Computer Programming

section week 11

Agenda

1. review(回顧上一週所做的)
2. github 功能介紹
3. 多人一起協同
4. git pull(conflict 處理)

What is the problem?

多人一起协同, how to do it ?

GitHub

Git

User

Origin image from here

Origin image from here

User by Luis Prado from The Noun Project

review

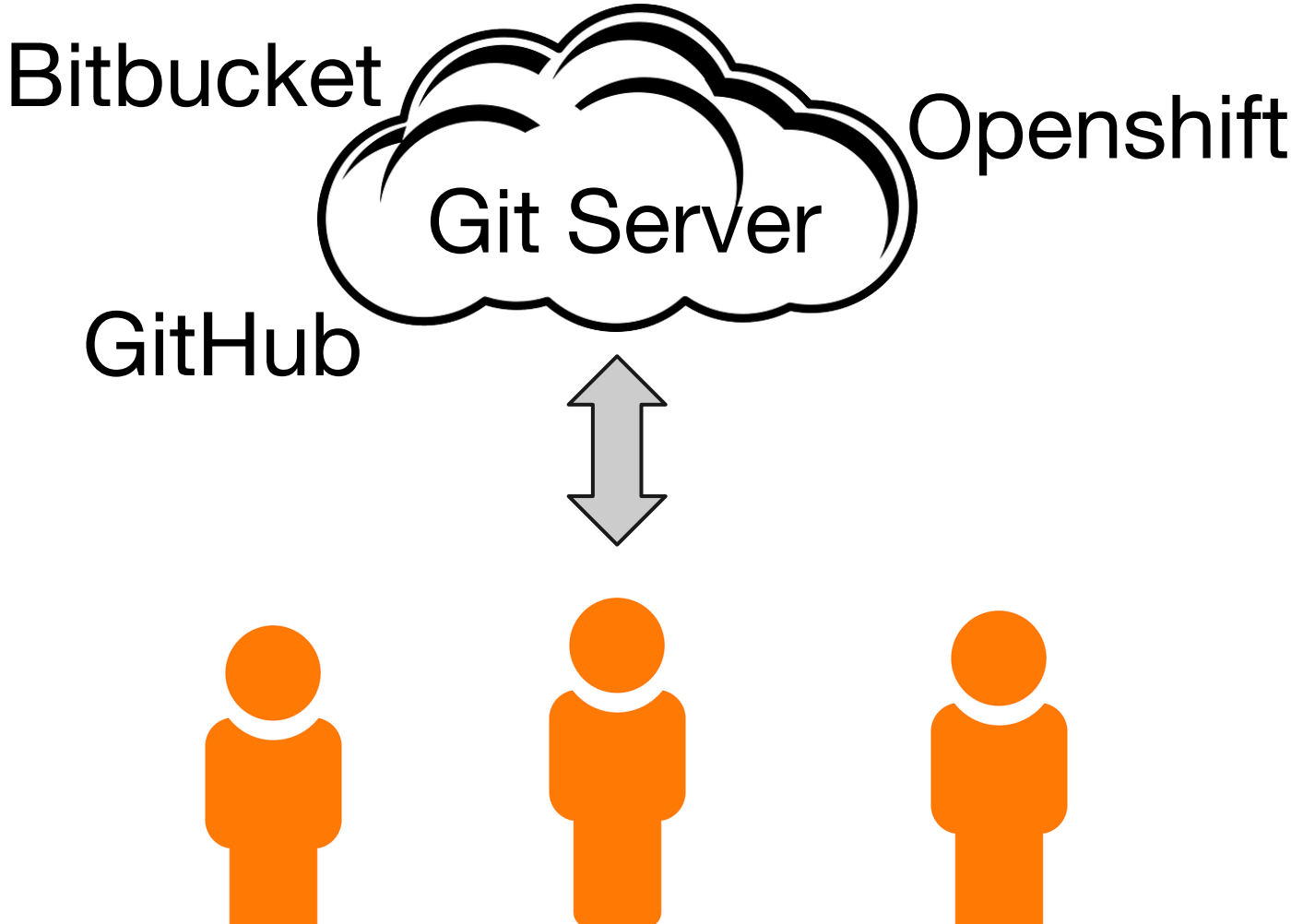
1. 為什麼要版次管理
2. 什麼是 Git , GitHub
3. Git 基本功能介紹
4. Git 與 GitHub 互動

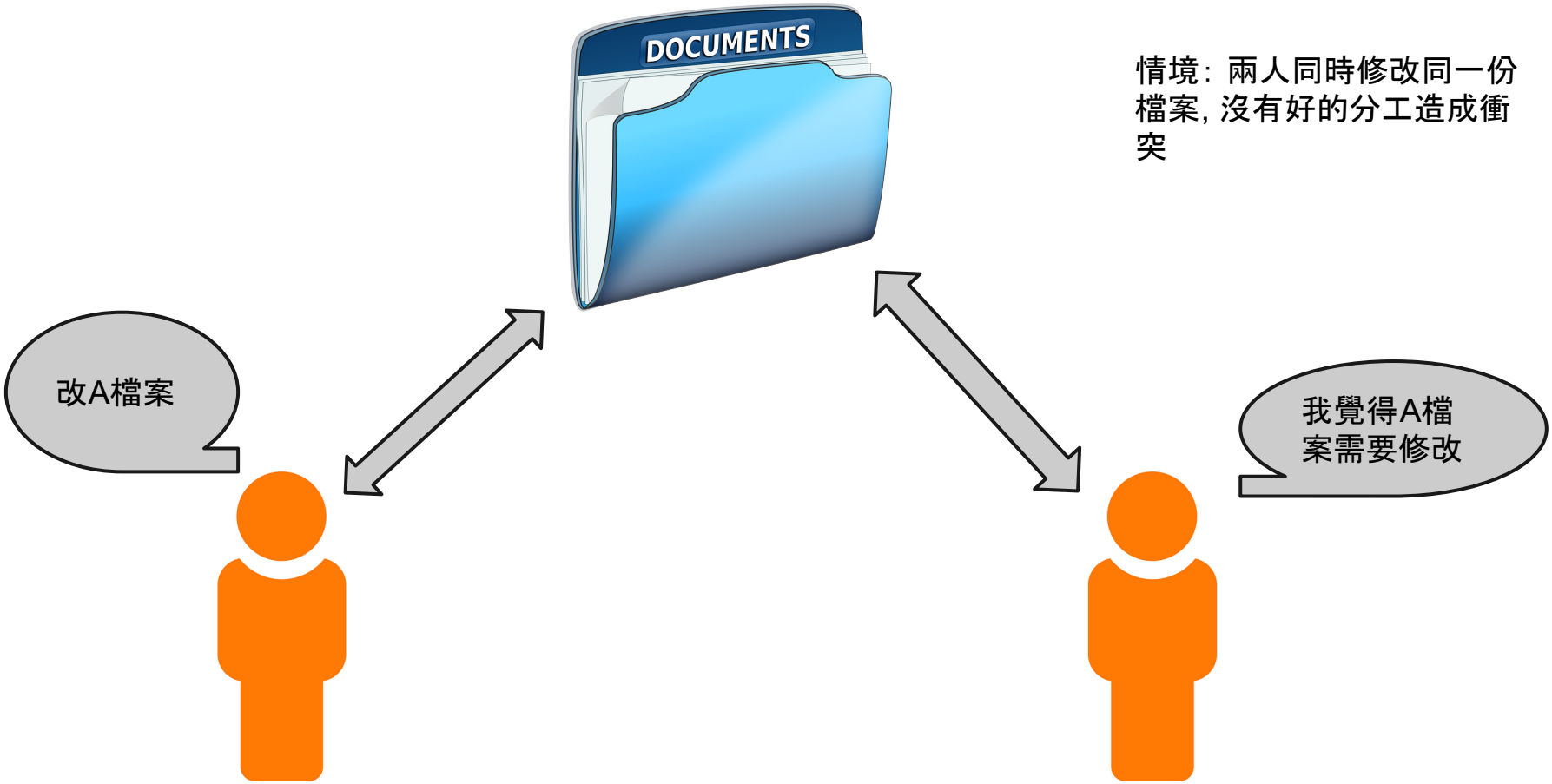
GitHub 功能介紹

- issues(問題回報）
- collaborators（協同者）
- deploy keys - use rsa encryption

多人一起協同

實做(製造情境 conflict 並處理)

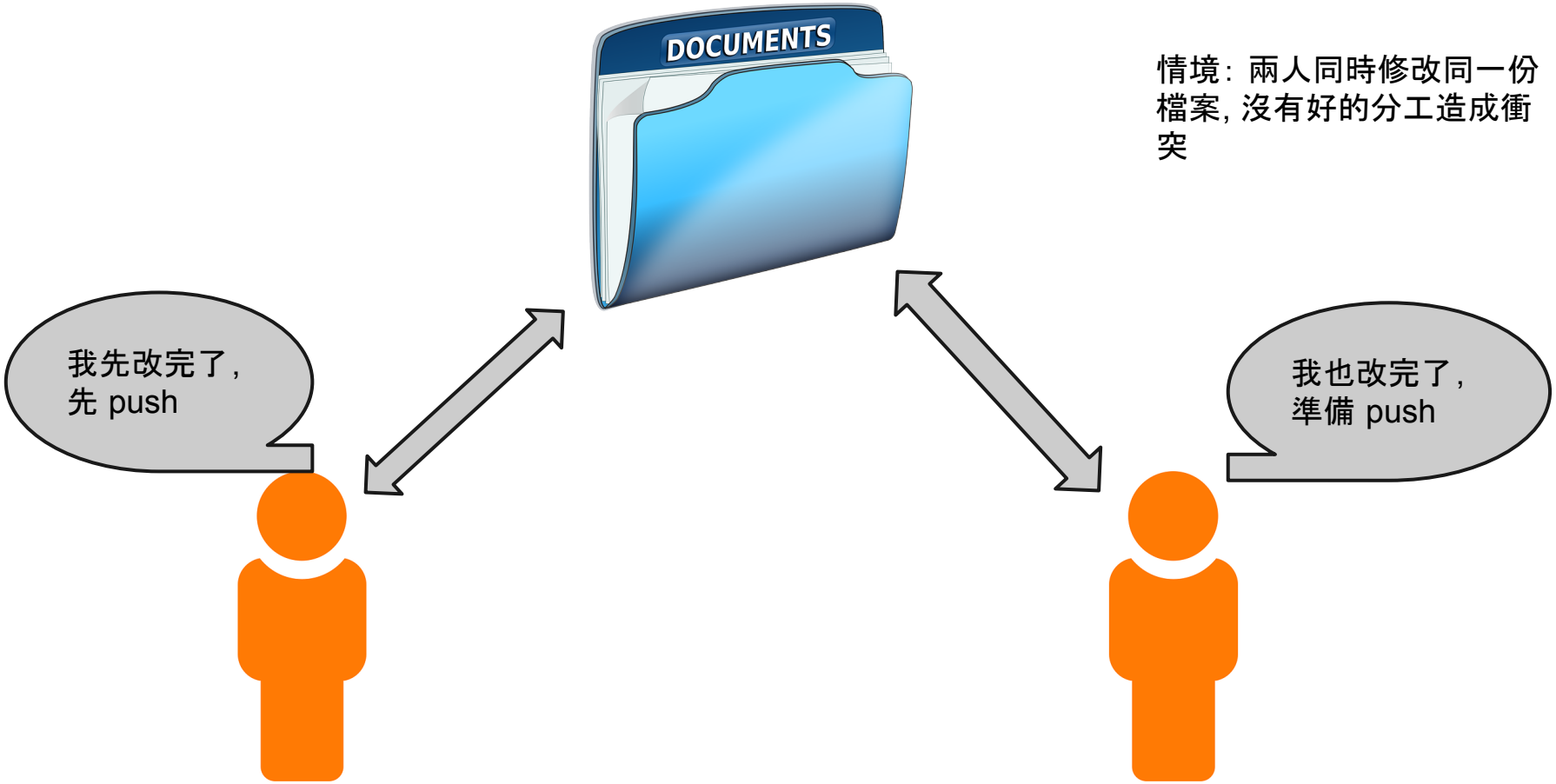

Bitbucket
Git Server
Openshift
GitHub

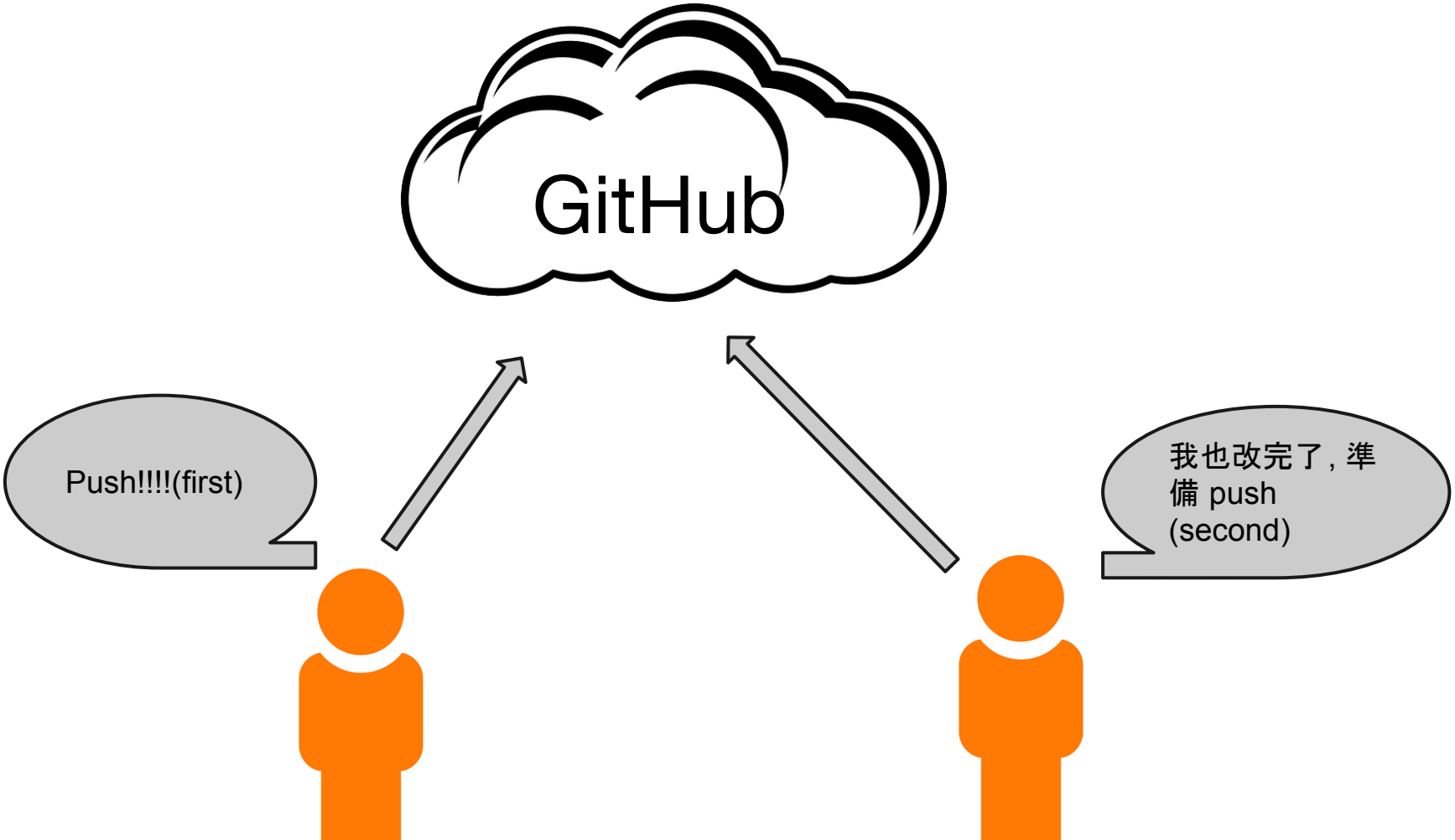


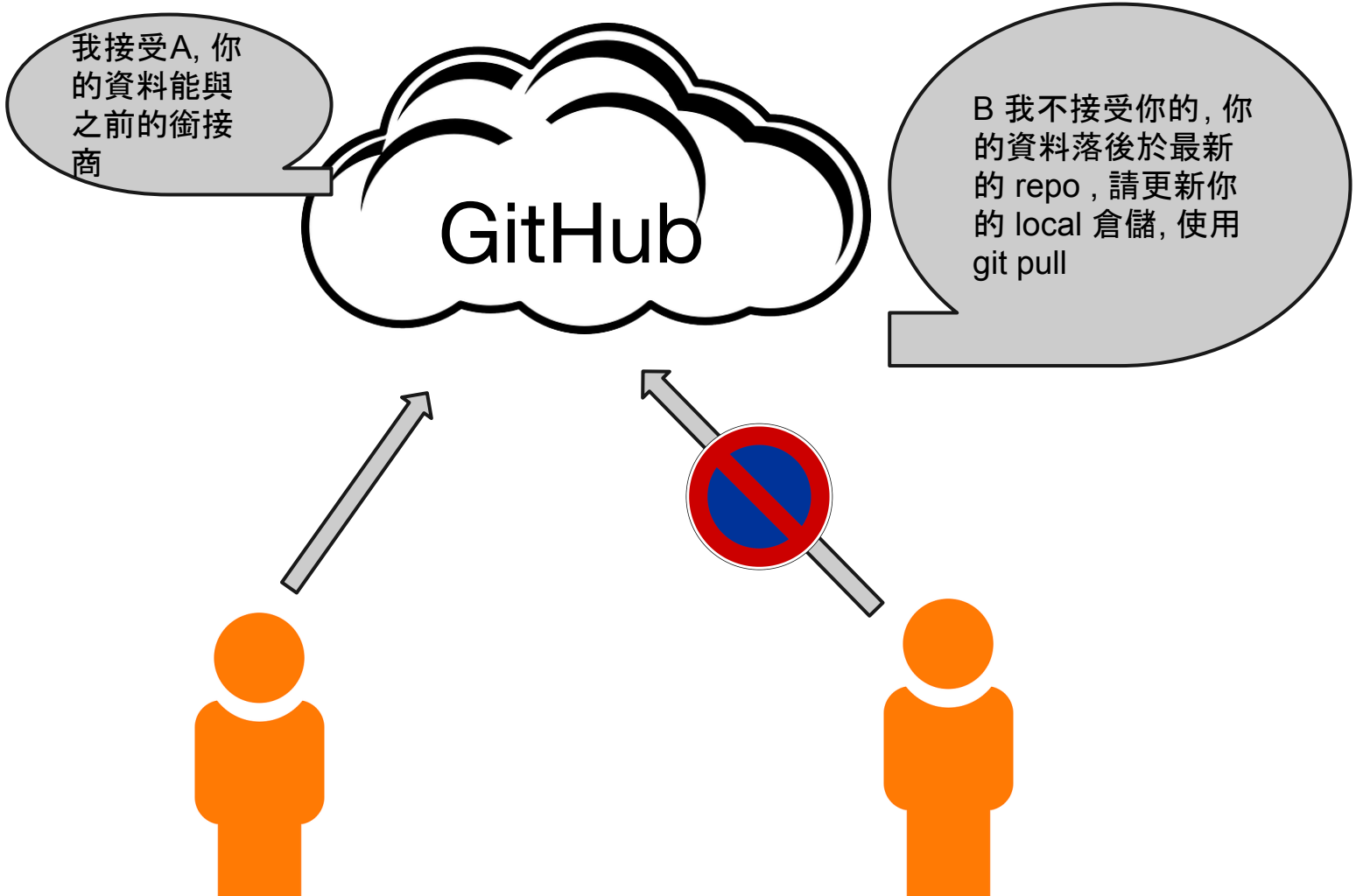
情境：兩人同時修改同一份檔案，沒有好的分工造成衝突

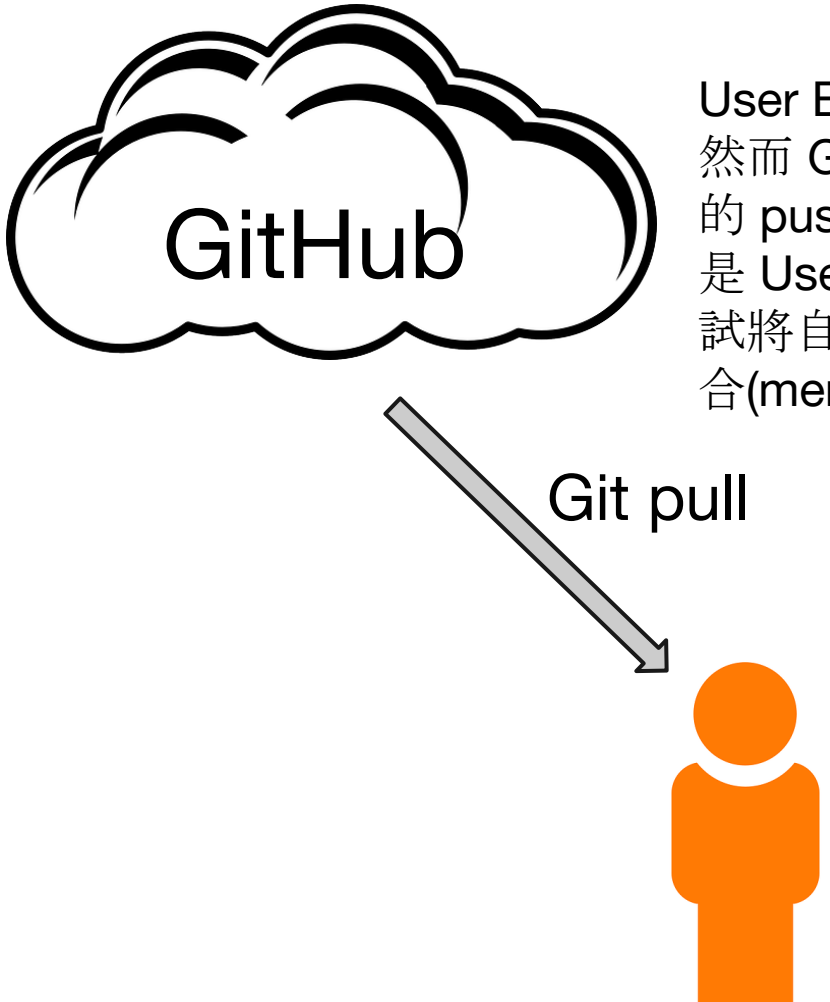
User A	User B	Origin
Today is good day. have fun!	Today is good day. have a good day!	Today is good day.

情境：兩人同時修改同一份檔案，沒有好的分工造成衝突




GitHub
Push!!!!(first)
我也改完了，準備 push (second)


我接受A, 你的資料能與之前的銜接商
GitHub
B 我不接受你的, 你的資料落後於最新的 repo , 請更新你的 local 倉儲, 使用 git pull



User B 的 A 檔案改過了, 然而 GitHub 接受 User A 的 push, 所以現在 A 檔案是 User A 的版本, Git 會嘗試將自己的與遠端的做融合(merge)

當 Git 不能自動融合時

Git 抱怨產生衝突, 請使用者決定

解決衝突

Today is good day. <<<<<<< 版本號碼 <<<< have fun! <<<<<<< local <<<<<< have a good day! <<<<<<< end <<<<<<<	Today is good day. have fun! have a good day!

再次 push to github

git commit and push

GitHub say….

總結今天所做的