

クラウド開発演習（課題解決型学習） Project-/Problem-Based Learning on Cloud Computing

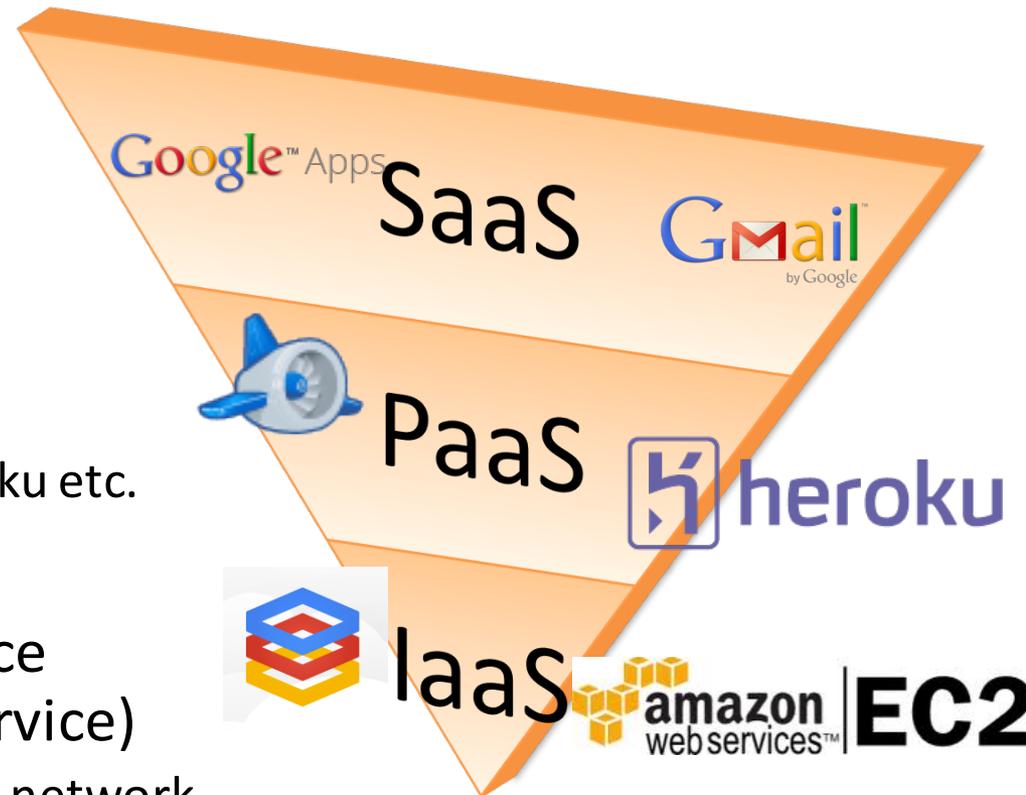
ソフトウェア・クラウド開発プロジェクト実践I

浅井大史

2016年4月15日

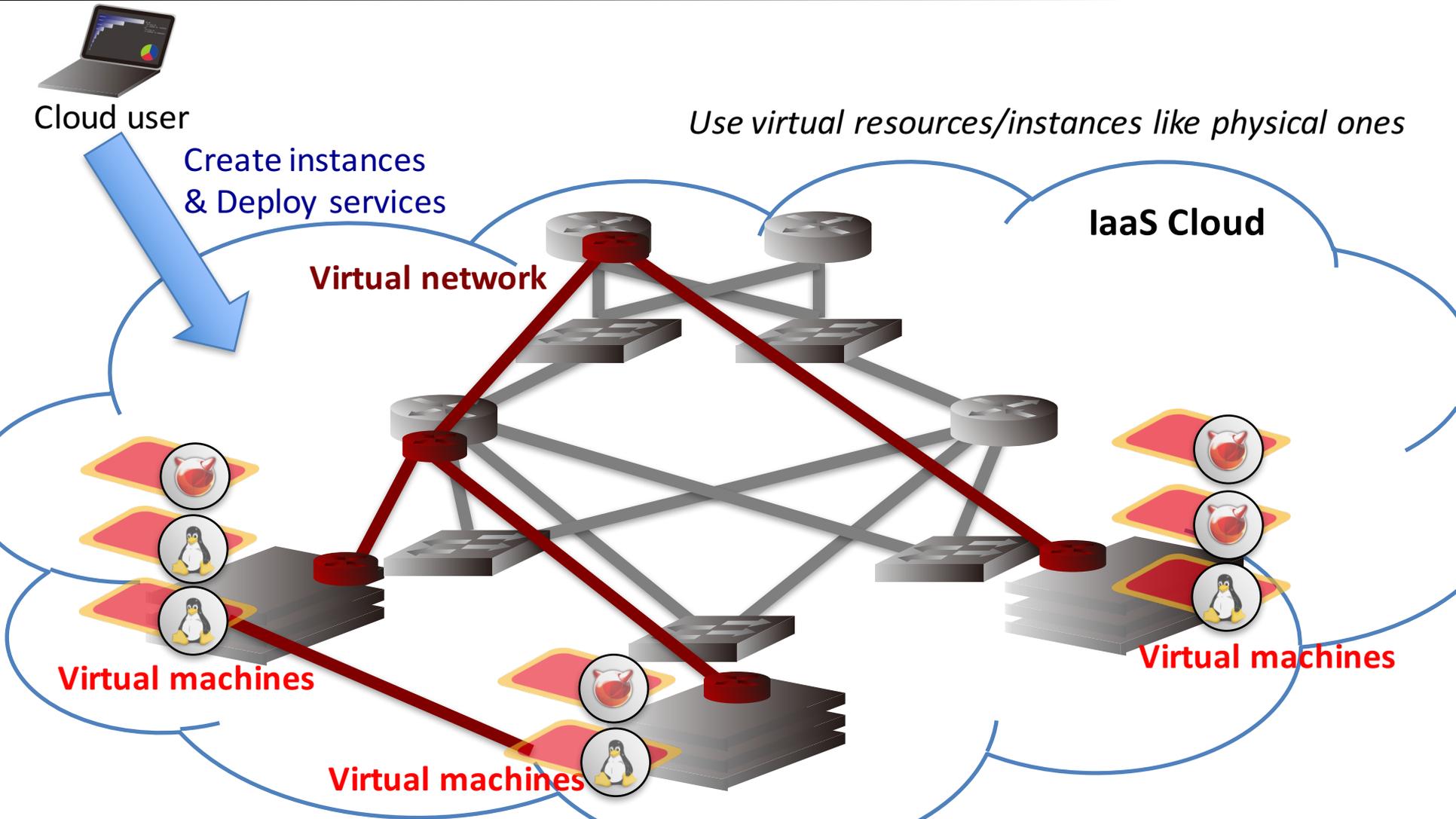
Cloud Service Models: X as a Service

- **SaaS**
 - Software as a Service
 - Google Apps, Gmail etc.
- **PaaS**
 - Platform as a Service
 - Google App Engine, heroku etc.
- **IaaS**
 - Infrastructure as a Service (a.k.a. Hardware as a Service)
 - Virtual machine, storage, network
 - Google Compute Engine, Amazon EC2 etc.



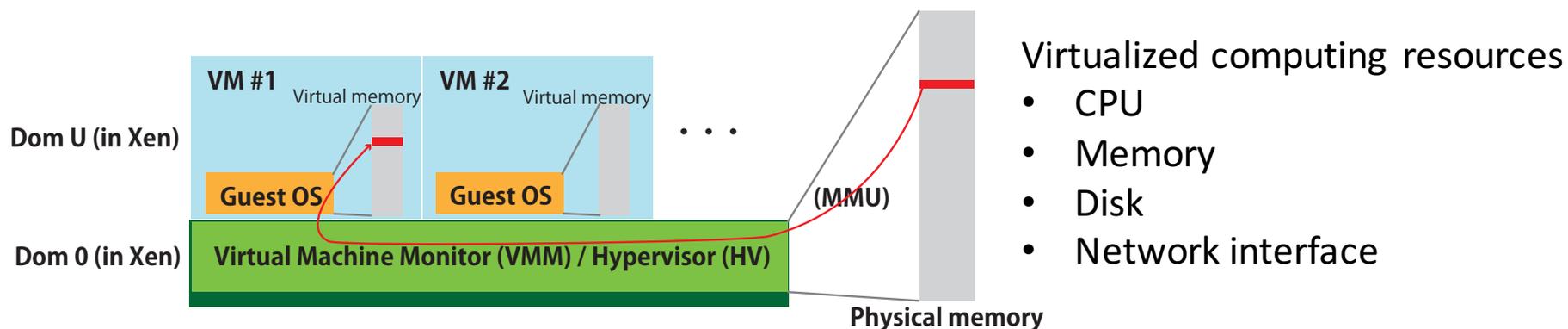
<http://mail.google.com/>
<http://www.google.com/apps/>
<http://developers.google.com/appengine/>
<http://heroku.com/>
<http://cloud.google.com/products/compute-engine>
<http://aws.amazon.com/ec2/>

Infrastructure as a Service



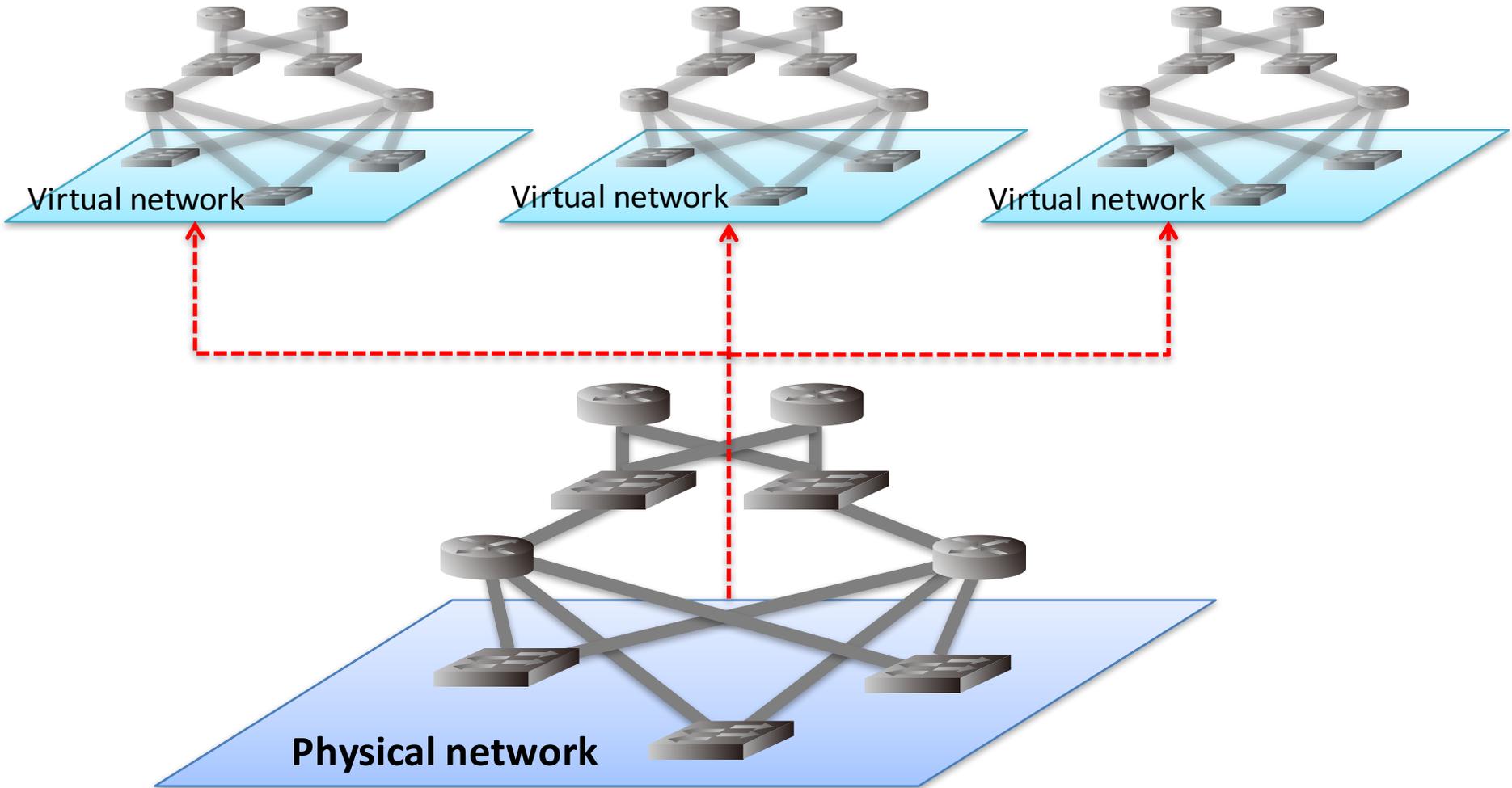
IaaS: Server Virtualization

Hypervisor: runs multiple virtual machines



- Hypervisor software
 - Xen
 - KVM
 - VMware vSphere Hypervisor
 - VirtualBox
- Related technologies
 - Hardware-assisted virtualization technologies
 - e.g., Intel® VT, VT-x, VT-d

IaaS: Network Virtualization



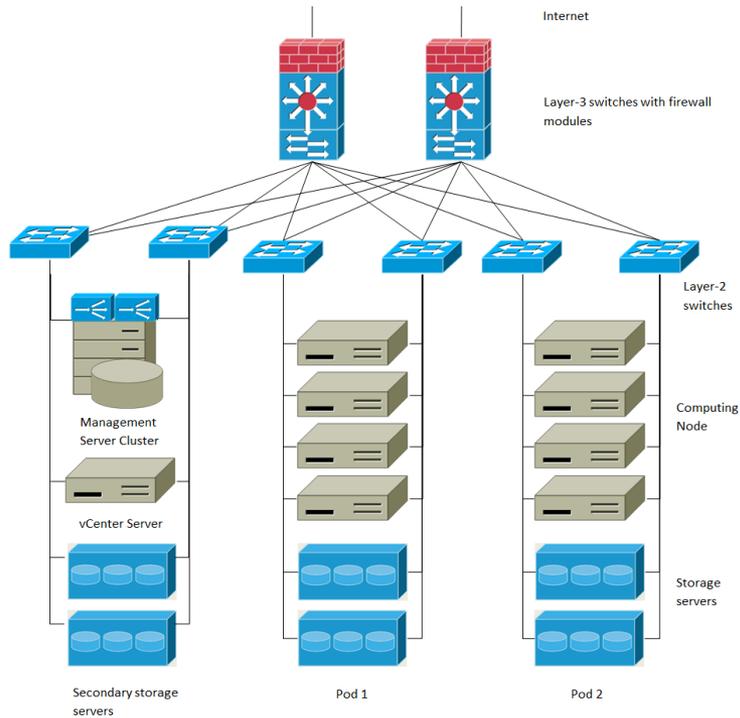
Network virtualization technologies:

VLAN, MPLS, VRF, L2TP, NV-GRE, STT, VXLAN, OpenFlow etc.

IaaS Cloud Management Software

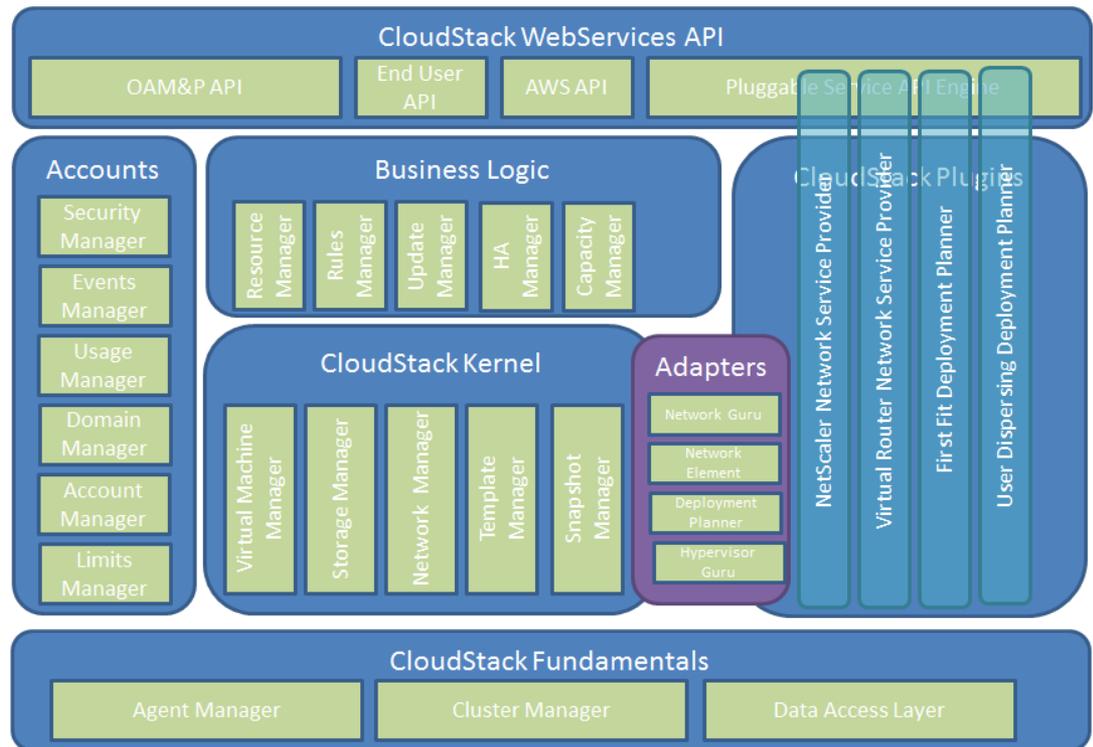
Integrated management softwares:

Eucalyptus, OpenNebula, OpenStack, WIDE Cloud Controller, Wakame, CloudStack etc.



Large-Scale Redundant Deployment

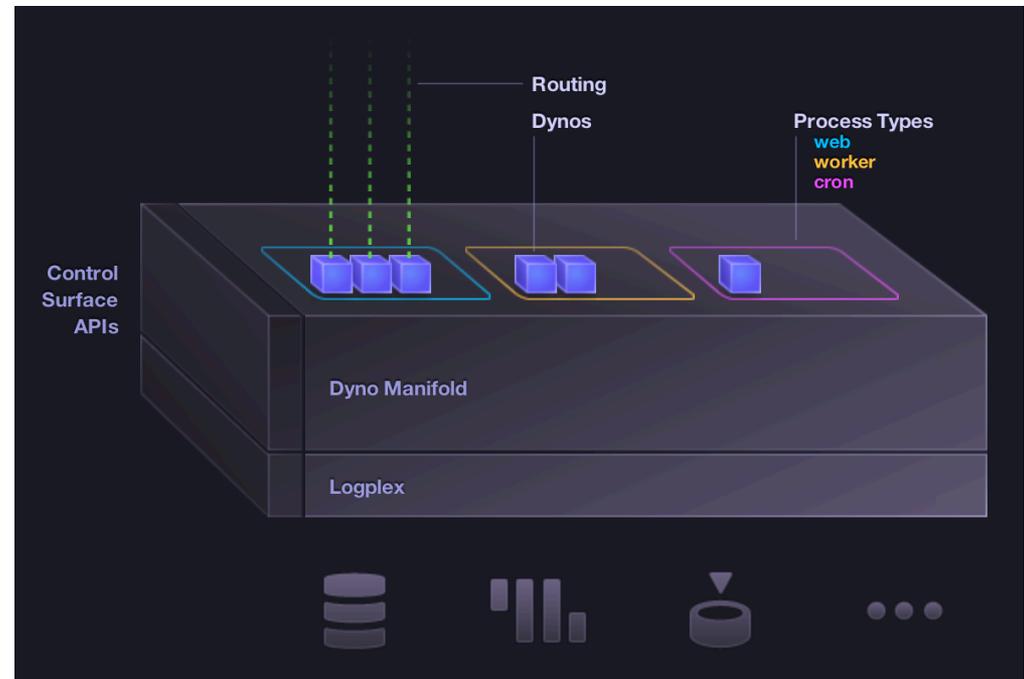
http://cloudstack.apache.org/docs/en-US/Apache_CloudStack/4.0.1-incubating/html/Installation_Guide/large_scale_redundant_setup.html



<https://cwiki.apache.org/CLOUDSTACK/development-101.html>

Platform as a Service

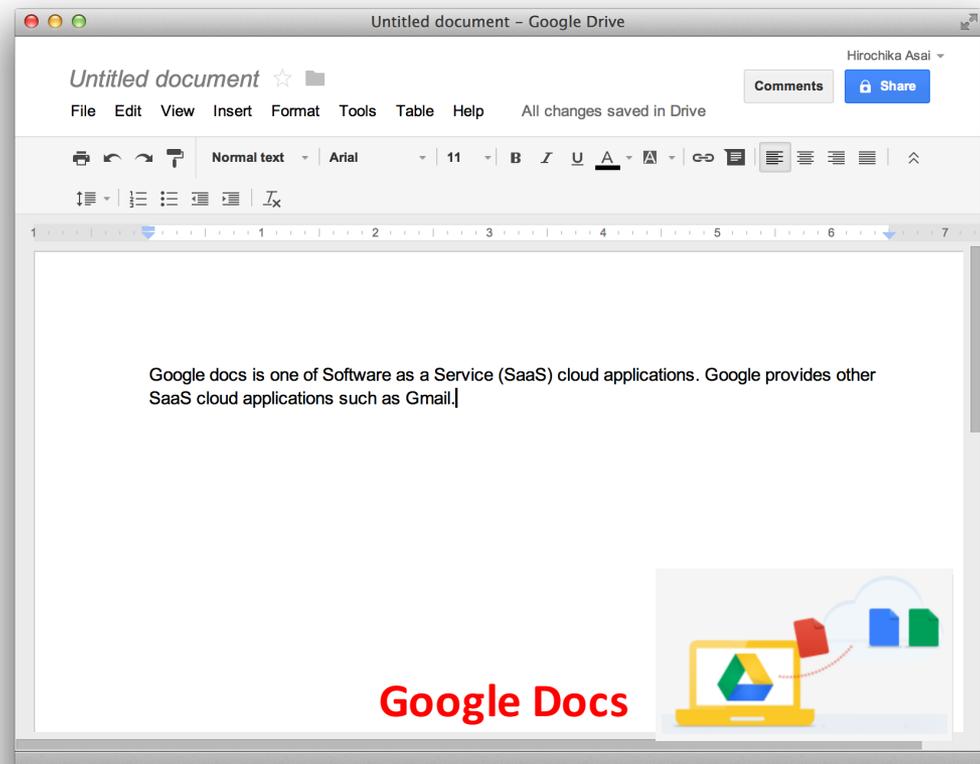
- PaaS
 - provides “platform”, hence PaaS users develop&deploy their applications on it
- PaaS providers
 - heroku
 - Google App Engine
 - Windows Azure
 - Amazon S3
 - (Storage as a Service)
- Related technologies
 - Distributed computing
 - Distributed storage



<https://www.heroku.com/how/scale>

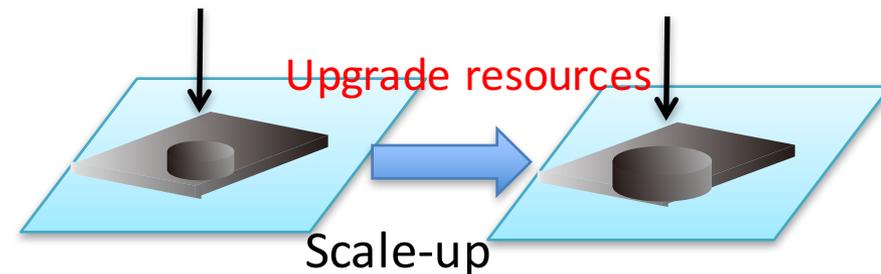
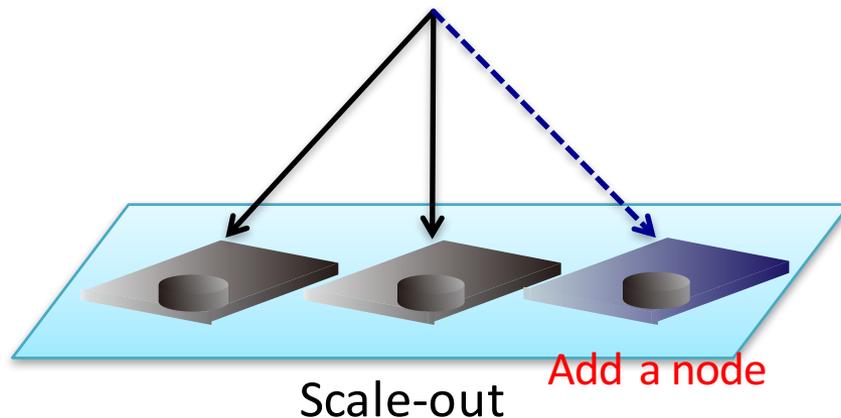
Software as a Service

- SaaS
 - provides “Software” usually directly accessed by end-users
- SaaS providers
 - Google Apps
 - Gmail
 - iCloud
 - github



Scale-out vs. Scale-up

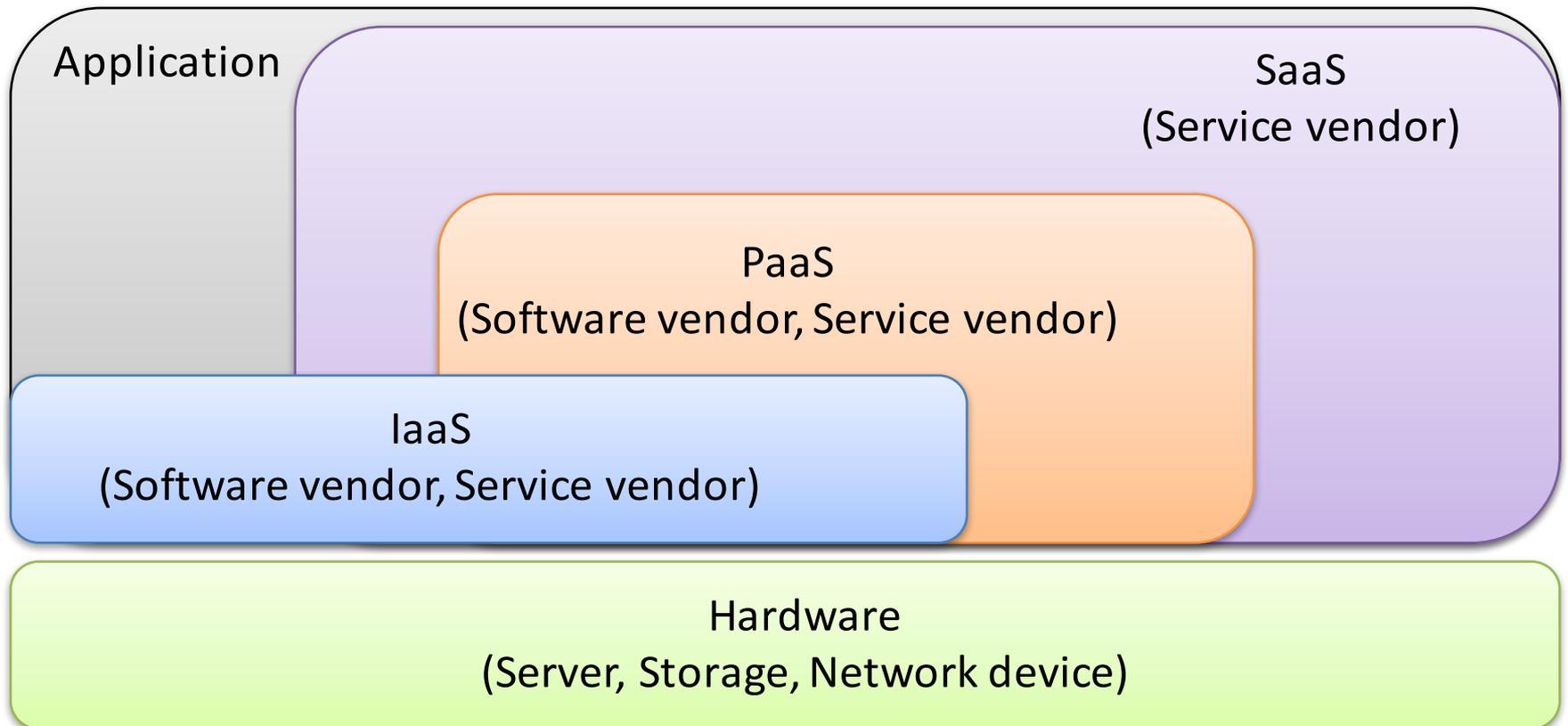
- Scale-out
 - a.k.a. horizontal scaling
 - low-cost
 - add nodes to a system
- Scale-up
 - a.k.a. vertical scaling
 - high-cost
 - add or upgrade resources to a single node



Abstraction level of cloud service model

- IaaS
 - Operating System
 - Process (CPU, Memory)
 - Filesystem (storage)
 - Network interface
 - Virtual network
- PaaS
 - Program (e.g., Hadoop, Ruby on Rails, Docker)
 - Database
- SaaS
 - User contents (data)

Summary of XaaS



Deployment Models

- Public cloud
 - Advantages
 - Lower initial cost (and operating cost)
 - Disadvantages
 - Not customizable
 - Lower privacy and security
- Hybrid cloud
- Private cloud
 - Advantages
 - Customizable
 - Higher privacy and security
 - Disadvantages
 - Higher initial cost (and operating cost)

Group PBL

- IaaS
 - (A1) Cloud controller
 - Hypervisor: KVM
 - Cloud controller: (using libvirt API)
 - (A2) Cloud storage
 - High-speed storage: 10GbE / RAID card + SSD
- PaaS
 - (B1) Hadoop cluster
 - Hadoop (MapReduce)
 - (B2) PaaS cloud
 - Linux container
- SaaS
 - (C1) SaaS application
 - Server: Scale-out capable application
 - Client: Browser



Project management for software development

- Project management
 - Ticket / Issue tracker
 - Gantt chart
 - Documentation
- Software development process
 - Waterfall model
 - Agile
 - eXtreme Programming (XP)
 - Scrum
 - Test-Driven Development etc.
- Team development tools
 - Source control management (SCM) system
 - Version control system (VCS)
 - RCS, CVS, SVN, Mercurial, Git

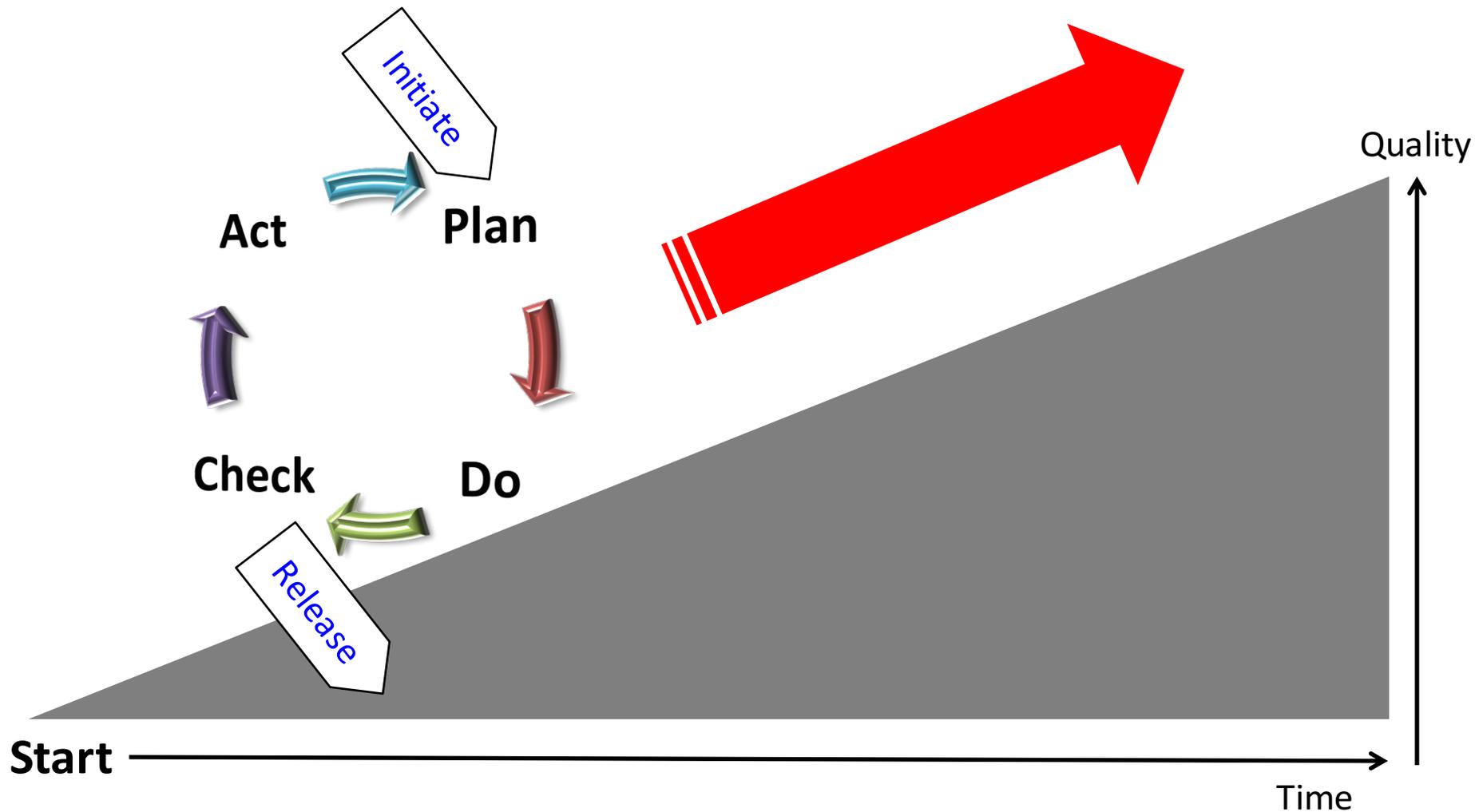


June 12: Software Engineering

Project management (an example, briefly)

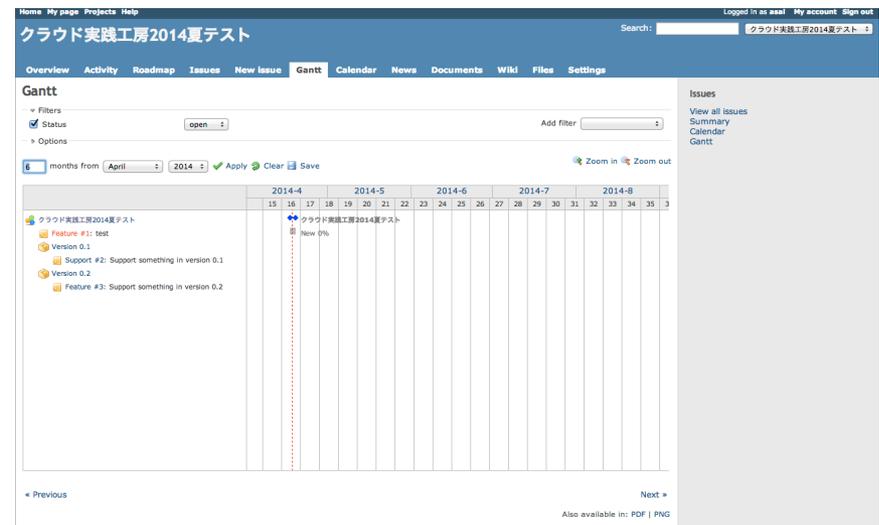
- Roadmap
 - Goal
 - Due (release date)
 - Interim milestones
 - Critical path
 - Approach
- PDCA cycle
 - Plan
 - Do
 - Check
 - Act / Adjust

Project management: PDCA-cycle



Project management tool

- Redmine <http://www.redmine.org/>
 - Open Source Software
 - Features
 - Issue tracking
 - Roadmap based on version
 - Gantt chart
 - Document uploader
 - Wiki



Team development tools

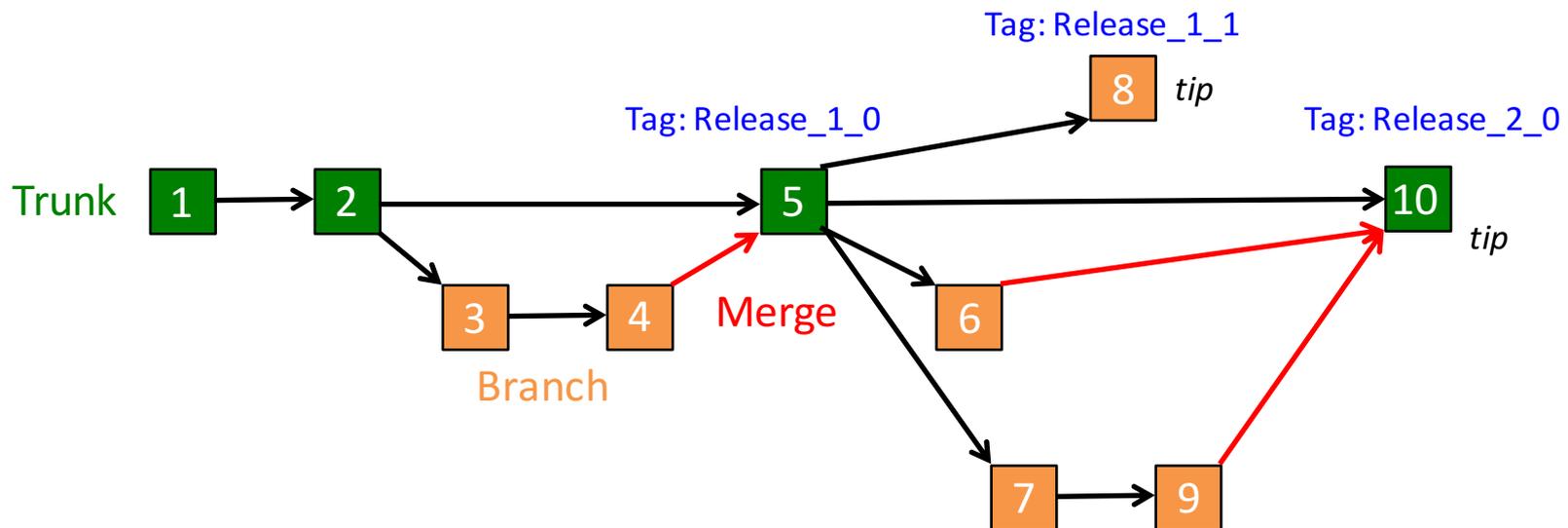
- Source control management (SCM) system
- Version control system (VCS)
 - e.g., RCS, CVS, SVN, Mercurial, Git

Fundamentals of Version Control Systems

- Terminology
 - Repository
 - where current files and histories are stored
 - Trunk
 - The main line of development (not branch lines)
 - Branch
 - Development lines forked from trunk or another branch
 - Tag
 - Important snapshot with human-friendly name
 - Head (or tip)
 - Most recent commit of trunk/branch
 - Commit
 - Write the changes to the repository
 - Merge
 - Merge two sets of changes
 - Conflict
 - occurs when multiple developers made changes to the same file

Revision structure

Directed Acyclic Graph (DAG) structure



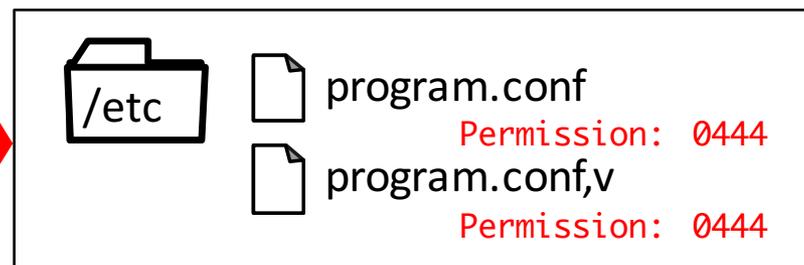
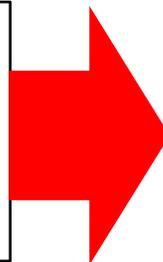
Comparison

	Target	Repository	Conflict handling (Multi-developer)
RCS	File	Local, Central	Lock/Unlock
CVS	Repository	Local/Remote, Central	Merge (commit)
Subversion	Repository	Local/Remote, Central	Merge (commit)
Mercurial	Repository	Local/Remote, Distributed	Merge (push/pull)
Git	Repository	Local/Remote, Distributed	Merge (push/pull)

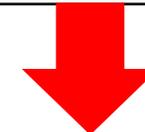
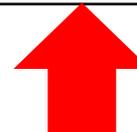
(Historical tool) RCS

Note: Still used to manage configuration files

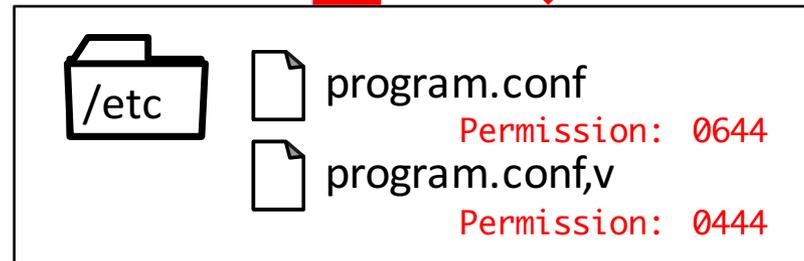
Check in (unlock)



**Check in
(unlock)**



**Check out
(lock)**



(Historical tool) RCS

Note: Still used to manage configuration files

```
sh-3.2$ vi test.txt ← Create a new file
sh-3.2$ ls -a
.      ..      test.txt
sh-3.2$ cat test.txt
RCS example 01
sh-3.2$ ci -u test.txt ← Check in and unlock the file
test.txt,v <-- test.txt
enter description, terminated with single '.' or end of file:
NOTE: This is NOT the log message!
>> Added test.txt
>> .
initial revision: 1.1
done
sh-3.2$ ls -a
.      ..      test.txt test.txt,v ← text.txt,v: RCS history file
```

(Historical tool) RCS

Note: Still used to manage configuration files

```
sh-3.2$ co -l test.txt ← Lock and check out the file
test.txt,v --> test.txt
revision 1.1 (locked)
done
sh-3.2$ vi test.txt ← Edit the file
sh-3.2$ cat test.txt
RCS example 01
Adding a line to the bottom.
sh-3.2$ ci -u test.txt ← Check in and unlock the file
test.txt,v <-- test.txt
new revision: 1.2; previous revision: 1.1
enter log message, terminated with single '.' or end of file:
>> Added a line at the bottom of the file.
>> .
done
sh-3.2$ ls -a
.      ..      test.txt test.txt,v
```

(Historical tool) RCS

Note: Still used to manage configuration files

```
sh-3.2$ rlog test.txt ← Check the commit log
RCS file: test.txt,v
Working file: test.txt
head: 1.2
branch:
locks: strict
access list:
symbolic names:
keyword substitution: kv
total revisions: 2; selected revisions: 2
description:
Added test.txt
-----
revision 1.2
date: 2014/04/18 01:15:56; author: asai; state: Exp; lines: +1 -0
Added a line at the bottom of the file.
-----
revision 1.1
date: 2014/04/18 01:13:57; author: asai; state: Exp;
Initial revision
=====
```

(Historical tool) RCS

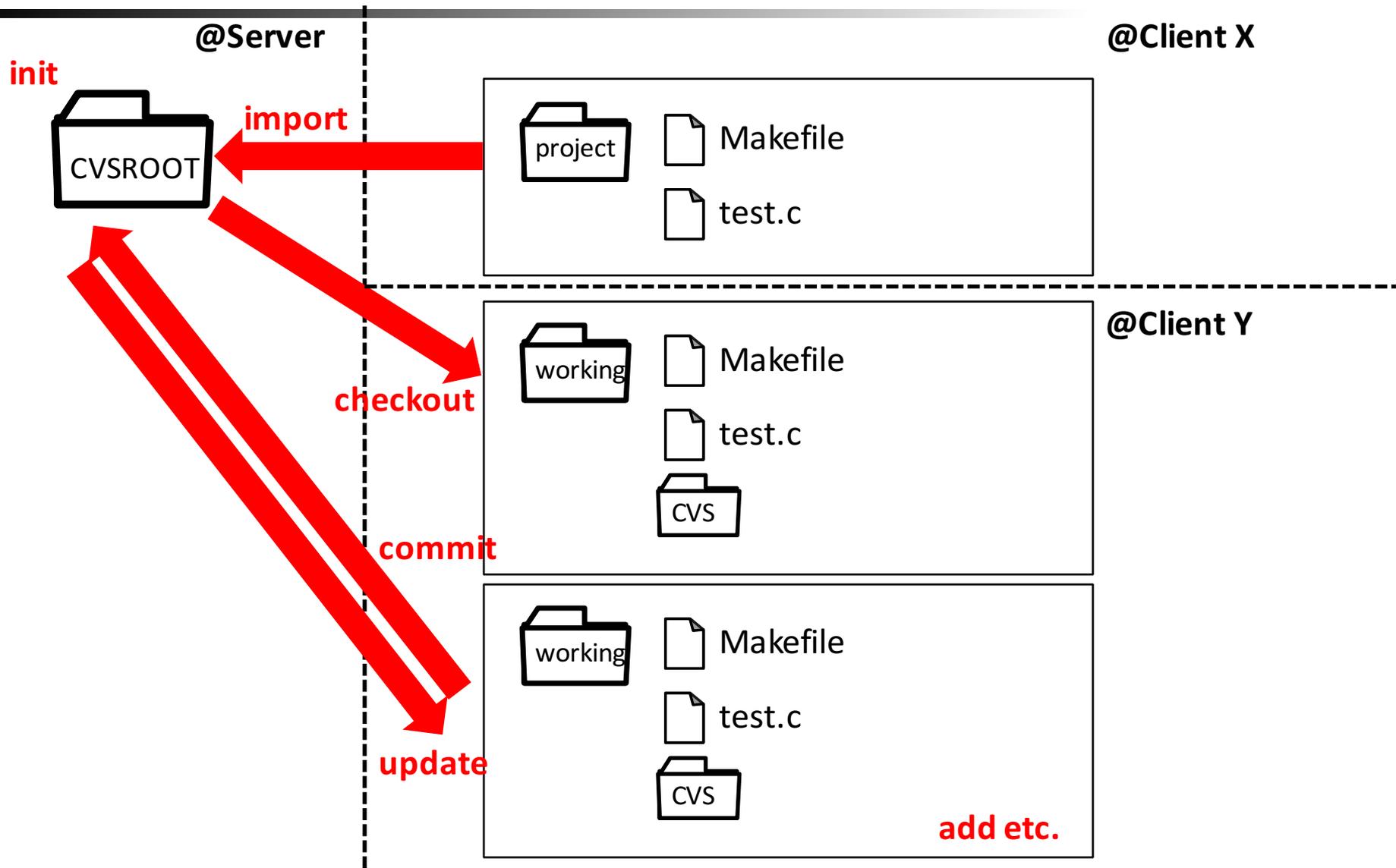
Note: Still used to manage configuration files

```
sh-3.2$ mkdir RCS ← Create RCS directory for history files
sh-3.2$ vi test.txt ← Create a file
sh-3.2$ ls -aR
.      ..      RCS      test.txt

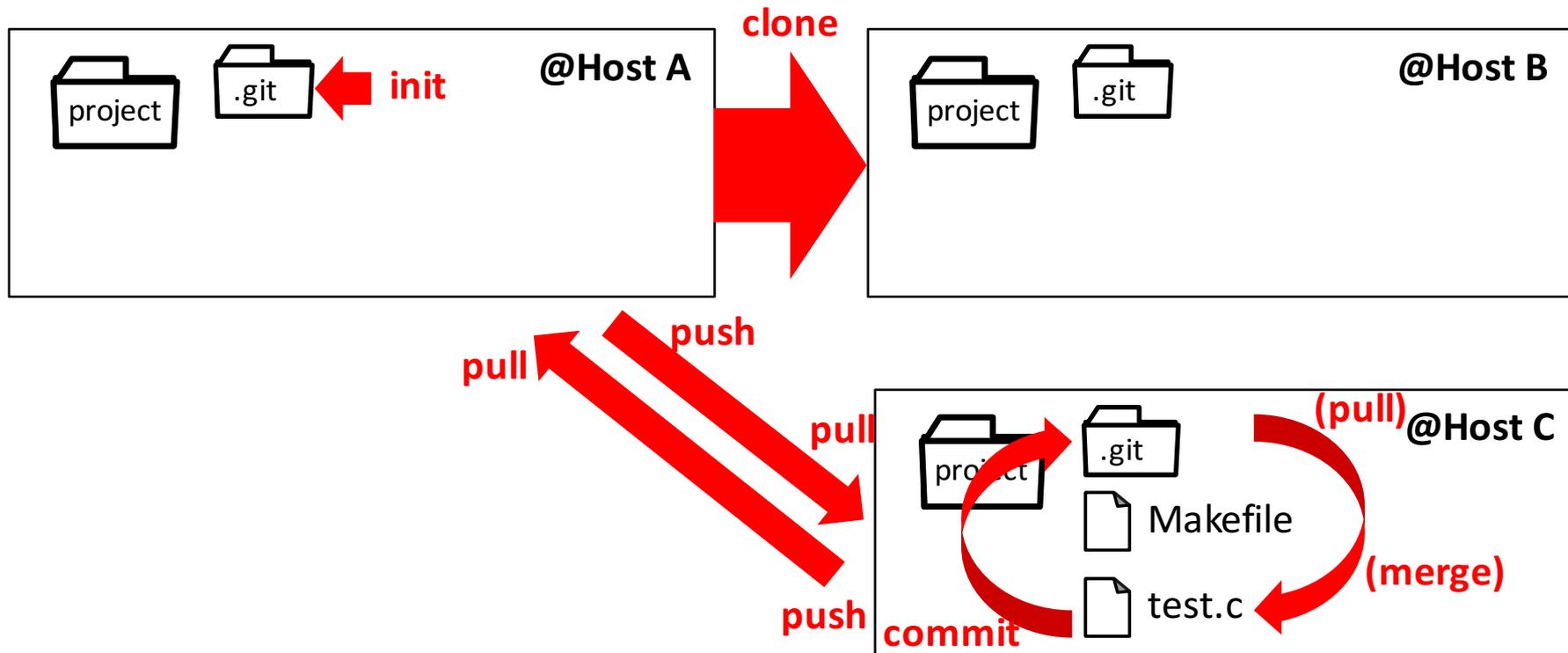
./RCS:
.      ..
sh-3.2$ cat test.txt
RCS example 02
sh-3.2$ ci -u test.txt ← Check in and unlock the file
RCS/test.txt,v <-- test.txt
enter description, terminated with single '.' or end of file:
NOTE: This is NOT the log message!
>> Added test.txt
>> .
initial revision: 1.1
done
sh-3.2$ ls -aR
.      ..      RCS      test.txt

./RCS:
.      ..      test.txt,v ← text.txt,v: RCS history file
```

(Historical tool) CVS/SVN



Mercurial/Git

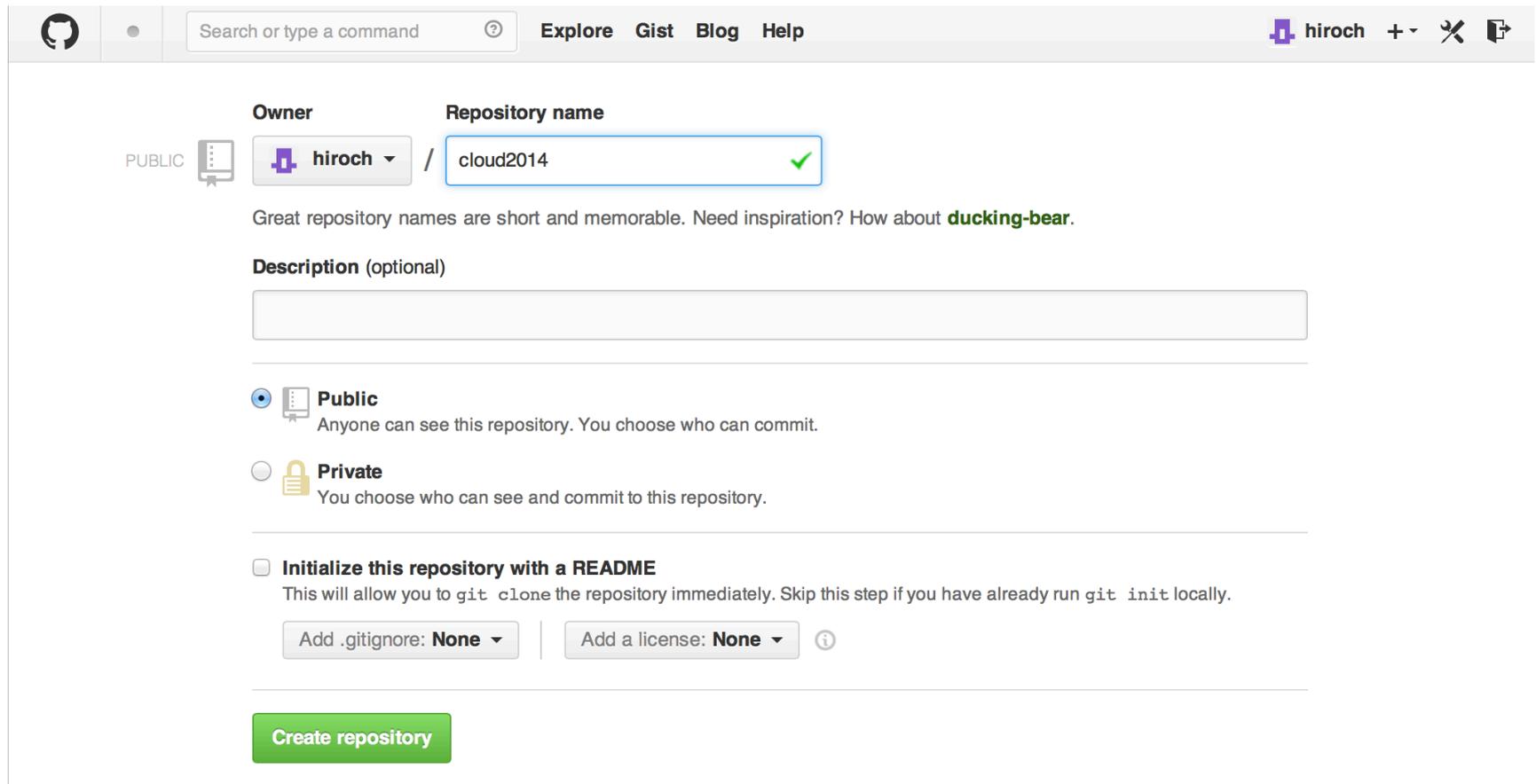


github

- GitHub
 - <https://github.com>
 - Git hosting service
 - Why github is widely used?
 - Social coding
 - Good place for open source projects
 - Pull request
 - Good to do code review
- (about git)
 - <http://git-scm.com/book/en/>

Playing with github

(1) Create a new repository



The screenshot shows the GitHub interface for creating a new repository. At the top, there is a search bar and navigation links for 'Explore', 'Gist', 'Blog', and 'Help'. The user's profile 'hiroch' is visible in the top right corner. The main form is titled 'Create a new repository' and includes the following elements:

- Owner:** A dropdown menu showing 'hiroch' with a purple GitHub icon.
- Repository name:** A text input field containing 'cloud2014' with a green checkmark to its right.
- Visibility:** A 'PUBLIC' label with a lock icon, indicating the repository is public.
- Description (optional):** A large empty text area for entering a description.
- Privacy options:** Two radio buttons: 'Public' (selected) and 'Private'. The 'Public' option is accompanied by a lock icon and the text 'Anyone can see this repository. You choose who can commit.' The 'Private' option is accompanied by a lock icon and the text 'You choose who can see and commit to this repository.'
- Initialization options:** A checkbox labeled 'Initialize this repository with a README'. Below it, there are two dropdown menus: 'Add .gitignore: None' and 'Add a license: None', along with an information icon.
- Create repository button:** A prominent green button at the bottom of the form.

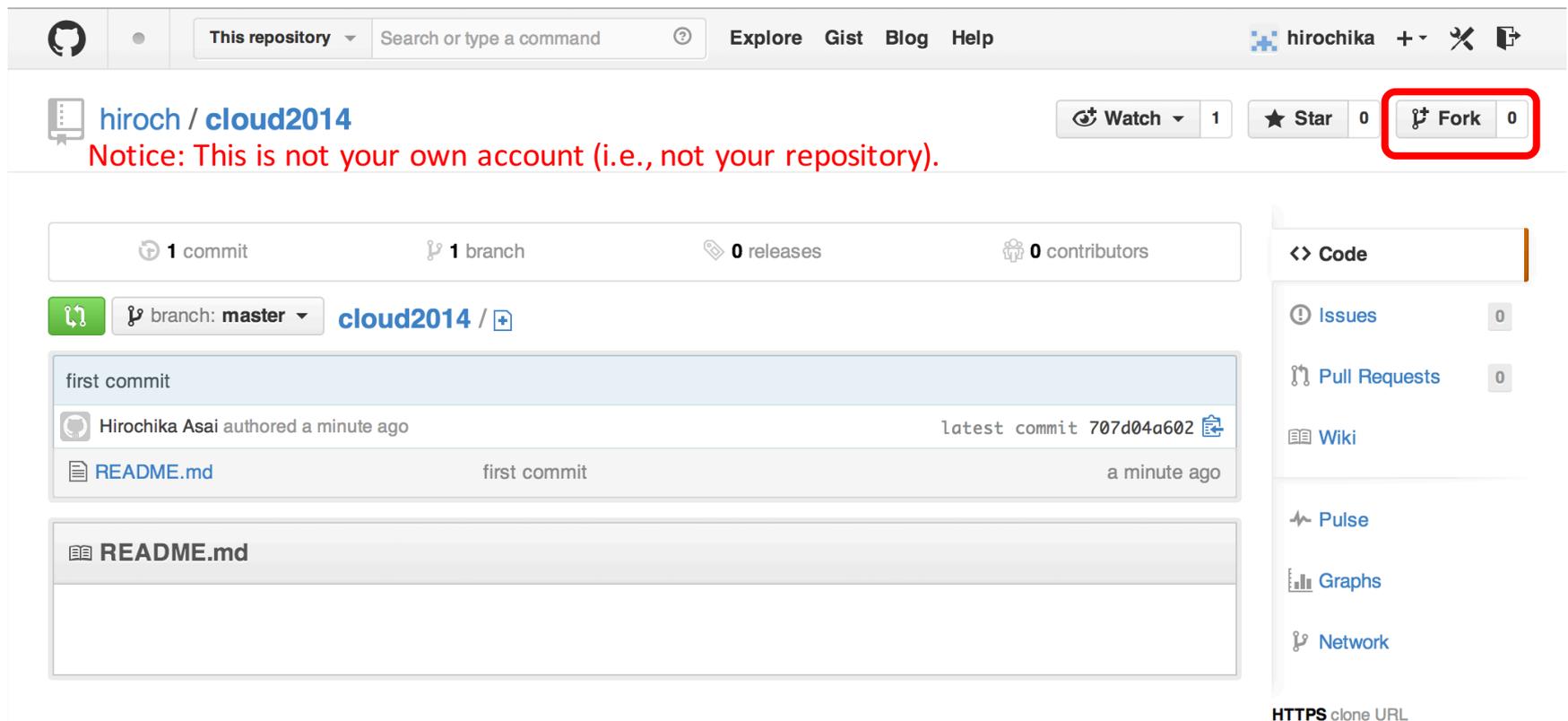
Playing with github

(2) Initialize the repository, add one file, and commit and push this master branch to origin/master at github server

```
sh-3.2$ mkdir myrepos
sh-3.2$ cd myrepos
sh-3.2$ touch README.md
sh-3.2$ git init
Initialized empty Git repository in /path/to/working/directory/myrepos/.git/
sh-3.2$ git add README.md
sh-3.2$ git commit -m "first commit"
[master (root-commit) ce2a2f4] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
sh-3.2$ git remote add origin git@github.com:yourname/myrepos.git
sh-3.2$ git push -u origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 217 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@github.com:yourname/myrepos.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

Playing with github

(3) Fork a repository of another user



This repository Search or type a command Explore Gist Blog Help hirochika

hiroch / **cloud2014** Watch 1 Star 0 Fork 0

Notice: This is not your own account (i.e., not your repository).

1 commit 1 branch 0 releases 0 contributors

branch: master cloud2014

first commit

Hirochika Asai authored a minute ago latest commit 707d04a602

README.md first commit a minute ago

README.md

Code

- Issues 0
- Pull Requests 0
- Wiki
- Pulse
- Graphs
- Network

HTTPS clone URL

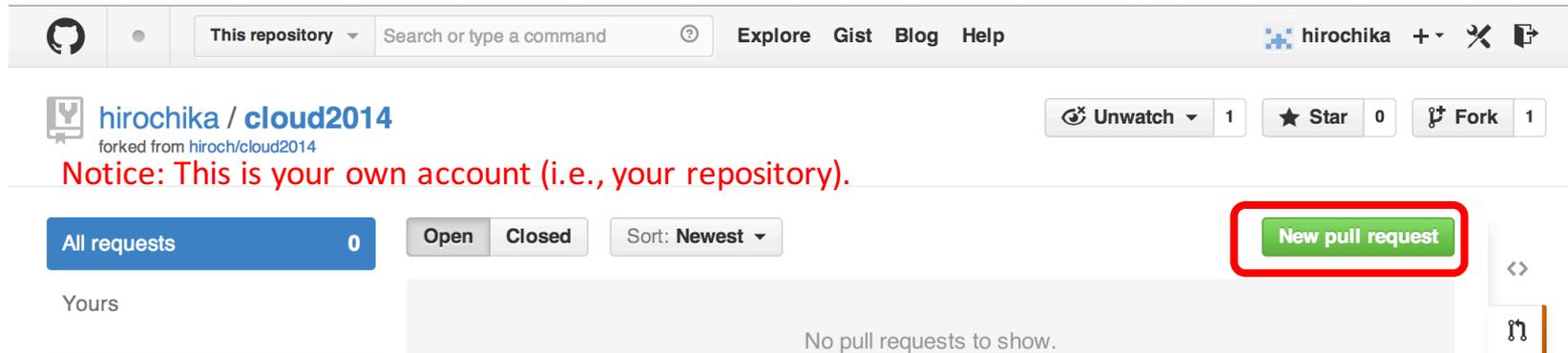
Playing with github

(4) Clone the forked repository, and add one file

```
bash-3.2$ git clone https://github.com/yourname/myrepos.git
Cloning into 'myrepos'...
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0)
bash-3.2$ git add test.txt
bash-3.2$ git commit -m "added a file."
[master ad52f75] added a file.his will be a modification to the forked repository
Committer: Hirochika
 1 file changed, 1 insertion(+)
 create mode 100644 test.txt
bash-3.2$ git push
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 342 bytes, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://hirochika@github.com/hirochika/cloud2014.git
 707d04a..ad52f75  master -> master
```

Playing with github

(5) Create a pull request



The screenshot shows the GitHub interface for a repository named 'cloud2014' by user 'hirochika'. The repository is a fork of 'hiroch/cloud2014'. The page displays the repository name, a search bar, and navigation links. Below the repository name, there are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (1). A red box highlights the 'New pull request' button. Below this, there are filters for 'All requests' (0), 'Open', 'Closed', and 'Sort: Newest'. The main content area shows 'Yours' and a message 'No pull requests to show.'.

hirochika / cloud2014
forked from hiroch/cloud2014

Notice: This is your own account (i.e., your repository).

All requests 0 Open Closed Sort: Newest

New pull request

Yours

No pull requests to show.

Playing with github

(5) Create a pull request

The screenshot shows a GitHub pull request interface. At the top, the repository is identified as 'hirochika / cloud2014', which is a fork of 'hiroch/cloud2014'. It shows 1 watch, 0 stars, and 1 fork. Below this, a comparison is shown between 'hiroch:master' and 'hirochika:master', with the repository names highlighted in a red box. A green 'Create Pull Request' button is also highlighted in a red box. The pull request description reads: 'Open a Pull Request for this comparison to discuss and review your changes with others.' Below the description, it shows '1 commit', '1 file changed', '0 comments', and '1 contributor'. The commit history shows a commit by 'Hirochika Asai' on 'Apr 18, 2014' with the message 'added a file.' and commit ID 'ad52f75'. A diff view for 'test.txt' shows a single addition: '+This will be a modification to the forked repository'. At the bottom, it states 'No commit comments for this range'.

hirochika / cloud2014
forked from hiroch/cloud2014

Unwatch 1 Star 0 Fork 1

Original repository from your repository

hiroch:master ... hirochika:master Edit

Create Pull Request Open a Pull Request for this comparison to discuss and review your changes with others.

1 commit 1 file changed 0 comments 1 contributor

Apr 18, 2014

Hirochika Asai added a file. ad52f75

Showing 1 changed file with 1 addition and 0 deletions. Show diff stats

1 test.txt Open View

```
... @@ -0,0 +1 @@
1 +This will be a modification to the forked repository
```

No commit comments for this range

Playing with github

(6) Accept or Reject pull requests

The screenshot displays the GitHub interface for a pull request. At the top, the repository name is `hiroch / cloud2014`. The pull request is titled `#1` and contains the message `added a file. Please use this patch!` by user `hirochika` from the `hirochika:master` branch. The interface includes a search bar, repository statistics (1 Unwatch, 0 Star, 1 Fork), and a list of pull requests with filters for 'Open' and 'Closed' status, and a 'Sort: Newest' dropdown. A 'New pull request' button is visible on the right side.

Playing with github

(6) Accept or Reject pull requests

The screenshot shows a GitHub pull request interface. At the top, there's a navigation bar with 'This repository' dropdown, a search bar, and links for 'Explore', 'Gist', 'Blog', and 'Help'. The repository name 'hiroch / cloud2014' is displayed, along with 'Unwatch' (1), 'Star' (0), and 'Fork' (1) buttons. The pull request title is 'added a file. #1'. Below the title, it says 'hirochika wants to merge 1 commit into hiroch:master from hirochika:master'. There are buttons for 'Open', 'Conversation' (0), 'Commits' (1), and 'Files changed' (1). A comment from 'hirochika' says 'Please use this patch!'. Below the comment, there's a commit 'added a file.' with hash 'ad52f75'. A green box highlights the 'Merge pull request' button. On the right, there are sections for 'Labels', 'Milestone', 'Assignee', and 'Notifications'. The 'Merge pull request' button is highlighted with a red box.

added a file. #1

Open hirochika wants to merge 1 commit into hiroch:master from hirochika:master

Conversation 0 Commits 1 Files changed 1

hirochika commented a minute ago

Please use this patch!

added a file. ad52f75

This pull request can be automatically merged. You can also merge branches on the [command line](#).

Merge pull request

Labels: None yet

Milestone: No milestone

Assignee: No one assigned

Notifications: Unsubscribe

You're receiving notifications because you're subscribed to this repository.

Playing with github

(7) Follow the original repository at the forked repository

```
bash-3.2$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
bash-3.2$ git remote add upstream https://github.com/original_committer/original_repos.git
bash-3.2$ git fetch upstream
remote: Counting objects: 7, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 4 (delta 0)
Unpacking objects: 100% (4/4), done.
From https://github.com/original_committer/original_repos
 * [new branch]      master      -> upstream/master
bash-3.2$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/upstream/master
```

↓ Add original repository as “upstream” branch

← Fetch the upstream branch

Playing with github

(7) Follow the original repository at the forked repository

```
bash-3.2$ ls
README.md test.txt typescript
bash-3.2$ cat test.txt
This will be a modification to the forked repository
bash-3.2$ git merge upstream/master ← Merge the upstream branch to current branch
Updating ad52f75..ce34d8f
Fast-forward
 test.txt | 1 +
 1 file changed, 1 insertion(+)
bash-3.2$ cat test.txt
This will be a modification to the forked repository
Added a line.
bash-3.2$ git push ← Push it to the forked repository
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 558 bytes, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://hirochika@github.com/yourname/myrepos.git
 ad52f75..ce34d8f master -> master
```

Playing with github

- Do it with other students