Git 簡介

Vincent Liu　2014/12/30

Agenda

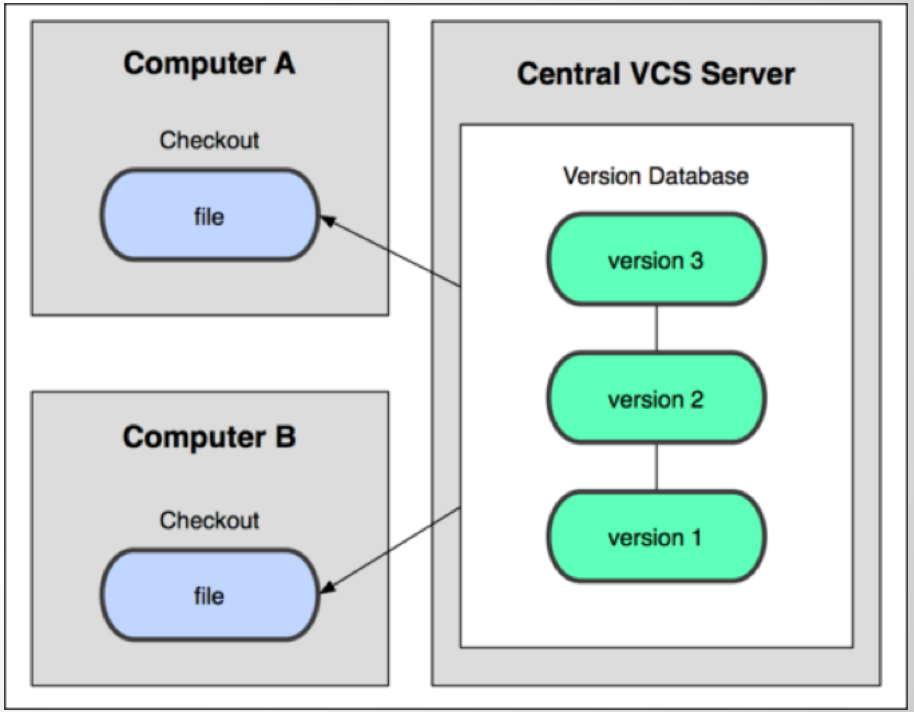
- 版本控制基本概念
- 版本控制系統類型:集中式與分散式
- Git Basics
- Git 常用指令
- Git branch
- 使用 branch 開發

為何需要版本控制系統

- 文件有不同版本需保存。
  - (可以用不同檔名或目錄名來保存)
- 文件需要分享。
  - (可以用網路分享)
- 文件存取要有權限。
  - (可以用系統權限設定)
- 需要知道誰修改了哪份文件。
- 文件需要共同編修。

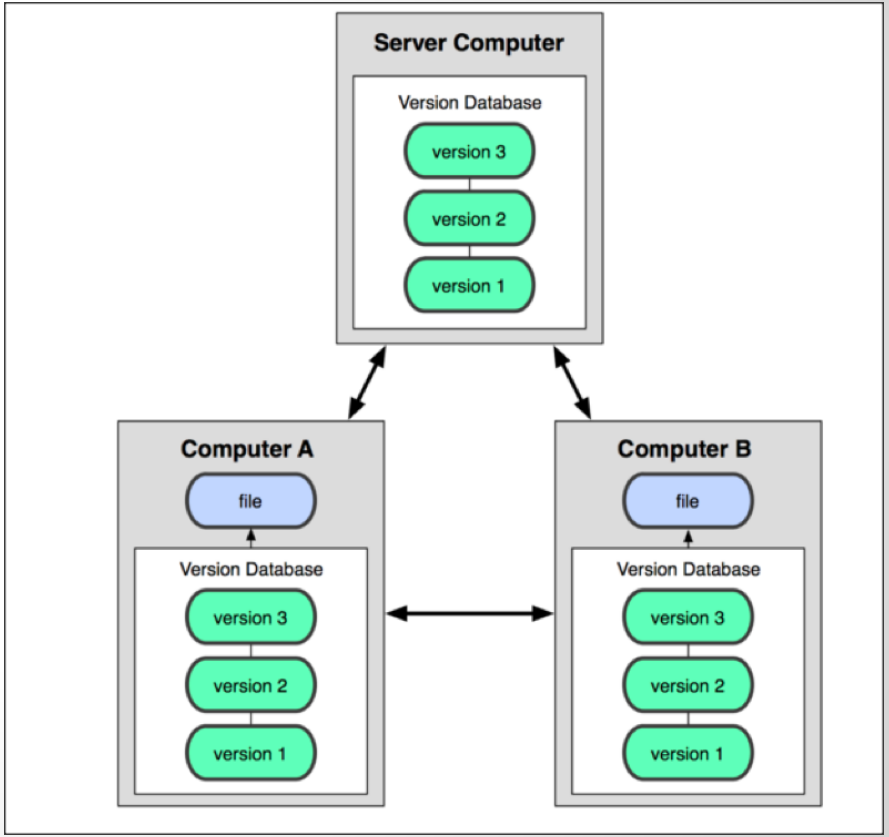
集中式版本管理系統

- 伺服器端才擁有完整資料。
- 幾乎所有動作都在伺服器端完成。
- 伺服器資料毀損, 所有歷程會消失。
- 常見的 SVN 及 CVS 是代表的系統。

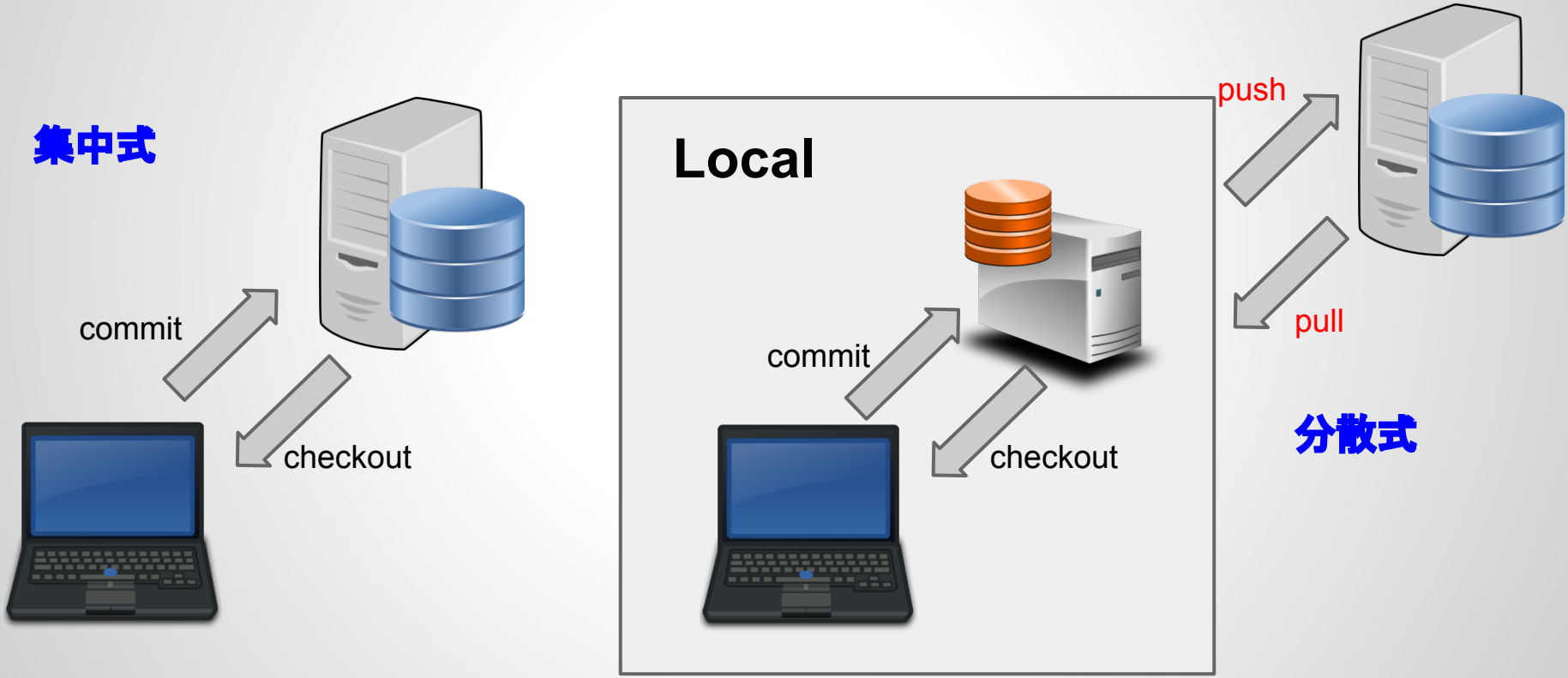


分散式版本管理系統

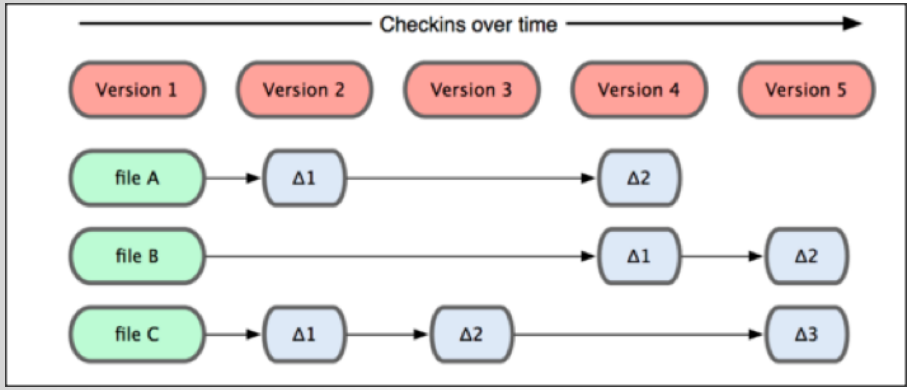
- 每個 Client 都有完整的資料。
- 幾乎所有的工作都可以在本地端完成。
- 適合多人異地平行開發。
- GIT 為代表系統。



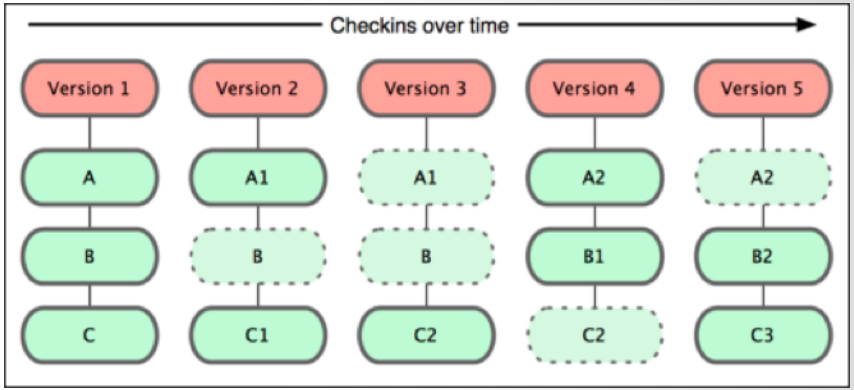
集中式與分散式圖示比較



GIT Basics

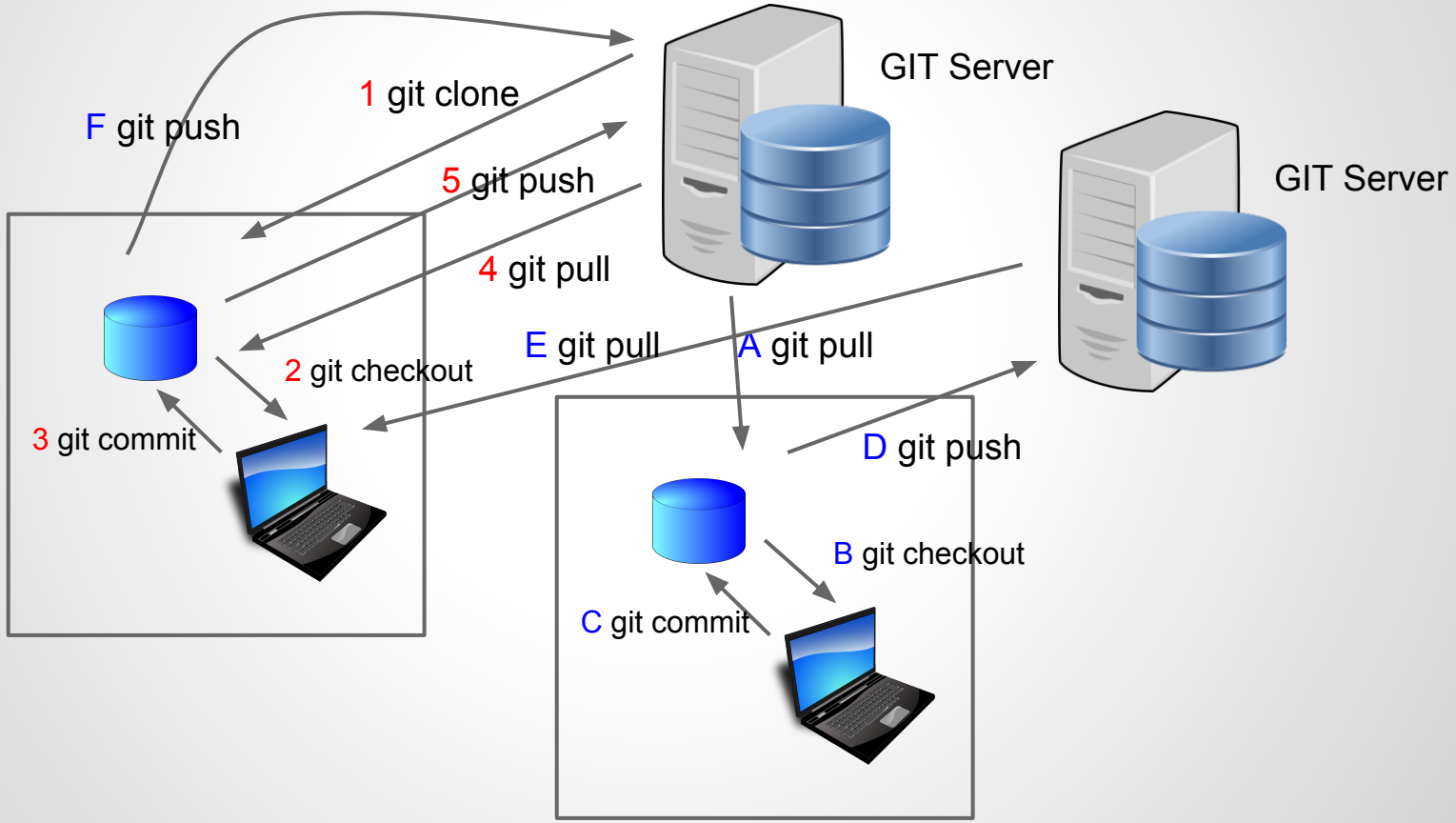


傳統版本控制系統記錄文件每次提交差異。

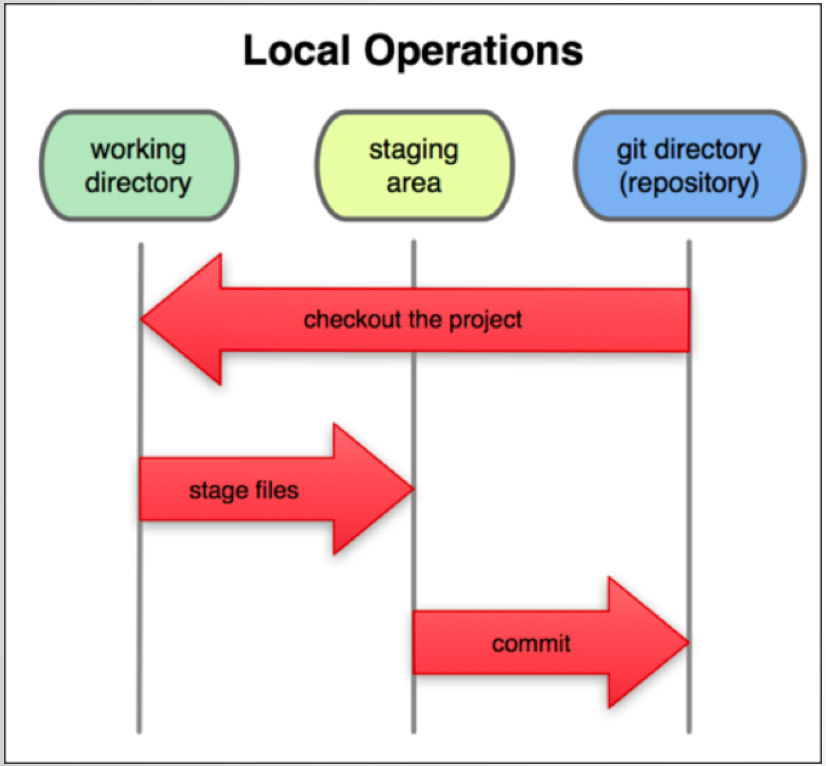


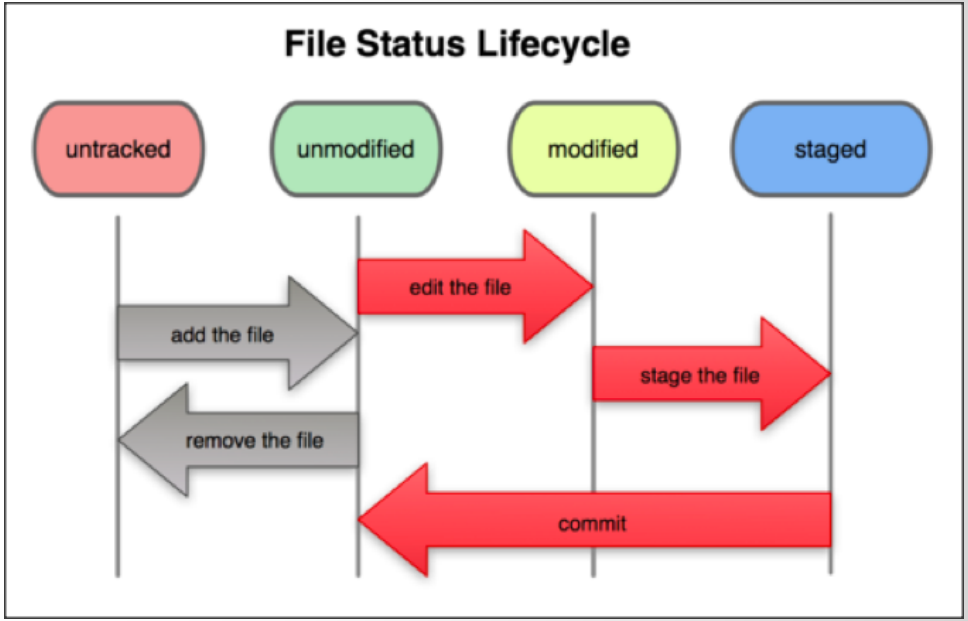
GIT 儲存每次提交文件的快照。

開發生命週期


GIT Server
GIT Server
1 git clone
F git push
5 git push
4 git pull
E git pull
A git pull
2 git checkout
3 git commit
D git push
B git checkout
C git commit

GIT Basics


Local Operations
working directory
staging area
git directory (repository)
checkout the project
stage files
commit


File Status Lifecycle
untracked
unmodified
modified
staged
edit the file
add the file
stage the file
remove the file
commit

GIT Stage

```
wkliu@ubuntu:~/testrepo$ git status
On branch master

Initial commit

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

        test02.js

nothing added to commit but untracked files present (use "git add" to track)
wkliu@ubuntu:~/testrepo$ git add test02.js
wkliu@ubuntu:~/testrepo$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   test02.js
```

GIT 建立倉庫(Repository)

- git init　建立空的倉庫
- git status 檢視目前狀態

```
wkliu@ubuntu:~$ git init myrepo
Initialized empty Git repository in /home/wkliu/myrepo/.git/
wkliu@ubuntu:~$ cd myrepo
wkliu@ubuntu:~/myrepo$ git status
On branch master

Initial commit

nothing to commit (create/copy files and use "git add" to track)
```

GIT 複製已存在倉庫

- git clone

```
wkliu@ubuntu:~$ git clone https://github.com/wkliu/testrepo.git
Cloning into 'testrepo'...
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 6 (delta 0)
Unpacking objects: 100% (6/6), done.
Checking connectivity... done.
```

GIT 常用指令

- git status 查看目前狀態

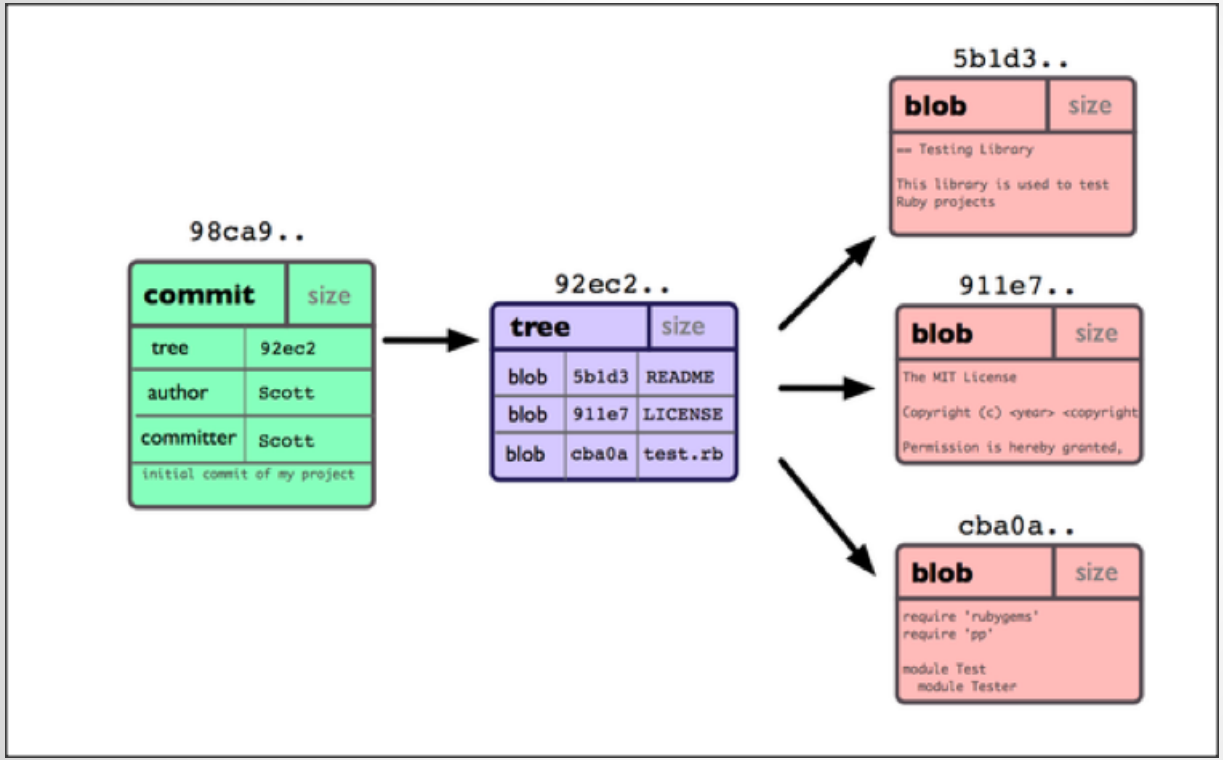
```
wkliu@ubuntu:~/CAG2$ git status
On branch fixbugs#6
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   apps/server/host/models/storage.js

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:   apps/server/host/models/eventDenormalizer.js

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

        apps/server/.DS_Store
        apps/server/._.DS_Store
        apps/server/host/.DS_Store
```

GIT 常用指令

- git log 查看已 committed 的歷史記錄。

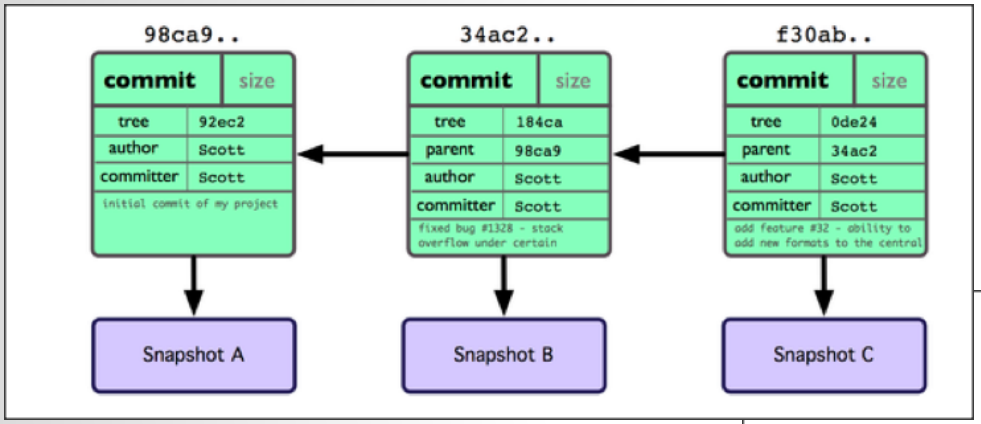
```
commit 5072ff6f287f047cd806e4a9907eb47404e0023f
Author: SYC <syc@syc-AOD260.(none)>
Date:   Tue Nov 18 16:20:31 2014 +0800

    新增錯誤訊息多語系Index

commit 048d29b5113bf17f22ae7c9817ffdbc212b721bb
Author: Timothy Huang <elsely@gmail.com>
Date:   Tue Nov 18 12:11:55 2014 +0800

    cloudsearch with environment CAGENV

    cloudsearch upload with cagenv and delete with cagenv finished.
    put the server-side formatcloudsearch datetime code from
    bower_components utility to server utility

commit cb484455aa7b825145bc441b60a8286ffe81bc12
Author: wkliu <wkliu228@gmail.com>
Date:   Tue Nov 18 04:21:08 2014 +0800

    Solve the issue which clicked the contact us link on Service page, the
```

GIT 分支(Branch)



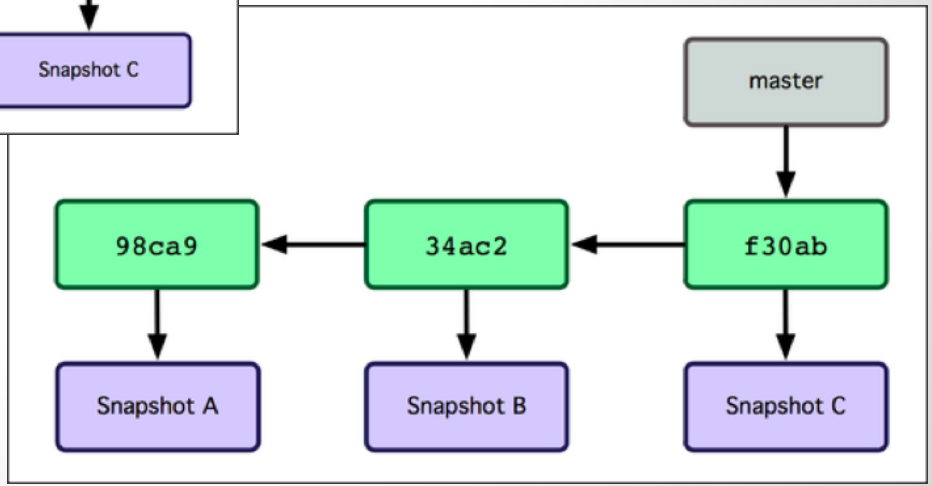
GIT commit 的資料結構示意圖

GIT Branch



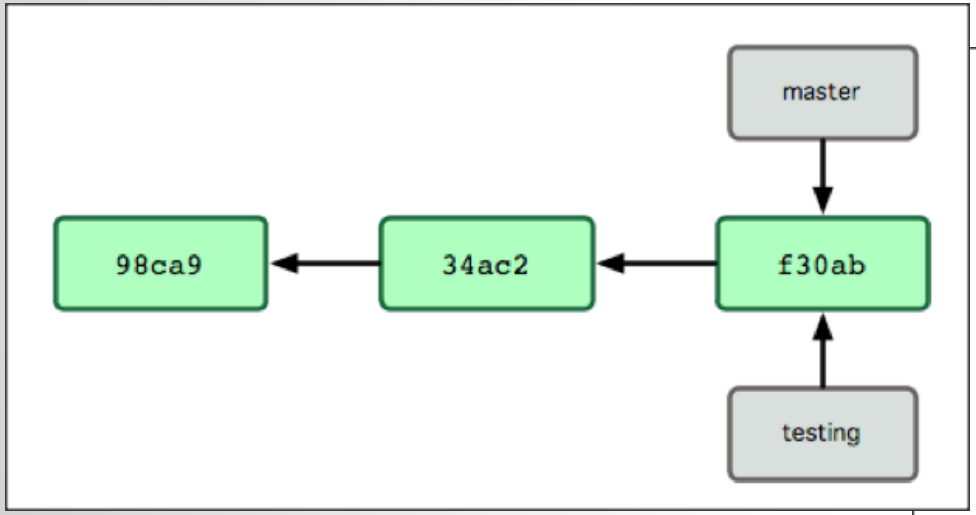
多次 Commit 後的圖示

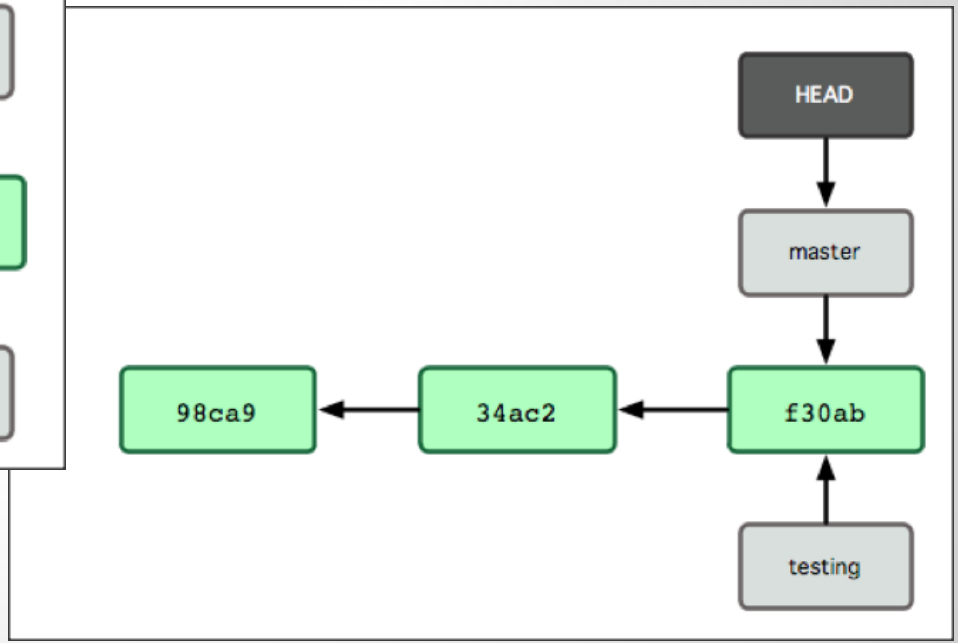
預設分支 master



GIT Branch

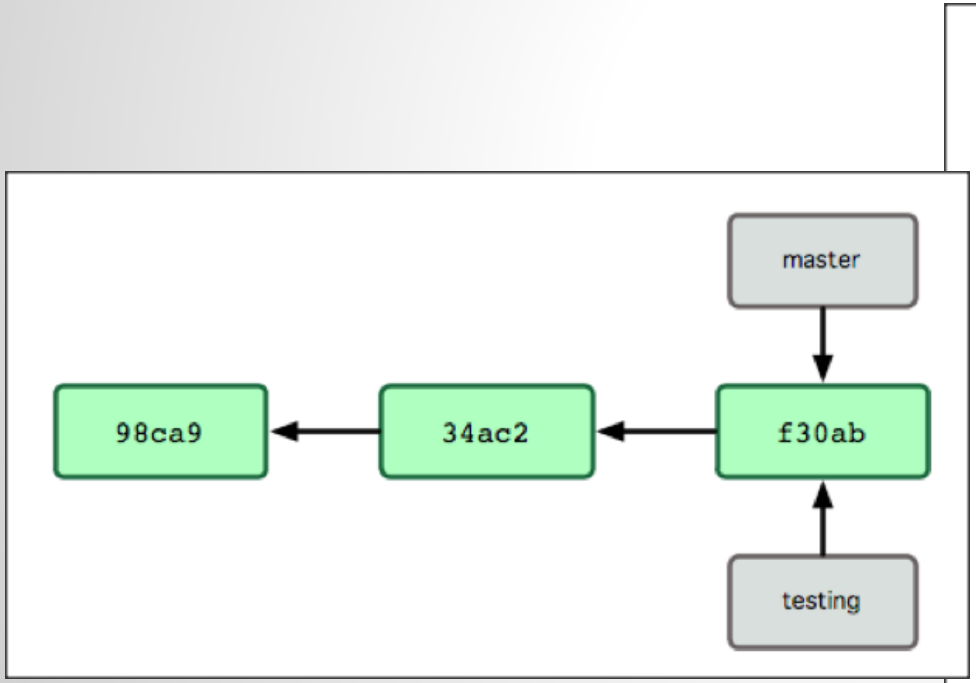
- git branch testing

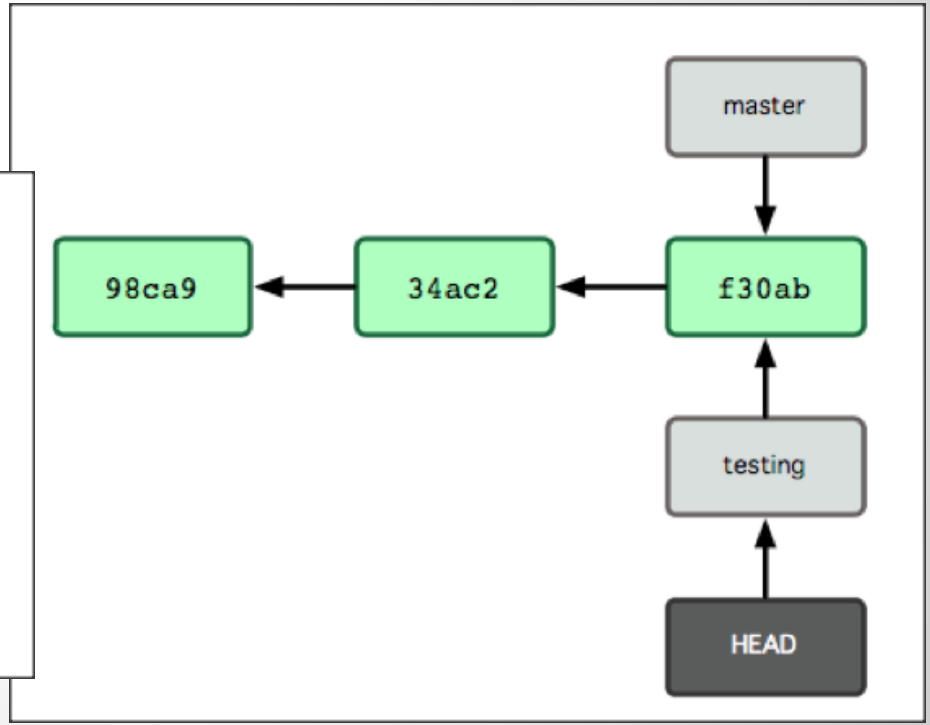




GIT Branch

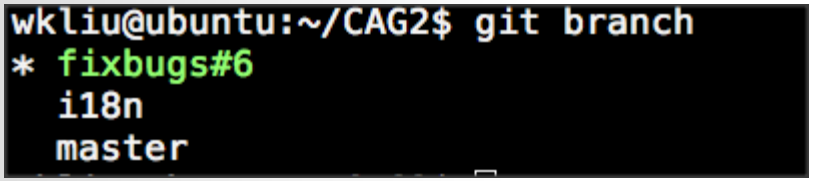
- git checkout -b testing
- git branch testing and git checkout testing





GIT 分支(Branch)

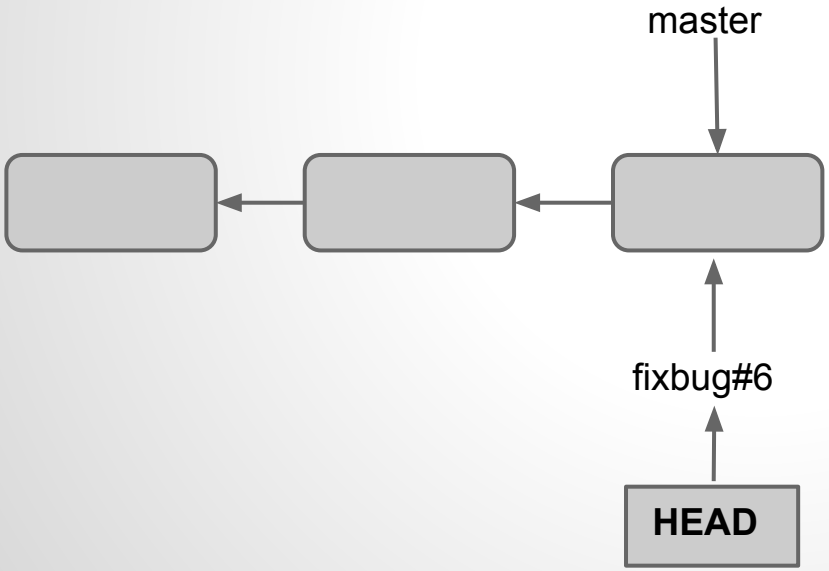
- git branch 查看目前所在分支與目前擁有分支。



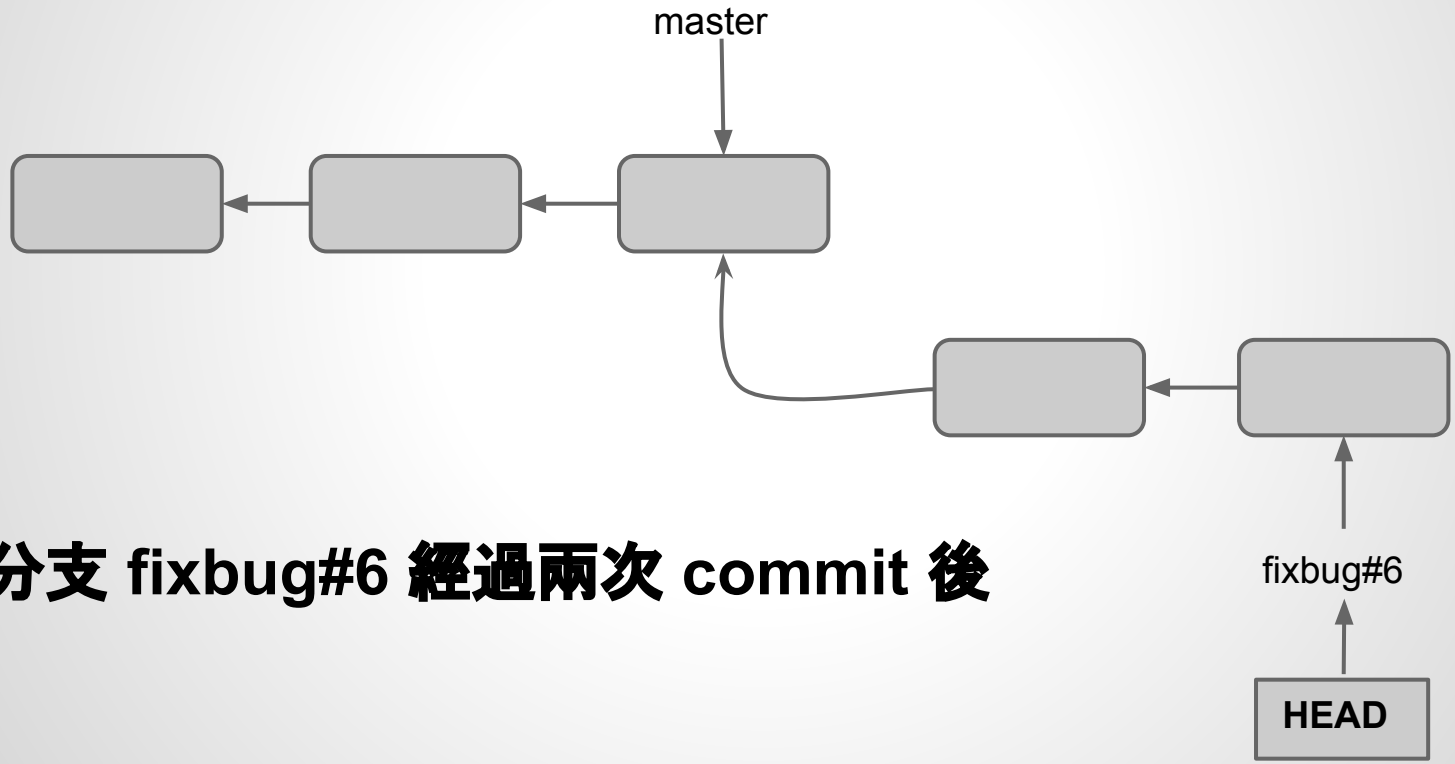
- git branch -d branch_name 刪除分支。
  - 不能刪除目前所在分支。
  - 不能刪除未合併的分支。(可改用 -D 強制刪除)

建議開發時使用 branch

- git checkout -b fixbug#6

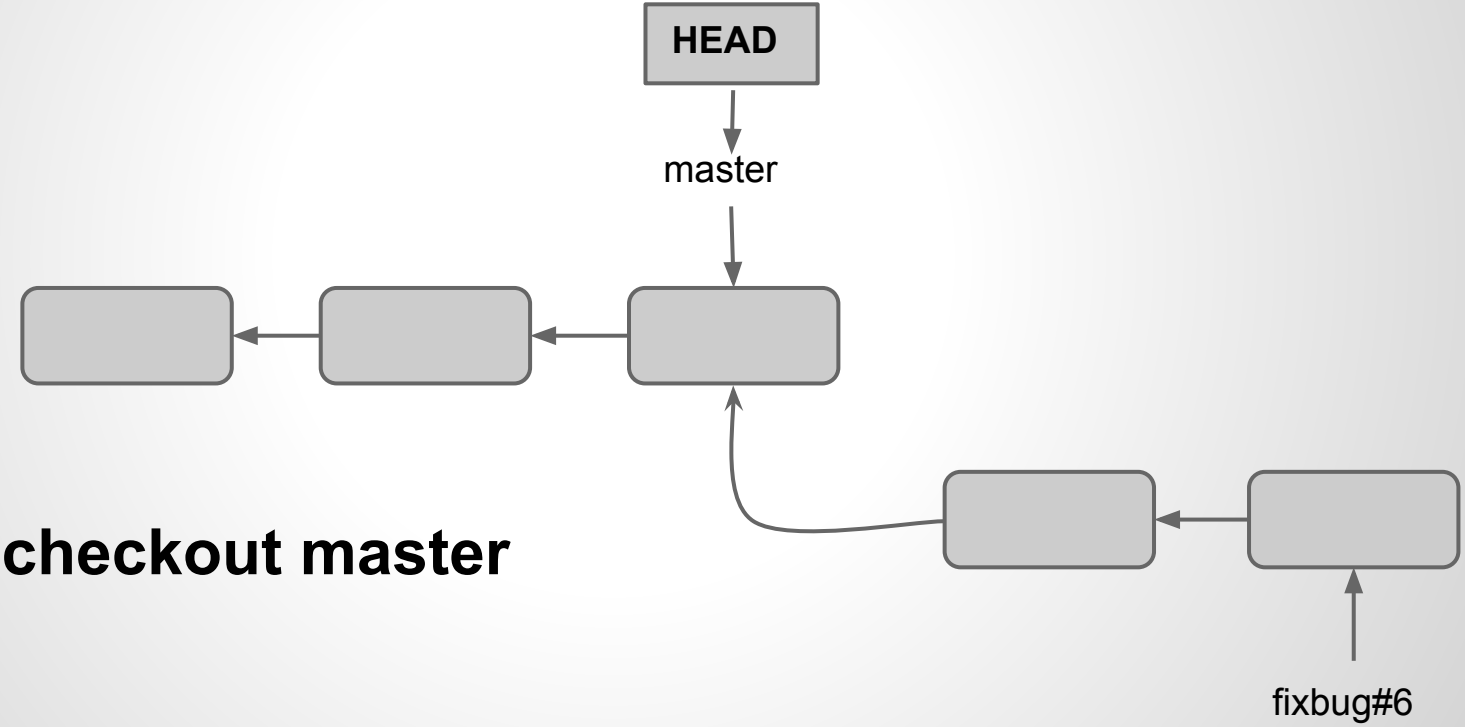


使用 branch 開發



在分支 fixbug#6 經過兩次 commit 後

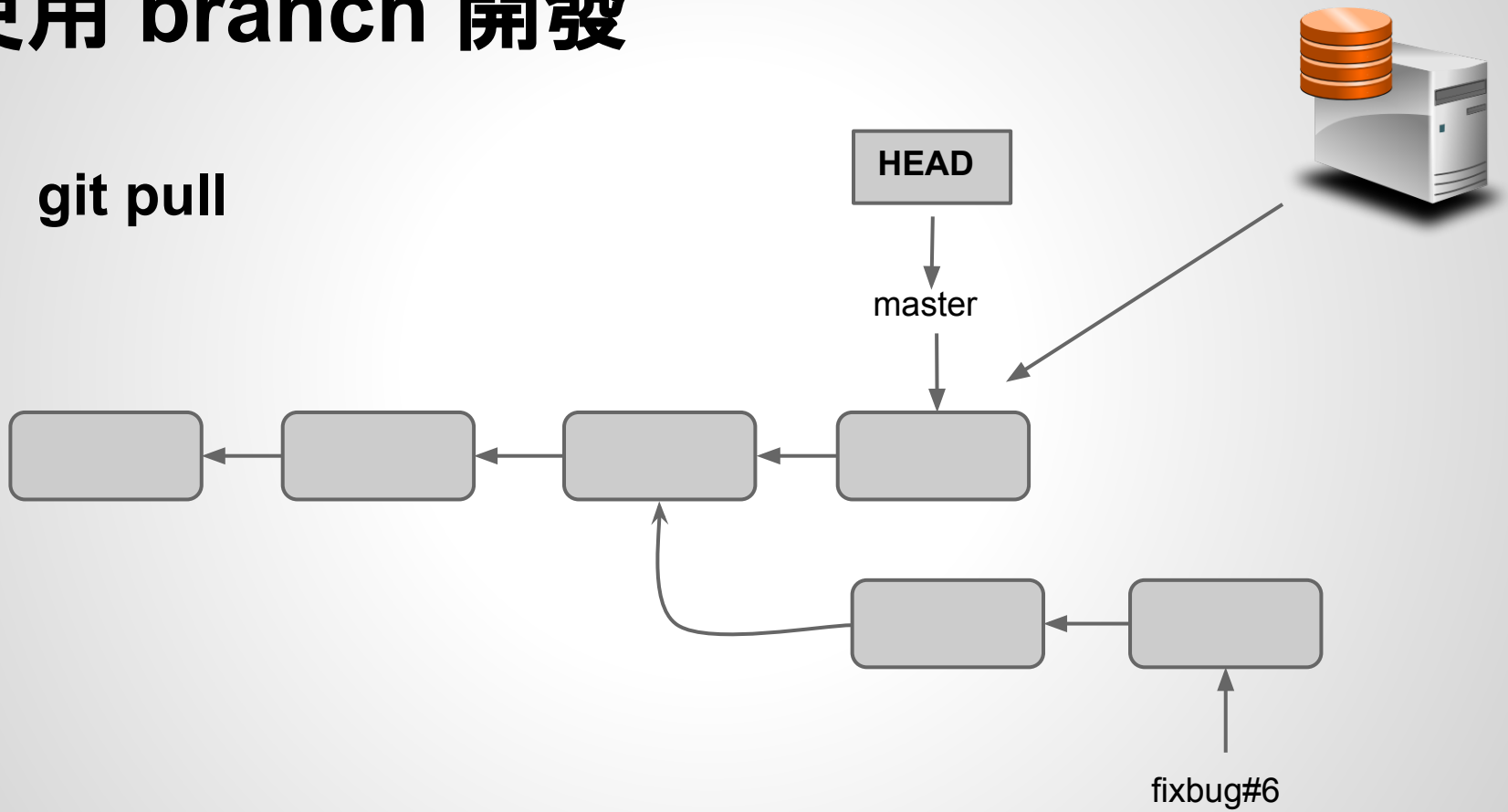
使用 branch 開發



- **git checkout master**

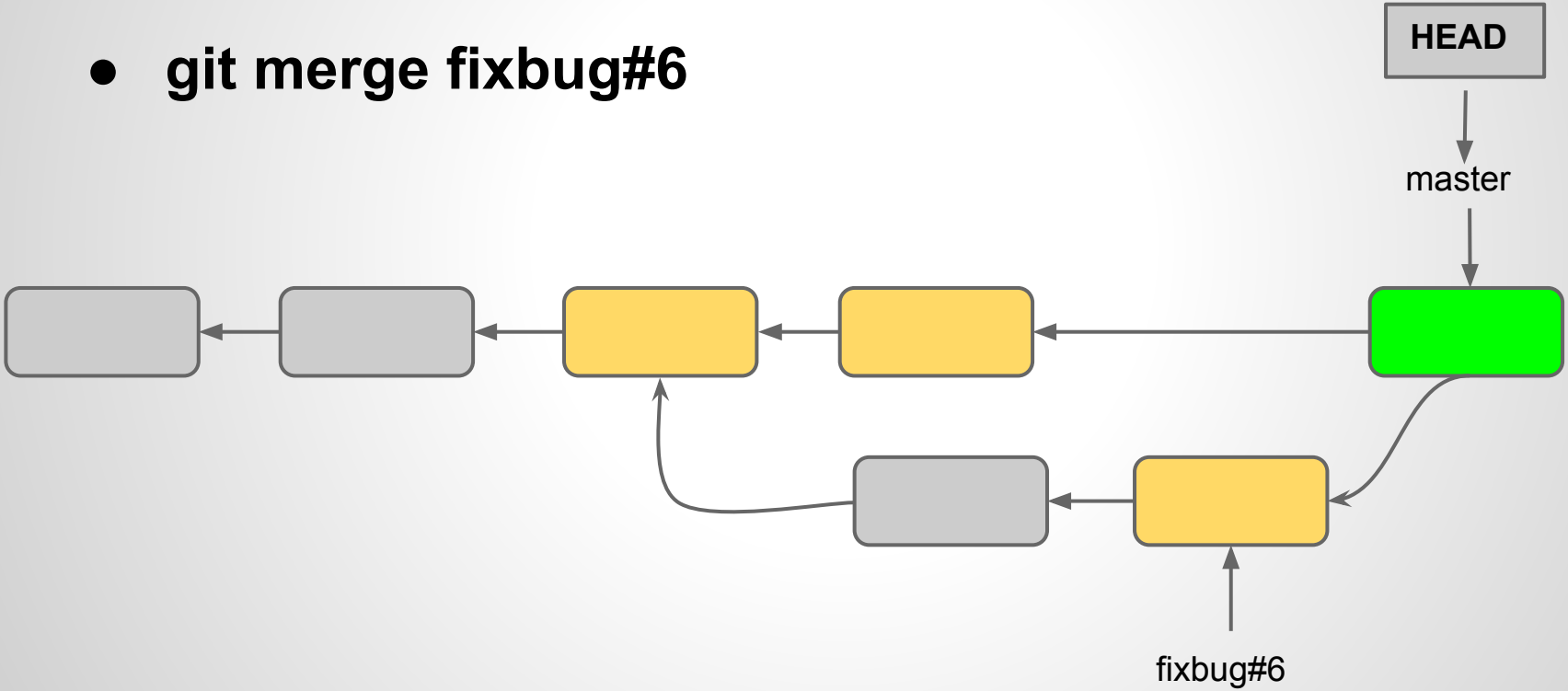
使用 branch 開發

- git pull



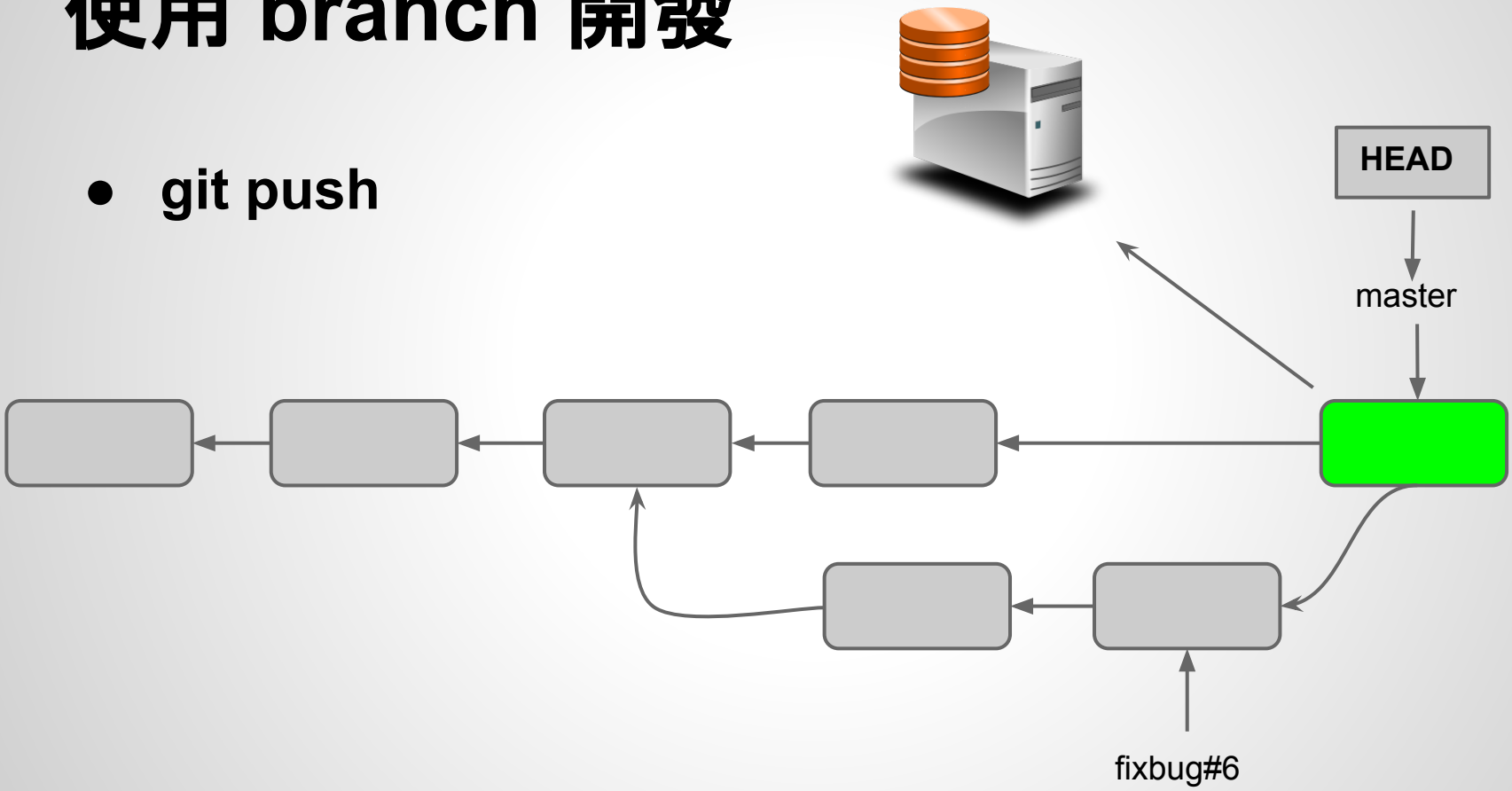
使用 branch 開發

- git merge fixbug#6



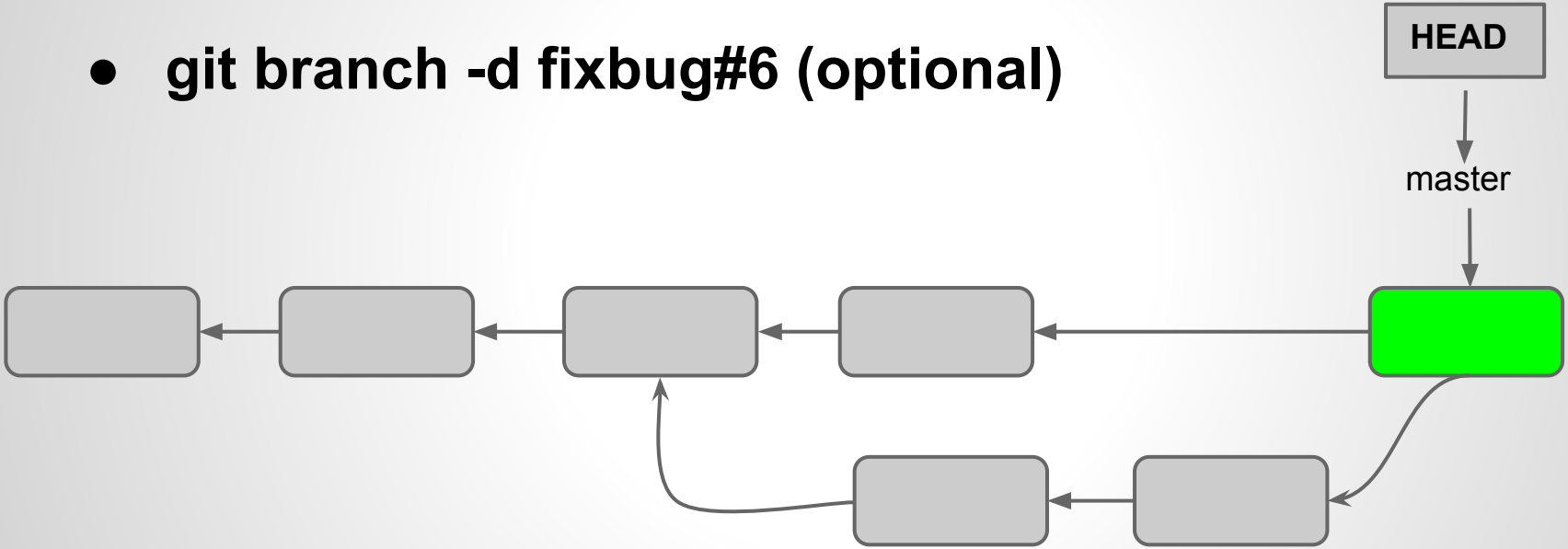
使用 branch 開發

- git push

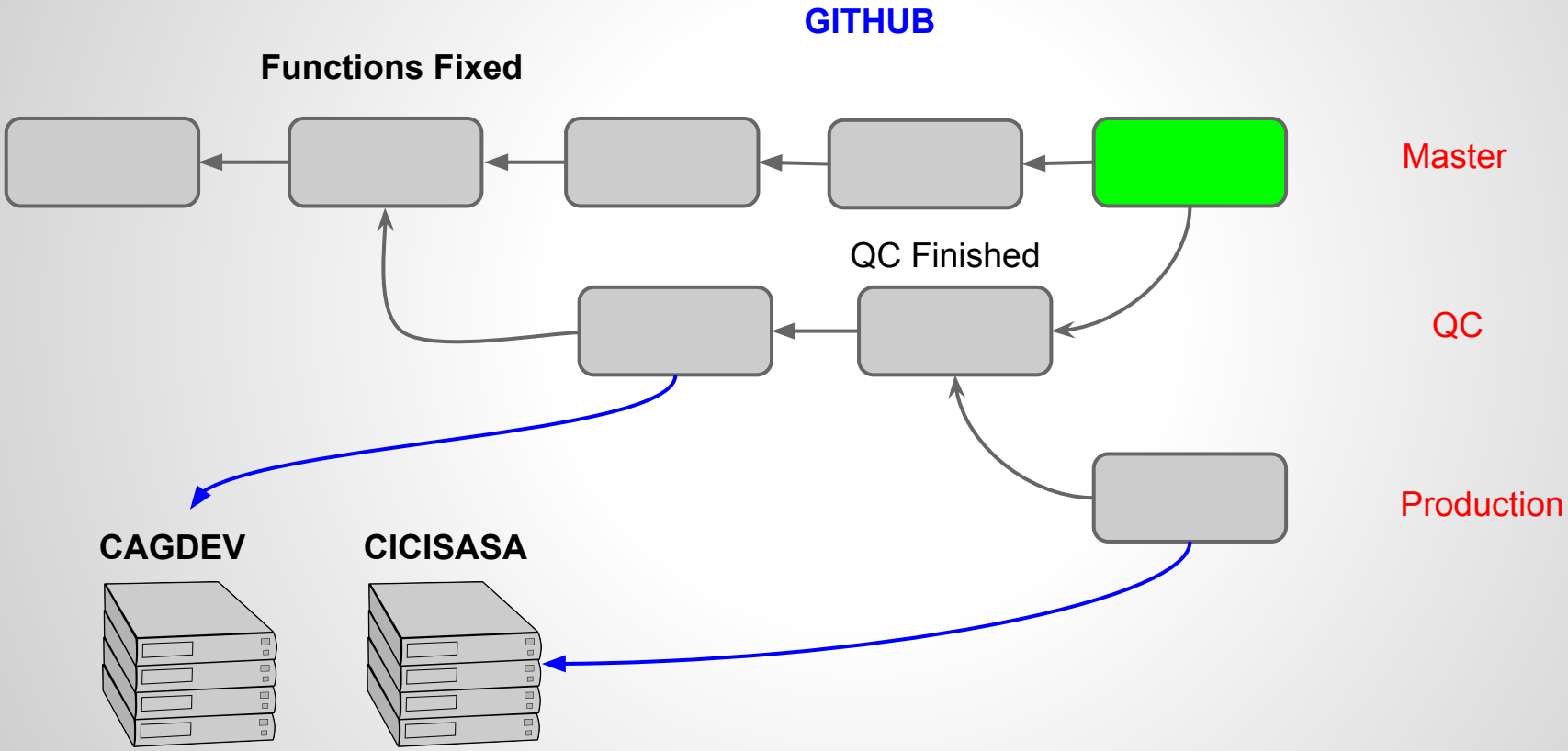


使用 branch 開發

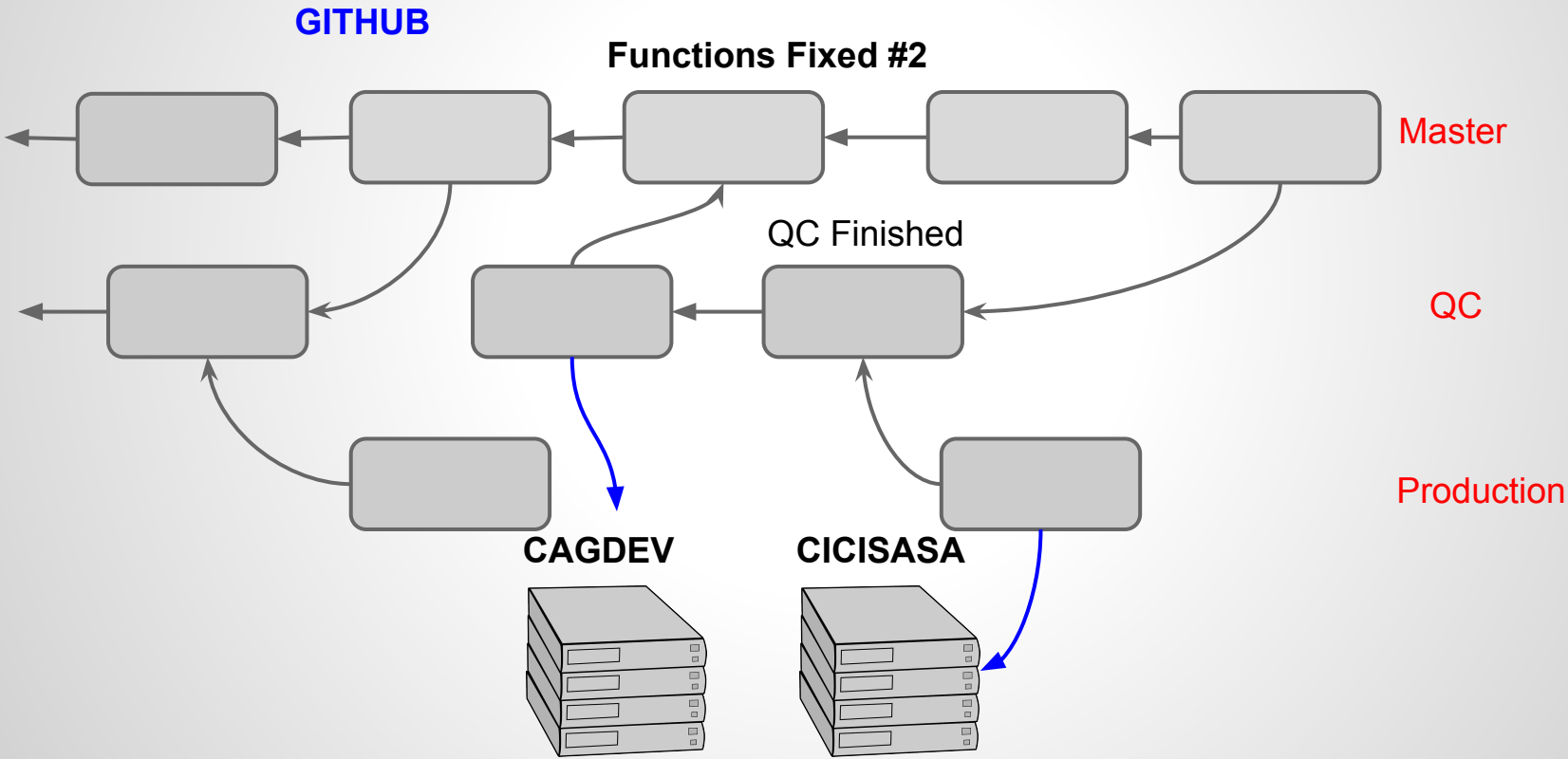
- git branch -d fixbug#6 (optional)



建議


GITHUB
Functions Fixed
Master
QC Finished
QC
Production
CAGDEV
CICISASA

建議


GITHUB
Functions Fixed #2
Master
QC Finished
QC
Production
CAGDEV
CICISASA