

GIT






Dilum Navanjana

Senior Software Engineer

99x Technology

DilumN@99x.lk

99XTechnology

dilumn_ 

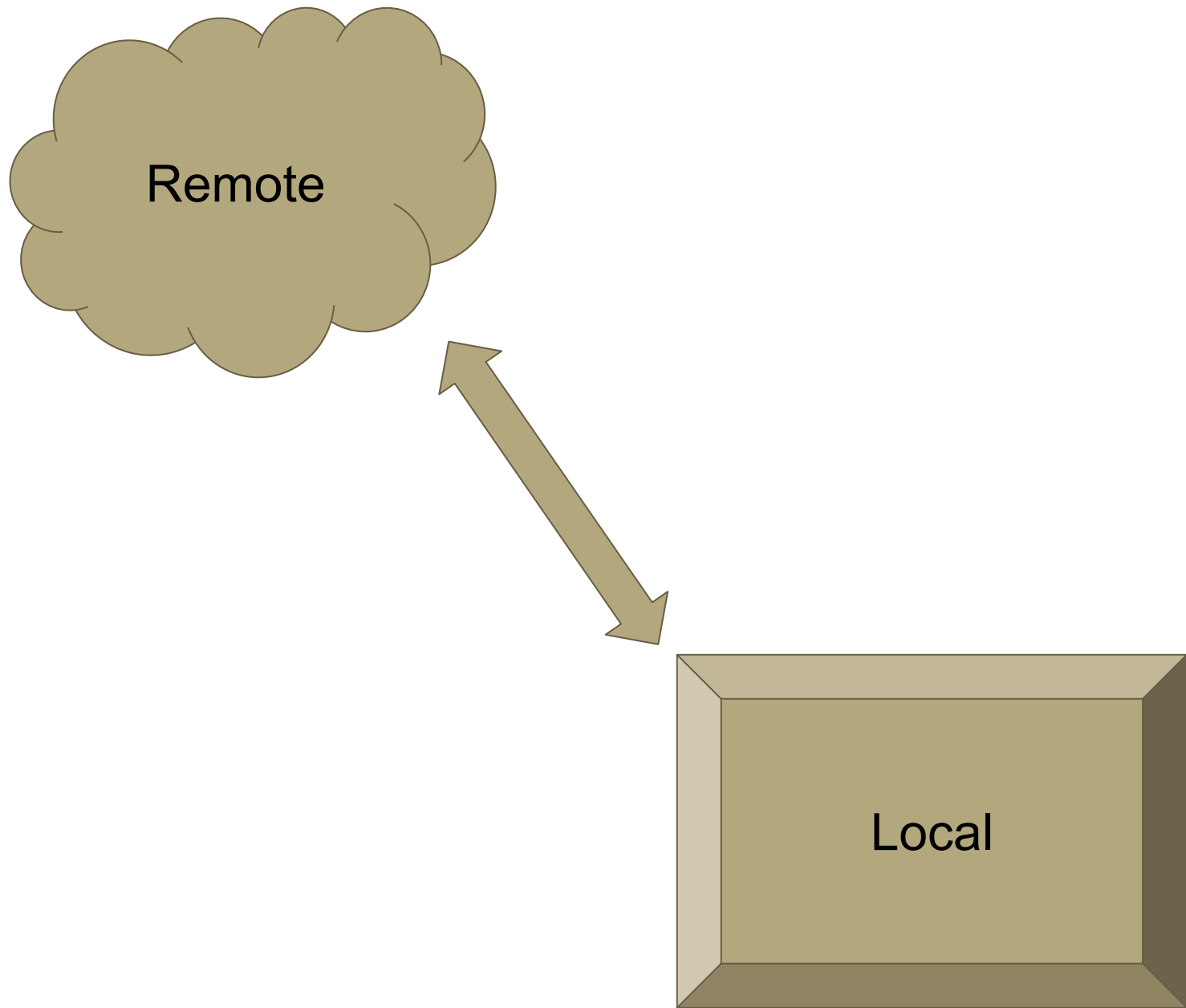
 dilumn

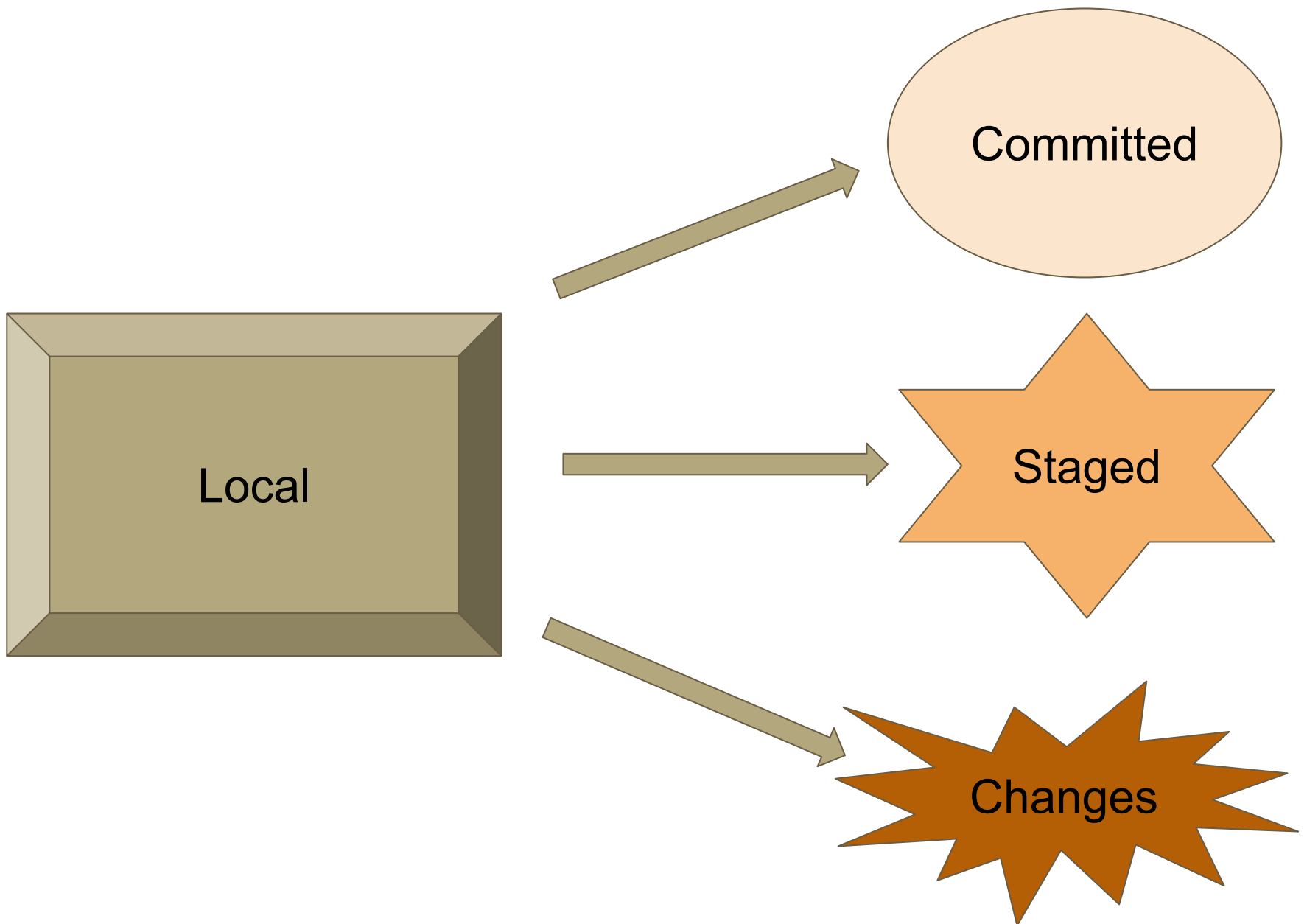
What is GIT

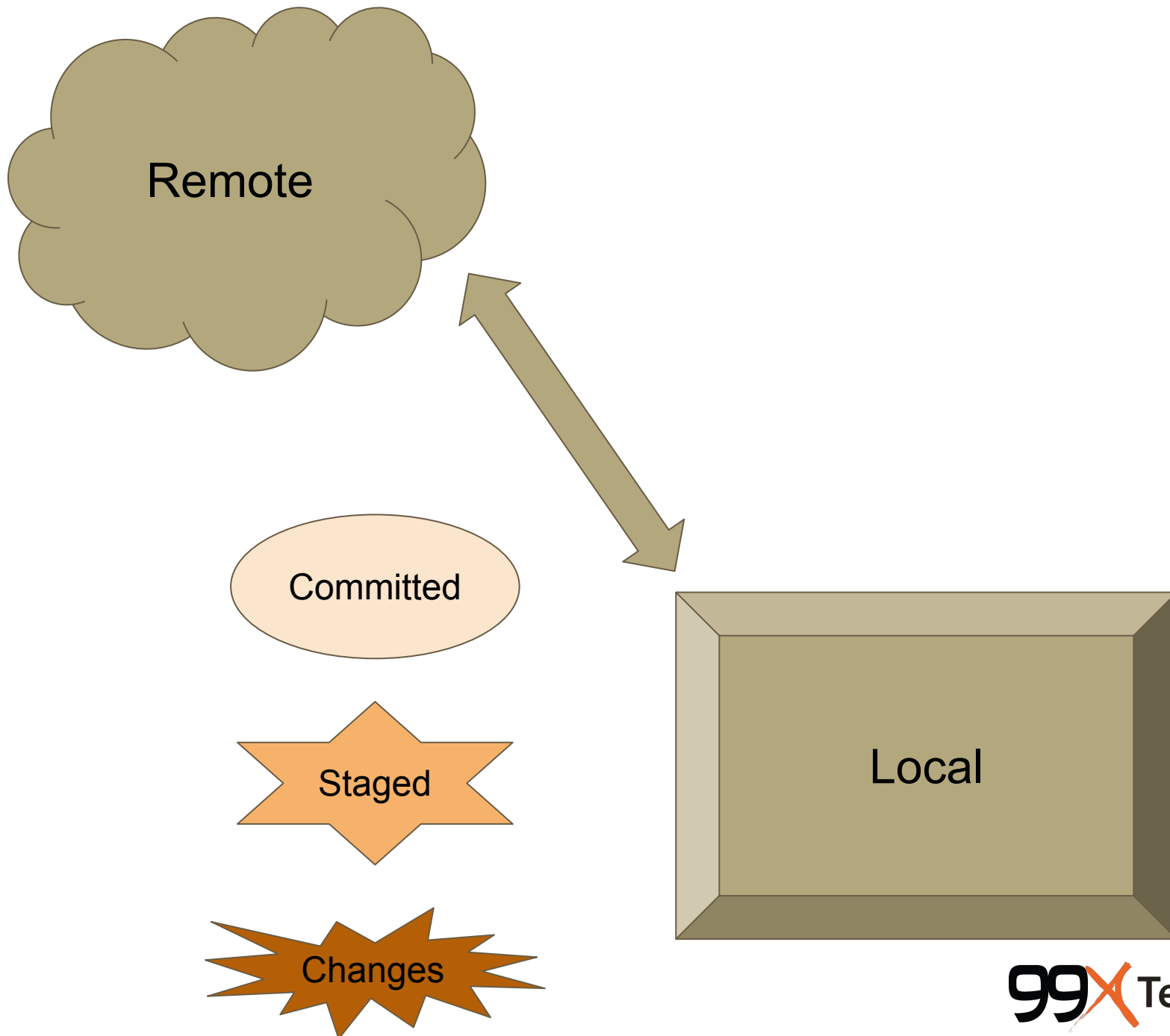


Why we use GIT











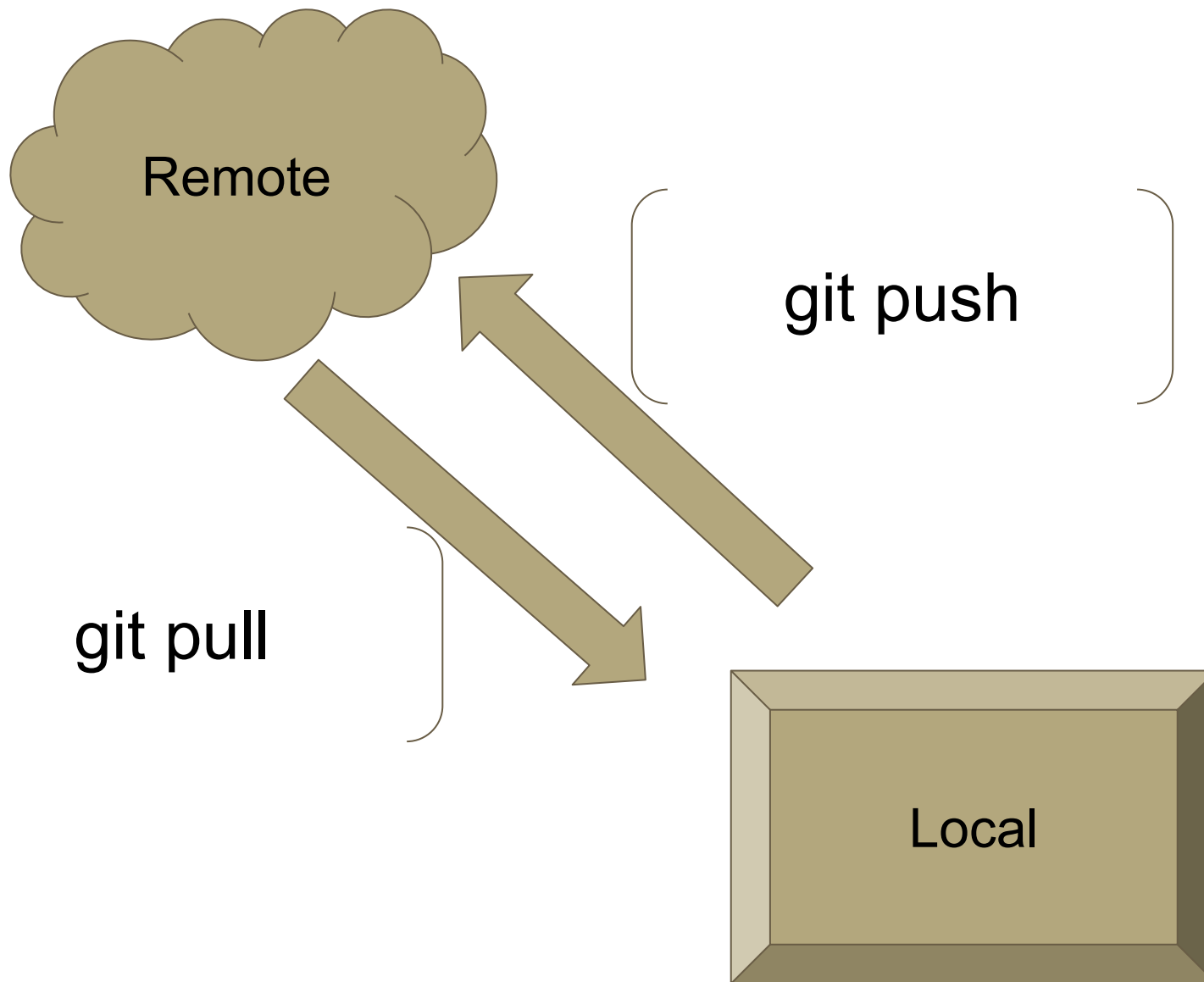
Committed

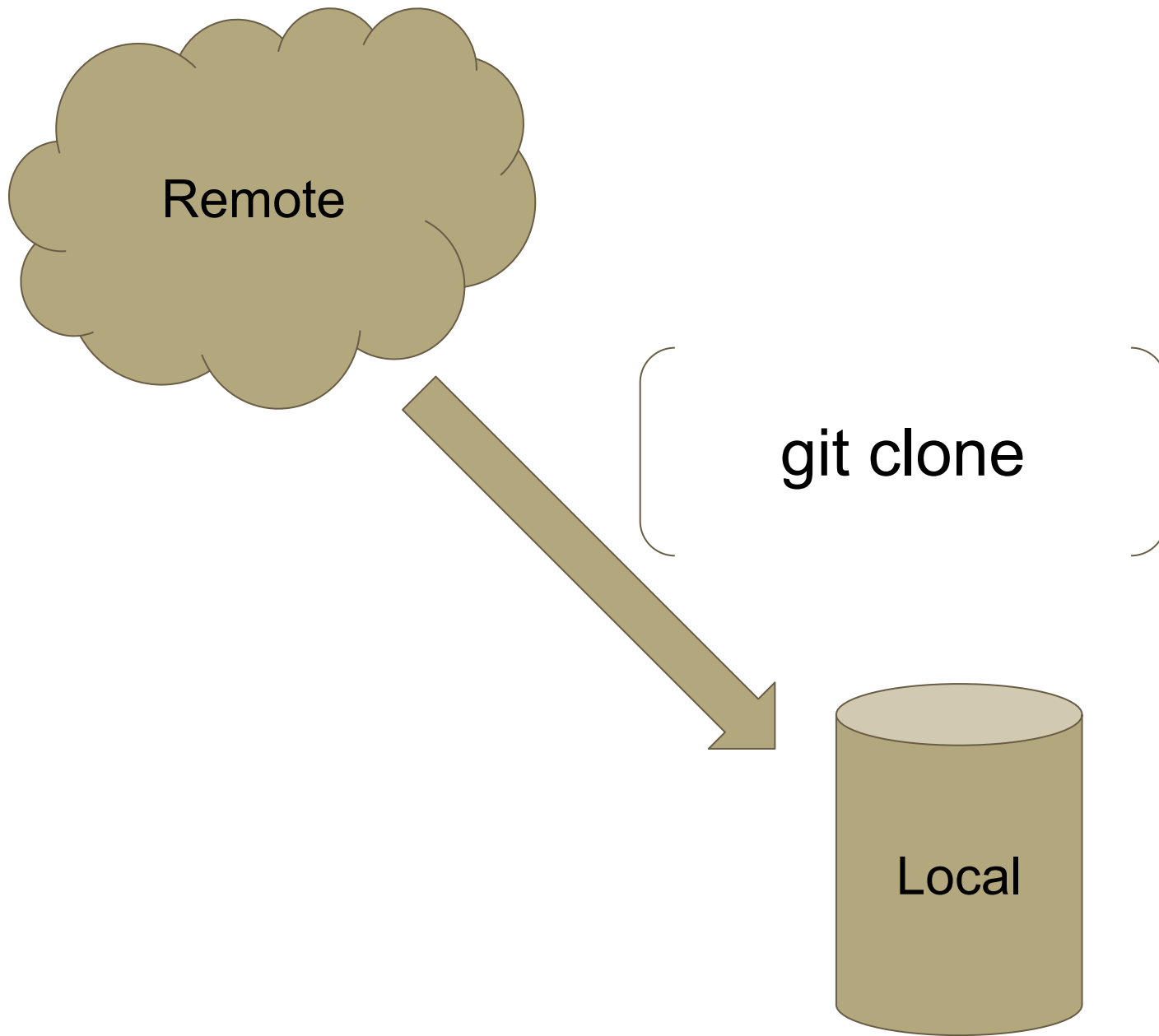
Staged

Changes

git commit

git add







Remote



Local



clone



add



commit



push



pull

Scenario 01

- Take a repository to the local for the first time

git clone <https://github.com/dilumn/git-plan.git>



Scan me

Scenario 02

- Take a repository to the local for the first time
- Make some changes
- Push those changes to remote

git clone <https://github.com/dilumn/git-plan.git>

git add .

git commit -m “commit message”

git push origin master

Scenario 03

- Push your existing project to Github

git init

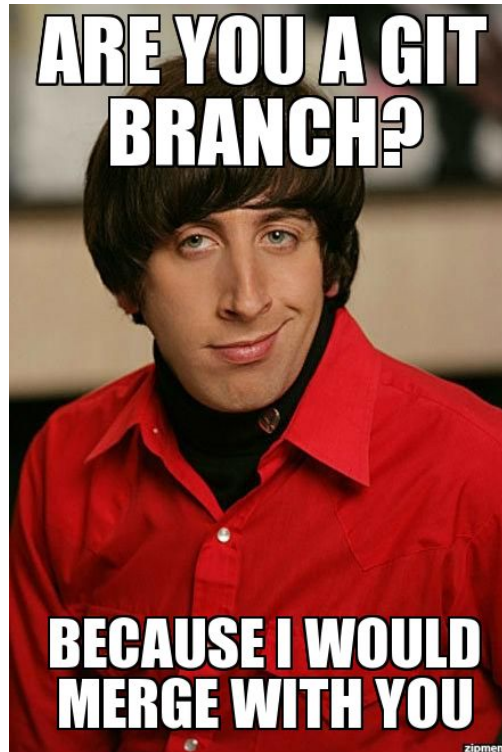
git remote add origin <https://github.com/dilumn/git-plan.git>

git add .

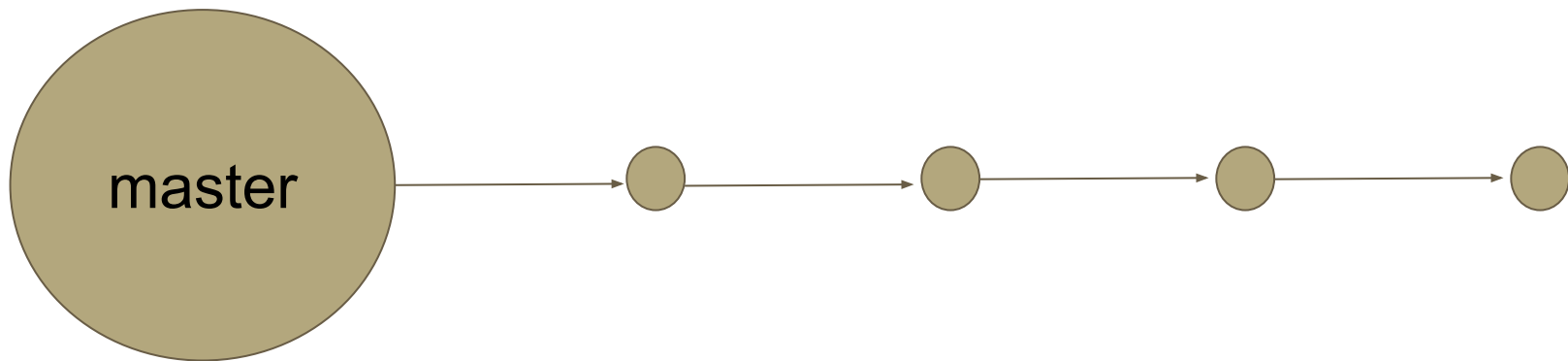
git commit -m “commit message”

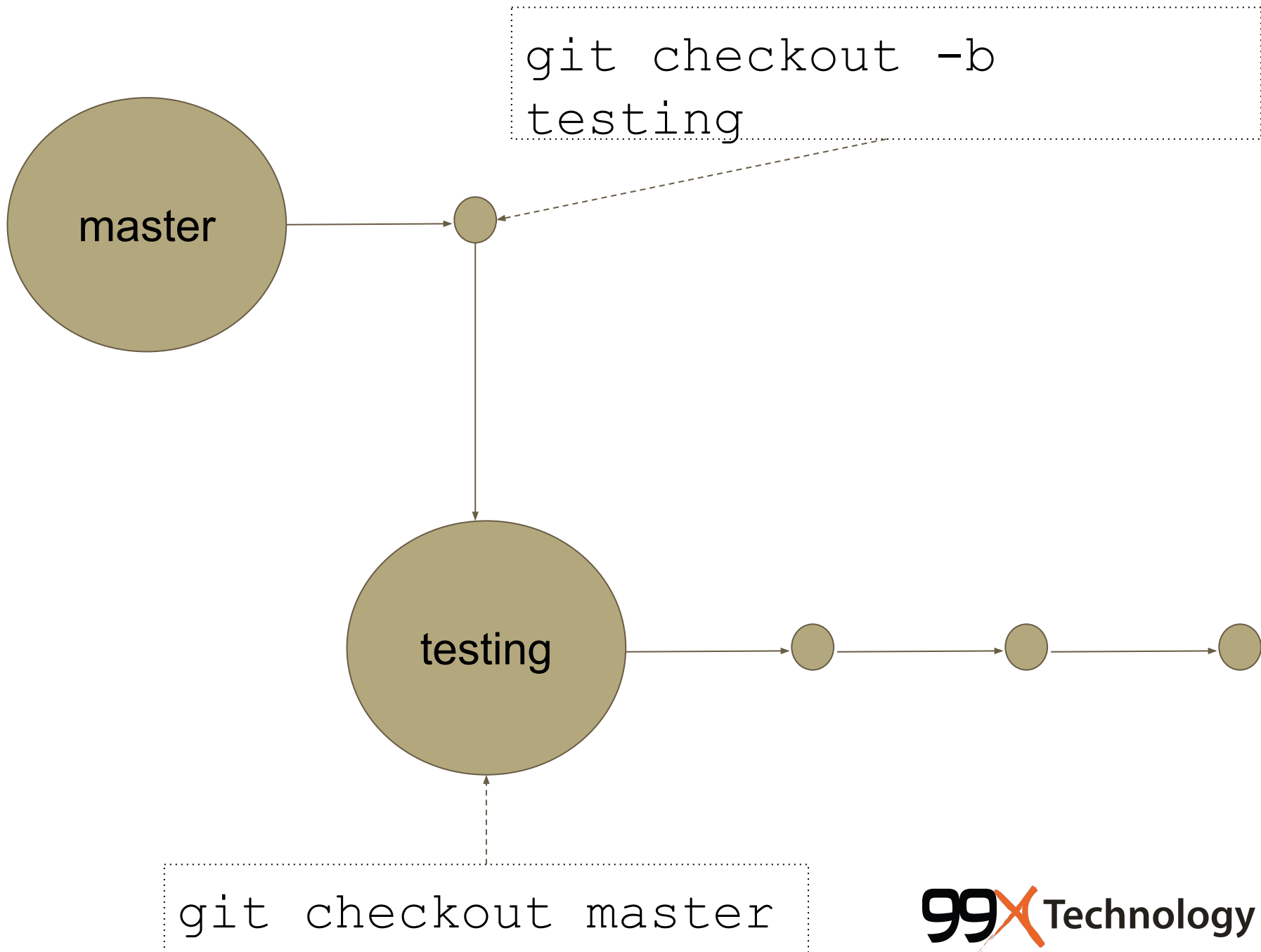
git push origin master

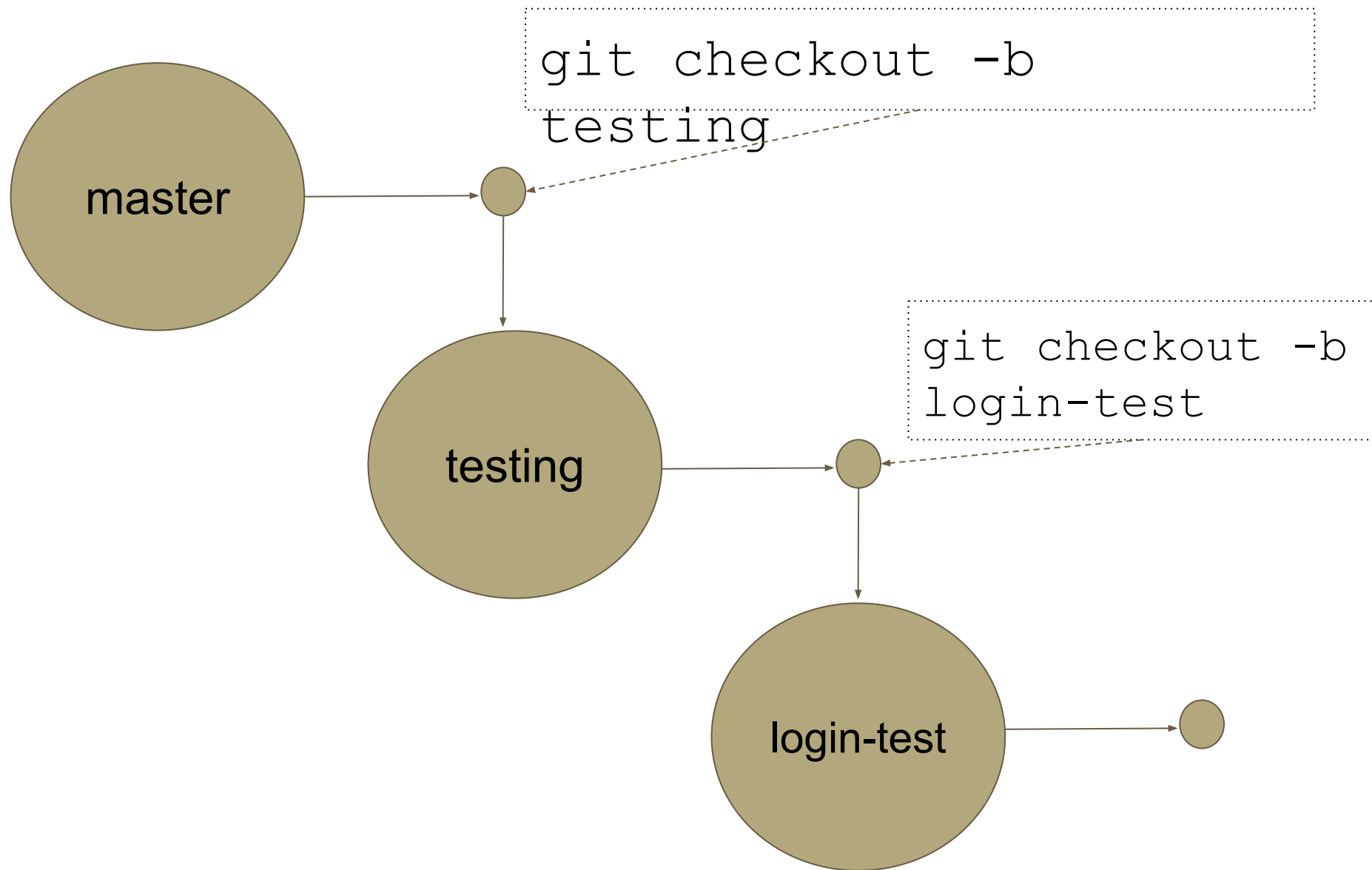
git branching

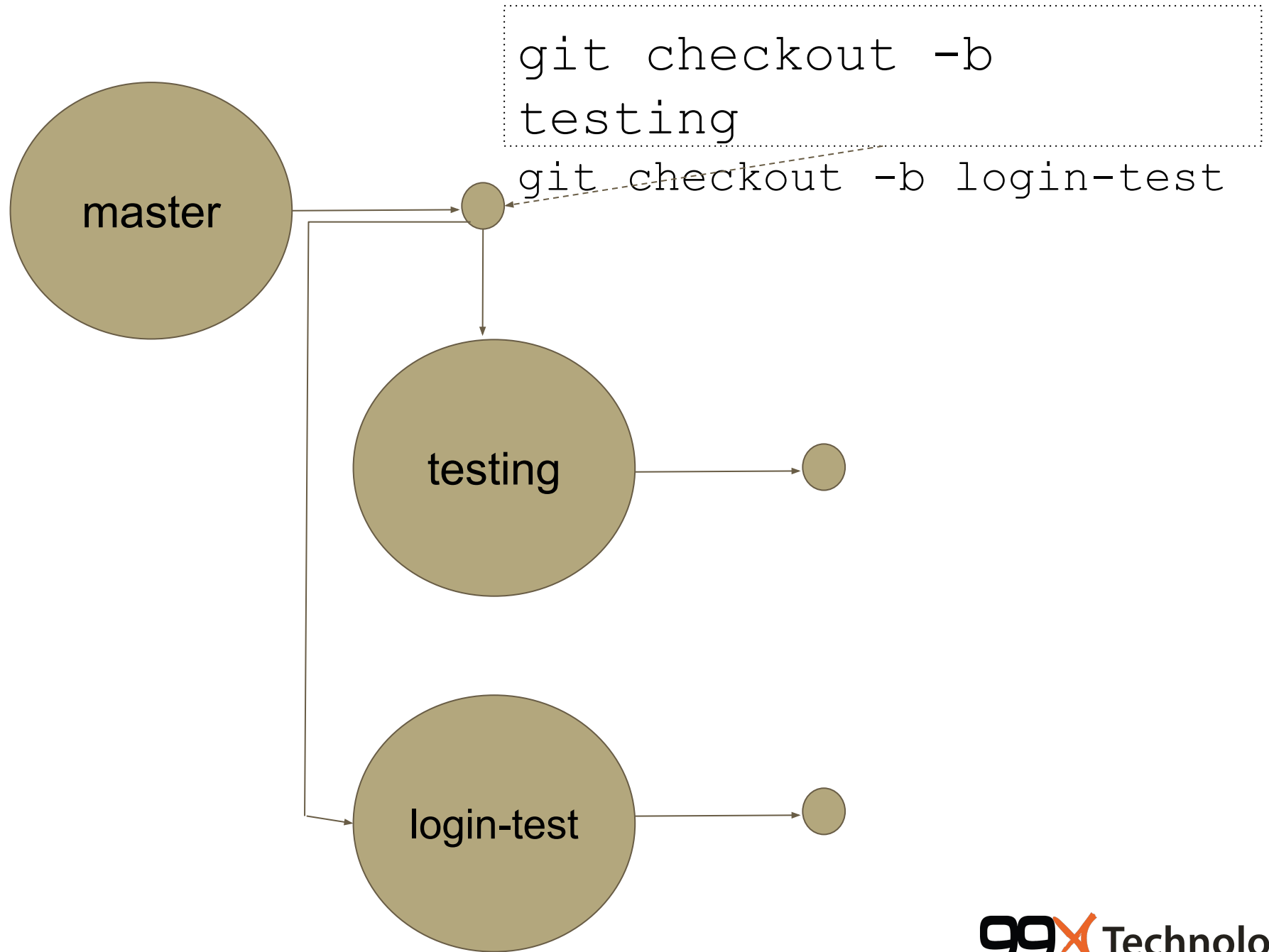








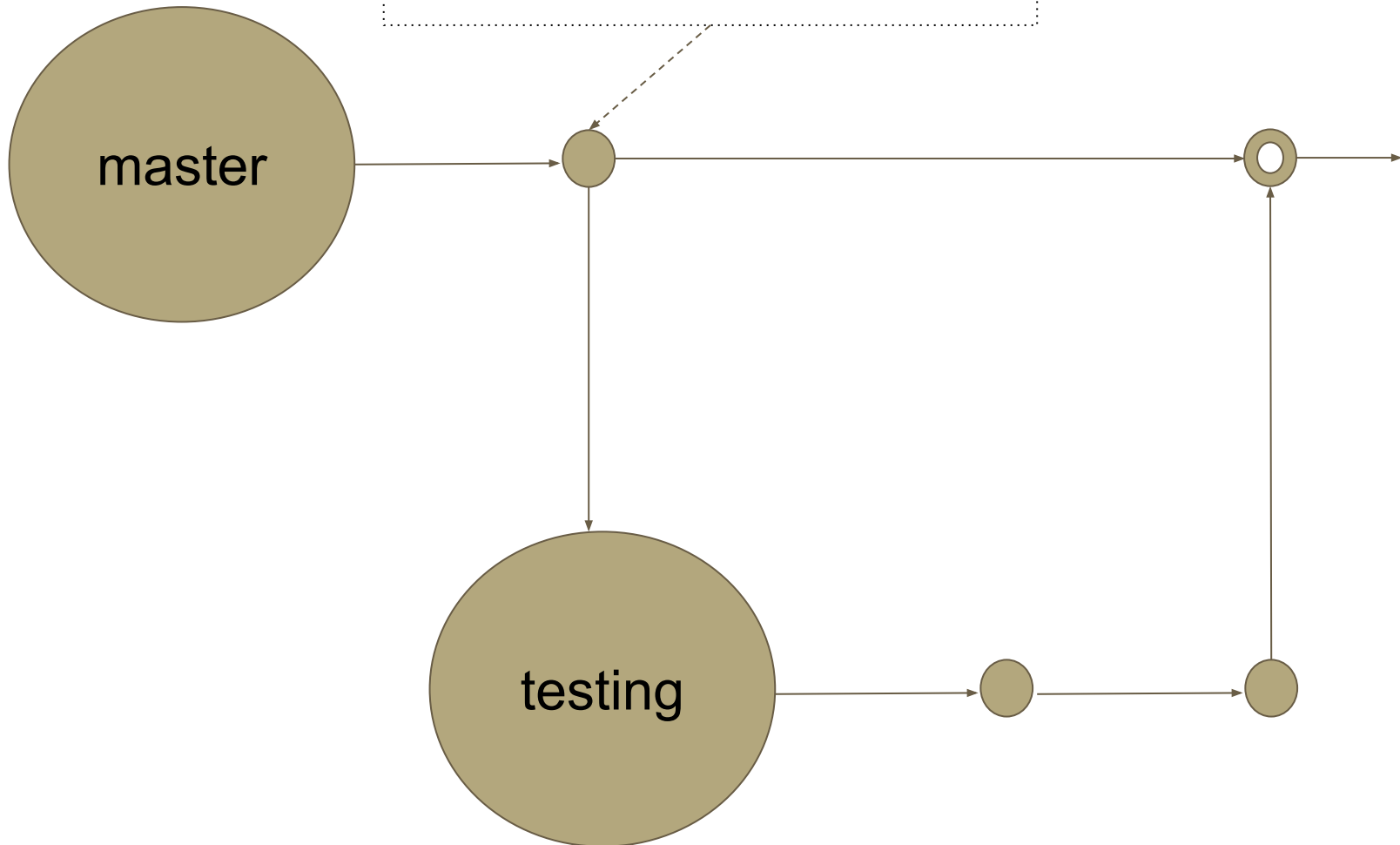




git merge



git merge testing



 Code

 Pull requests 0

 Projects 0

 Wiki

 Pulse

 Graphs

 Settings

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



base: **master** ▾

...

compare: **0-15-stable** ▾

✗ Can't automatically merge. Don't worry, you can still create the pull request.

 **Create pull request**

Discuss and review the changes in this comparison with others.












 9 commits

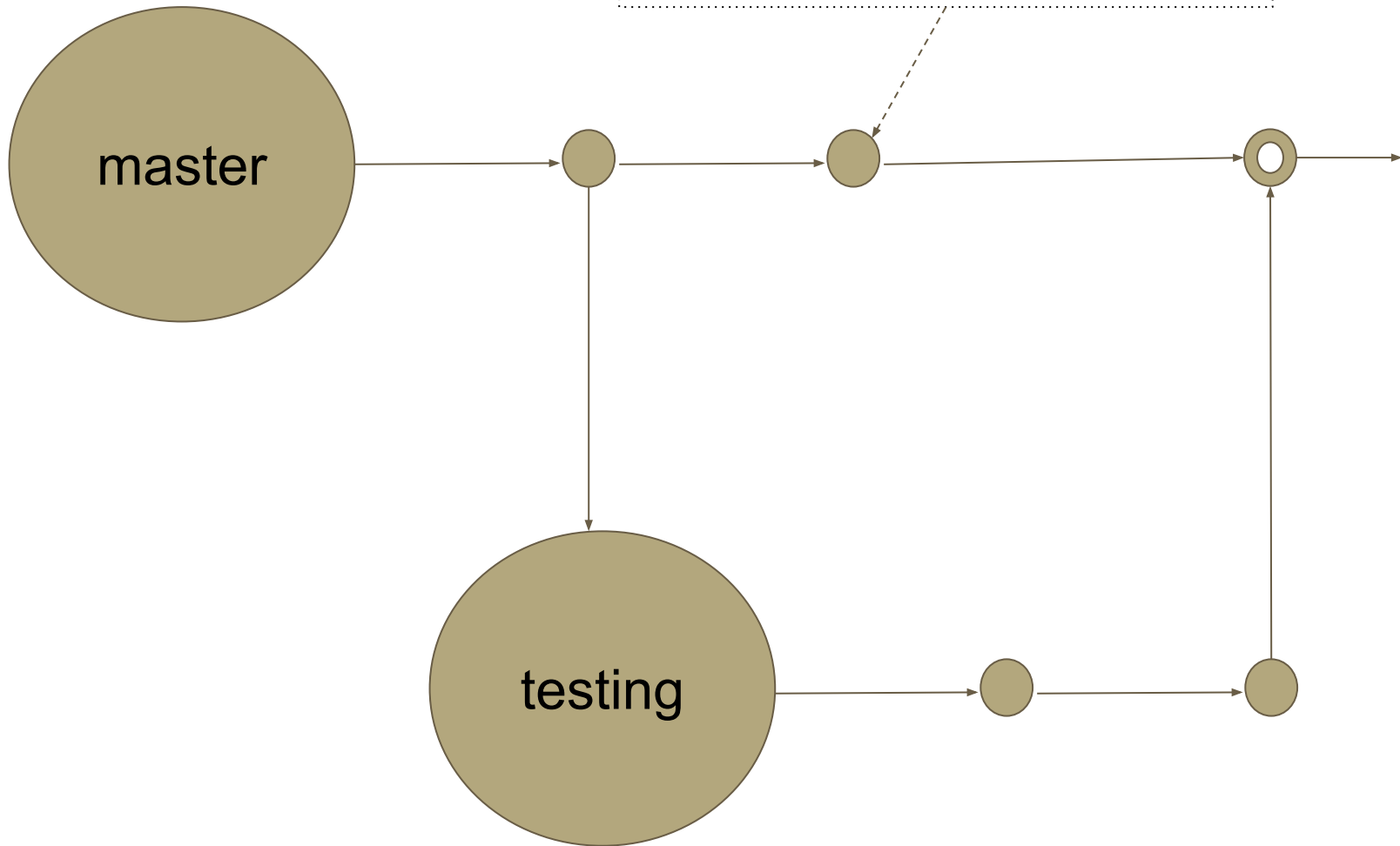
 9 files changed

 0 commit comments

 2 contributors

 Commits on Sep 05, 2013			
	halorgium	Allow PoolManager to incorrectly resume a task from within a task ...	0888ae9
 Commits on Sep 06, 2013			
	tarcieri	Revert "Allow PoolManager to incorrectly resume a task from within a ..."	20ecb50
	tarcieri	Warn for nested task resumption in debug mode	846bec6
	halorgium	Revert "Warn for nested task resumption in debug mode" ...	871690b
	halorgium	Allow PoolManager to incorrectly resume a task from within a task ...	0cf1588
	halorgium	Only raise if \$CELLULOID_DEBUG is set ...	cf09728
	tarcieri	Bump version to 0.15.1	cd7847d

git merge testing



merge conflicts

Demo

Scenario 04

— Create a new branch “branch1” from master
Do some changes to a file. Commit & push them —
Go to master branch, change the same file, commit
& push the changes
Create a pull request from “branch1” to master
merge “branch1” to master

```
git checkout -b branch1
```

```
git add .
```

```
git commit -m "changes from branch1"
```

```
git push origin branch1
```

```
git checkout master
```

```
git add .
```

```
git commit -m "changes from master"
```

```
git push origin master
```

```
git merge branch1
```

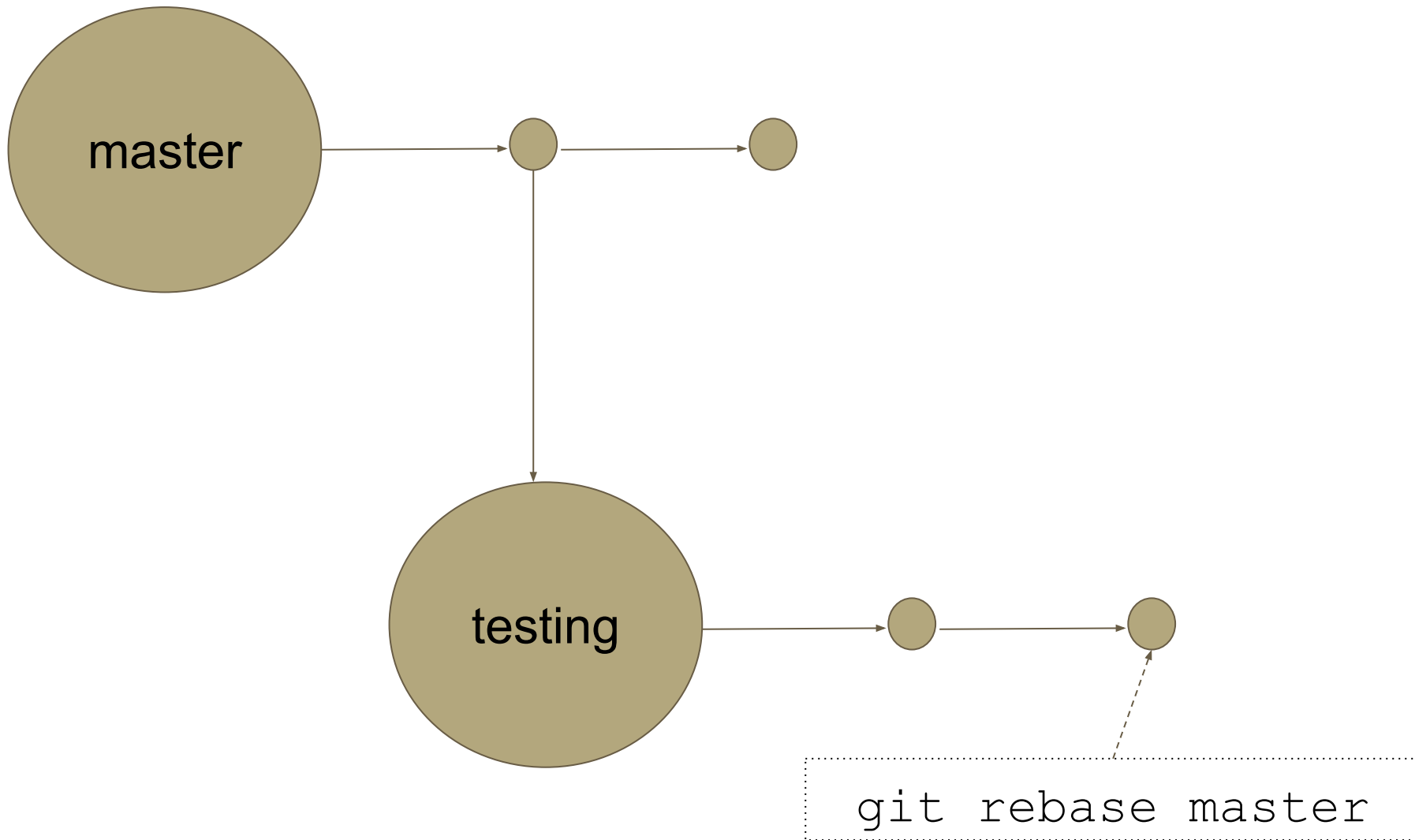
```
git add .
```

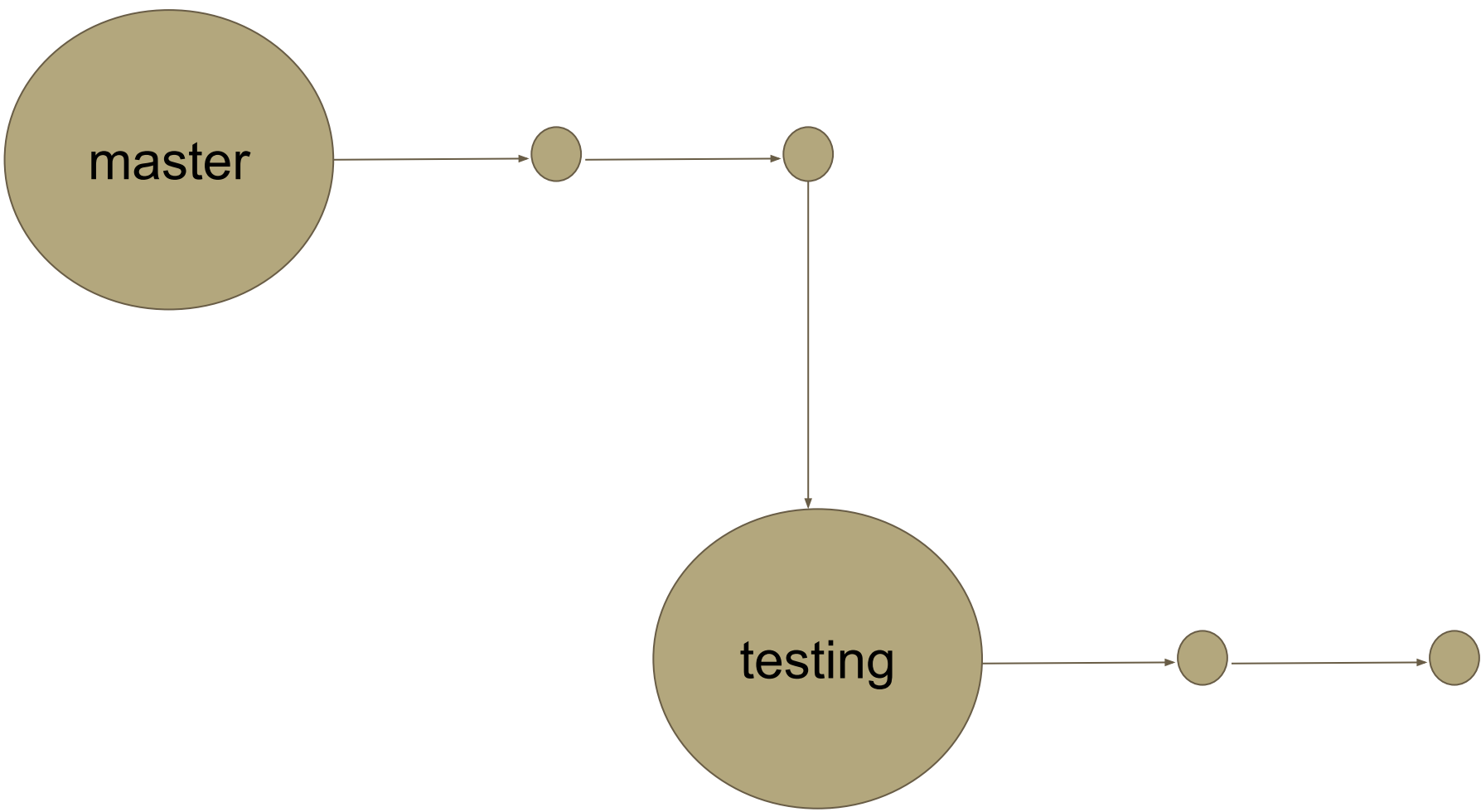
```
git commit -m "merge conflict fixes"
```

```
git push origin master
```

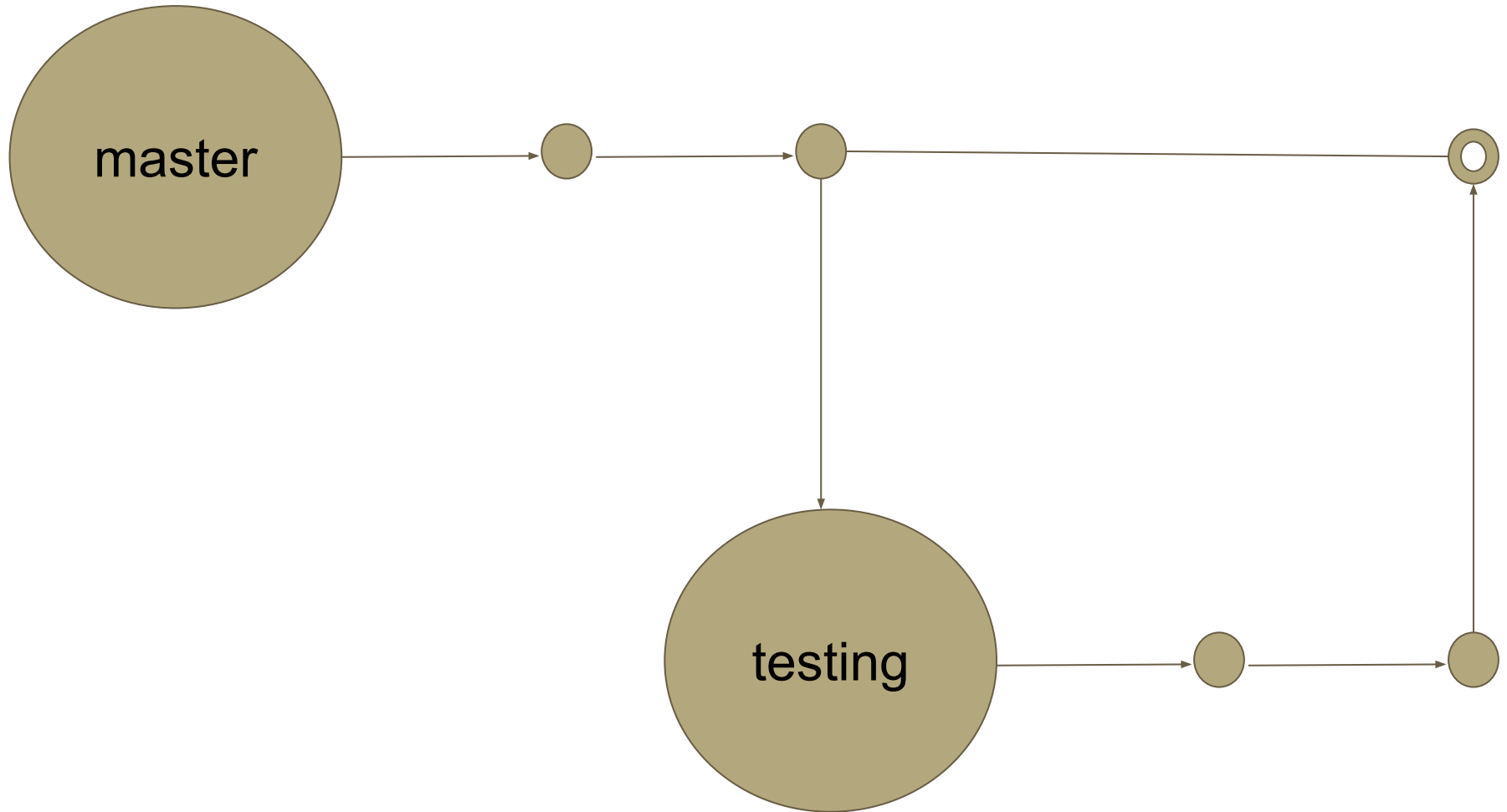
git rebase







git rebase & merge



Demo

Scenario 05

Create a new branch "branch2" from master
Do some changes to a file. Commit & push them
Go to master branch, change the same file, commit
& push the changes
Rebase "branch2" from master
merge "branch2" to master

git checkout -b branch2

git add .

git commit -m "changes from branch2"

git push origin branch2

git checkout master

git add .

git commit -m "changes from master"

git push origin master

git checkout branch2

git rebase master

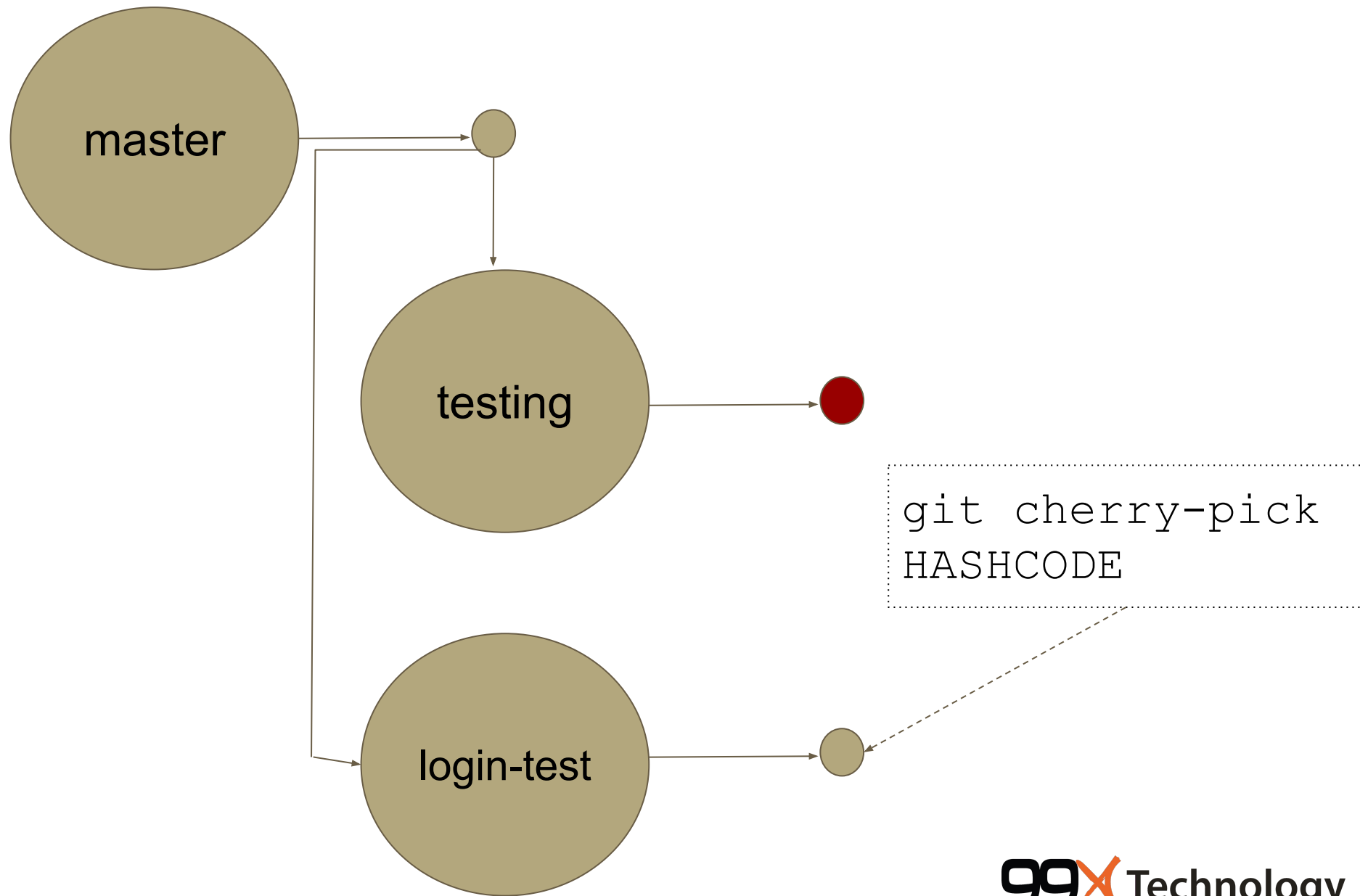
git add .

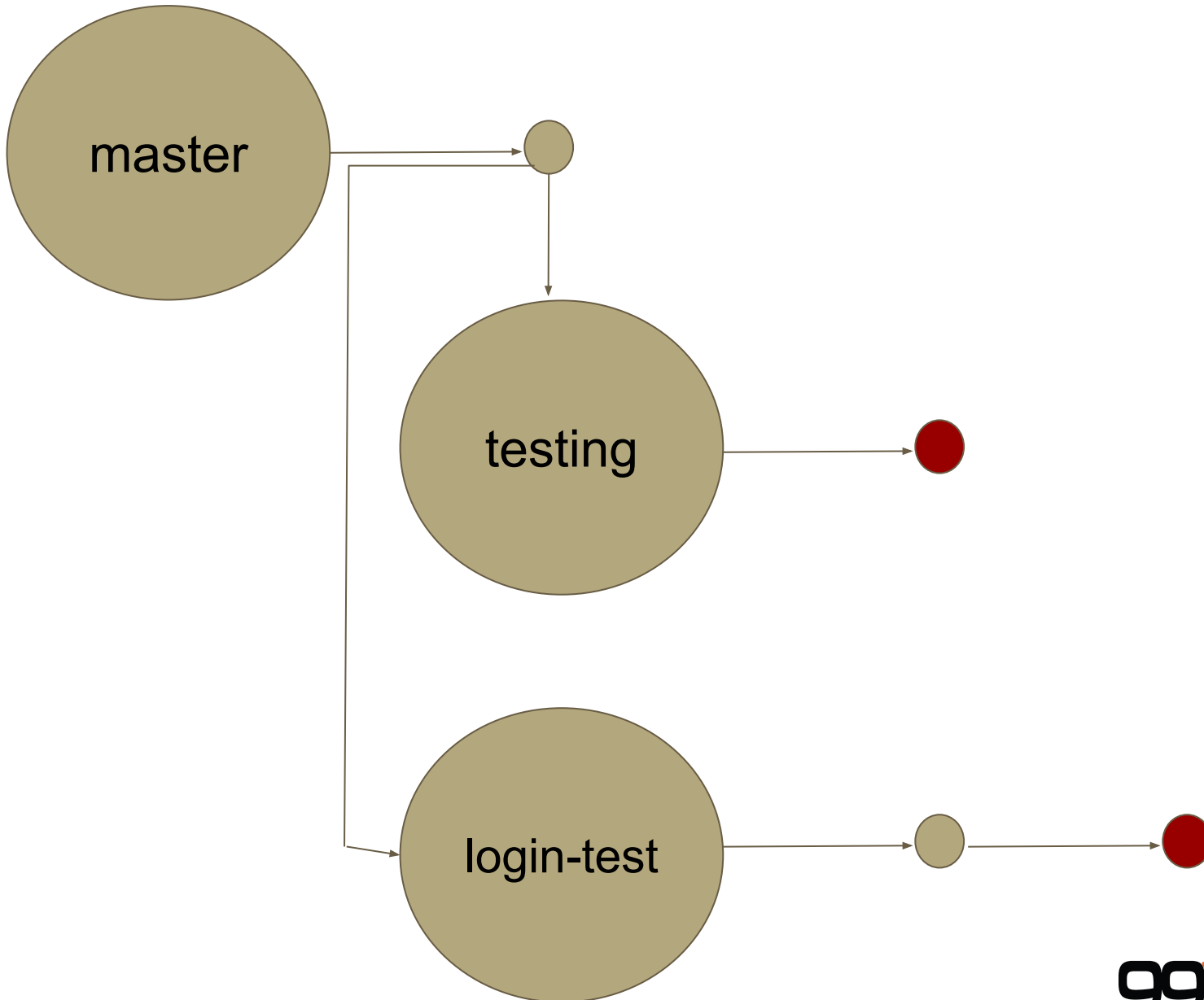
git rebase --continue

git push origin branch2 --force

git cherry-pick







Scenario 06

Cherry pick branch1 5kw32
commit to branch2

```
git checkout branch2
```

```
git cherry-pick 5kw32
```

git stash



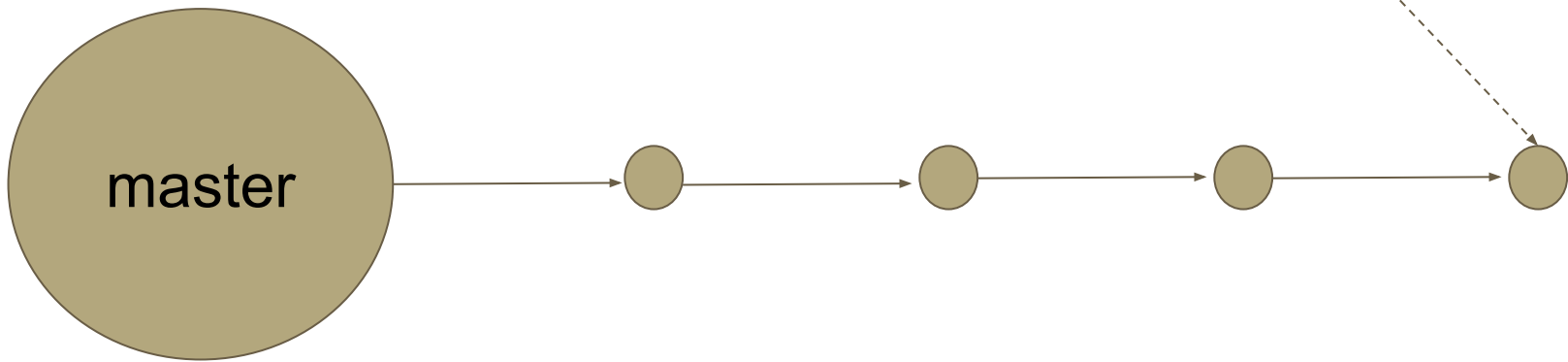
**KEEP
CALM
AND
GIT
STASH**

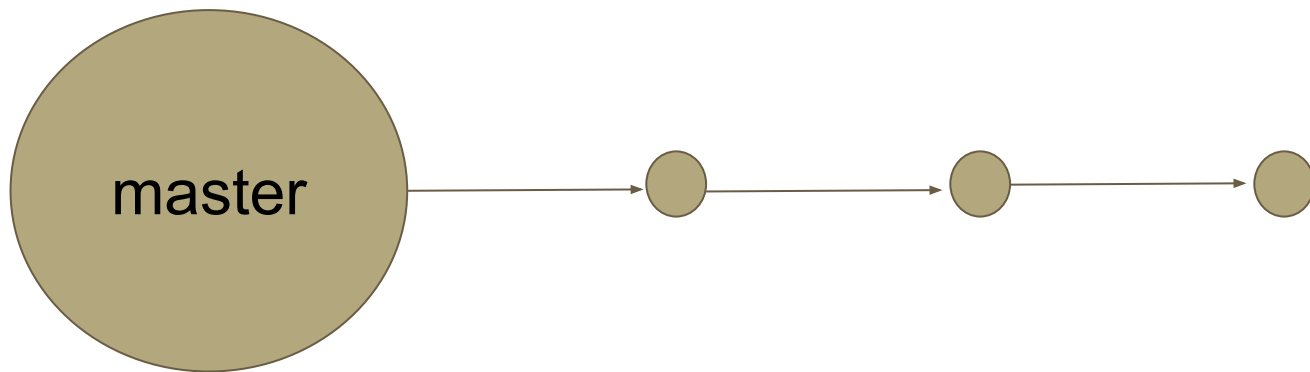
Demo

git reset



```
git reset --hard HEAD~
```





The diagram illustrates the Git workflow stages. It features three shapes on the left: a light orange oval at the top labeled 'Committed', an orange six-pointed star in the middle labeled 'Staged', and a dark orange multi-pointed star at the bottom labeled 'Changes'. To the right of these shapes are two curved arrows pointing upwards. The top arrow, between 'Staged' and 'Committed', is labeled 'git commit' in a large bracket. The bottom arrow, between 'Changes' and 'Staged', is labeled 'git add' in a large bracket. The entire diagram is set against a white background with a teal footer bar.

Committed

Staged

Changes

git commit

git add

Committed

Staged

Changes

`git reset --soft HEAD~`

`git reset`

`git checkout FILE_NAME`

Scenario 07

- Do some changes and commit them
- Discard that last commit
- Commit some changes again
- Take them back to staged state
- Take them to changes state
- Discard those changed files

```
git add .
```

```
git commit -m "commit message"
```

```
git reset --hard HEAD~
```

```
git add .
```

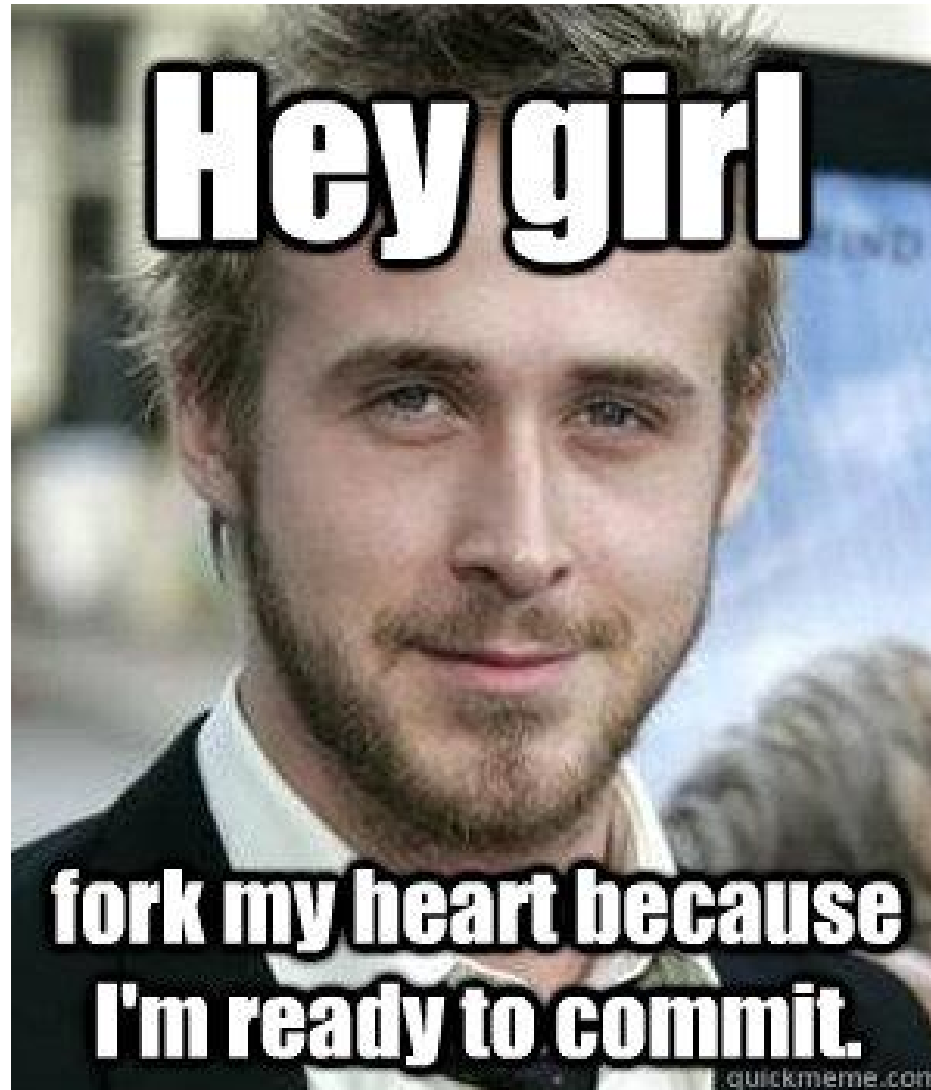
```
git commit -m "second commit"
```

```
git reset --soft HEAD~
```

```
git reset
```

```
git checkout FILE_NAMES
```

fork repository





“Stay hungry, stay foolish”

Steve Jobs



dilumn

dilumn_





KEEP

CALM

PRESENTATION IS OVER

ANY

QUESTIONS?



**THANK YOU FOR
LISTENING**

TO MY PRESENTATION!!