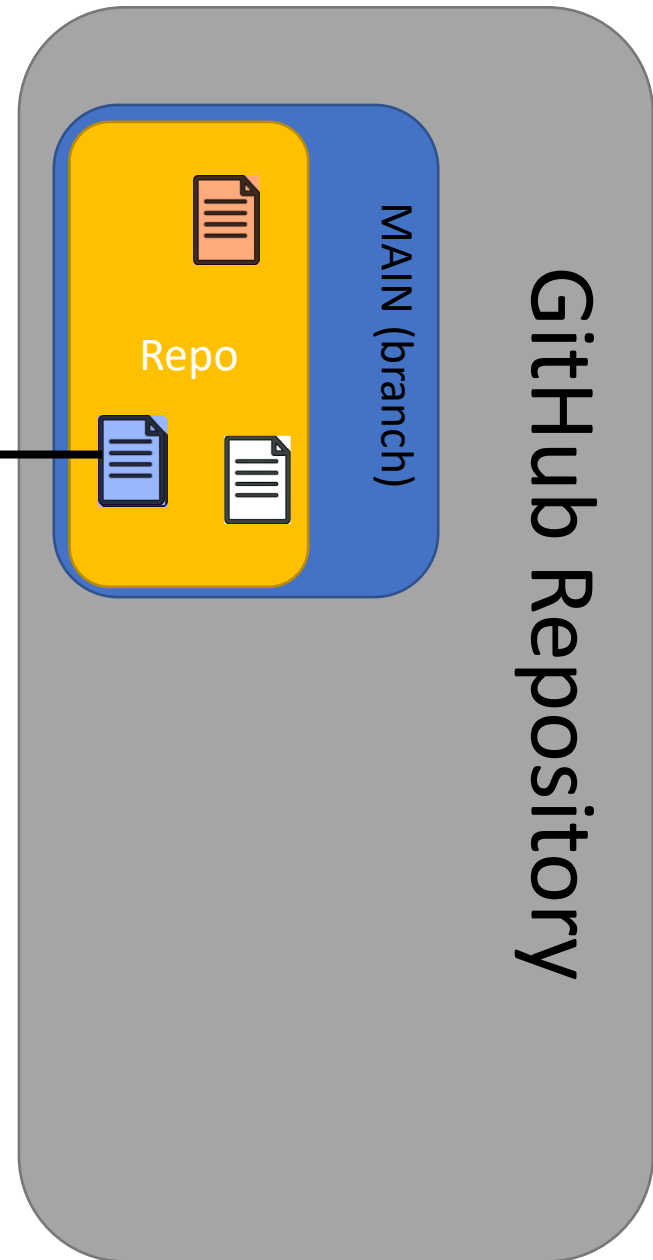
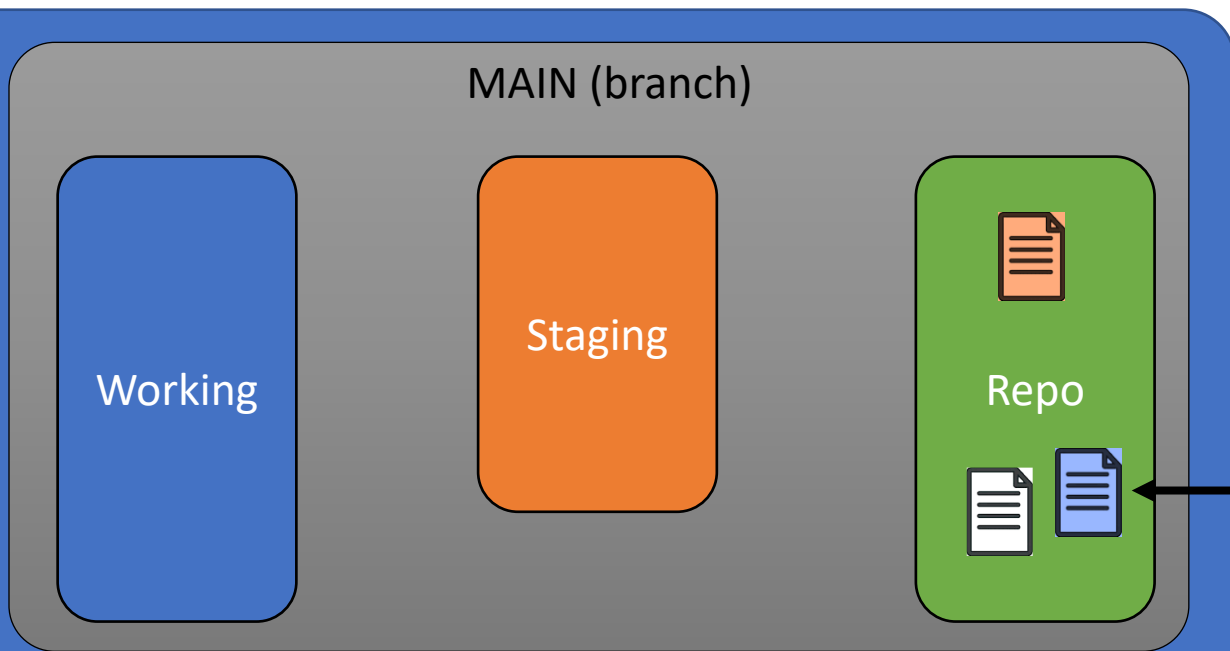
Local Repository



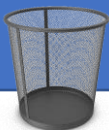
git pull



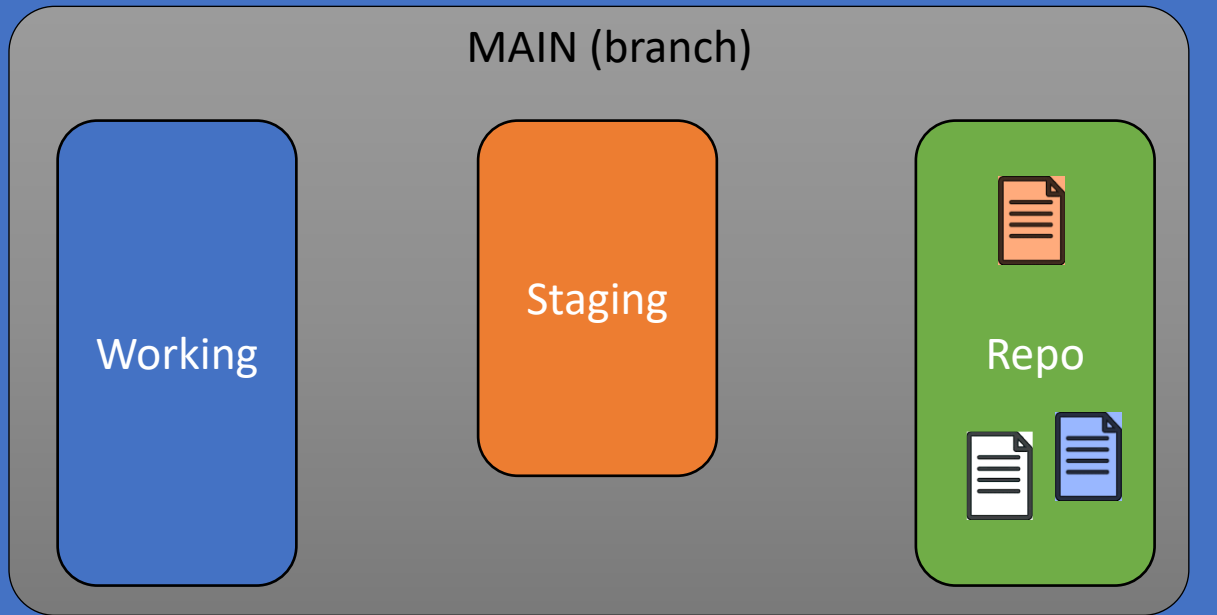
Local Repository



git pull

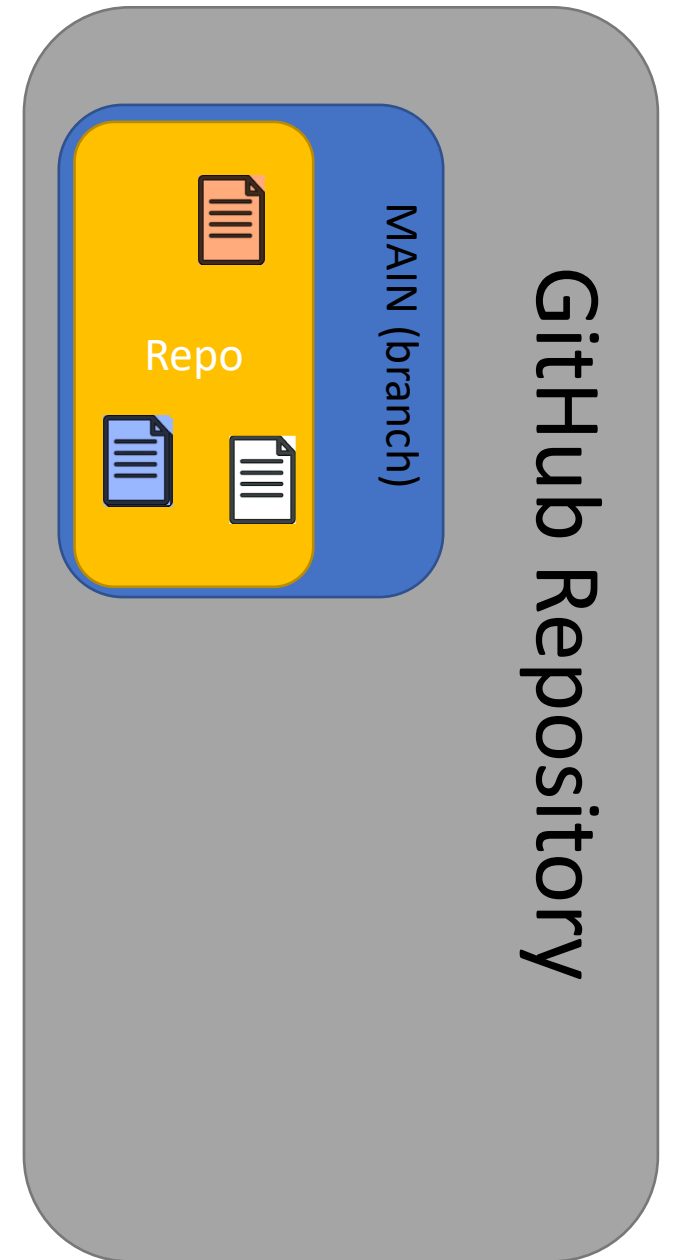


Local Repository



git status:
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

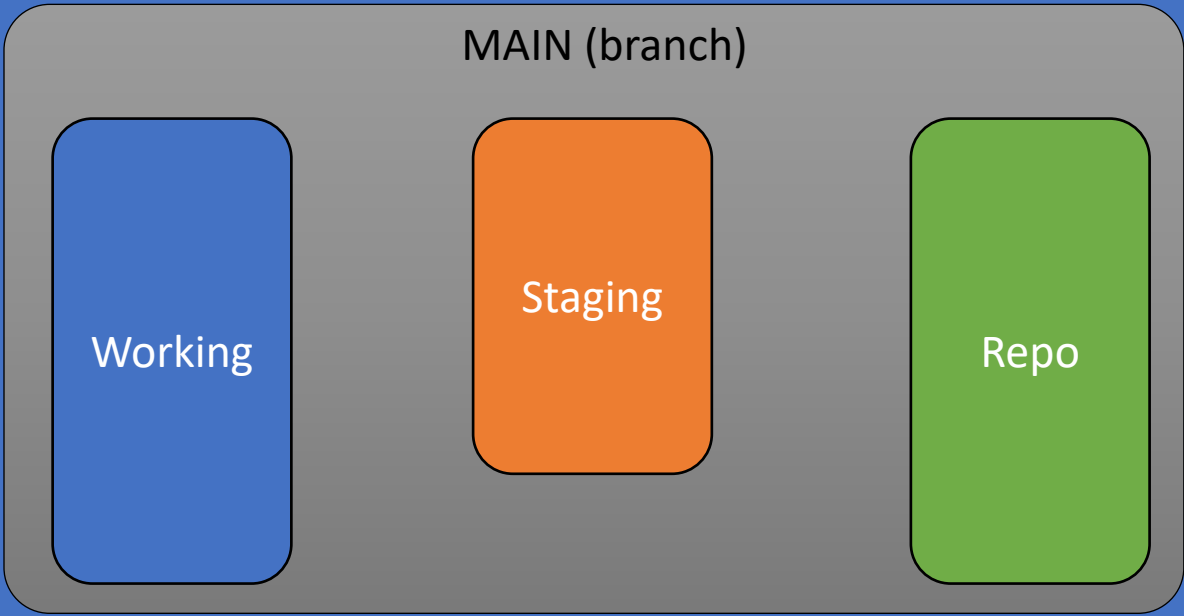


GitHub Repository

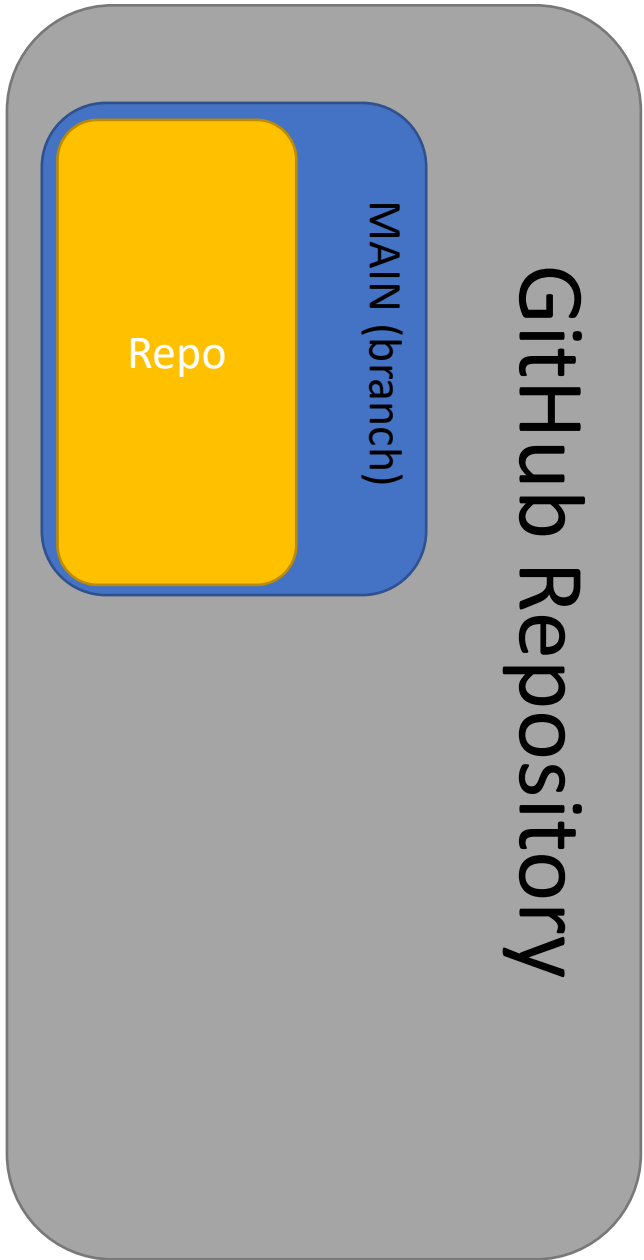
Using git with multiple branches

- git branch *newbranchname*
- git checkout *newbranchname*
- git add . (or git add filename)
- git commit -m "One line description of what was changed"
- git push
- Pull request (*from the GitHub side*)
- Merge (*from the GitHub side*)
- git fetch
- git pull

Local Repository



GitHub Repository



Local Repository

git branch "changes"

Working

Staging

Repo

MAIN (branch)



GitHub Repository

Repo

MAIN (branch)

Local Repository

git branch "changes"

Working

MAIN (branch)

Staging

Repo

Working

Staging

Repo

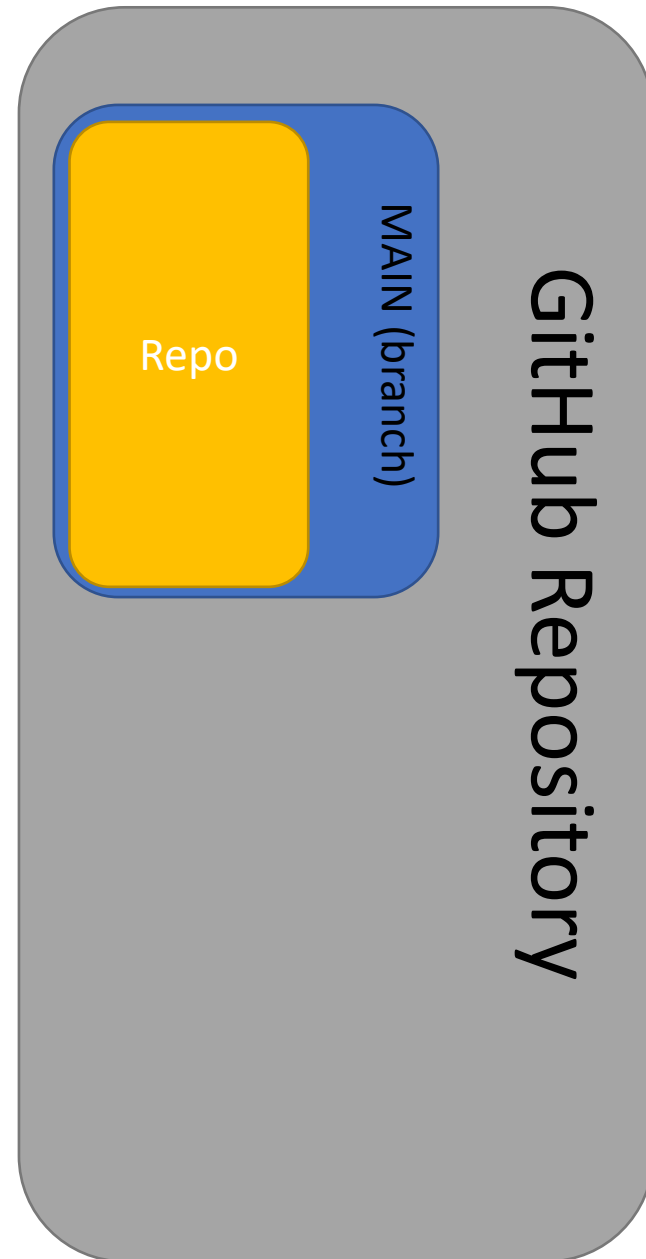
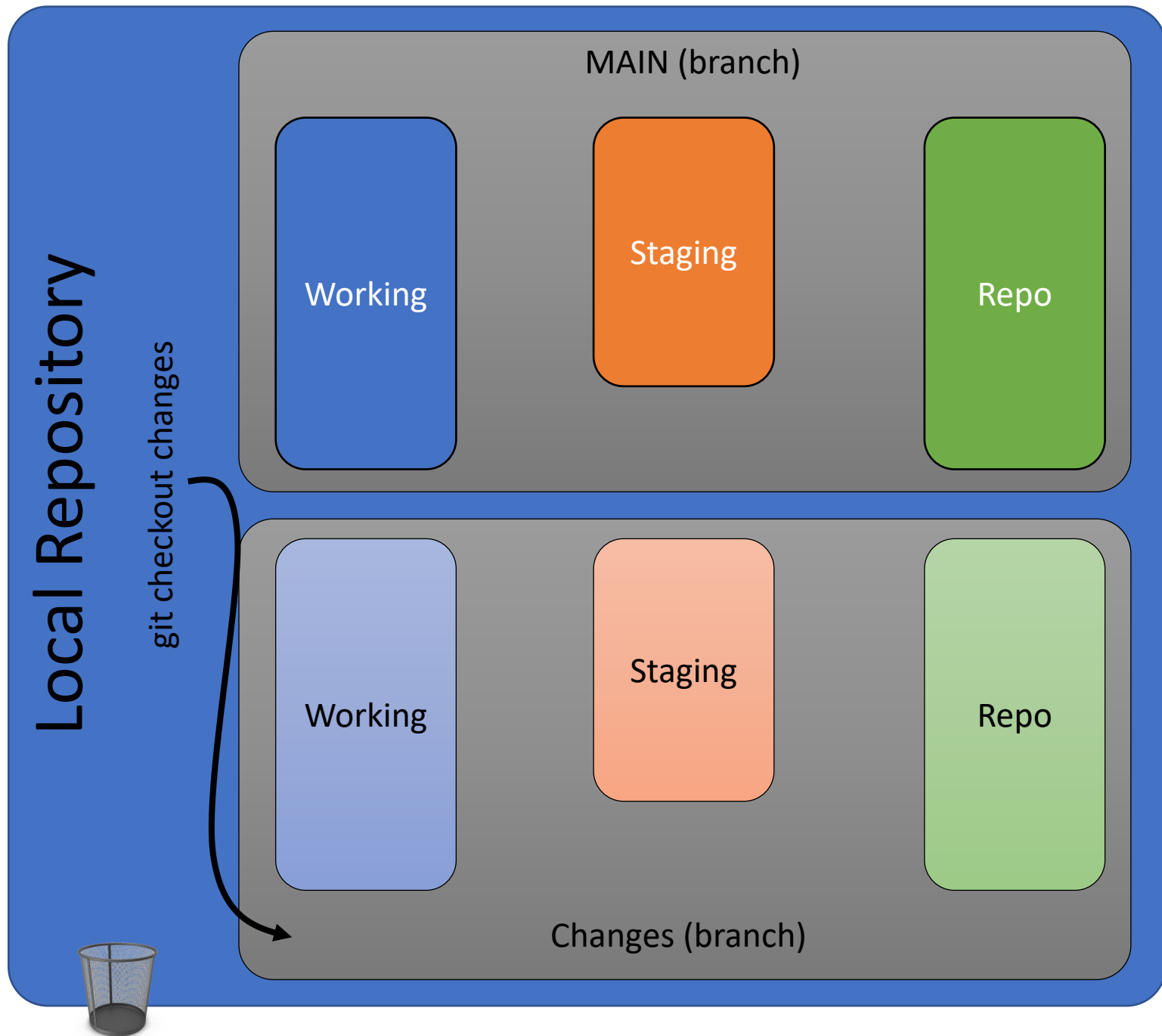
Changes (branch)

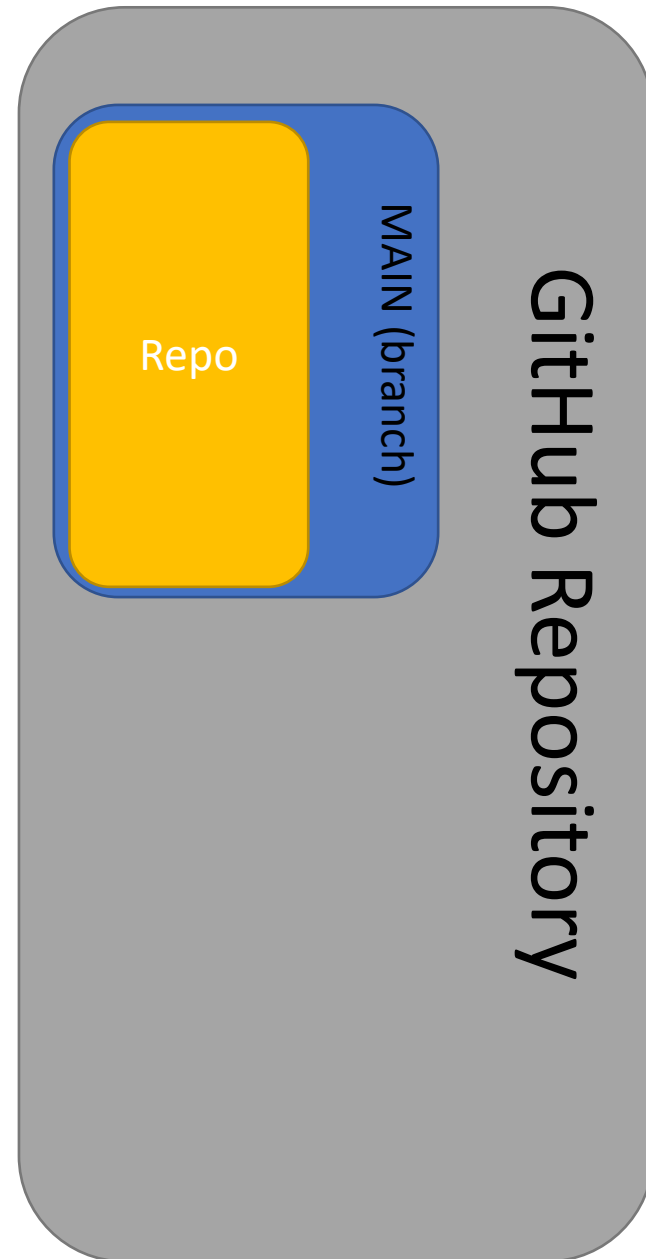
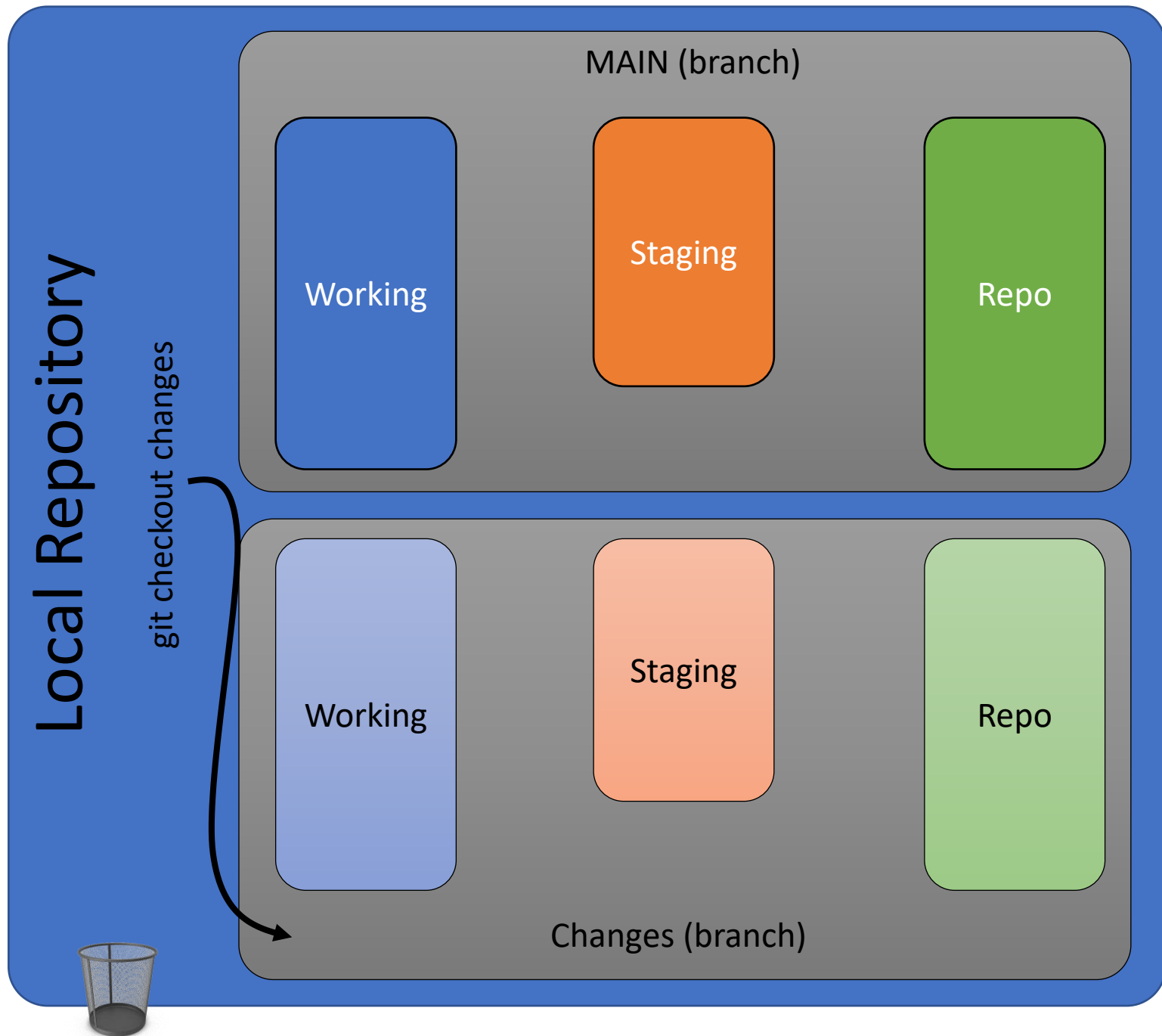


Repo

MAIN (branch)

GitHub Repository





Local Repository

MAIN (branch)

Working

Staging

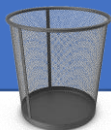
Repo

Working

Staging

Repo

Changes (branch)

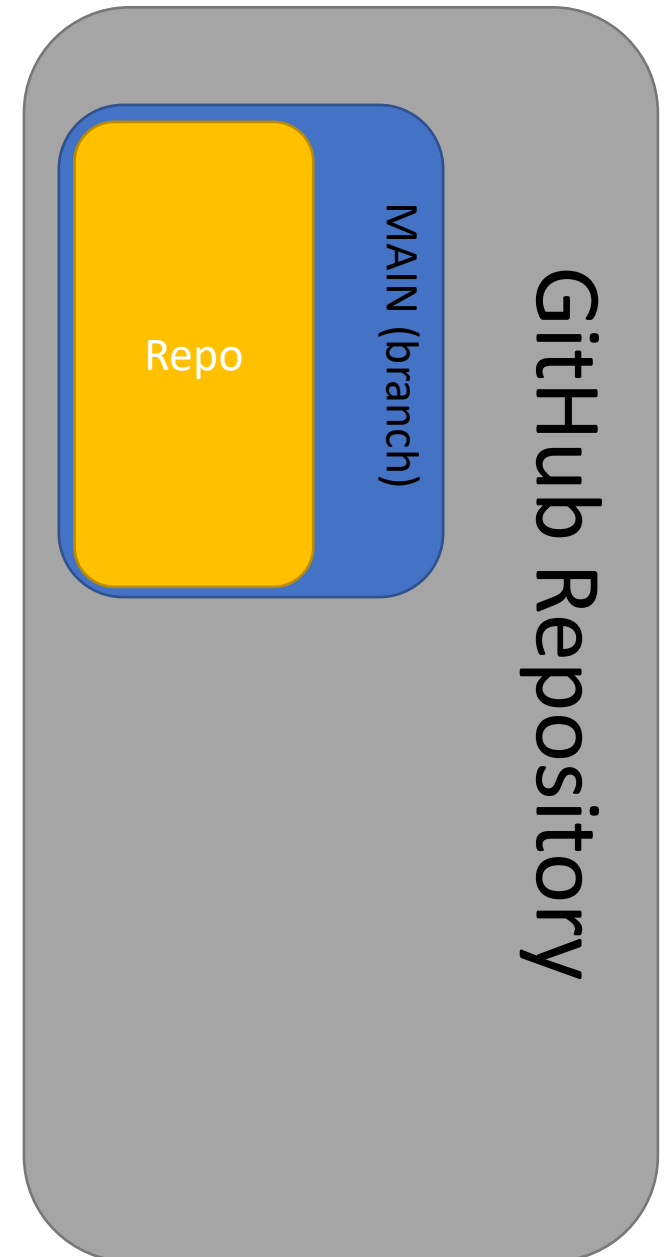
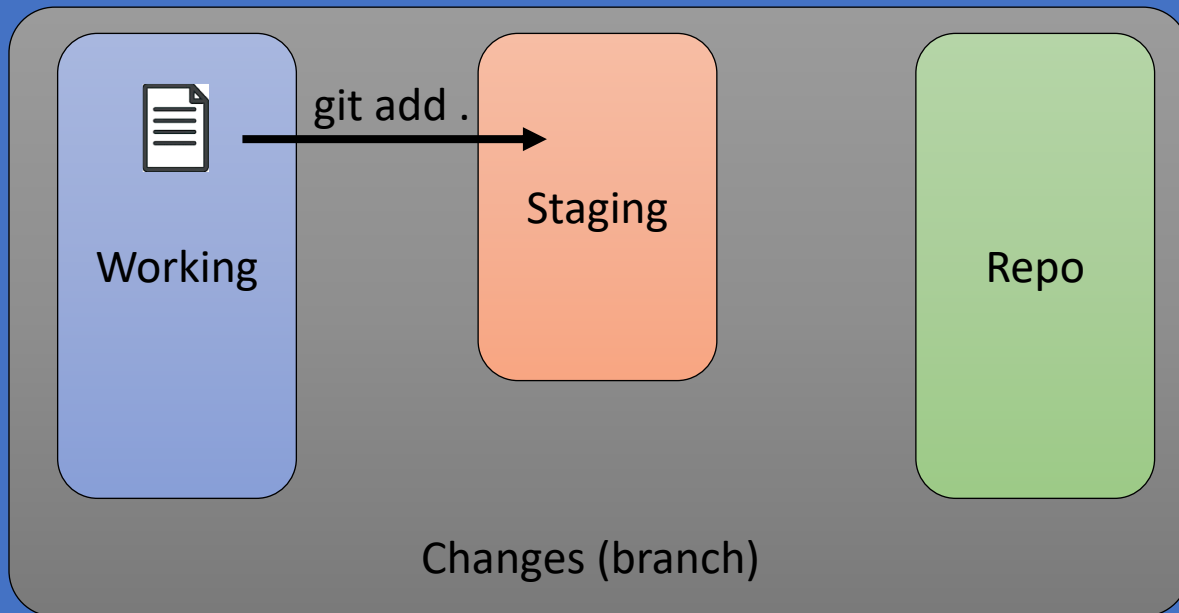
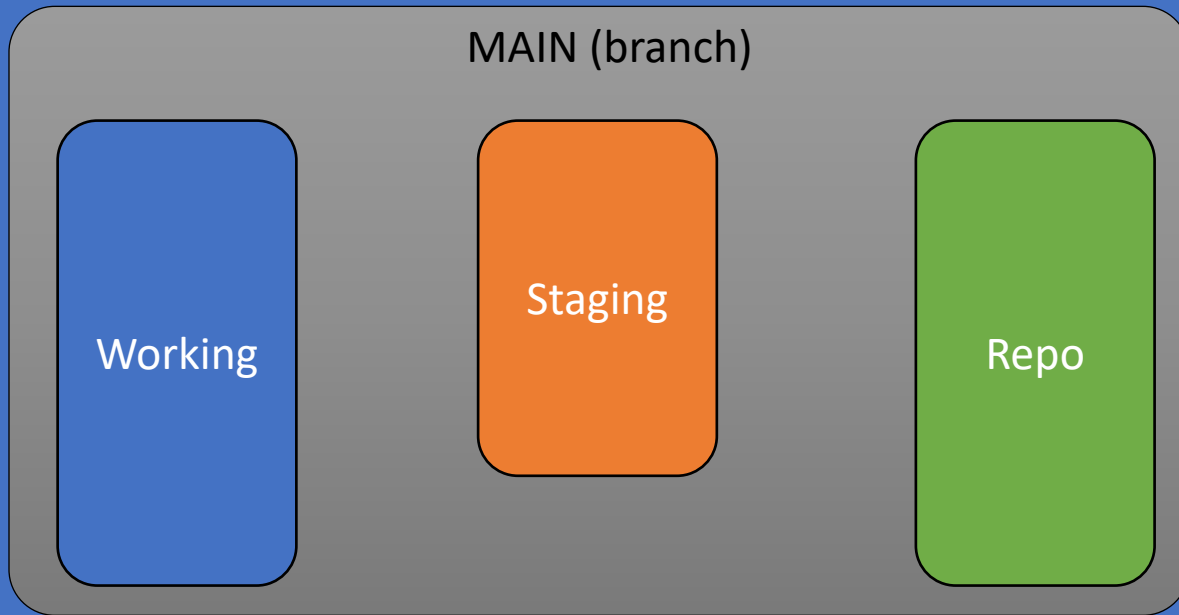


GitHub Repository

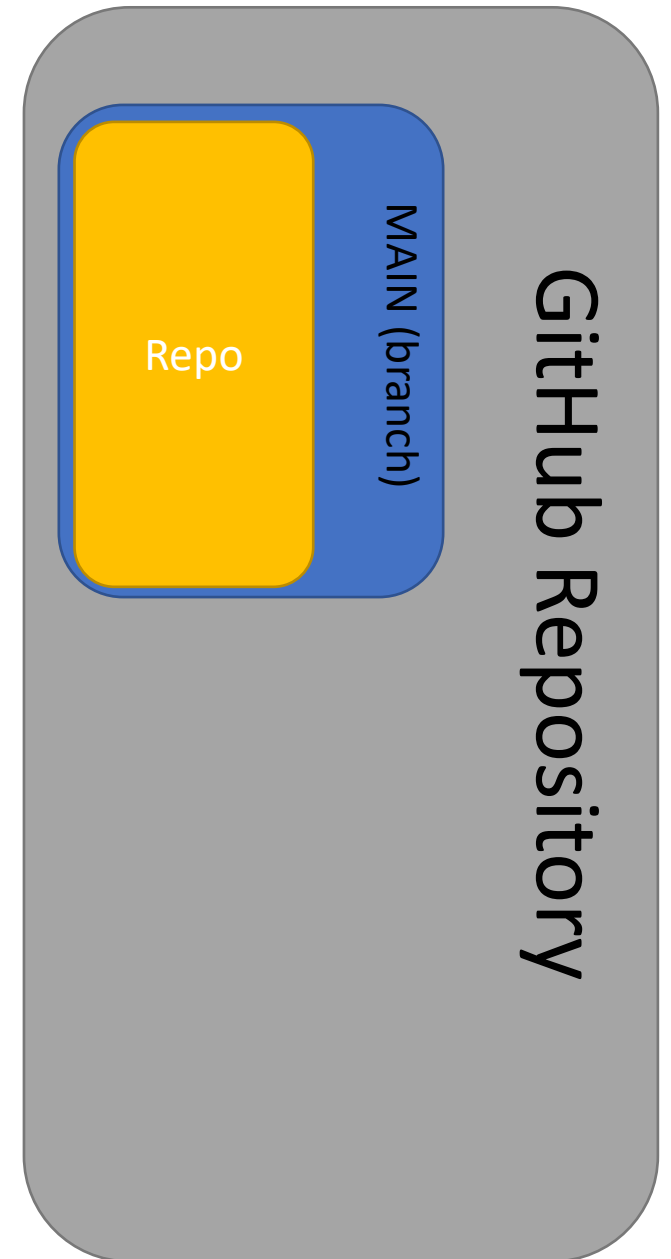
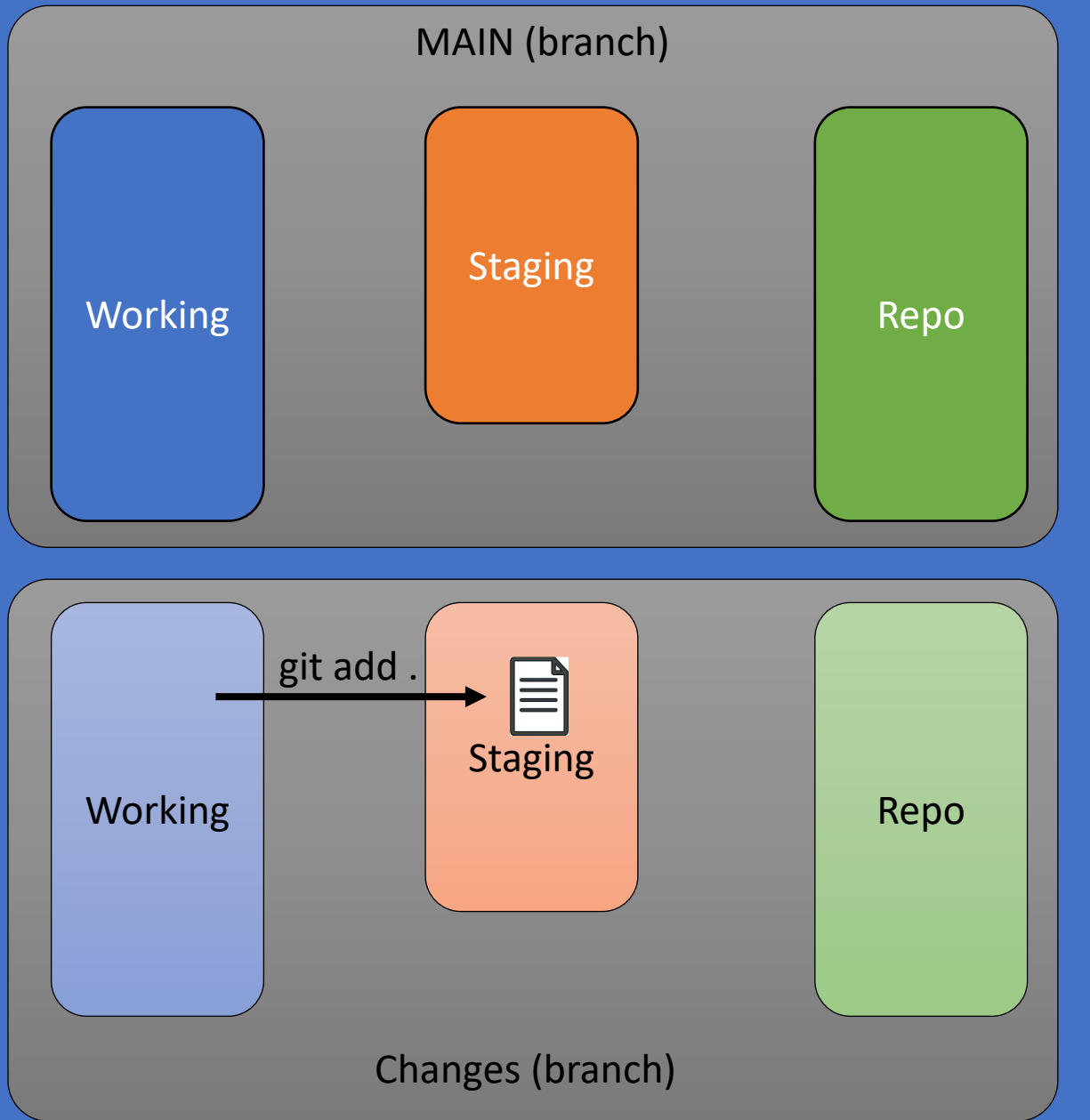
MAIN (branch)

Repo

Local Repository



Local Repository



Local Repository

MAIN (branch)

Working

Staging

Repo

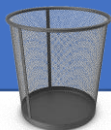
Working

Staging

Repo

git commit -m "message"

Changes (branch)

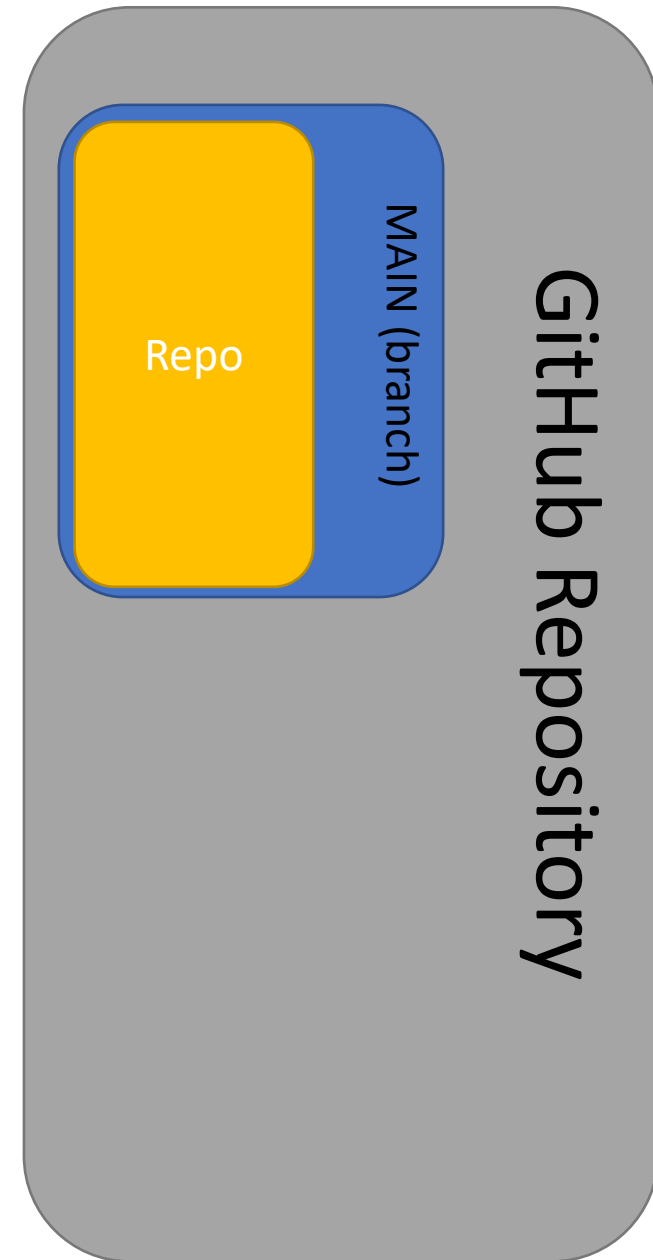
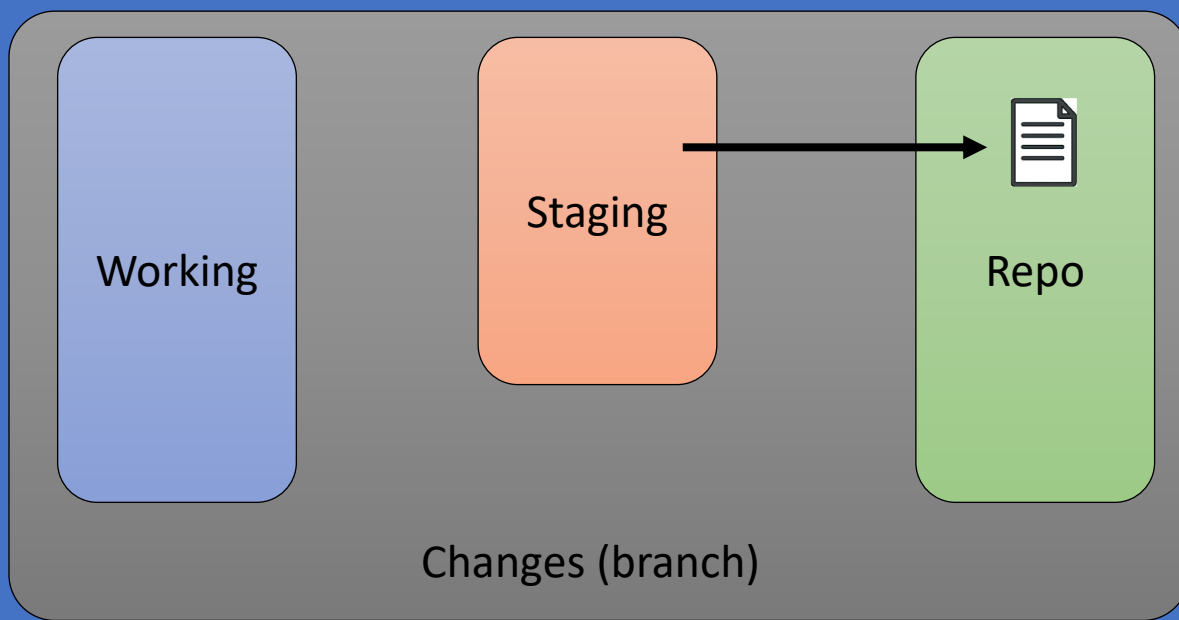
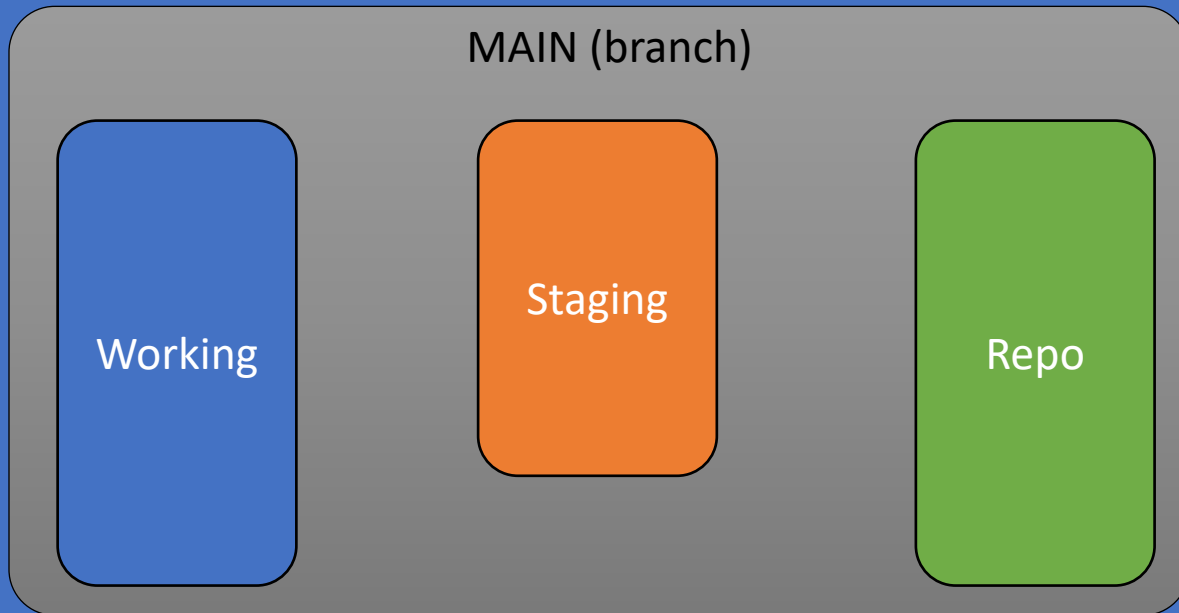


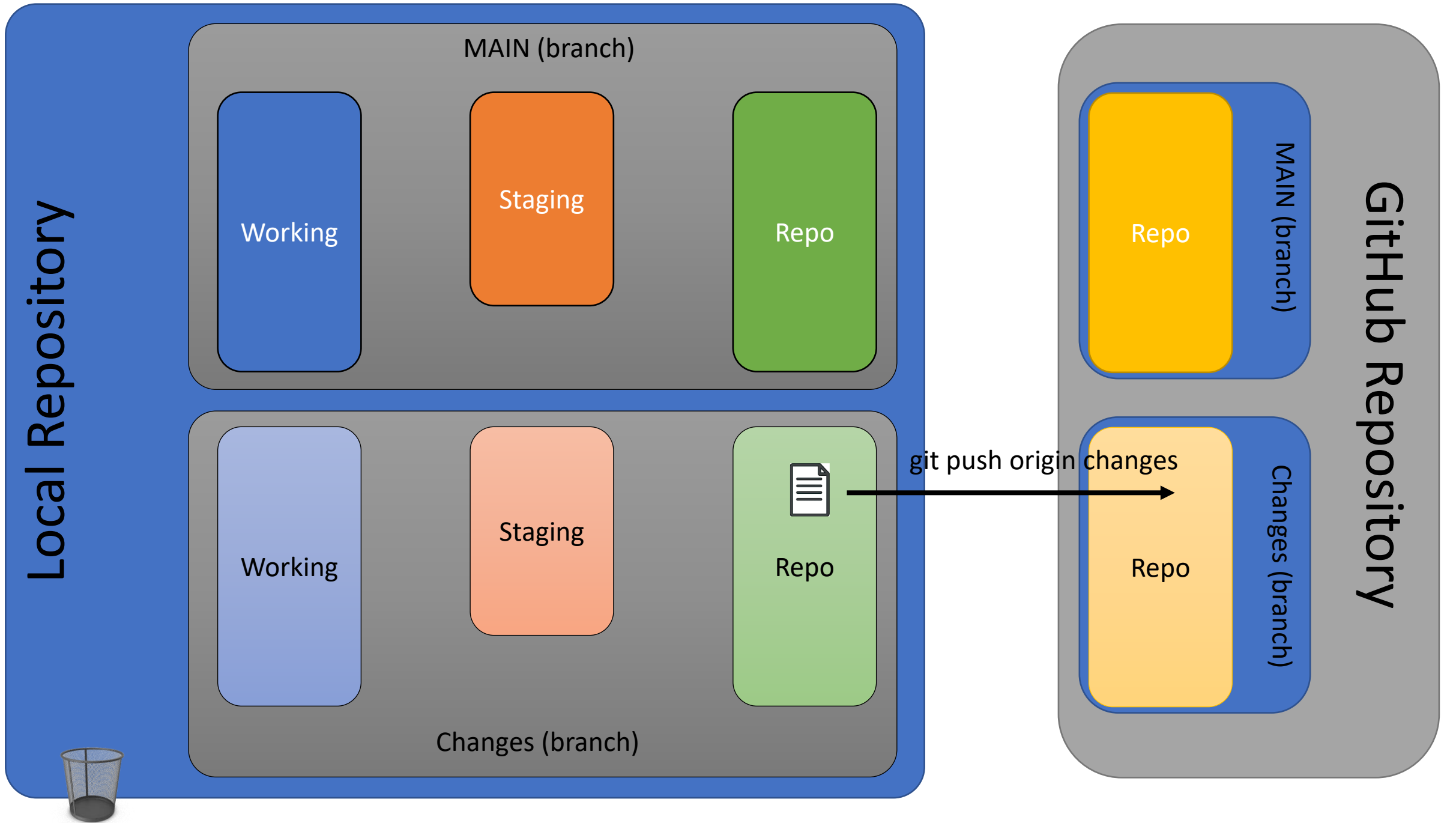
GitHub Repository

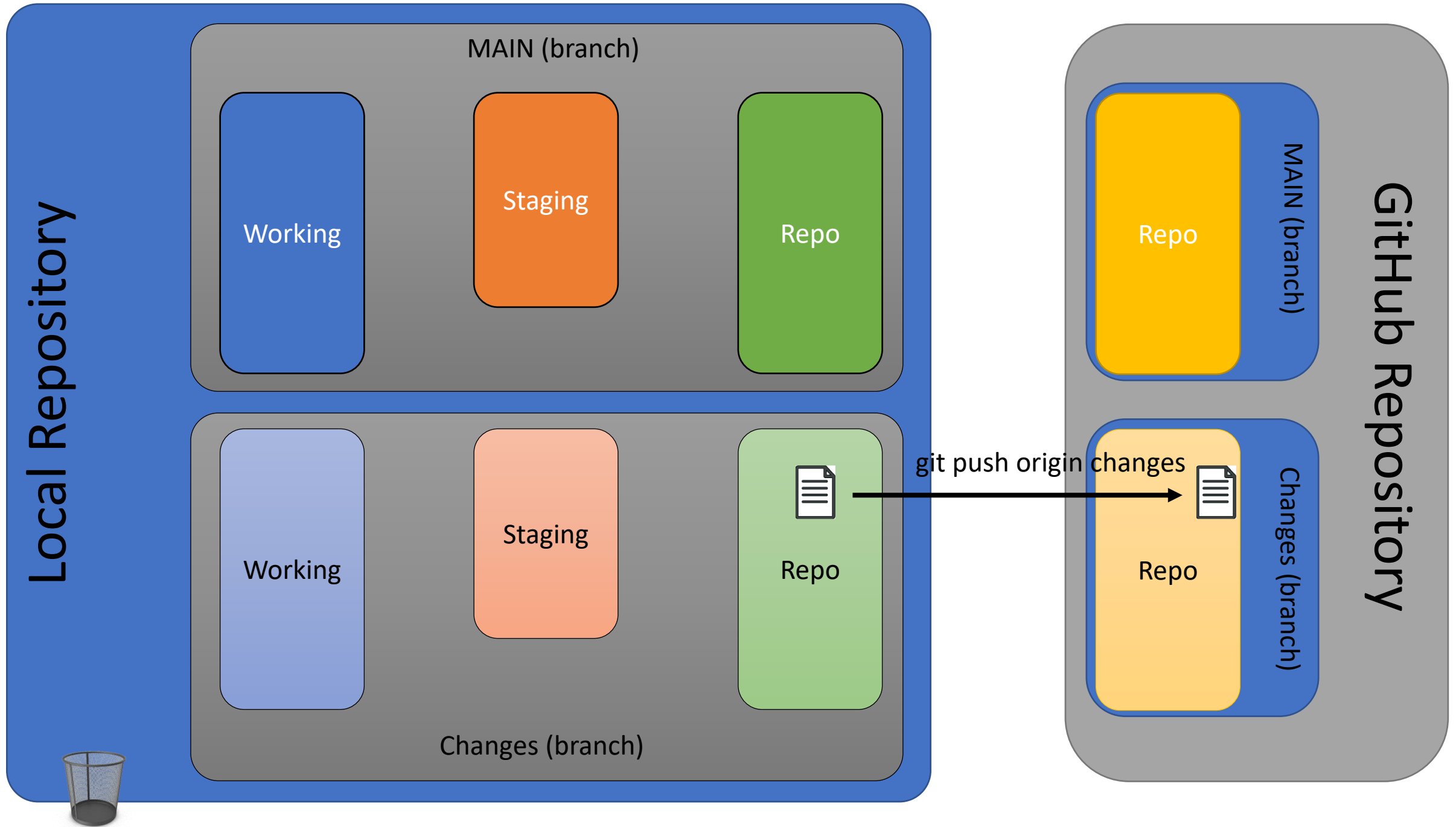
MAIN (branch)

Repo

Local Repository







Local Repository

MAIN (branch)

Working

Staging

Repo

Working

Staging

Repo

Changes (branch)



GitHub Repository

MAIN (branch)

Repo

Changes (branch)

Make
Pull
Request
Repo

Local Repository

MAIN (branch)

Working

Staging

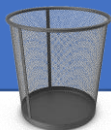
Repo

Working

Staging

Repo

Changes (branch)



GitHub Repository

MAIN (branch)

Repo

Changes (branch)

Repo

merge



Local Repository

MAIN (branch)

Working

Staging

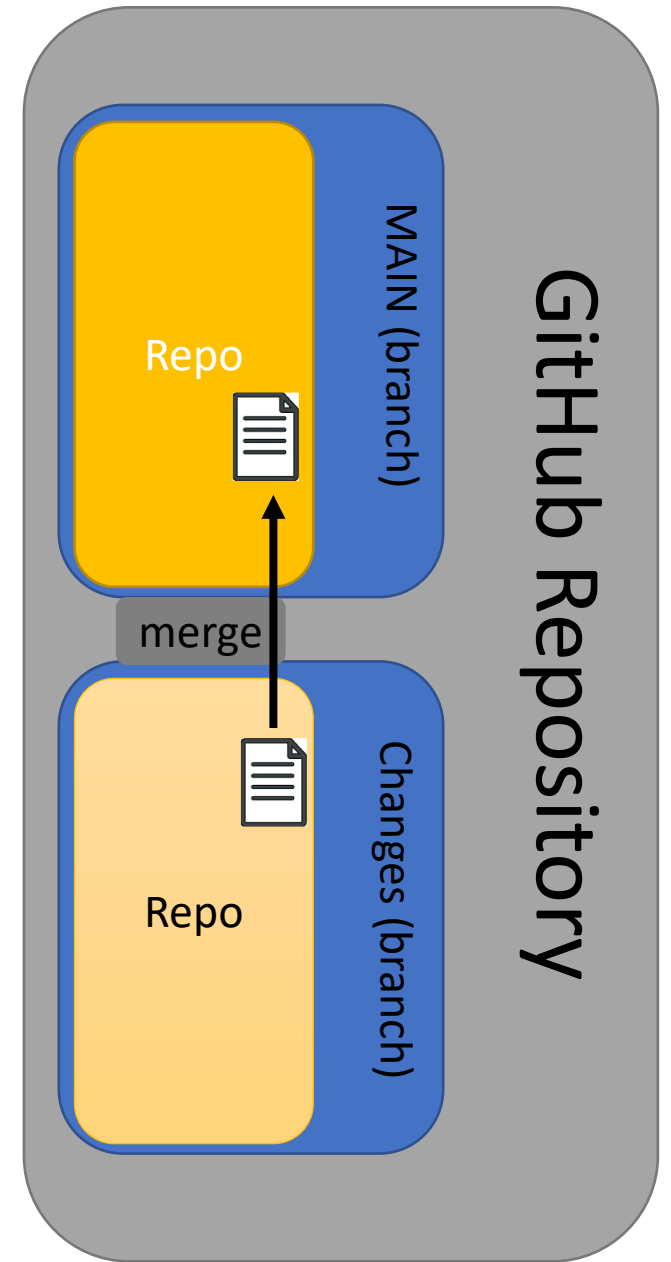
Repo

Working

Staging

Repo

Changes (branch)



GitHub Repository

Local Repository

MAIN (branch)

Working

Staging

Repo

Working

Staging

Repo

Changes (branch)



GitHub Repository

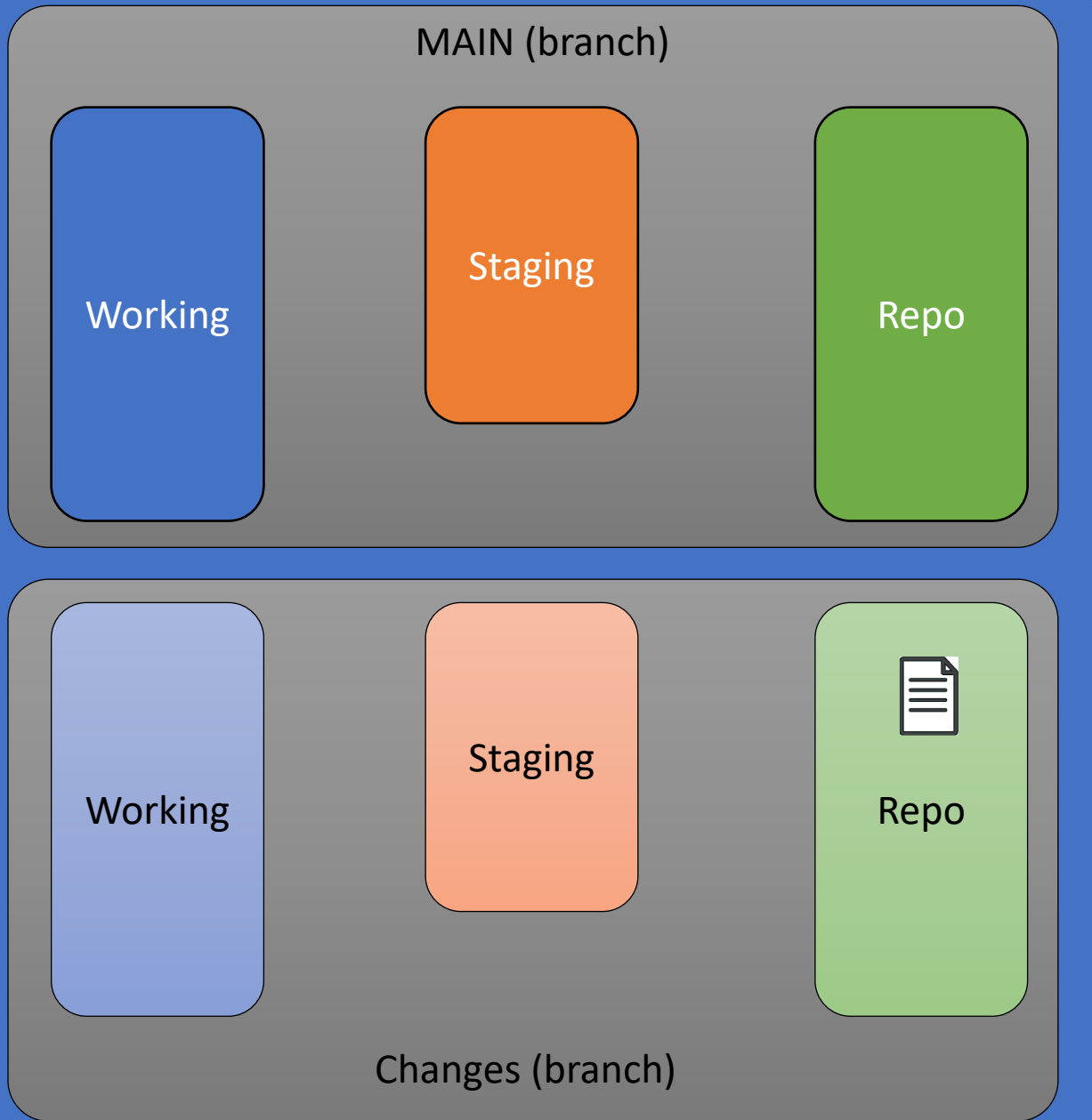
MAIN (branch)

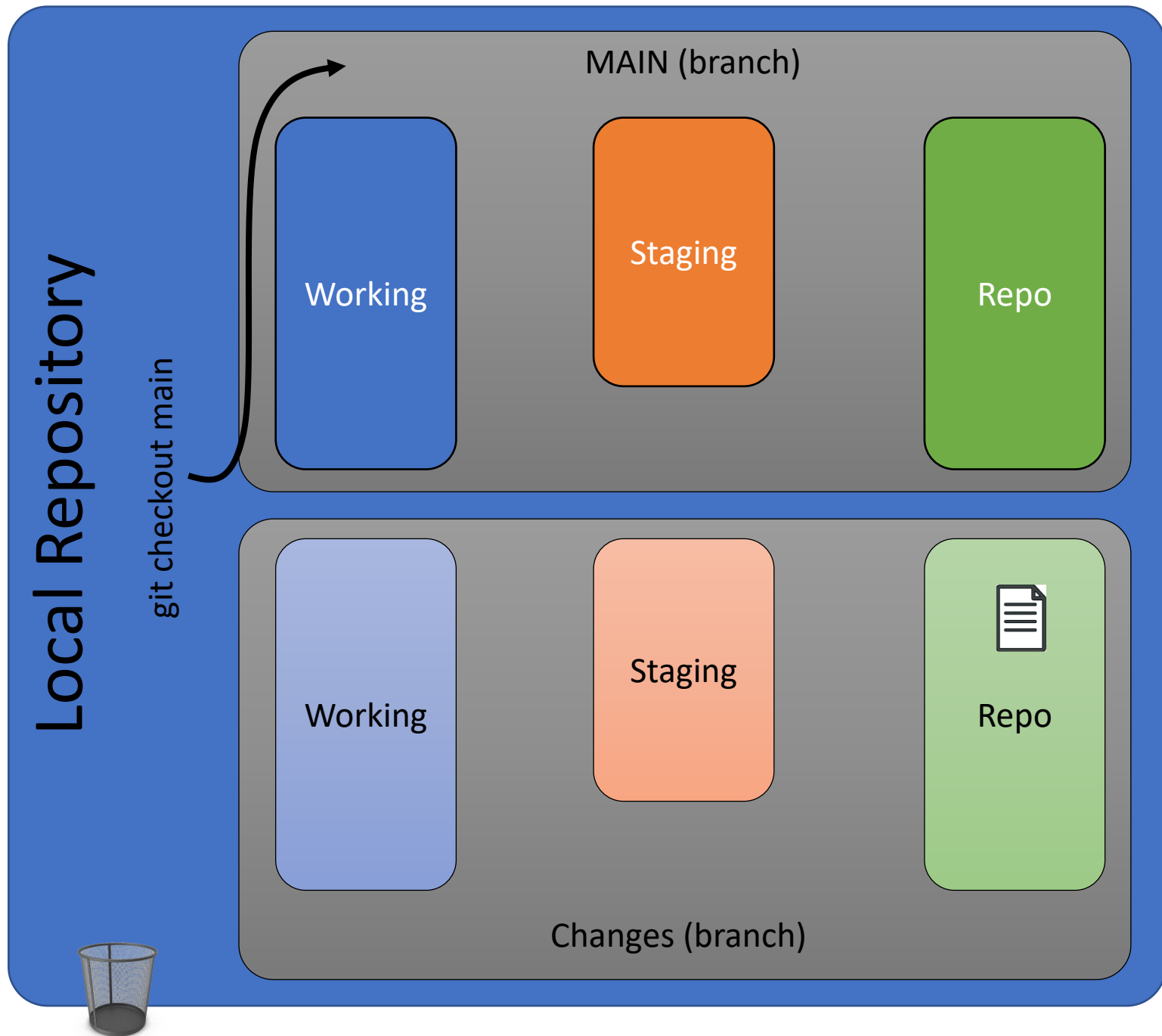
Repo

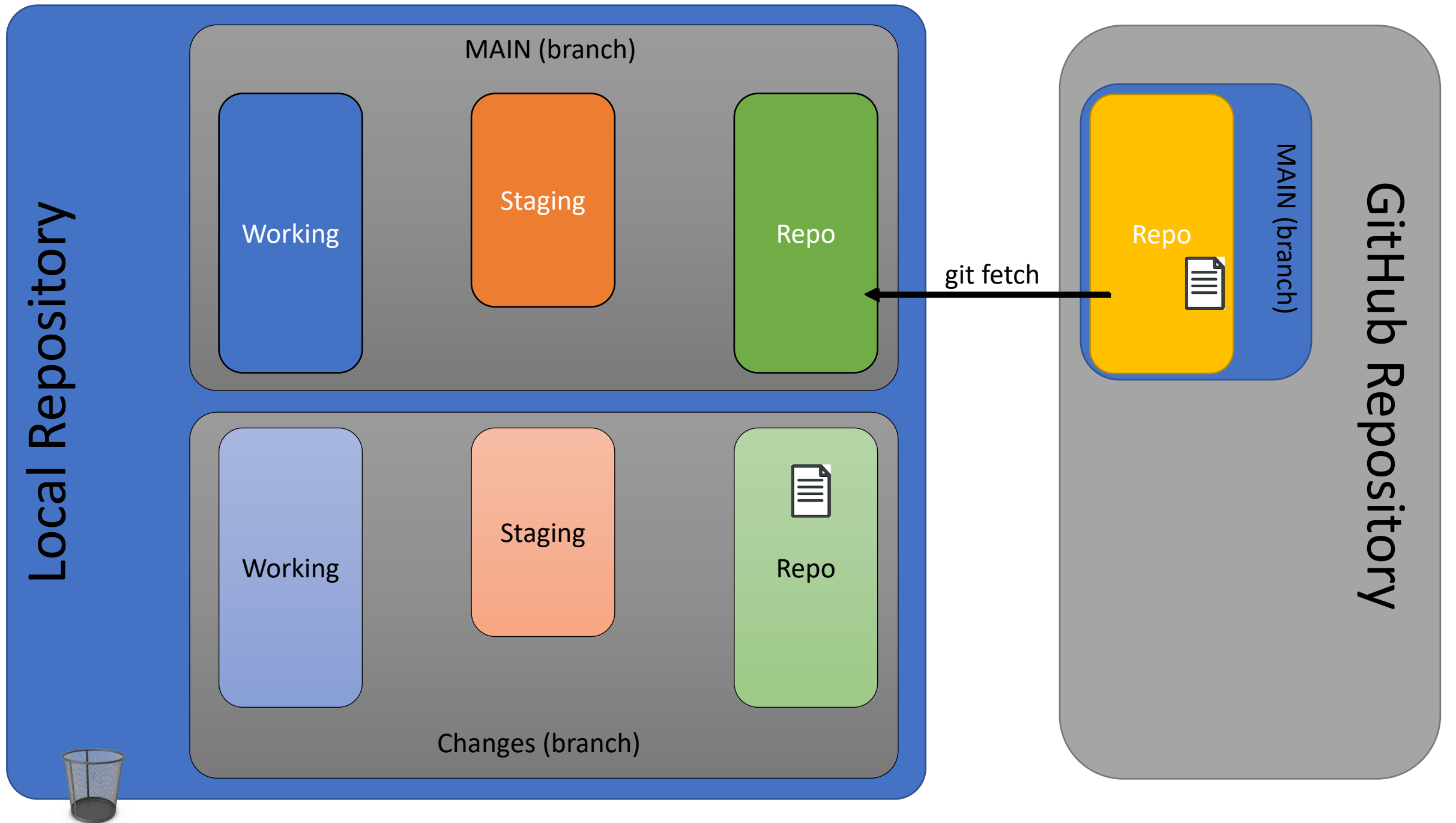
Changes (branch)

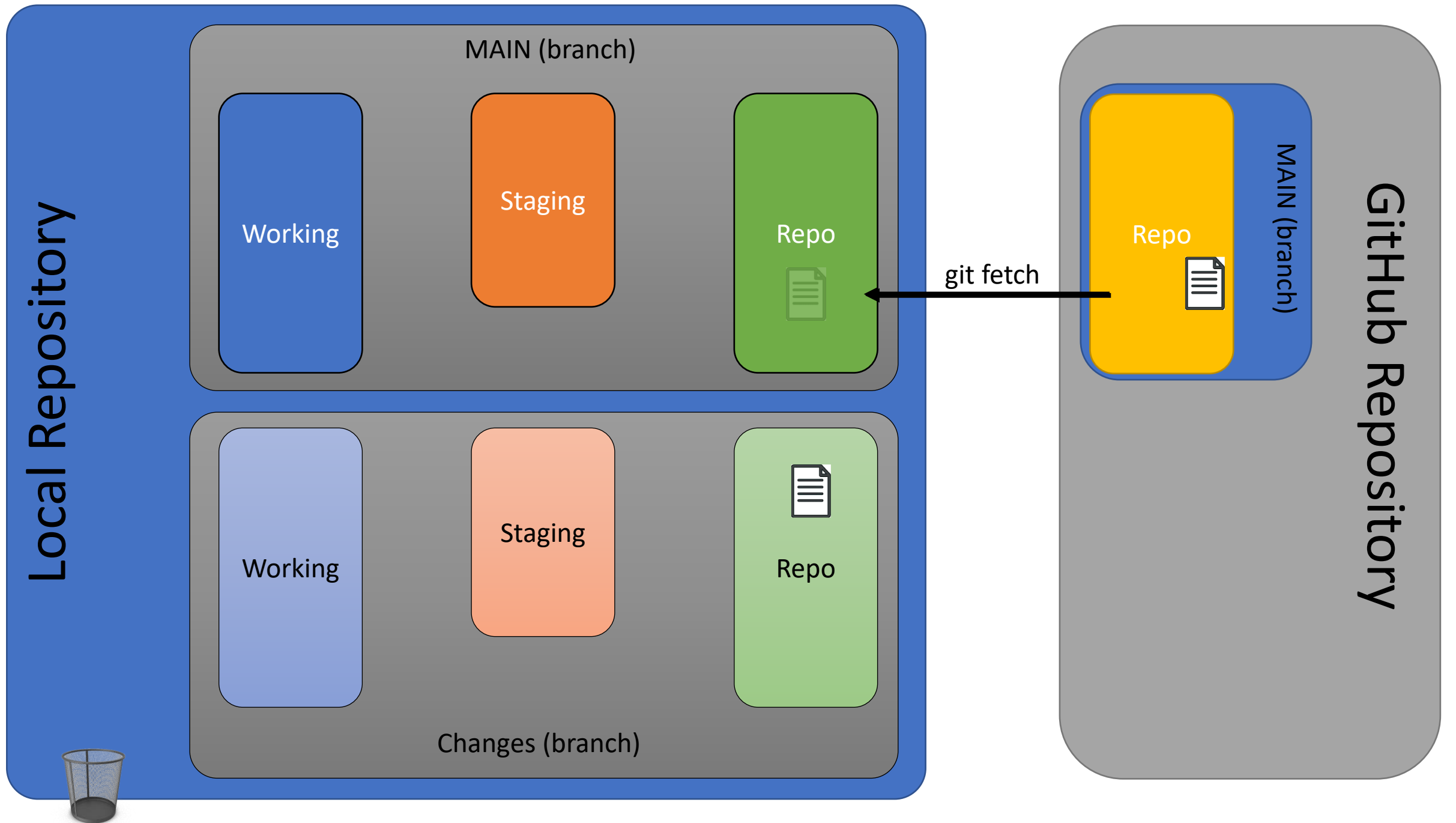
Repo

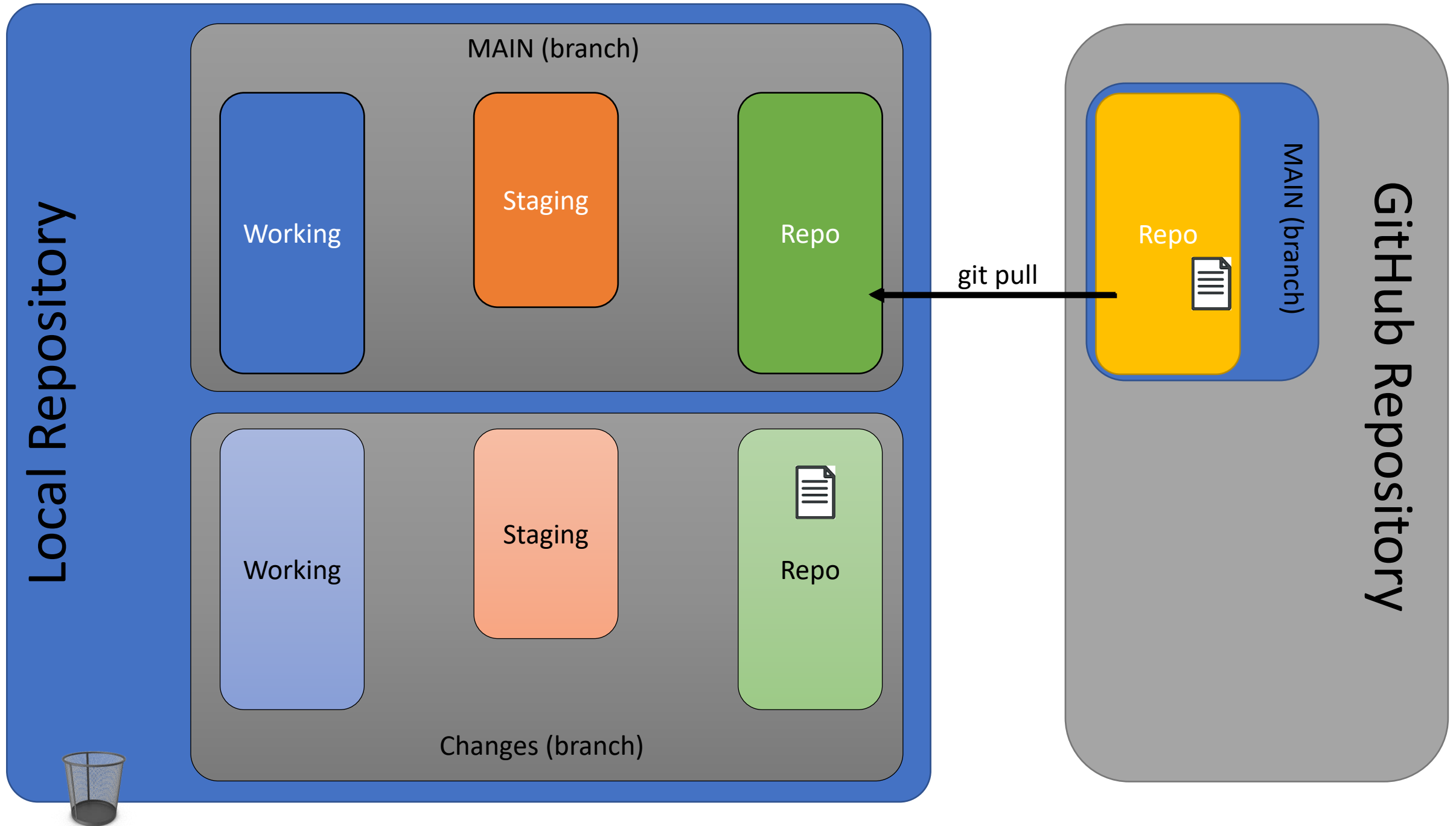
Local Repository

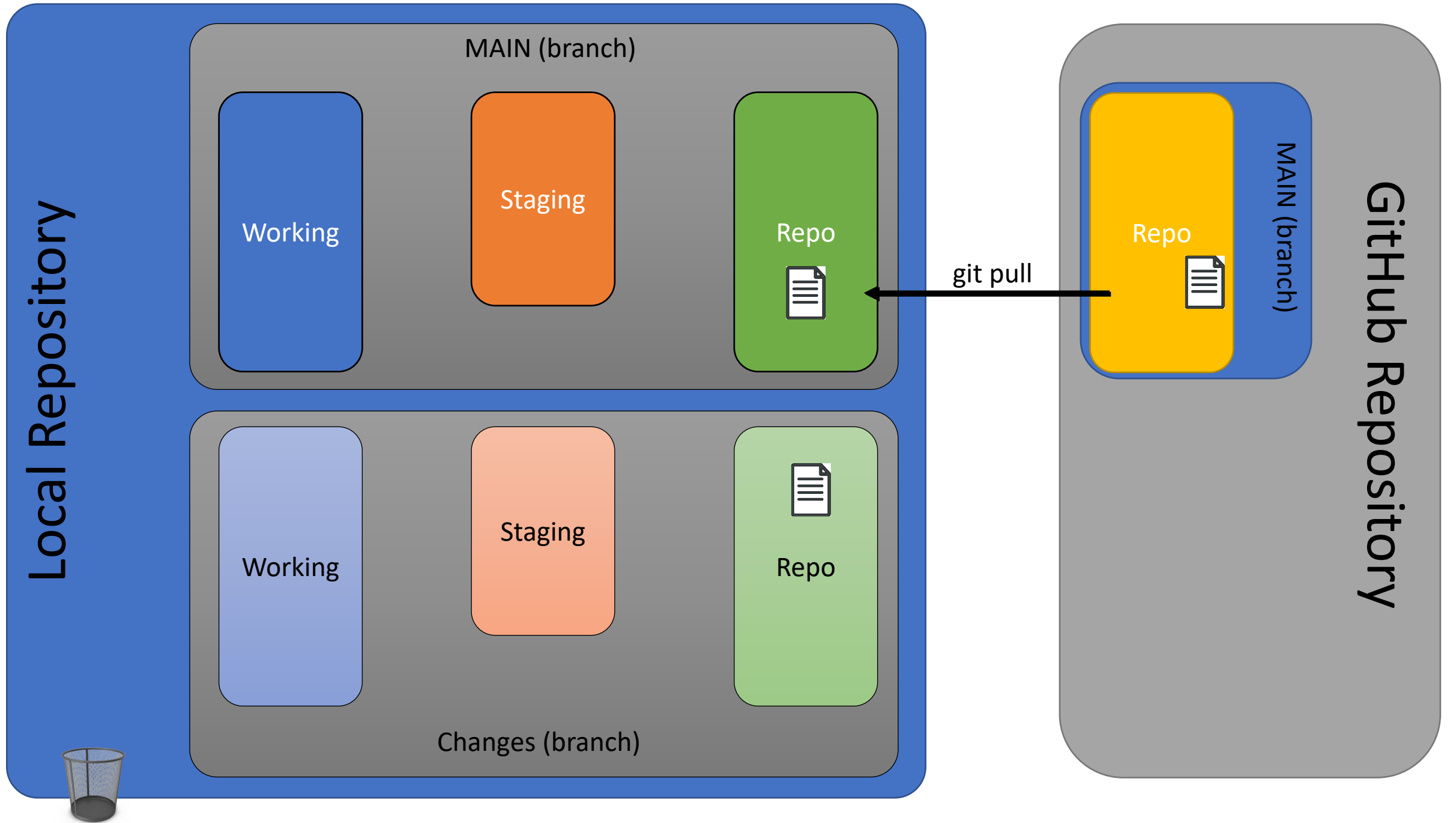




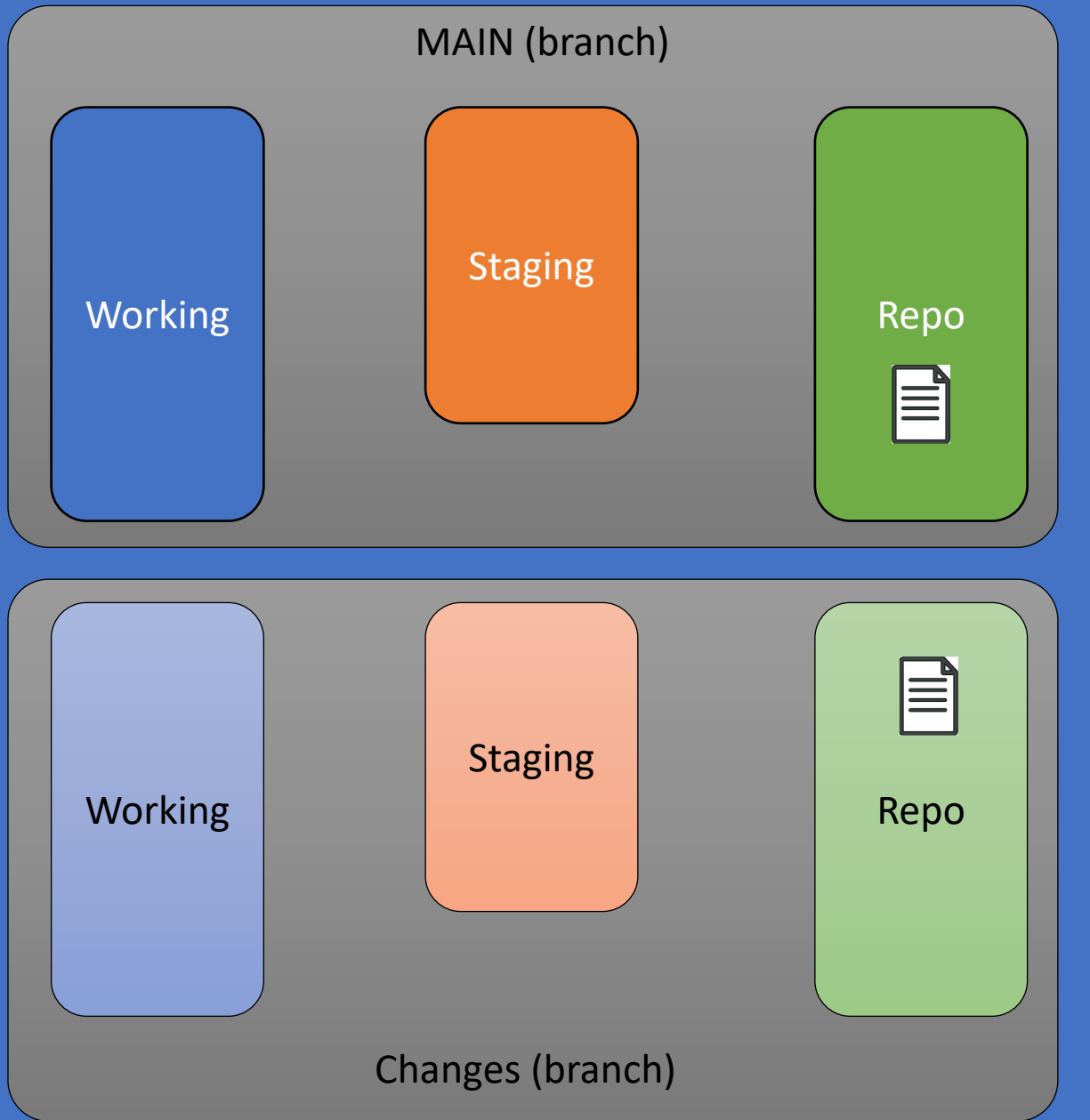






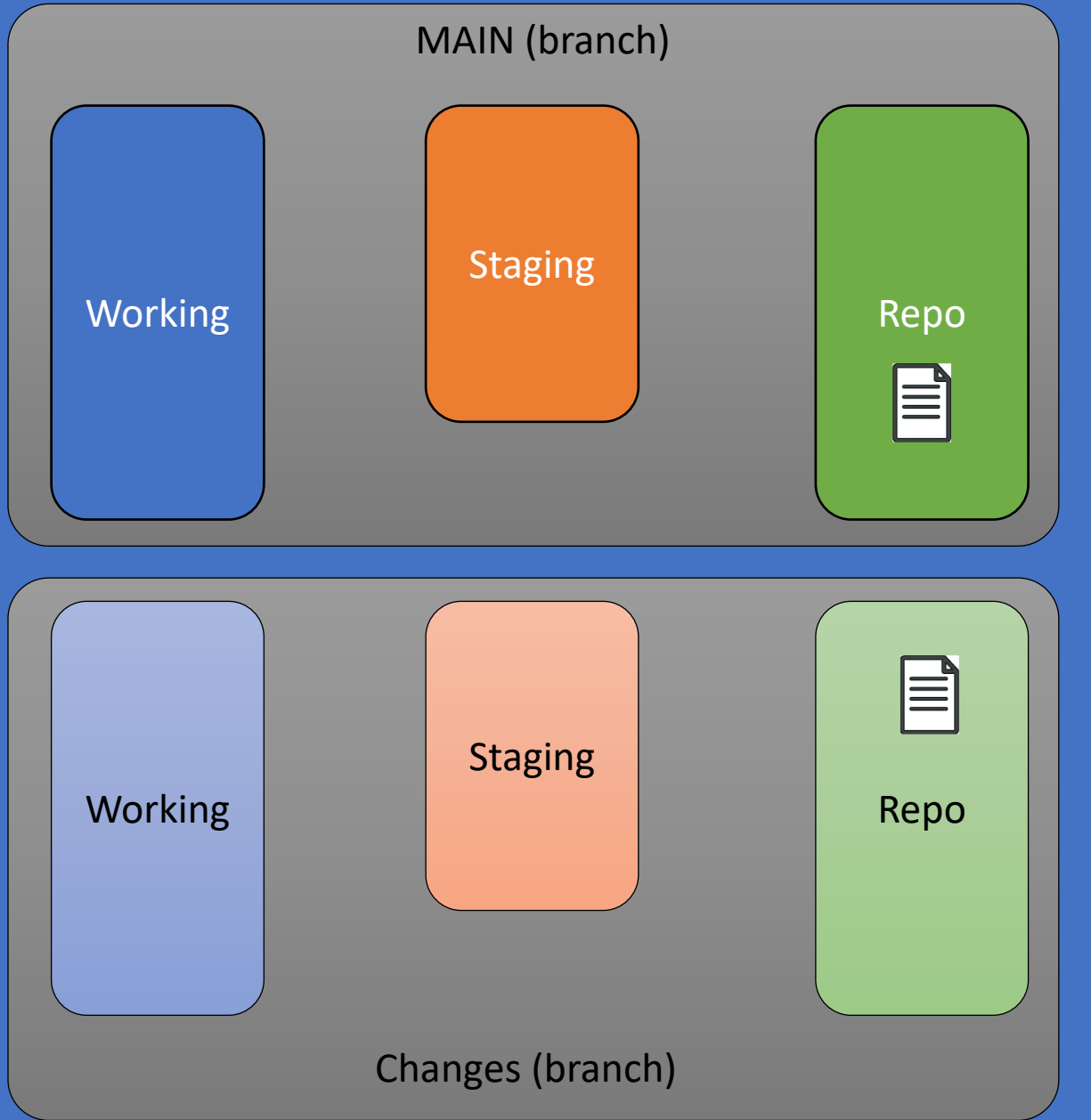


Local Repository

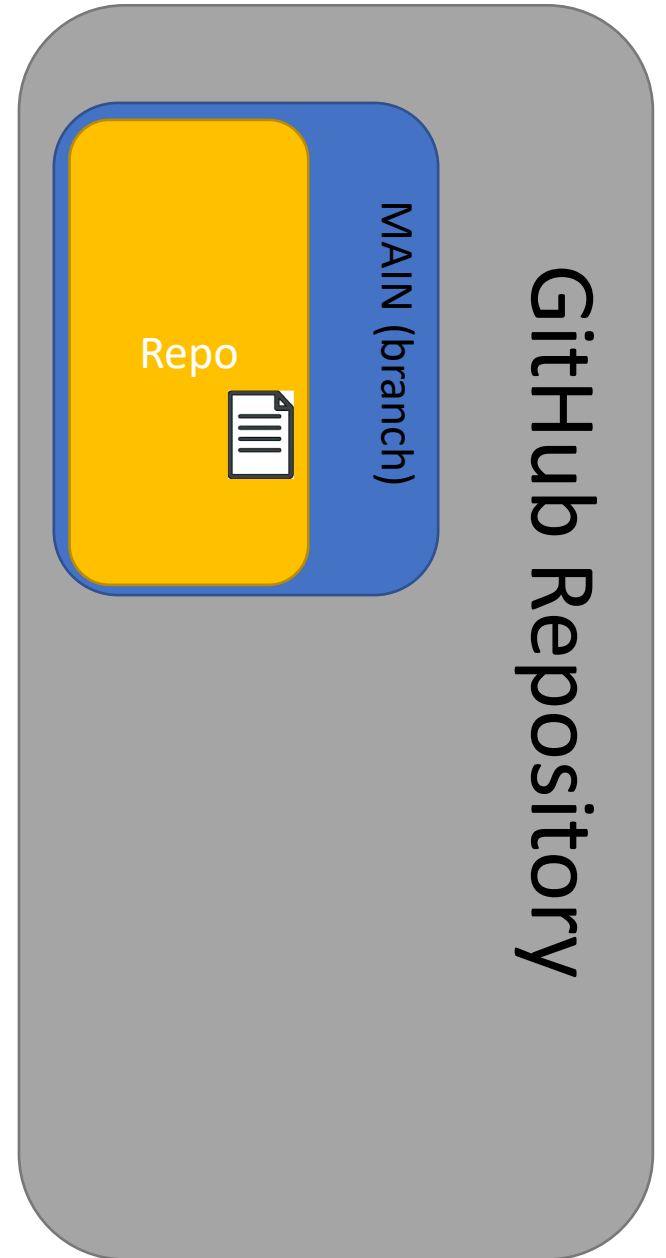


Local Repository

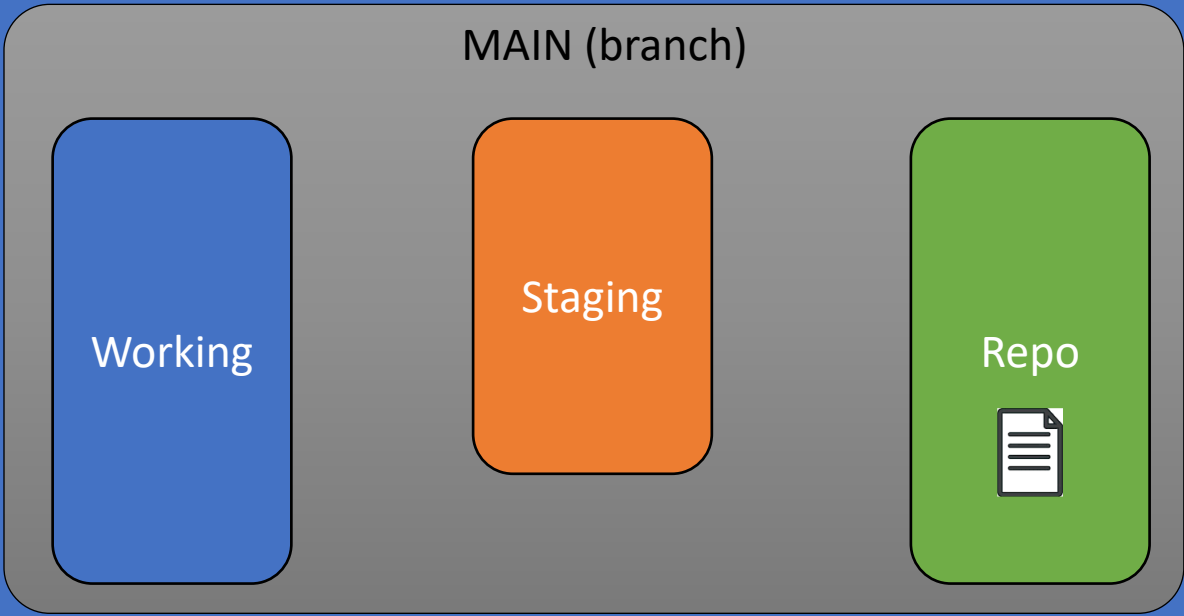
git branch -d changes



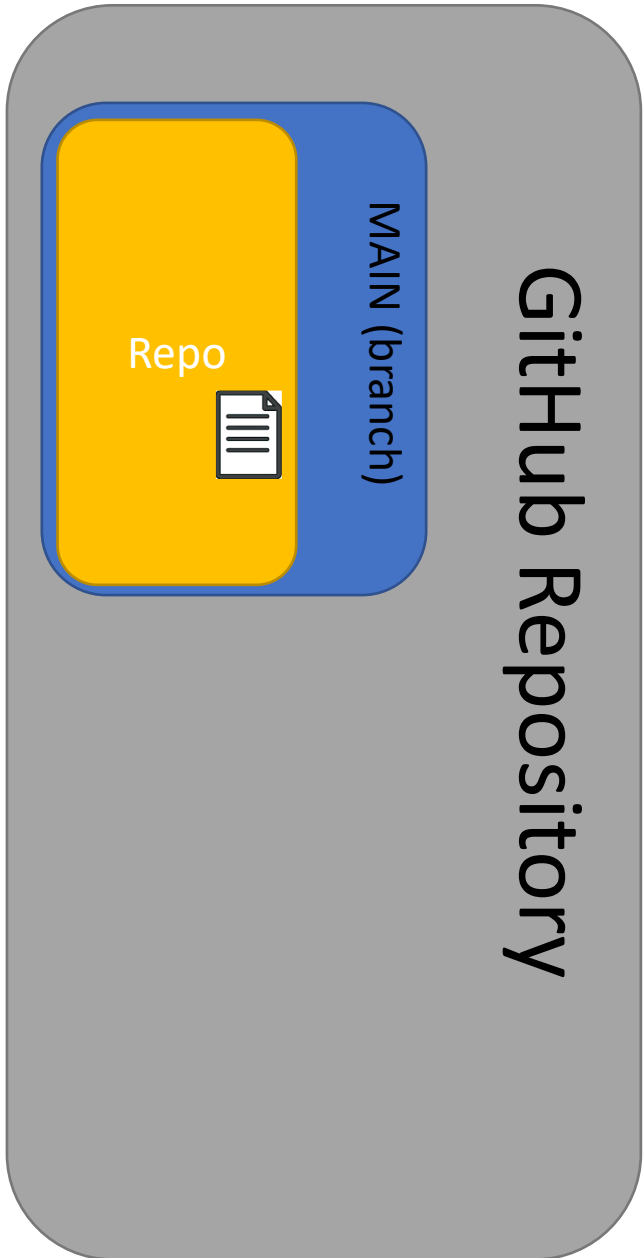
GitHub Repository



Local Repository



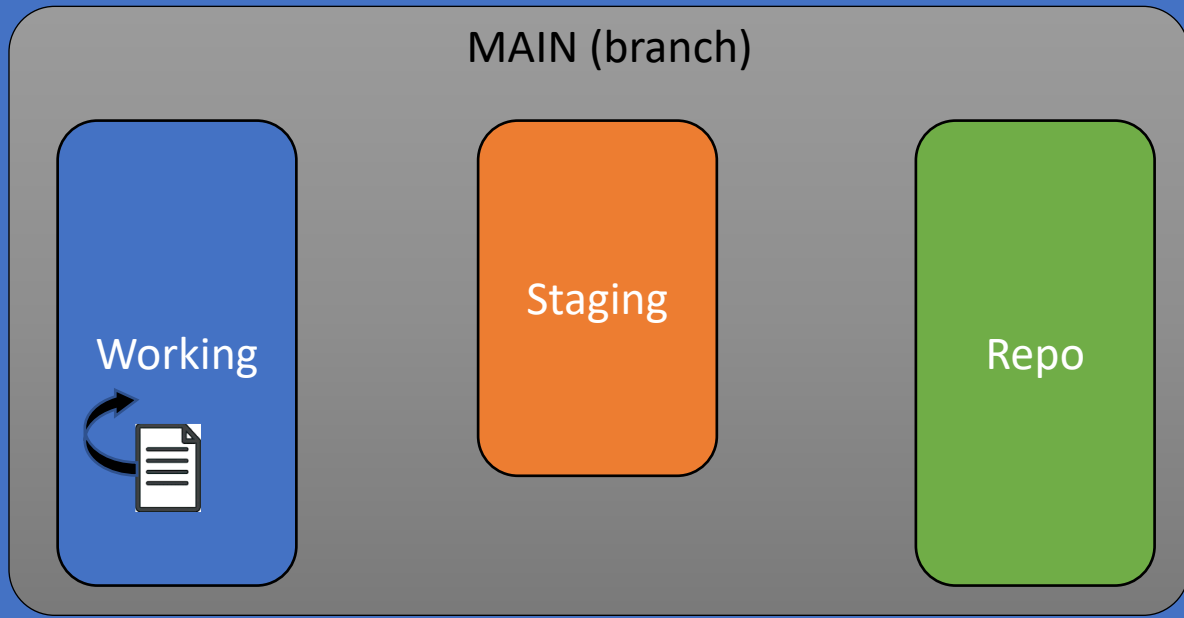
GitHub Repository



Reverse git commands

- git restore
- git reset

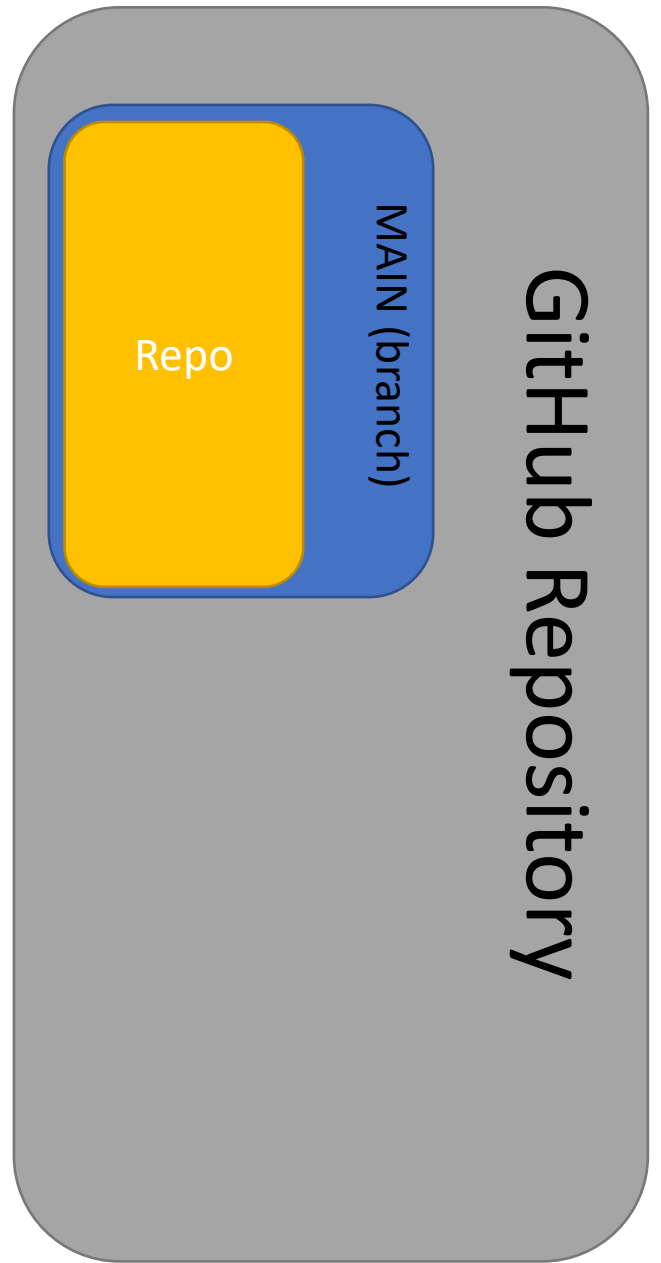
Local Repository



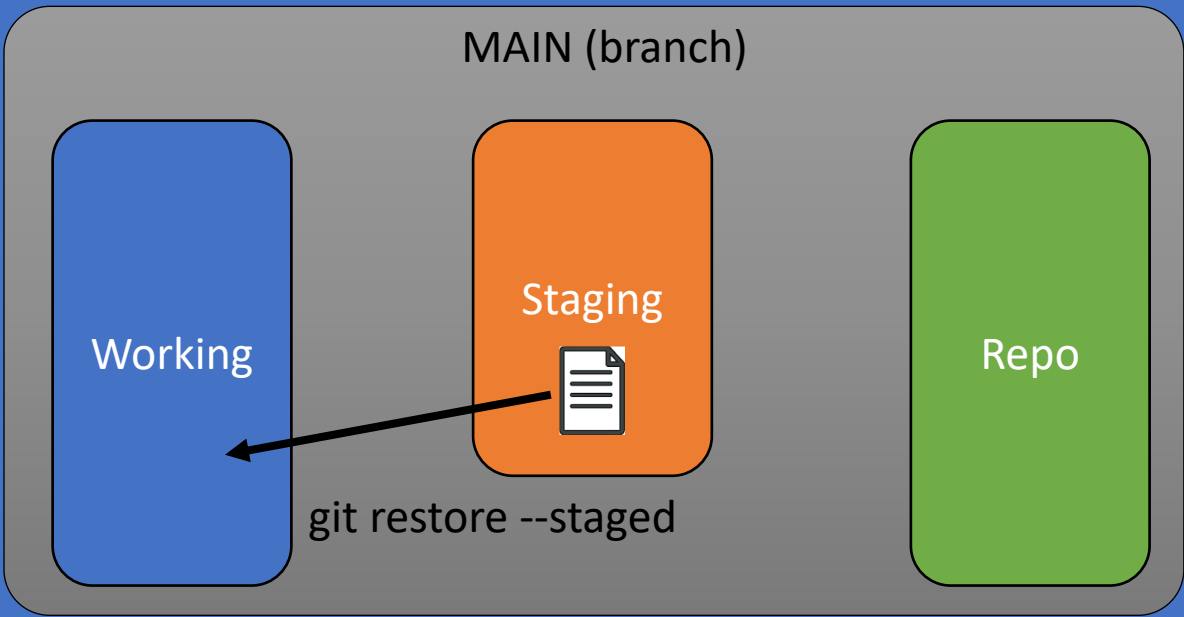
git reset --hard



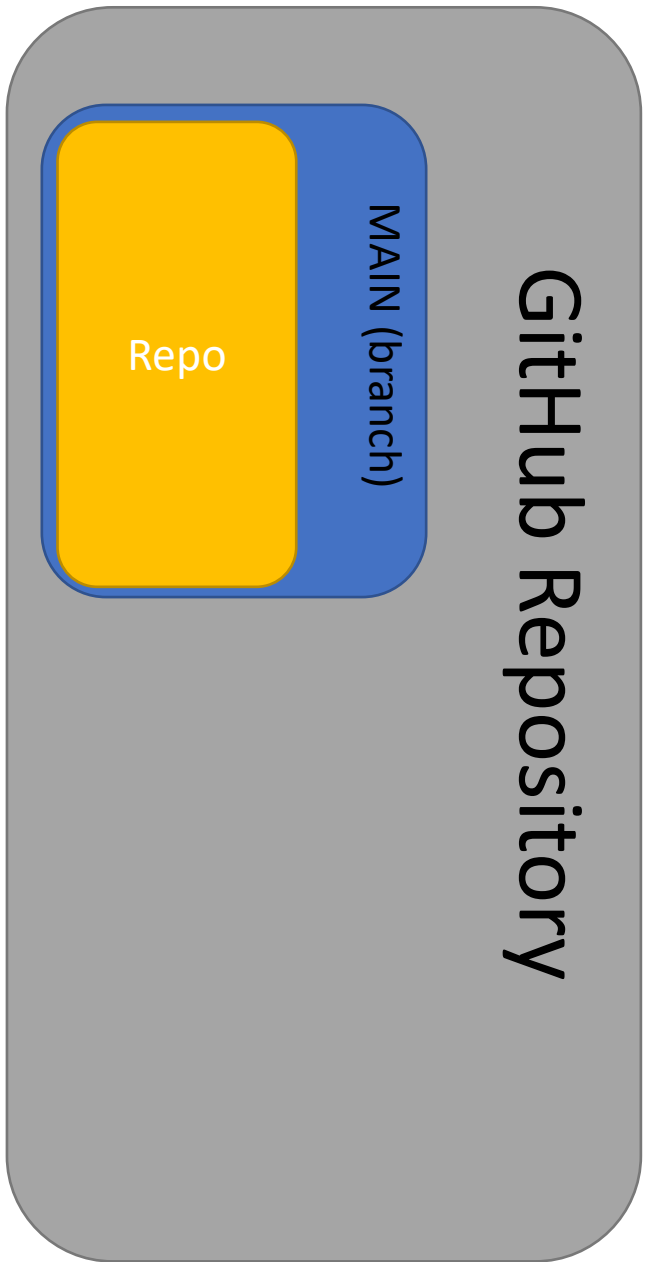
GitHub Repository



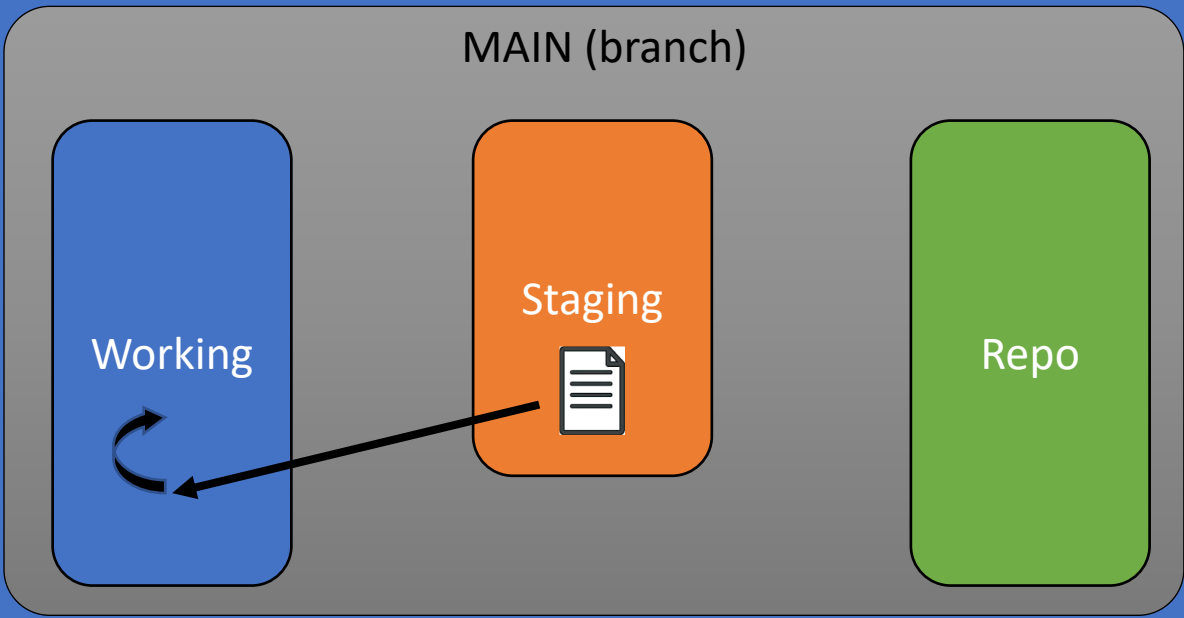
Local Repository



GitHub Repository



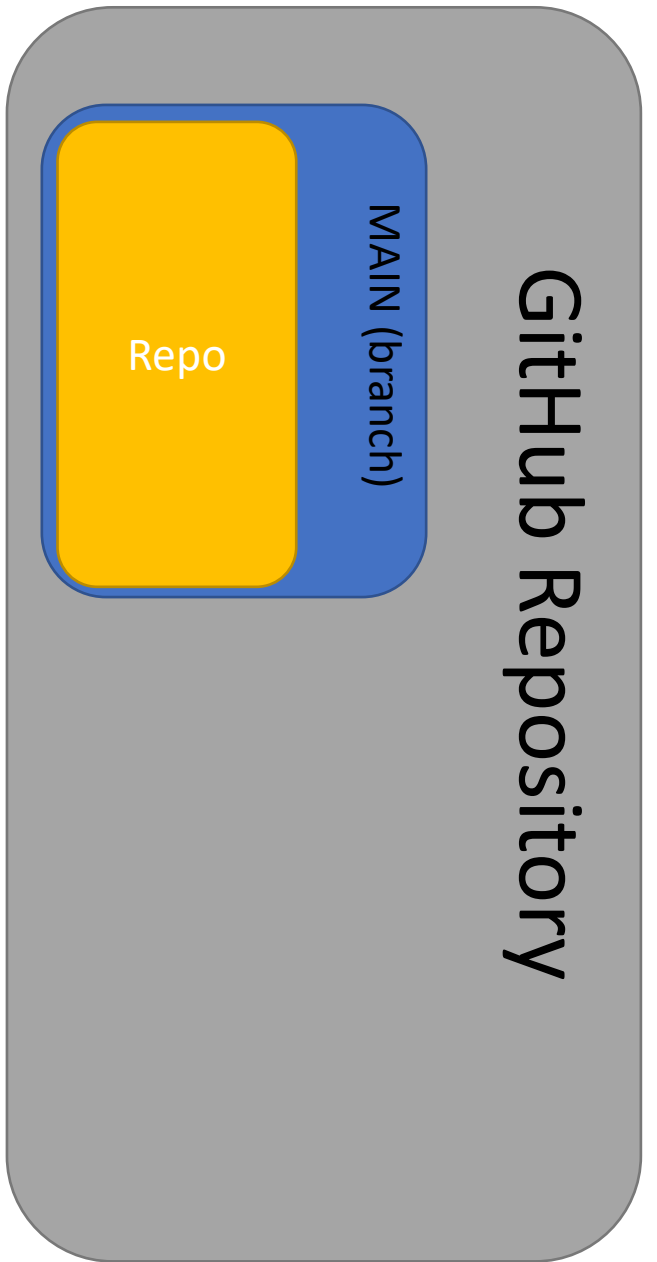
Local Repository



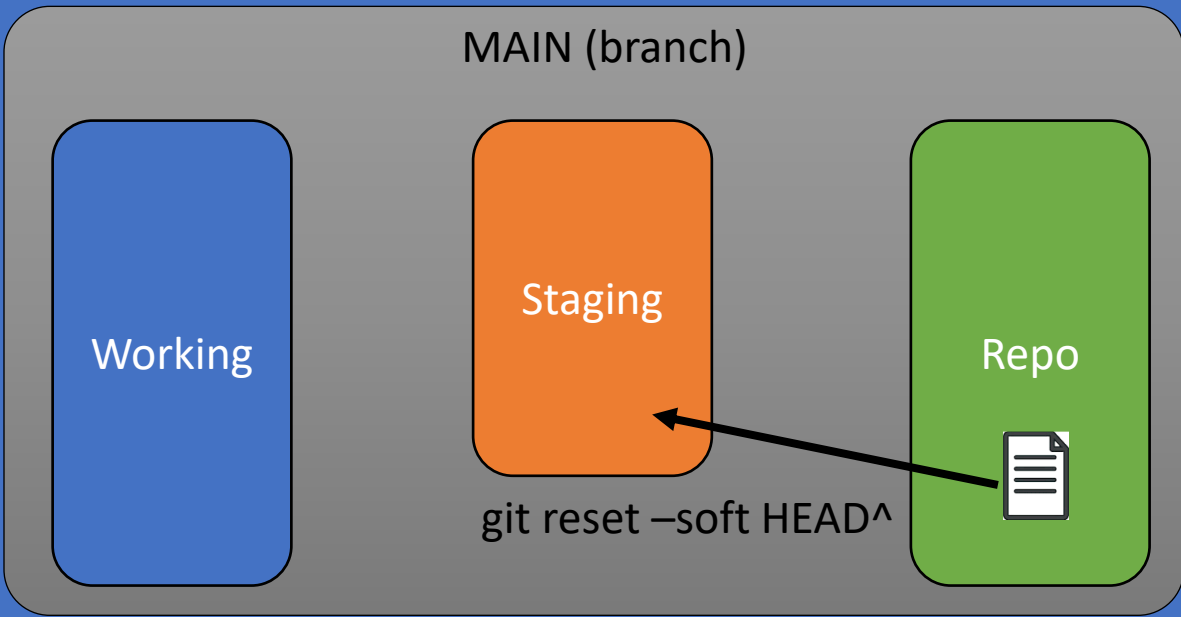
git reset --hard



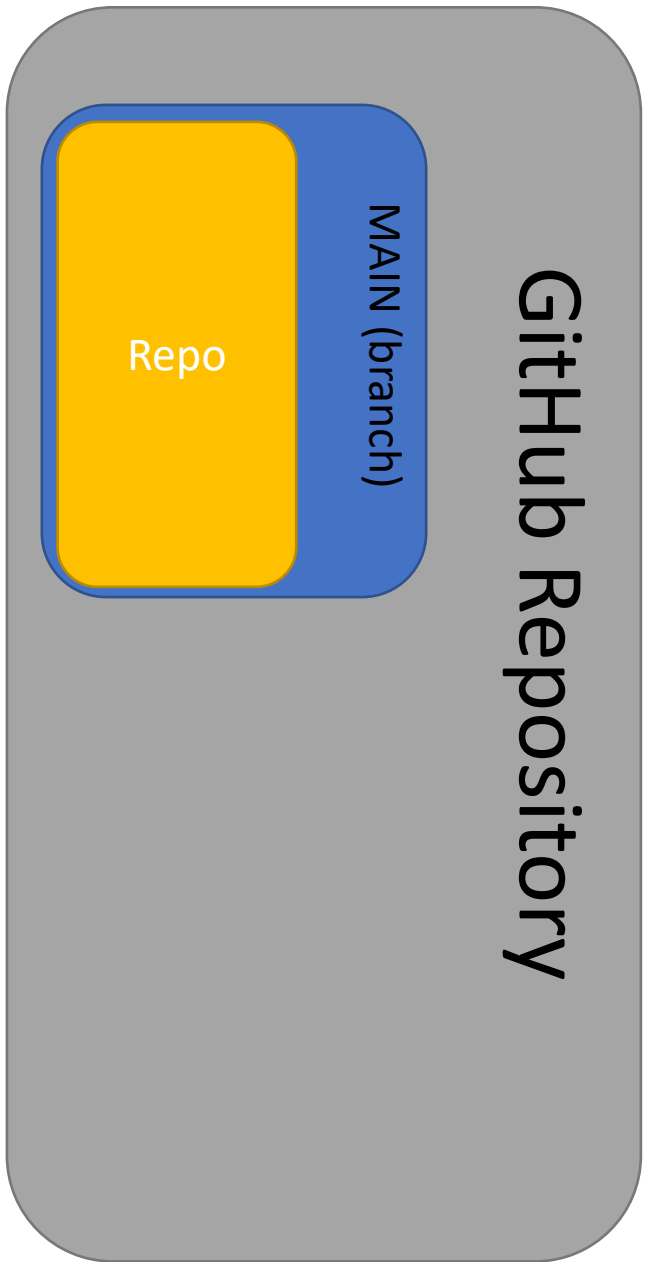
GitHub Repository



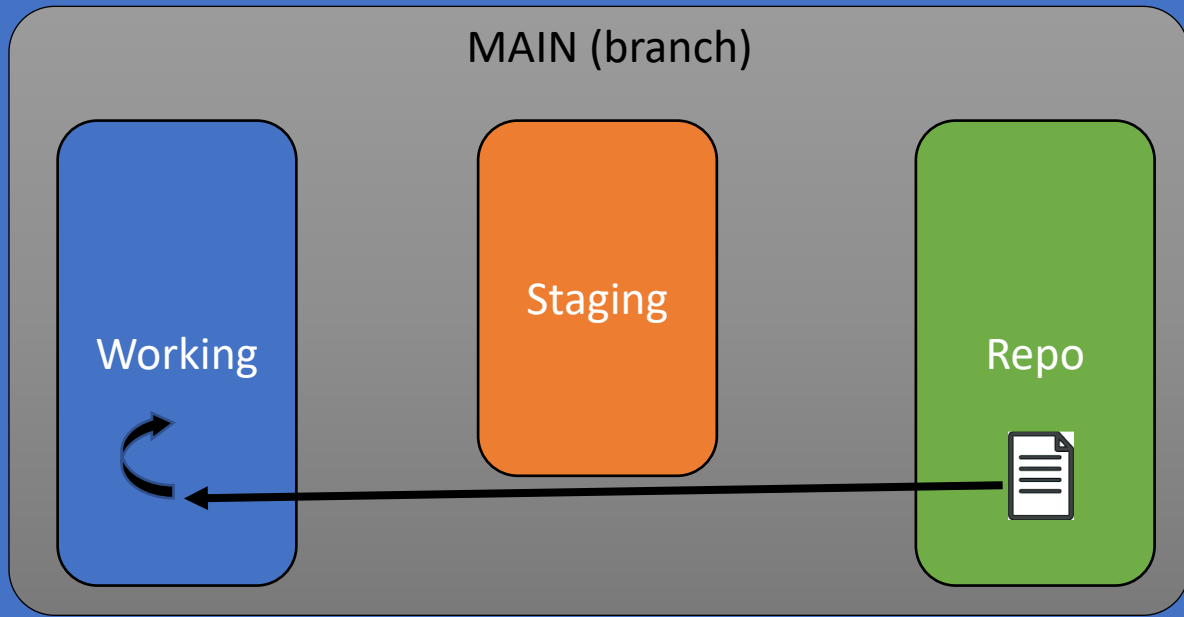
Local Repository



GitHub Repository



Local Repository



`git reset --hard HEAD^`



GitHub Repository

