



Software Park Thailand
/>Code Camp>
CODE CAMP#5

Git



Coach Bhornumnard Wanarisun

Outline

- Git ?
- Git host มีใครบ้าง
- Git register and create repository
- Git state
- Git command
- Git flow
- Git flow with team



Learning Outcome

เพื่อให้นักเรียนมีความรู้ความเข้าใจในการใช้ Git และสามารถใช้งาน Git ได้อย่างมีประสิทธิภาพ เข้าใจหลักการทำงานของ Git และสามารถใช้ทำงานร่วมกับผู้อื่นได้



What is Git?

Git คือ Version Control แบบ Distributed ตัวหนึ่ง เป็นระบบที่ใช้จัดเก็บและควบคุมการเปลี่ยนแปลงที่เกิดขึ้นกับไฟล์ชนิดใดก็ได้ ไม่ว่าจะเป็น Text File หรือ Binary File (จากนี้จะเรียก Text File หรือ Binary File รวมกันว่า Source Code)



การจัดการ source code ในอีต

ในอีตการจัดการงานต่างๆนั้นยังไม่มีระบบการจัดการที่ไฟล์ที่มีประสิทธิภาพมากนัก ข้อเสียต่างๆที่สามารถเห็นได้ชัดจากการจัดการไฟล์ คือ มีความซับซ้อน มีความซ้ำซ้อน การเก็บข้อมูลใน local workstation

ยุคระบบจัดการซอฟต์แวร์ เวอร์ชัน หรือ แบบดั้งเดิม



presentatim.ppt



presentation-v.2.ppt



presentatim-bak.ppt

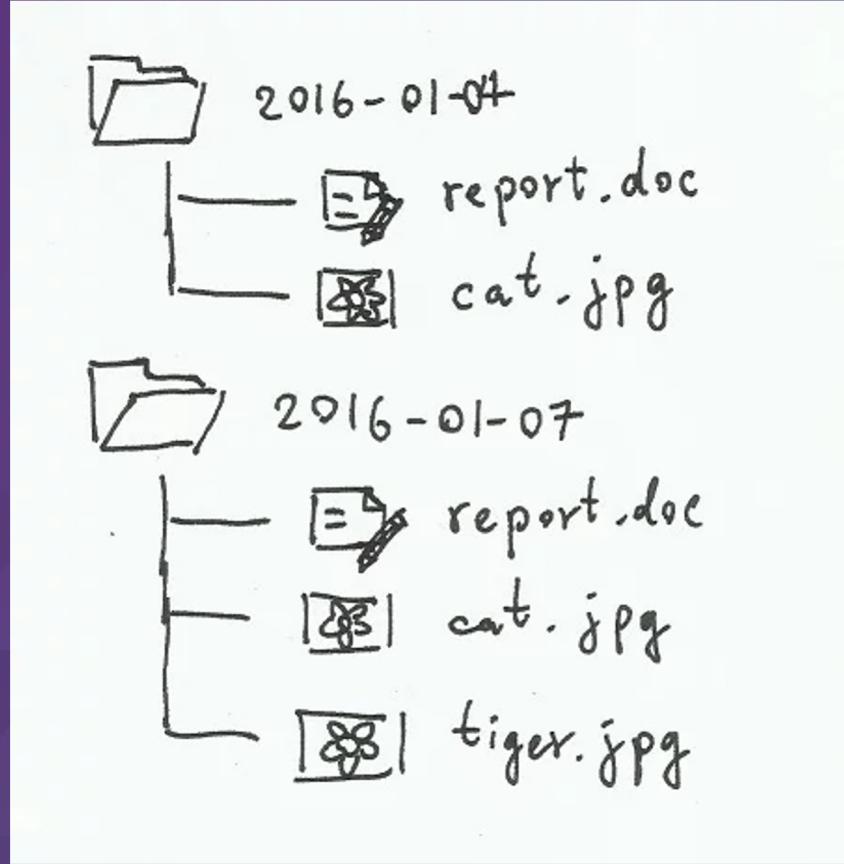


presentation-final.ppt

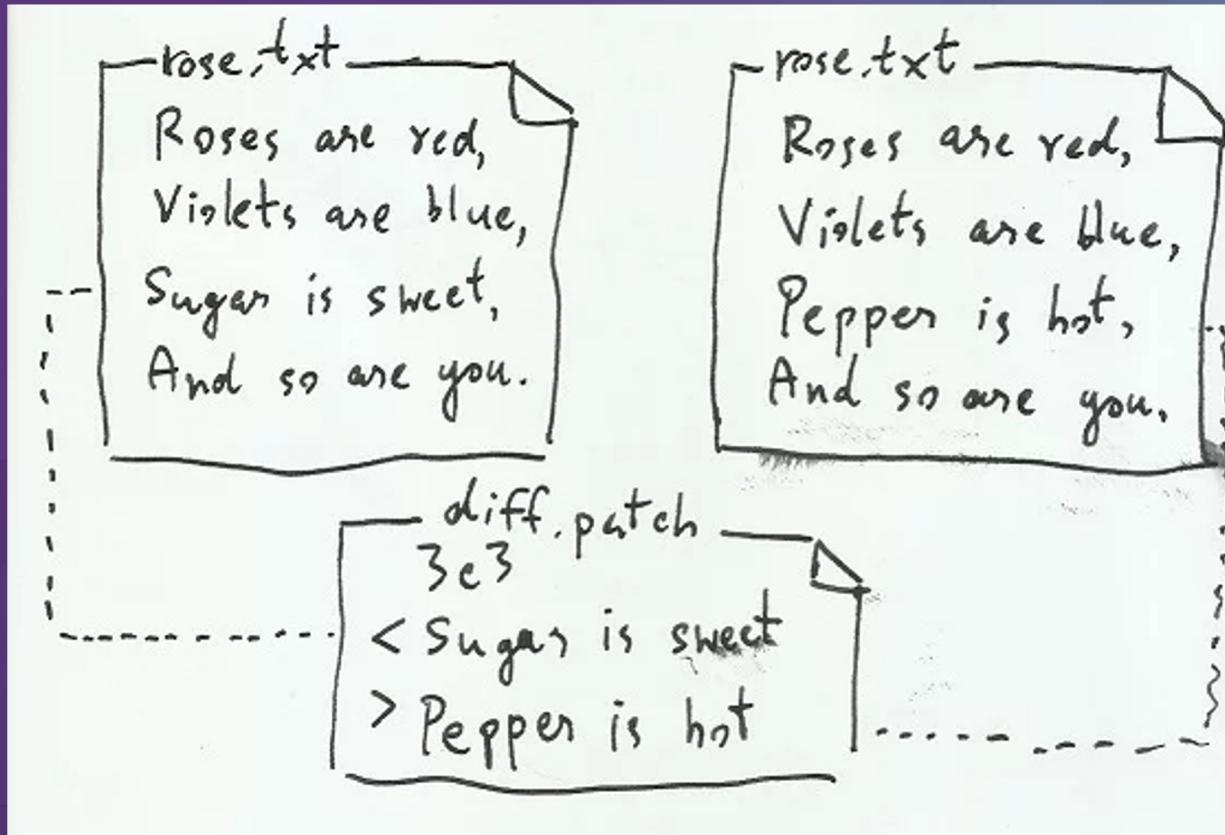


presentatim-latest.ppt

ยุคระบบจัดการซอร์สเดียว folder ไฟล์วันที่

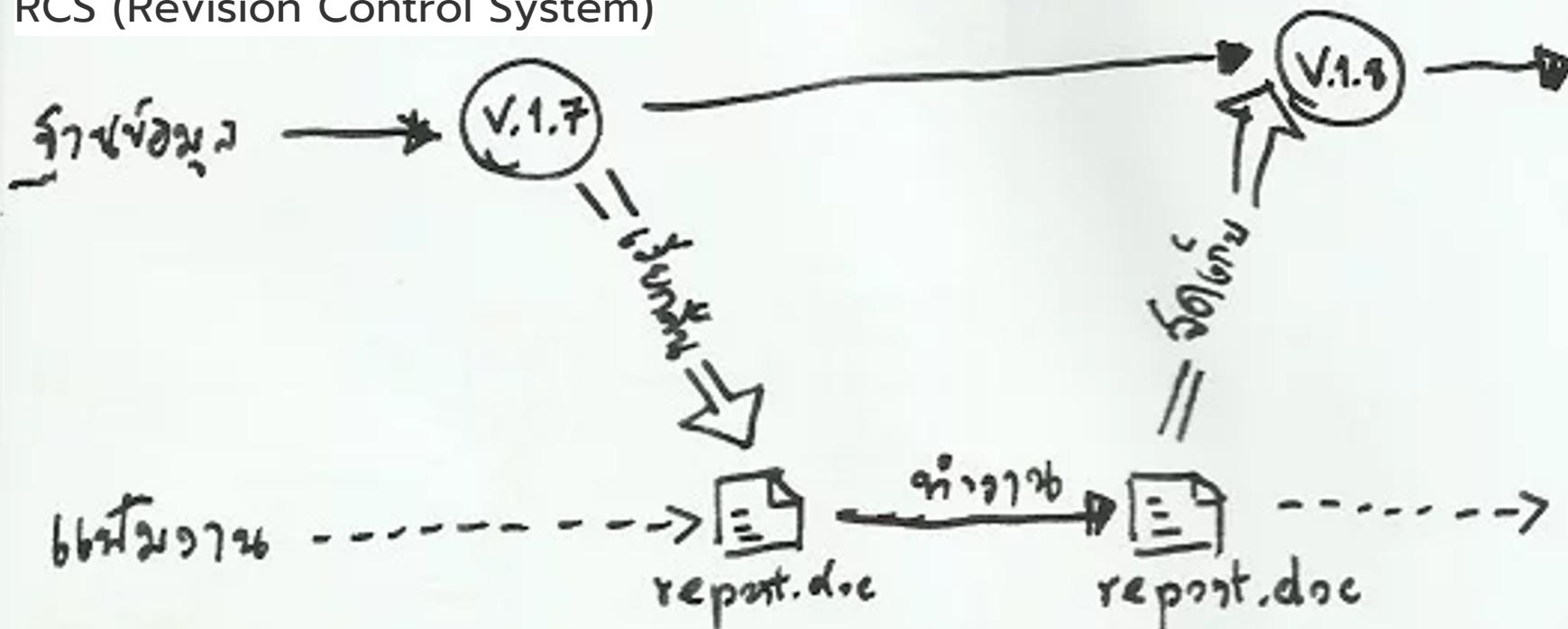


যুক্তিবৰ্বদ্ধণ পদ্ধতির জন্য patch(স্বীকৃত এবং অস্বীকৃত)

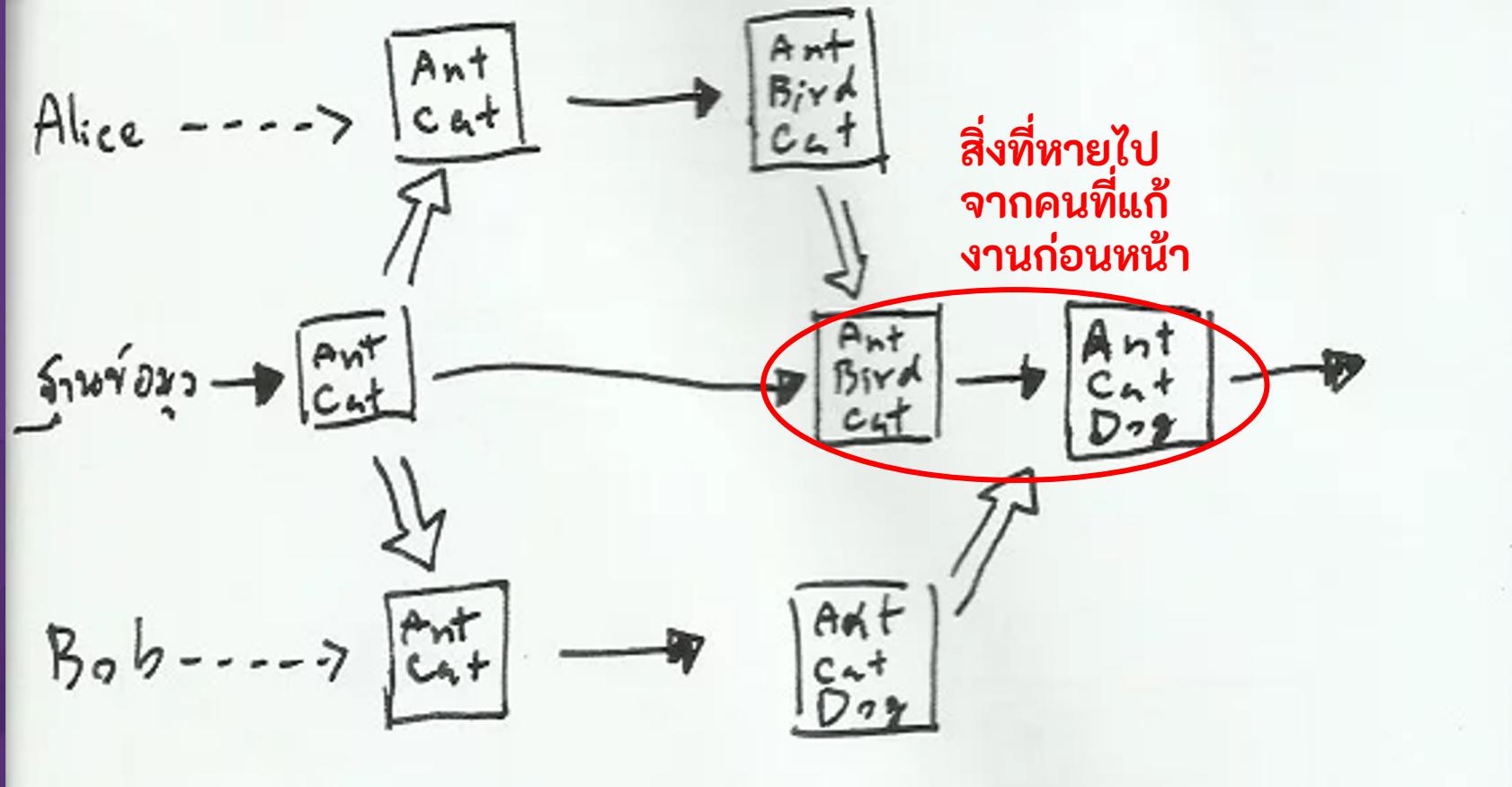


ຢູ່ຄະບບຈັດການຊອຮສແບບເວພາກ໌ (Local Version Control System)

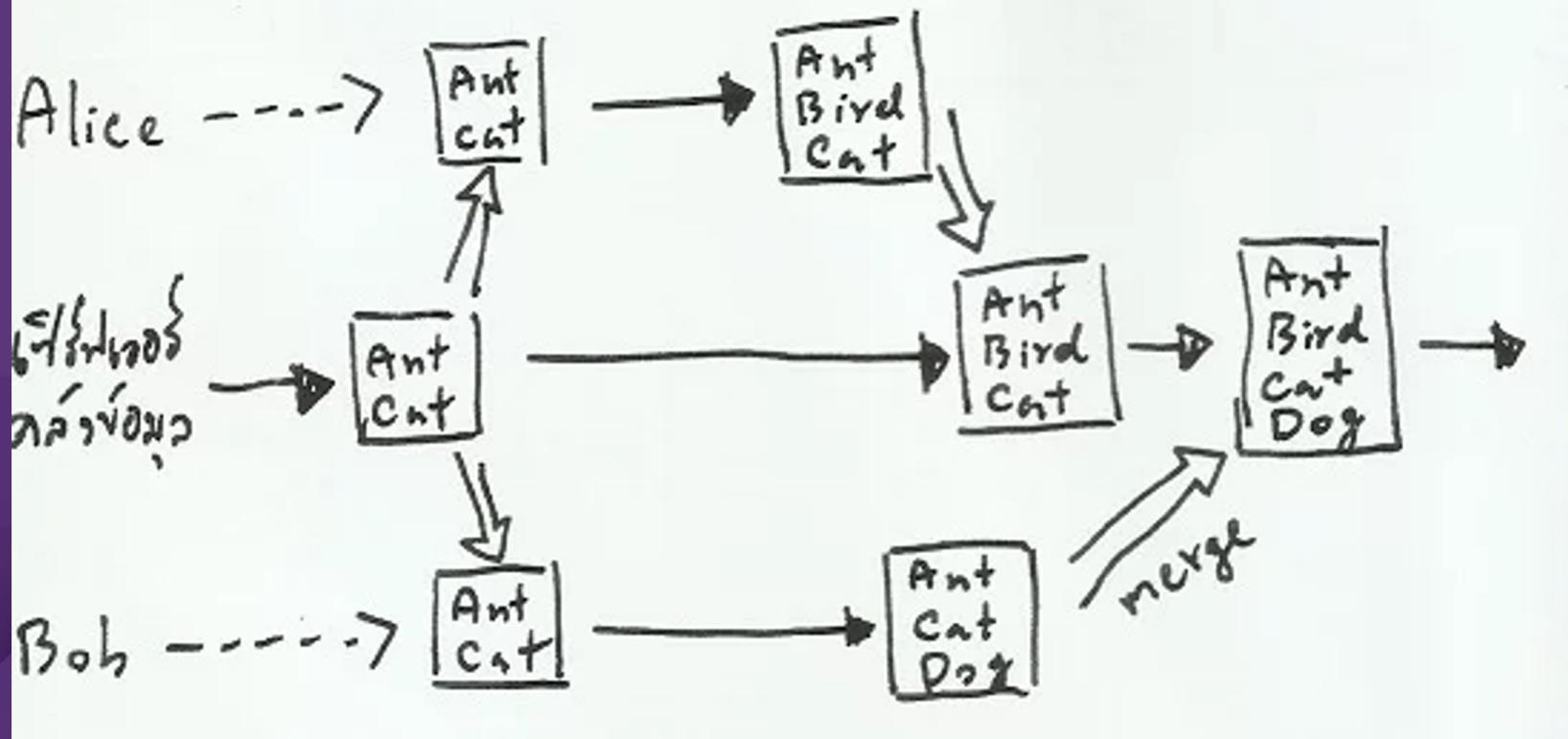
RCS (Revision Control System)



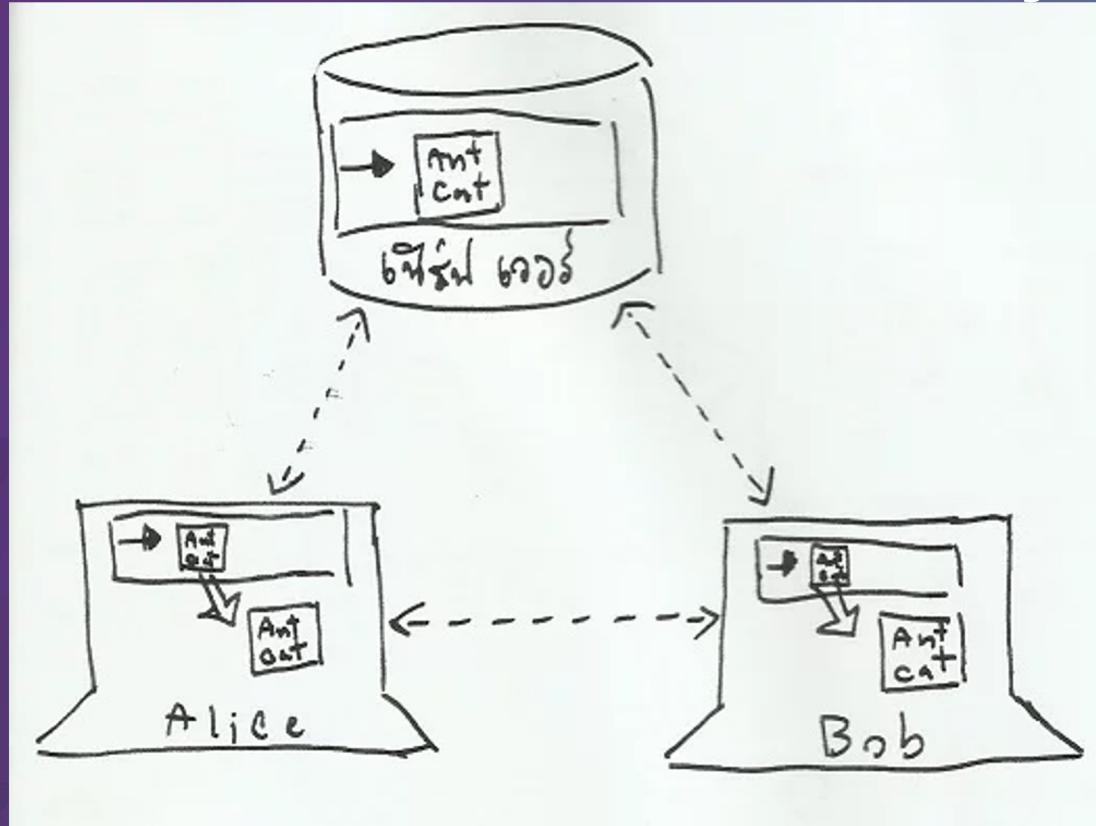
ปัญหาที่เกิดขึ้นจากการทำงานร่วมกันเป็นทีม



ยุคระบบจัดการซอฟต์แวร์รวมศูนย์ (Centralized Version Control System)



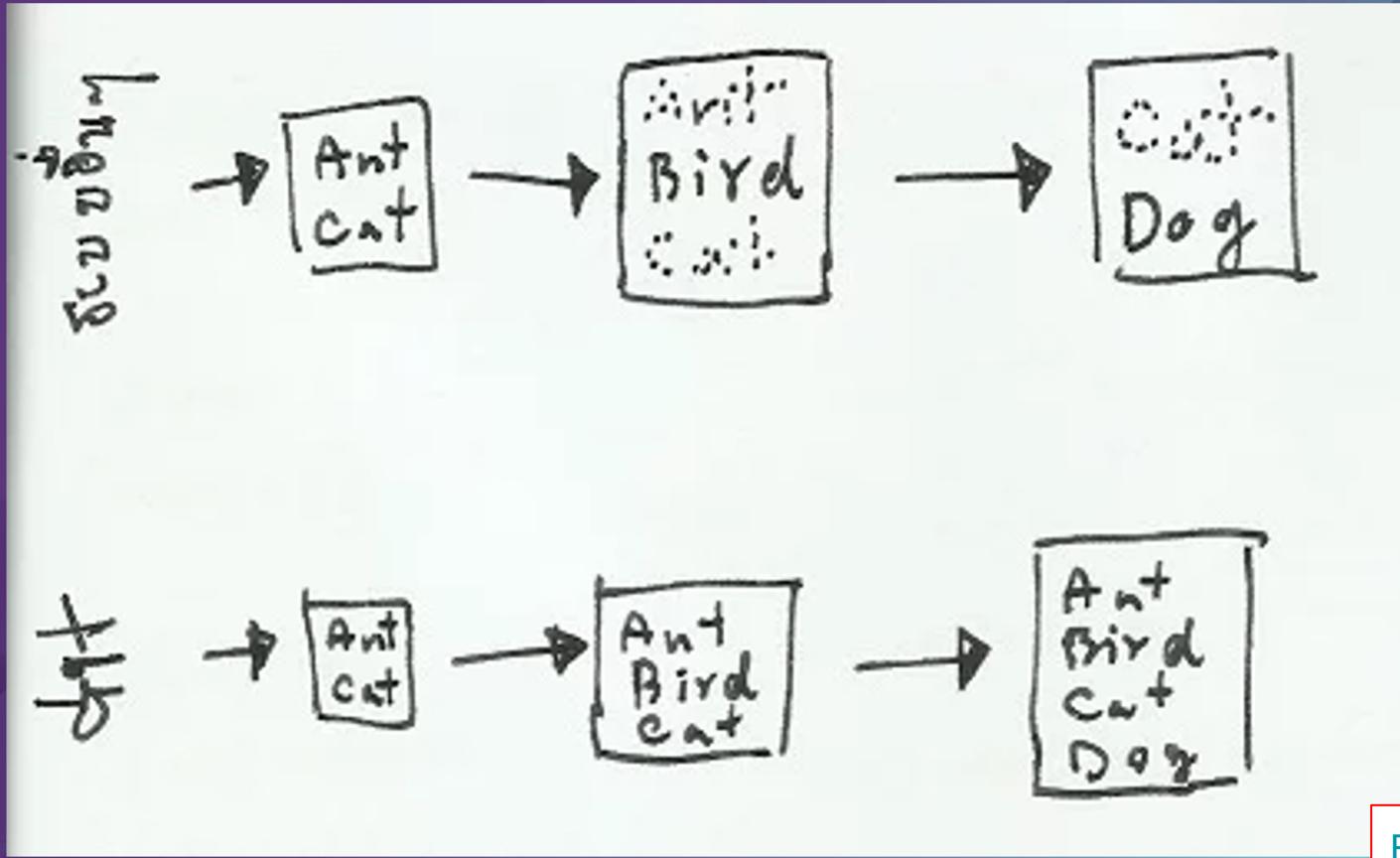
ยุคระบบจัดการชอร์สแบบกระจายศูนย์ (Distributed Version Control System)



ສຸດກ້າຍ git ກົດສ່າງຂຶ້ນໂດຍຊຸມບຸນ linux



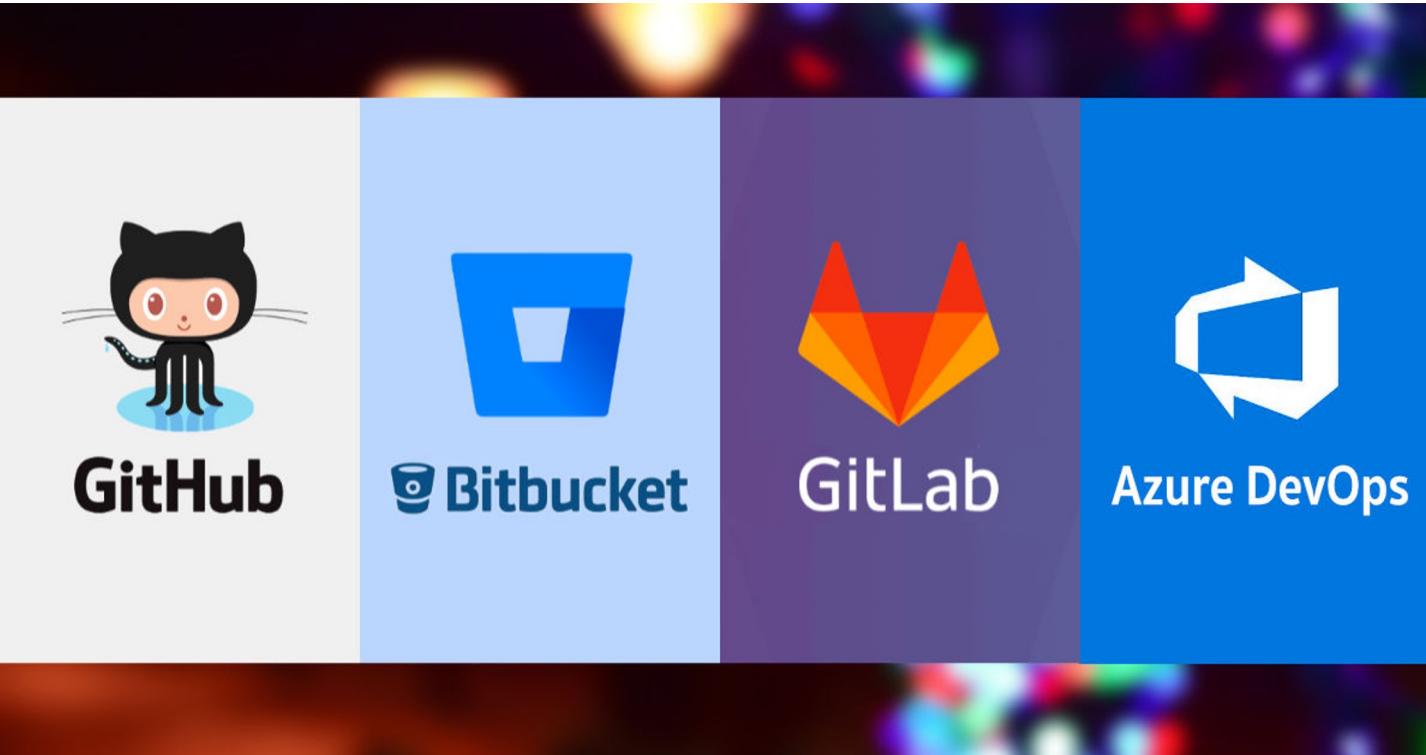
แนวคิดการมองภาพรุ่นของ Git เมื่อเทียบกับระบบอื่นๆ





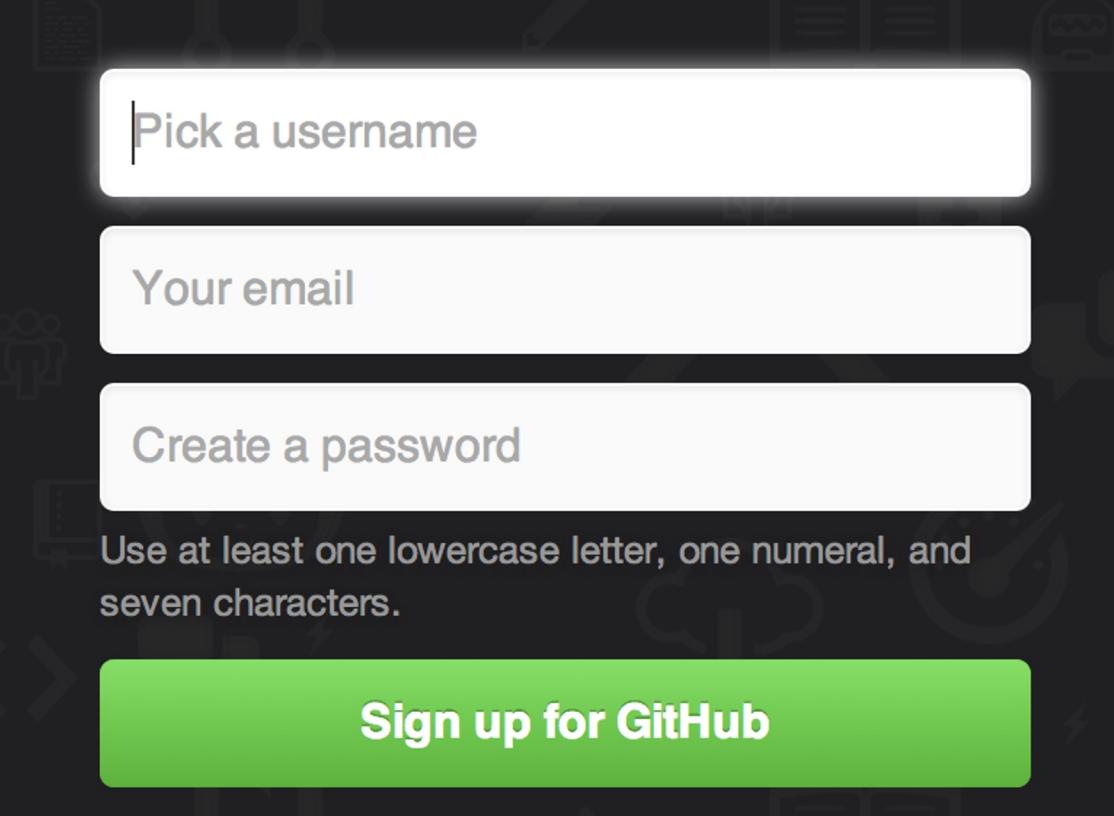
Software Park Thailand
</Code Camp>

Git host ใน ปัจจุบันมีครับ้าง





github register



The image shows a GitHub registration form on a dark background. It consists of three input fields: 'Pick a username', 'Your email', and 'Create a password'. Below the password field is a note: 'Use at least one lowercase letter, one numeral, and seven characters.' At the bottom is a large green button labeled 'Sign up for GitHub'.

Pick a username

Your email

Create a password

Use at least one lowercase letter, one numeral, and seven characters.

Sign up for GitHub



git lab register

The screenshot shows the GitLab.com sign-in page at https://gitlab.com/users/sign_in. The page features a large orange logo at the top right. Below it, a message states: "GitLab.com offers free unlimited (private) repositories and unlimited collaborators." A list of links follows:

- [Explore projects on GitLab.com](#) (no login needed)
- [More information about GitLab.com](#)
- [GitLab.com Support Forum](#)
- [GitLab Homepage](#)

Below this, a note says: "By signing up for and by signing in to this service you accept our:" followed by two links:

- [Privacy policy](#)
- [GitLab.com Terms.](#)

The right side of the page contains a sign-in form with fields for "Username or email" and "Password", a "Remember me" checkbox, and a "Sign in" button. Below the form is a link to "Forgot your password?". Further down, a message says "Don't have an account yet? [Register now](#)". At the bottom, there's a "Sign in with" section featuring icons for Google, GitHub, Twitter, Bitbucket, and Salesforce. The GitHub icon is highlighted with a red box.

git lab register with github



Repository

repository คือ folder ที่ใช้สำหรับเก็บงาน project ของเรา ซึ่งเราสามารถสร้าง branch (folder ย่อย) เพื่อแยกไปเขียน code ก่อนที่เราจะเป็นเวอร์ชันสมบูรณ์



github create repository

Signed in as
Bhornumnard

 Set status

Find a repository... Type: All Language

Repositories 6 Projects 0 Packages 0 Stars 0 Followers 2 Following 0

precodemcamp
HTML Updated 2 days ago

codecamp
HTML Updated on Dec 21, 2019

Project-day03 Private
code camp project test day 03
HTML Updated on Oct 24, 2019

Star

Your profile

Your repositories

Your projects

Your stars

Your gists

Feature preview

Help

Settings

Sign out





github create repository

Screenshot of a GitHub profile page showing repositories. A red box highlights the 'New' button in the top right corner of the repository search bar.

Search or jump to... Pull requests Issues Marketplace Explore

Overview Repositories 6 Projects 0 Packages 0 Stars 0 Followers 2 Following 0

Find a repository... Type: All Language: All [New](#)

precodecamp
HTML Updated 2 days ago [Star](#)

codecamp
HTML Updated on Dec 21, 2019 [Star](#)

Project-day03 Private
code camp project test day 03 [Star](#)
HTML Updated on Oct 24, 2019

Gittest [Star](#)



github create repository

Owner /

Repository name *

Great repository names are short and memorable. Need inspiration? How about [expert-adventure?](#)

Description (optional)

 **Public**
Anyone can see this repository. You choose who can commit.

 **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer.





git bash (git command line)

https://git-scm.com/downloads

Downloads

Mac OS X Windows
Linux/Unix

Latest source Release
2.25.0
Release Notes (2020-01-13)
Download 2.25.0 for Windows

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

GUI Clients

Git comes with built-in GUI tools (`git-gui`, `gitk`), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

Git via Git

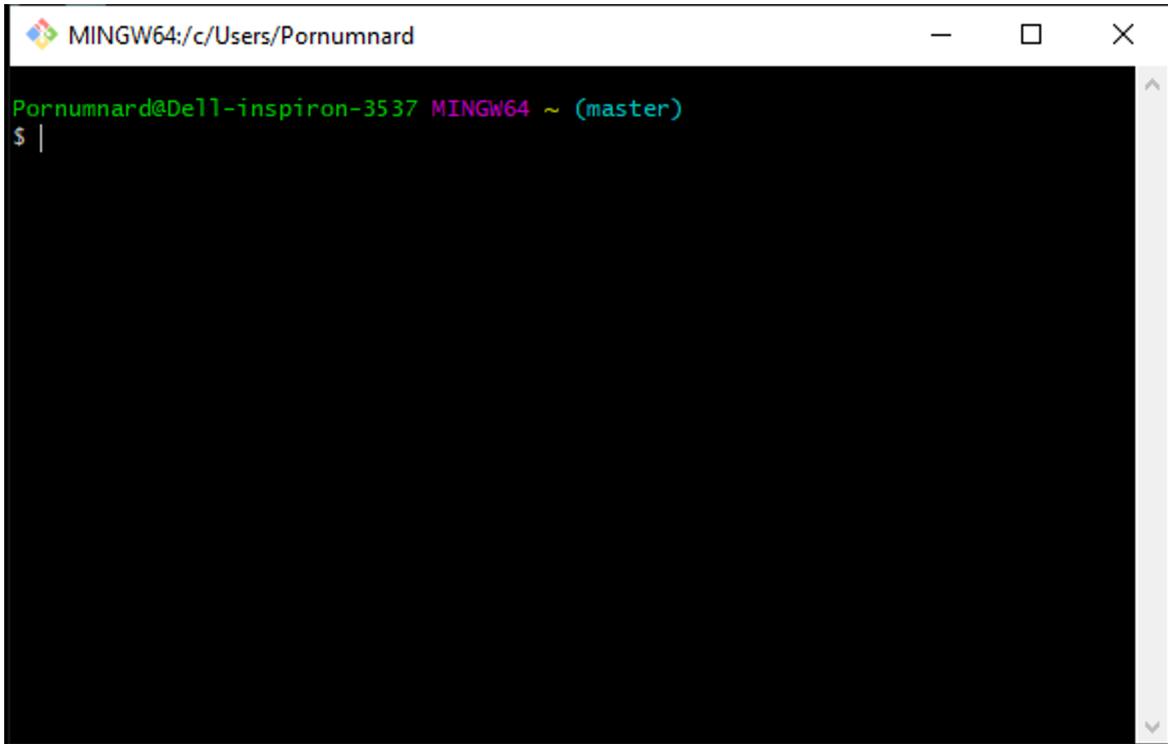
If you already have Git installed, you can get the latest development version via Git itself:

[gitbash download](#)

git bash display



Software Park Thailand
〈Code Camp〉
CODE CAMP#5



```
MINGW64:/c/Users/Pornumnard
Pornumnard@Dell-inspiron-3537 MINGW64 ~ (master)
$ |
```

basic command



Basic command (should known)

- git init : สร้างไฟล์ .git เพื่อเก็บ state และ ความเปลี่ยนแปลง
- git clone : คัดลอก repo มาลงในเครื่องของเรา
- git add : เพิ่ม state เพื่อเตรียม commit
- git commit : เตรียมข้อมูลการเปลี่ยนแปลง และ คำอธิบาย
- git status : เช็คสถานะของ state ณ ปัจจุบัน
- git merge : รวมไฟล์ระหว่าง 2 branch
- git pull : download ข้อมูล จาก repo ลงเครื่อง
- git push : upload ข้อมูล จากเครื่องขึ้น repo

[สรุป git command pdf](#)

Lab



New repository

- ให้สร้าง git repository ใหม่ ขึ้นมาว่า git test
- ทำการสร้าง folder ที่จะเก็บไฟล์ภายในเครื่อง
- คลิกขวาที่ folder ที่สร้างขึ้นมาใหม่และรัน git bash
- ทำการเชื่อม github กับ computer เรา
- สร้างไฟล์ Readme.md



markdown

markdown คือ markup language ชนิดหนึ่งที่เป็นการแสดงรูปแบบข้อความ ลำดับ และลิ้งค์ต่างๆ คล้ายๆ กับ HTML markdown นิยมใช้ในการแสดงคำอธิบายต่างๆ เพราะว่ามีรูปแบบการจัดการข้อความที่ง่าย และ แสดงผลข้อมูลได้อย่างมีประสิทธิภาพ

Ref. จดประสังค์ในการสร้าง readme.md

การเขียน markdown

#h1 = <h1> </h1>

##h2 = <h2> </h2>

###h3 = <h3> </h3>

####h4 = <h4> </h4>

#####h5 = <h5> </h5>

#####h6 = <h6> </h6>

2 spacebar enter =

การเขียน markdown

content, __content__ =

content, _content_ = <i> </i>

content, ___content___ = <i> </i>

__*content*__, **_content_*__ = <i> </i>

> content = | content

>> content = nested blockquotes

การเขียน markdown



1. content
3. content
4. content

+ content
- content
*content
tab * content

1. content
2. content
3. content

- content
- content
- content
 - content



การเขียน markdown

![detail pic] (url) = picture

***, ---, ----- = <hr>

[content](url) = content

Lab

Markdown



- ให้ลองเขียน markdown file
- สร้าง README.md
- ให้ใส่รายละเอียด repository หรือ ข้อมูลพื้นฐานต่างๆ

- ใช้คำสั่ง header
- สร้าง link
- สร้าง list
- สร้าง ตัวหนา ตัวเอียง
- สร้าง <hr>
- สร้าง blockquotes



git state

git state



Software Park Thailand
Code Camp
CODE CAMP#5

Local

working
directory

staging
area

localrepo

Remote

remote
repo

git flow



Git flow

upload file (work directory => remote repro)

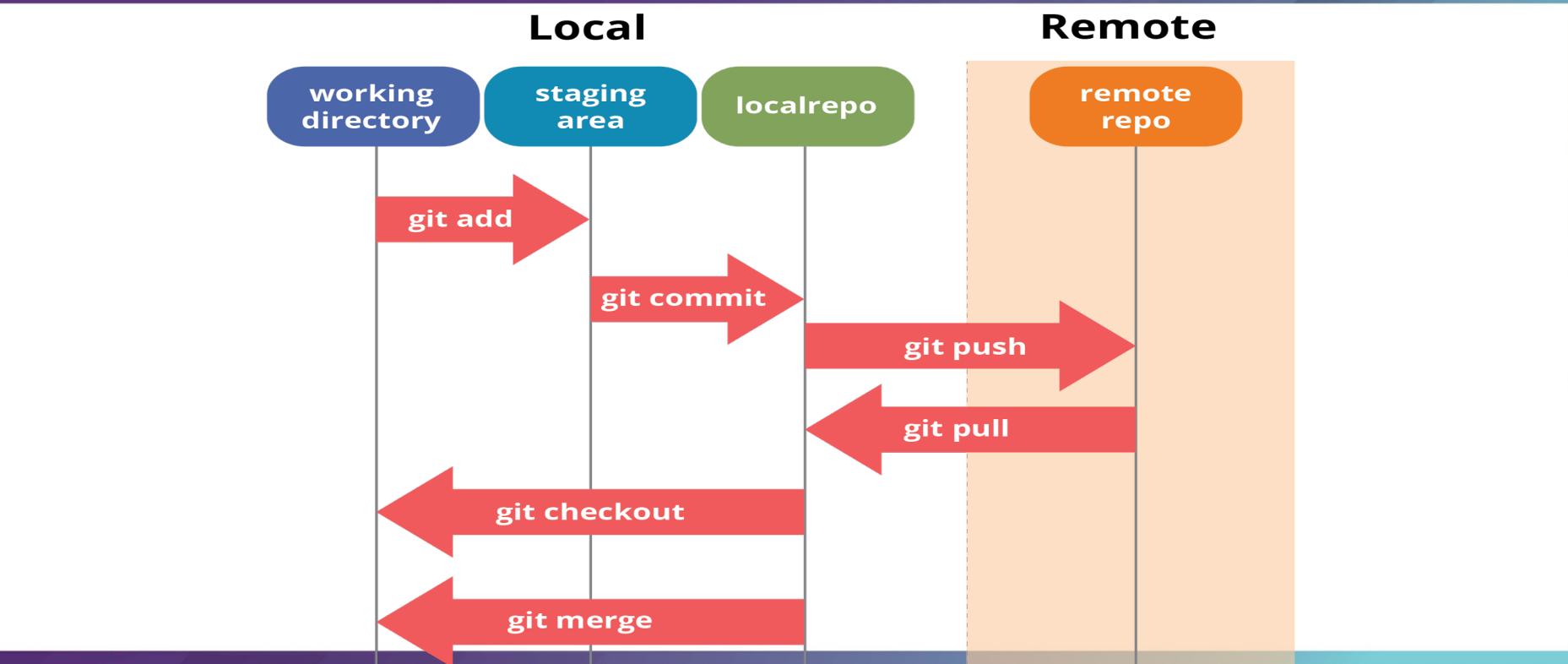
1. edit file
2. git add .
3. git commit -m "title" -m "description"
4. git push

download file (remote repro => work directory)

1. git pull



Git flow





Git graph

```
Have fun!
$ git commit
$ git commit
$ git commit
$ git commit
```

HEAD: master



master
HEAD

Lab

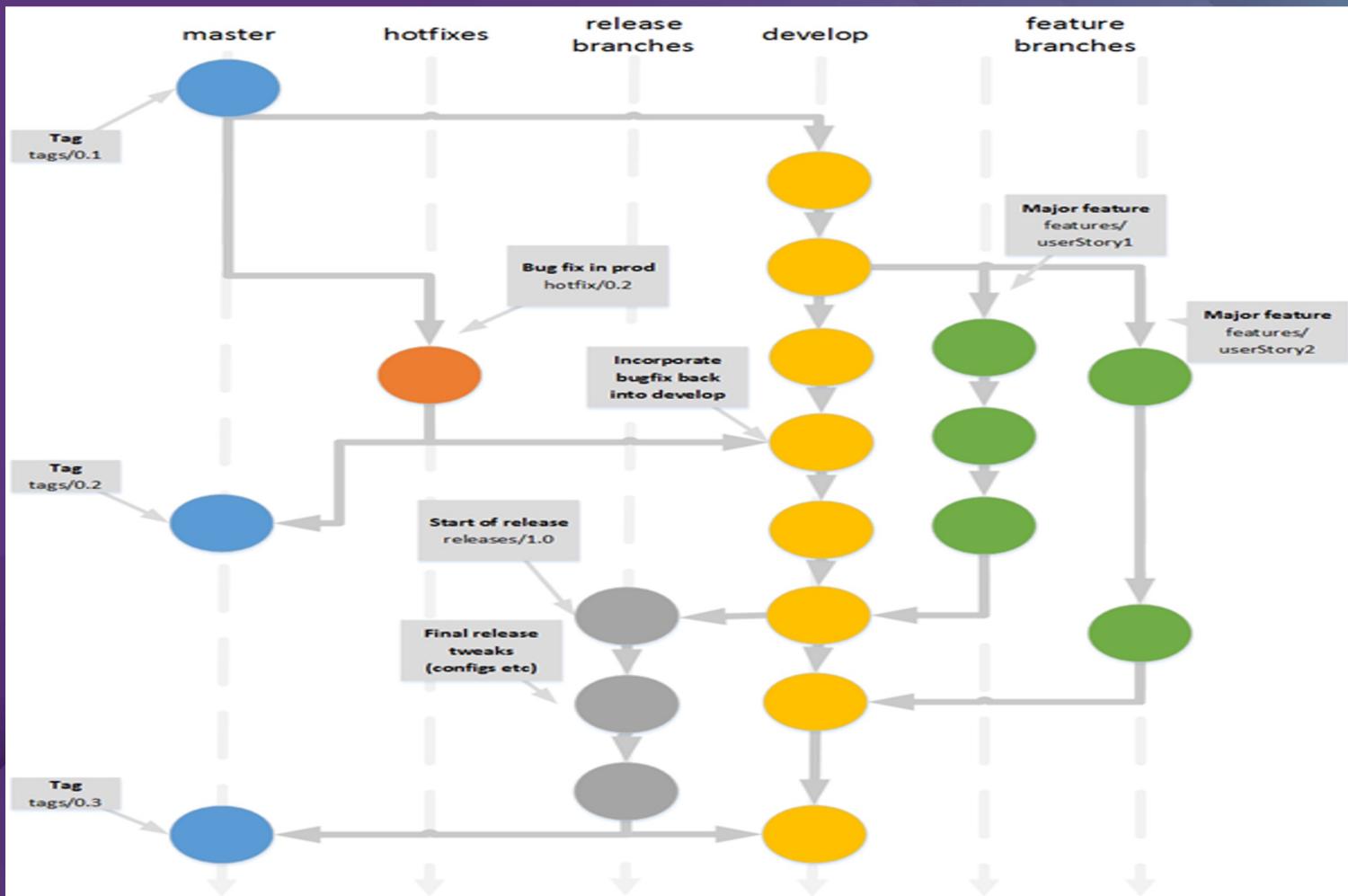
git flow

- ให้ลองเขียน สร้างไฟล์ .txt
- พิมพ์ข้อความอะไรก็ได้
- save และ ทำการ add file ขึ้น repository ของเรา
- เข้าไปเช็คว่าไฟล์ของเรามาขึ้นไป หรือไม่ที่เว็บ git
- ลองลบไฟล์ .txt ออก
- ลอง git restore จาก repo มาที่เครื่องเรา

git add .
git commit -m "title" -m
"description"
git push

git restore .

git with team



git branch



Branch in git

branch (สาขา) สร้างขึ้นแบ่งการทำงานออกเป็นส่วนๆ เพื่อป้องกัน และ ดัก code ที่มีปัญหา ก่อนการ deploy (เผยแพร่)



การแตก branch และ การสลับ branch

`git branch --list` : เป็นการเช็ค branch ว่ามี branch อะไรบ้าง

`git branch branch_name` : เป็นการแตก branch ใหม่

`git checkout branch` : เป็นการสลับ branch ในการทำงาน

`git checkout -b branch_name` : เป็นการสร้าง branch ใหม่ และ สลับไป branch พร้อมกัน

Lab

branch

- สร้าง branch ใหม่ด้วย git branch
- checkout ไป branch ใหม่
- สร้างไฟล์ .txt
- แก้ไขไฟล์ .txt
- upload file ขึ้น branch ใหม่
- สลับกลับมา master
- ดูการเปลี่ยนแปลงที่เกิดขึ้น

- au branch ด้วย git branch -d branch_name
- ลองใช้คำสั่ง git checkout -b branch_name
- สร้างไฟล์ .txt upload
- สลับมา master
- ดูการเปลี่ยนแปลง



การรวมข้อมูลระหว่าง branch

git merge เป็นการรวมข้อมูล จาก branch ไปอีก branch เช่น เราต้องการ นำข้อมูล จาก develope ไปรวมกับ master

1. เราต้อง commit และ push ทุกอย่าง ของ develop ก่อน
2. สลับไปหน้า master git pull ให้เรียบร้อย
3. ใช้คำสั่ง git merge develope แก้ไข conflict



การรวมข้อมูลระหว่าง branch

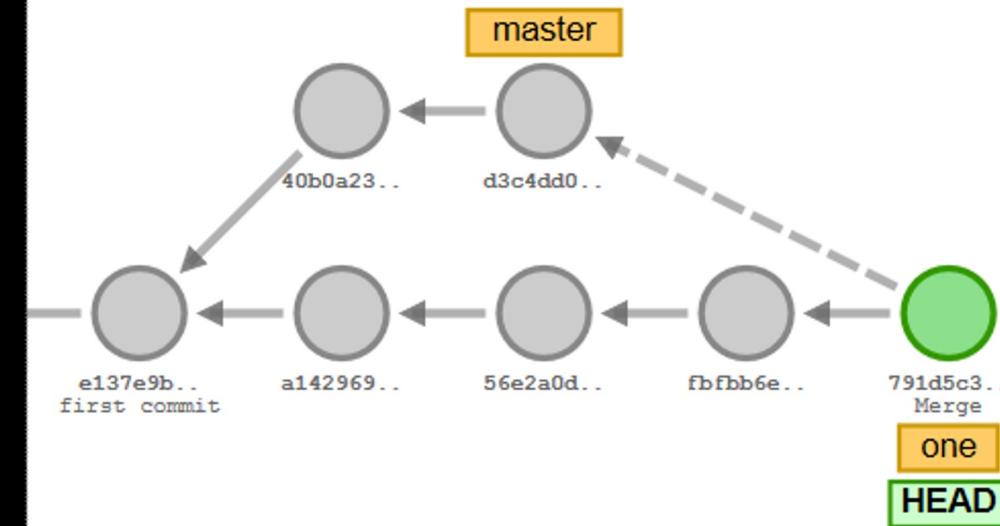
4. เมื่อแก้ไข conflict แล้ว ให้ save file นั้นๆ
5. git flow (git add => git push)
6. success รวมข้อมูลเรียบร้อย

Free Explore

```
Have fun!  
$ git branch one  
$ git checkout one  
$ git commit  
$ git commit  
$ git commit  
$ git checkout master  
$ git commit  
$ git commit  
$ git checkout one  
$ git merge master
```

Local Repository

HEAD: one



Lab

git merge



merge

กรณีมี conflict

- แก้ไข file .txt ใน master
- push ขึ้น repo
- สลับไป branch ที่สร้างเมื่อ กี้
- แก้ไข file.txt ให้ไม่เหมือน ใน master
- push ขึ้น repo
- ทำการ merge เข้ากับ master

กรณี ไม่มี conflict

- สลับมา branch ใหม่
 - แก้ไขไฟล์ .txt
 - push ขึ้น repo
 - ทำการ merge เข้ากับ master อีกรอบ
- กรณีมี การเพิ่ม comment
- สลับมา branch ใหม่
 - แก้ไขไฟล์ .txt แบบ comment
 - push ขึ้น repo
 - ทำการ merge เข้ากับ master
 - อีกรอบ



git command trick

```
git commit -am "title" -m "description"  
git checkout -b name  
git branch -m old_name new_name  
git log --oneline --decorate --graph --all  
git merge branch_name -m "description"  
git merge branch_name --no--ff (no fast  
forward)
```

มาลองเล่น git กัน



git rebase



git rebase

rebase เป็นการย้ายข้อมูล จาก branch
หนึ่งมาต่ออีก branch

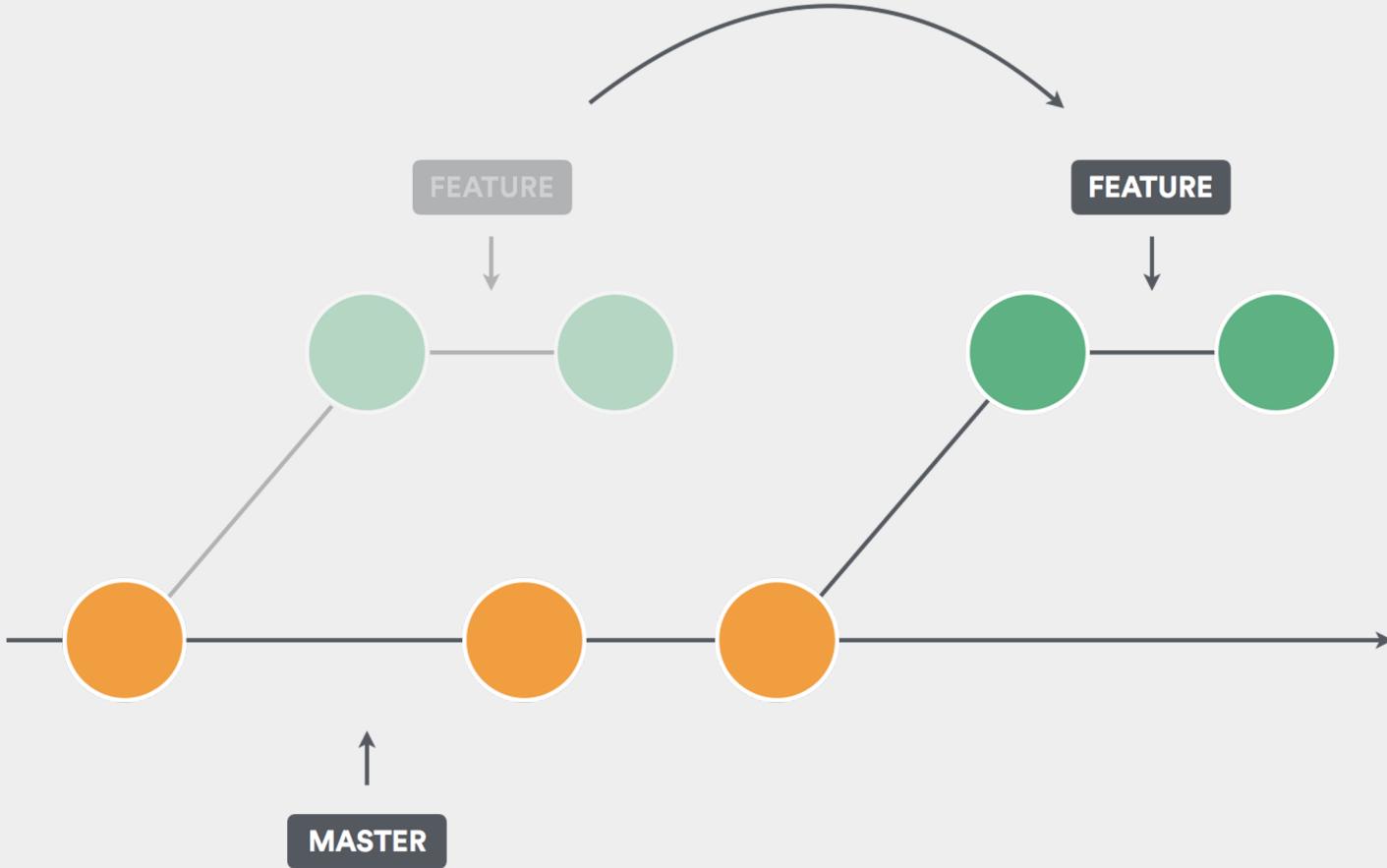
ขั้นตอนการทำ rebase จาก dev to master

1. update repo
2. checkout master
3. git pull
4. git rebase dev



git rebase

6. ตรวจสอบ conflict
7. ถ้ามีให้แก้ไข save และ git add
8. ถ้าไม่มีให้ git add ได้เลย
9. git rebase --continue
10. git commit and push

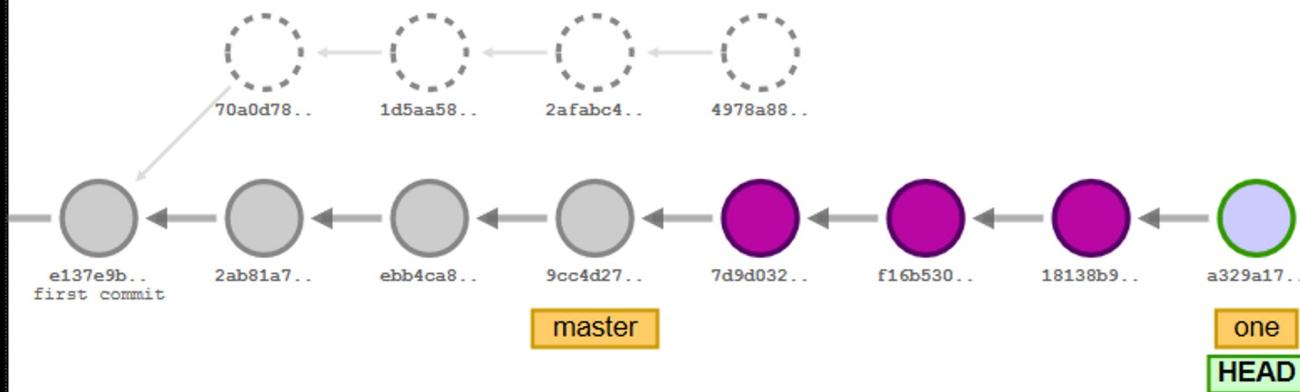


Free Explore

```
Have fun!
$ git checkout one
Cannot find commit: one
$ git branch one
$ git commit
$ git commit
$ git commit
$ git checkout master
$ git checkout one
$ git commit
$ git rebase master
```

Local Repository

HEAD: one



Lab

rebase



Software Park Thailand
Code Camp
CODE CAMP#5

กรณีมี conflict

- สร้าง branch ใหม่
- แก้ไข file .txt ใน master
- push ขึ้น repo
- สลับไป branch ที่สร้างเมื่อ กี้
- แก้ไข file.txt ให้ไม่เหมือน ใน master
- push ขึ้น repo
- ทำการ rebase เข้ากับ master

กรณี ไม่มี conflict

- สลับมา branch ใหม่
 - แก้ไขไฟล์ .txt
 - push ขึ้น repo
 - ทำการ rebase เข้ากับ master
- อีกรอบ

git ignore

git ignore

[ข้อมูลการใช้เพิ่มเติม](#)

[การสร้าง gitignore](#)



Software Park Thailand
Code Camp
CODE CAMP #5

git ignore เป็นการสร้างไฟล์ .gitignore ขึ้นมาเพื่อเป็นการกันไฟล์ที่ไม่ต้องการบางตัวขึ้น git repo ที่เราได้ทำการ Remote ไว้

การเขียน gitignore

name/ : ทุก folder ที่มีชื่อนี้

/name/ : ทุก folder ที่มี name ในชื่อ

*.file : ทุกไฟล์ที่มีนามสกุลนี้

name.* : ไฟล์ที่ขึ้นต้นด้วยชื่อนี้ ใน path เดียวกัน

.file: ทุกไฟล์ที่มีนามสกุล .file และ .fileอย่างอื่นด้วย

!\! file.txt ! : ยกเลิกการ ignore ไฟล์ .txt ที่ขึ้นต้นด้วย file กั้งหนด

**/ name : เฉพาะ file ชื่อ name หรือ folder ชื่อ name

a/** or a/**/b: ทุกไฟล์ใน folder a กั้งหนด หรือ folder b ใน a

* = file
** = folder
! = ยกเลิก

Lab

git ignore

ລອງ ສ້າງໄຟຟ້າ .html, .css, .js
push repo

ignore ໄວ້າ .html
git rm --cached file_name
push repo

edit ignore ໄວ້າ .css
push repo
git rm --chached
file_name

edit ignore ໄວ້າ .js
push repo
git rm --cached
file_name
push repo

