docker容器技術課程

GitLab CI 應用

Philipz
鄭淳尹

home

ModernWeb'17

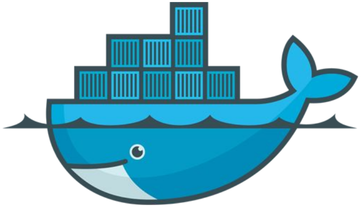
401 演講廳（8/11）

Track F

3:30~13:55	**Docker + CI pipeline 的高效率 ChatBot 開發方法** 鄭淳尹（Philipz）/ 臺北榮總
13:55~14:20	**Snapshot Everything - 從元件測試點燃新一代測試觀念**

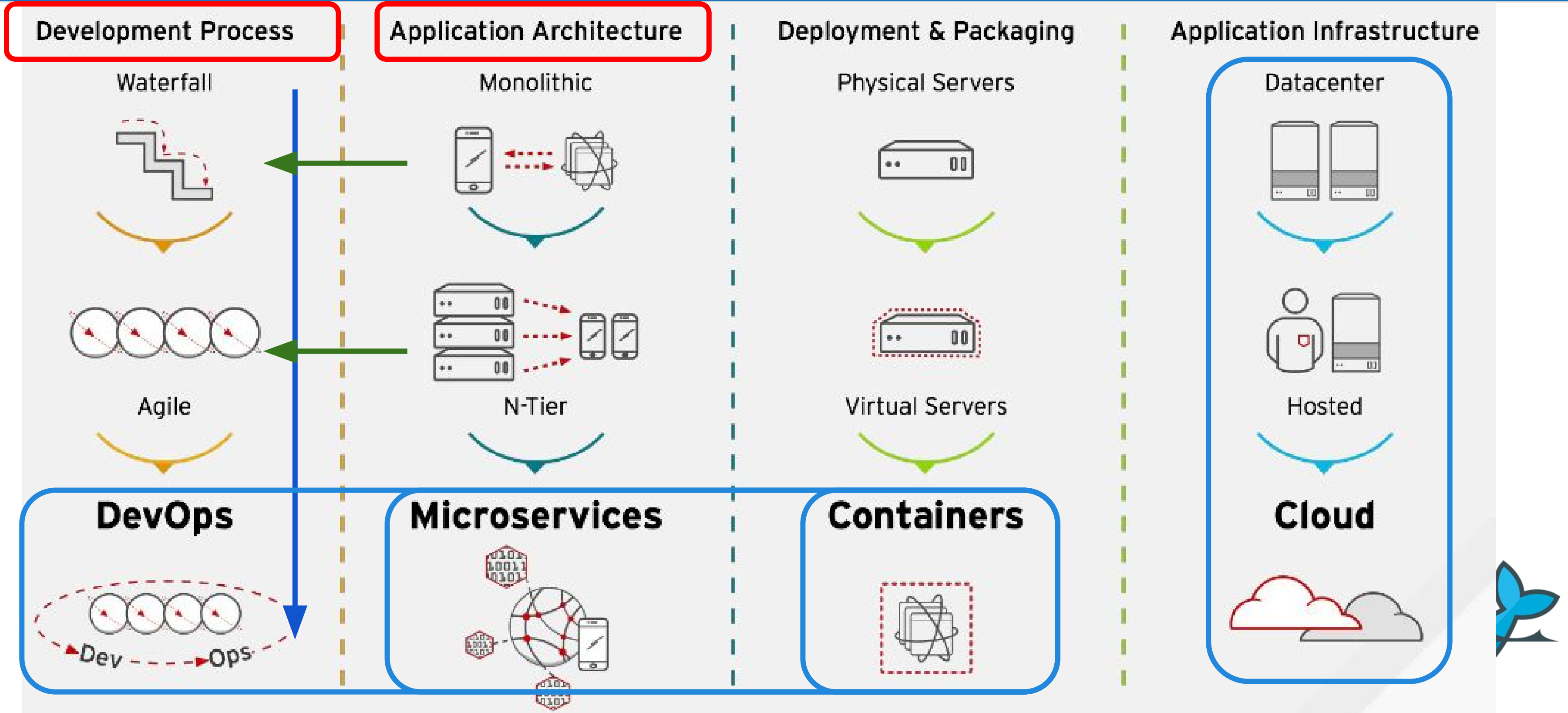
課程大綱

1. GitLab 簡介
2. GitLab CI 使用說明
3. GitLab CI 與雲端 Azure PaaS 整合
4. GitLab CI 與資料科學整合
5. GitLab CI 與 IoT 整合
6. GitLab CI 與 K8S 整合
7. 結語



1. GitLab 簡介

容器式系統架構



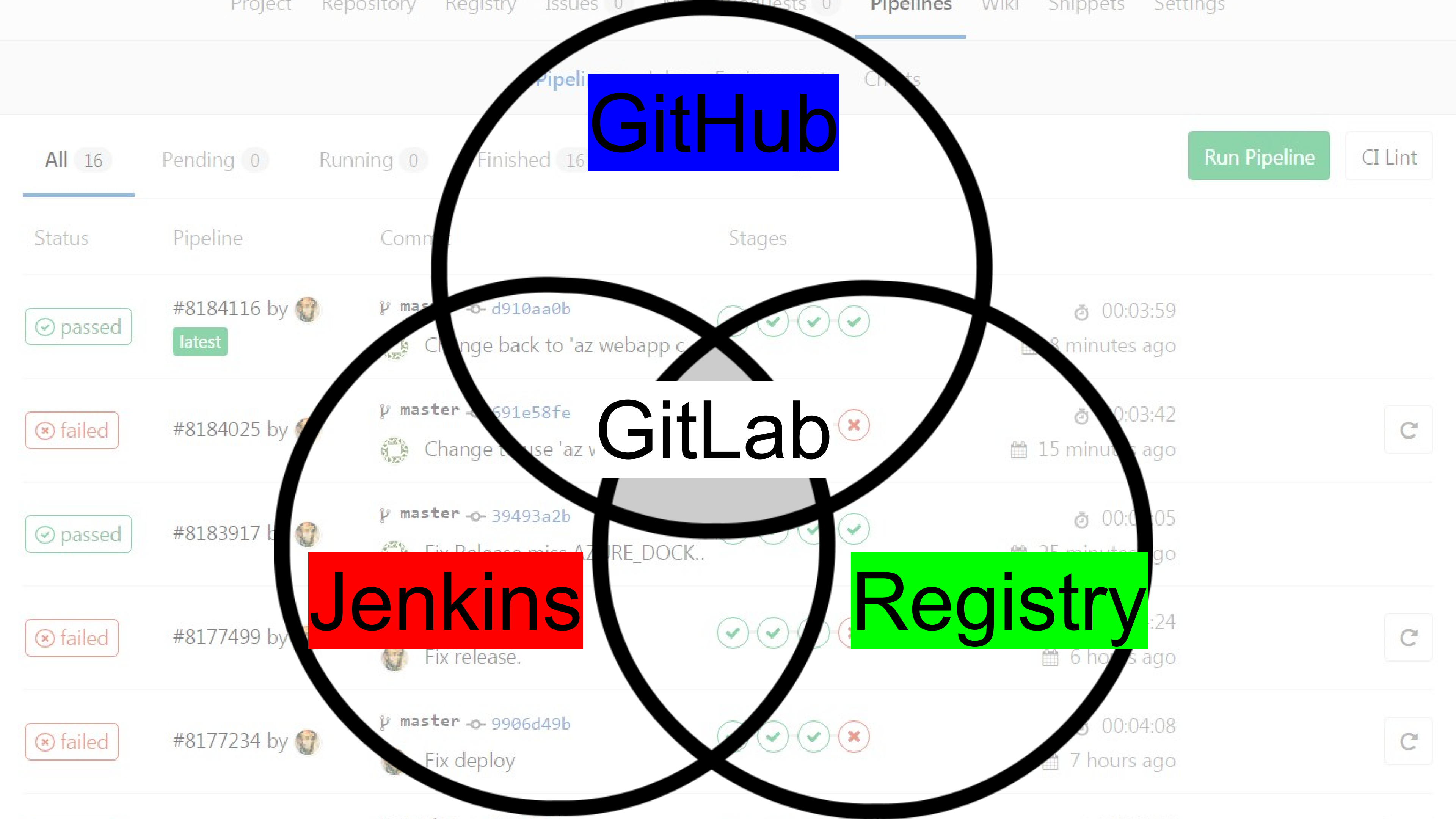
Development Process
Waterfall
Agile
DevOps
Dev
Ops
Application Architecture
Monolithic
N-Tier
Microservices
Deployment & Packaging
Physical Servers
Virtual Servers
Containers
Application Infrastructure
Datacenter
Hosted
Cloud

持續整合(Continuous Integration)

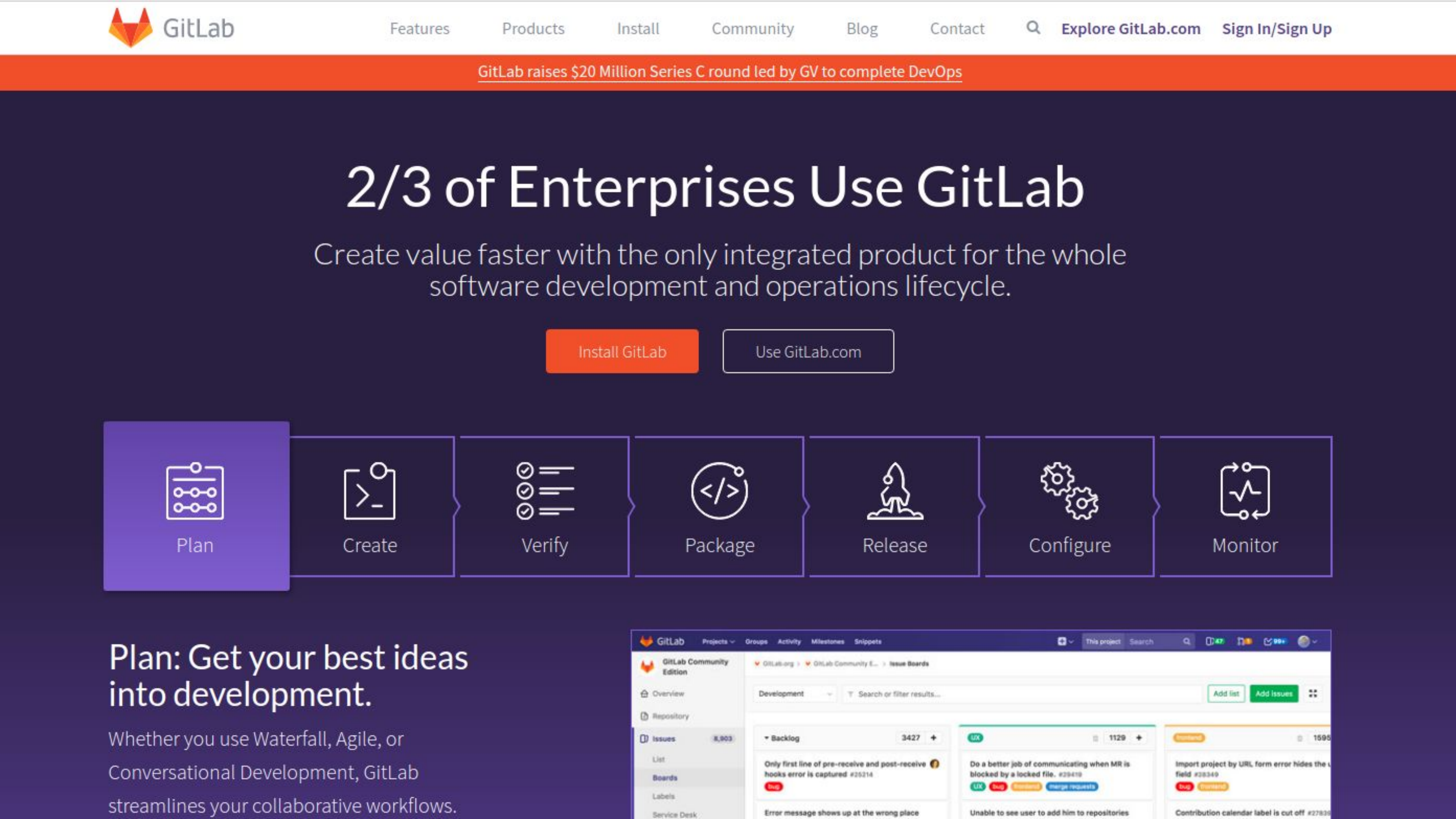
- 虛擬機方式
  - Jenkins
  - TravisCI
  - 舊式、肥大
- 容器方式
  - GitLab
  - CircleCI
  - 新式、輕量

解決軟體開發長久以來常見的痛

Docker更容易實現基礎架構程式化



Status	Pipeline	Comm	Stages	
passed	#8184116 by latest	mas d910aa0b Change back to 'az webapp c		00:03:59 8 minutes ago
failed	#8184025 by	master 691e58fe Change to use 'az w		0:03:42 15 minutes ago
passed	#8183917 b	master 39493a2b Fix Release miss AZ RE_DOCK..		00:0 05 25 minutes ago
failed	#8177499 by	Fix release.		:24 6 ho s ago
failed	#8177234 by	master 9906d49b Fix deploy		00:04:08 7 hours ago



GitLab raises $20 Million Series C round led by GV to complete DevOps

2/3 of Enterprises Use GitLab

Create value faster with the only integrated product for the whole software development and operations lifecycle.

Plan: Get your best ideas into development.

Whether you use Waterfall, Agile, or Conversational Development, GitLab streamlines your collaborative workflows.

Only first line of pre-receive and post-receive hooks error is captured #25214 bug

Error message shows up at the wrong place

Do a better job of communicating when MR is blocked by a locked file. #29419 UX bug frontend merge requests

Unable to see user to add him to repositories

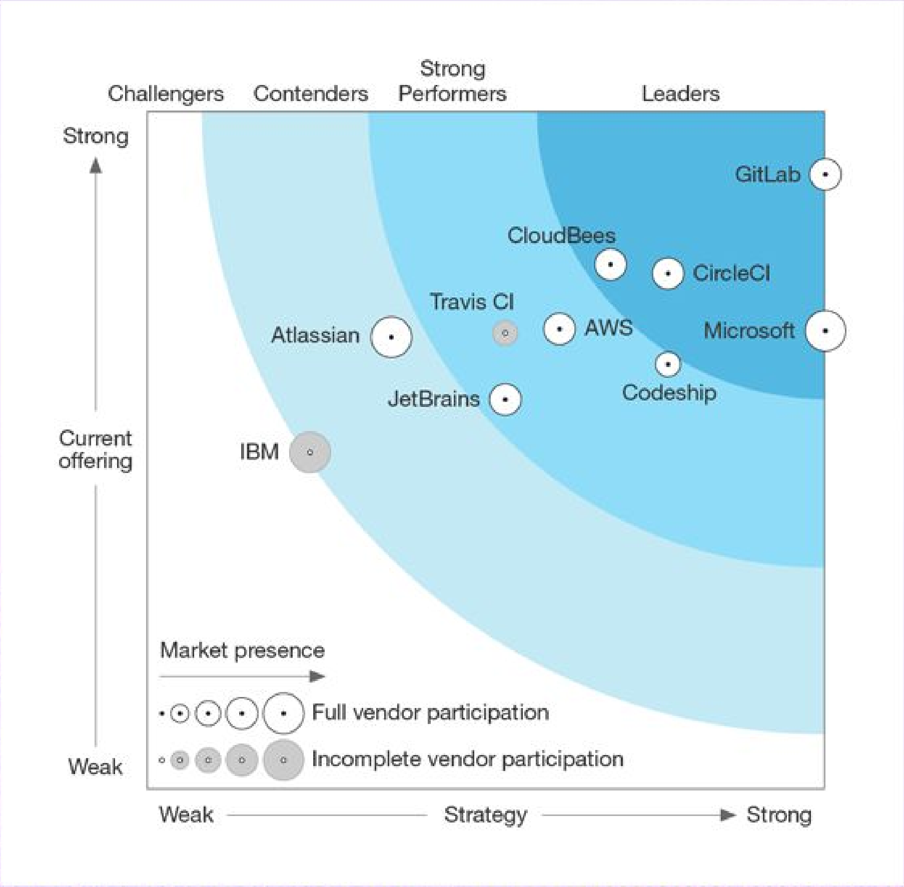
Import project by URL form error hides the u field #28349 bug frontend

Contribution calendar label is cut off #2783

GitLab named a Leader in the Forrester Wave™

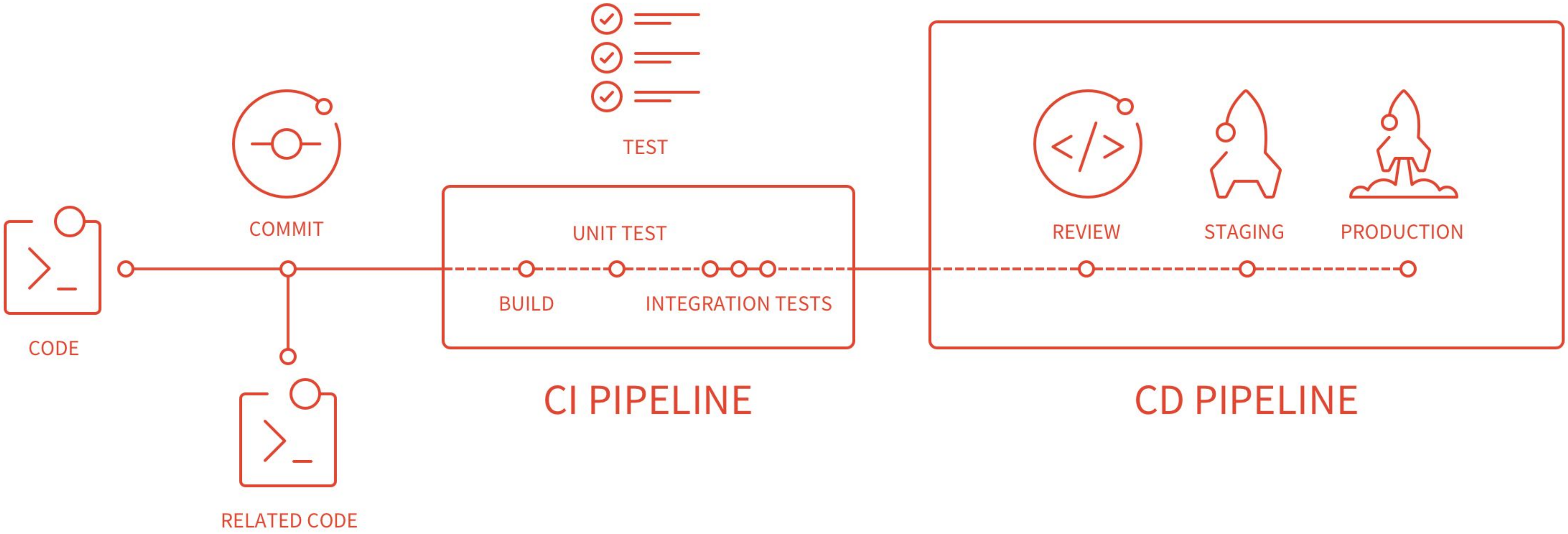
Receiving the highest score in Forrester's Current Offering evaluation, GitLab was named as a Leader in Continuous Integration in The Forrester Wave™: Continuous Integration Tools, Q3 2017 report. According to the report, "GitLab delivers ease of use, scalability, integration, and innovation."

Read the Report



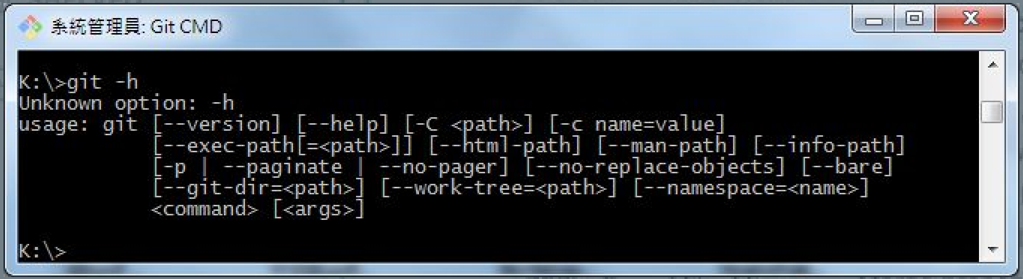
2. GitLab CI 使用說明

持續整合(CI)/持續交付(CD)



Install Git

- sudo apt-get install git
- Git cmd for windows
- SourceTree is best choice!
- GitHub is a git web-UI and repository.

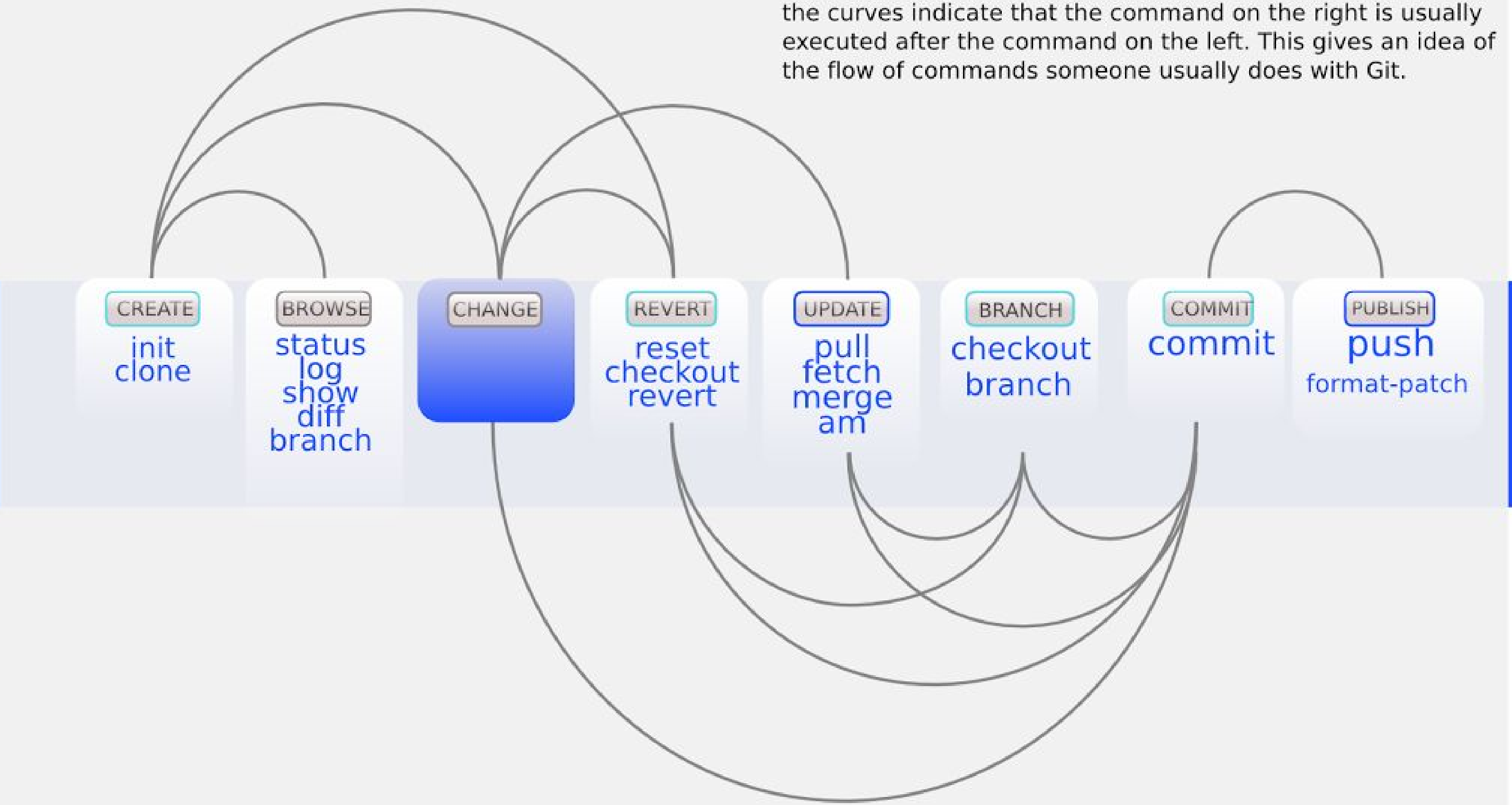


系統管理員: Git CMD

```
K:\>git -h
Unknown option: -h
usage: git [--version] [--help] [-C <path>] [-c name=value]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]

K:\>
```

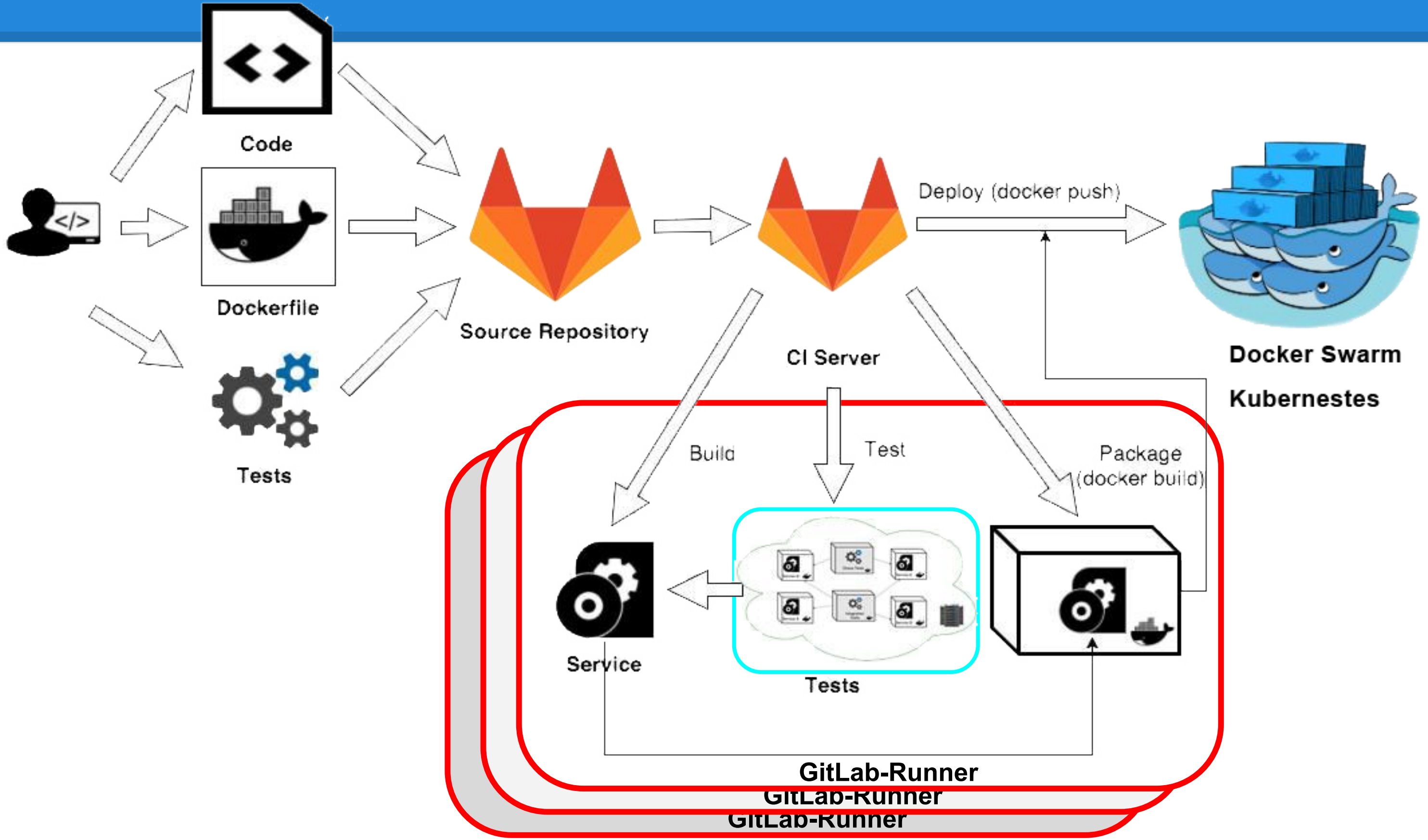
Commands Sequence

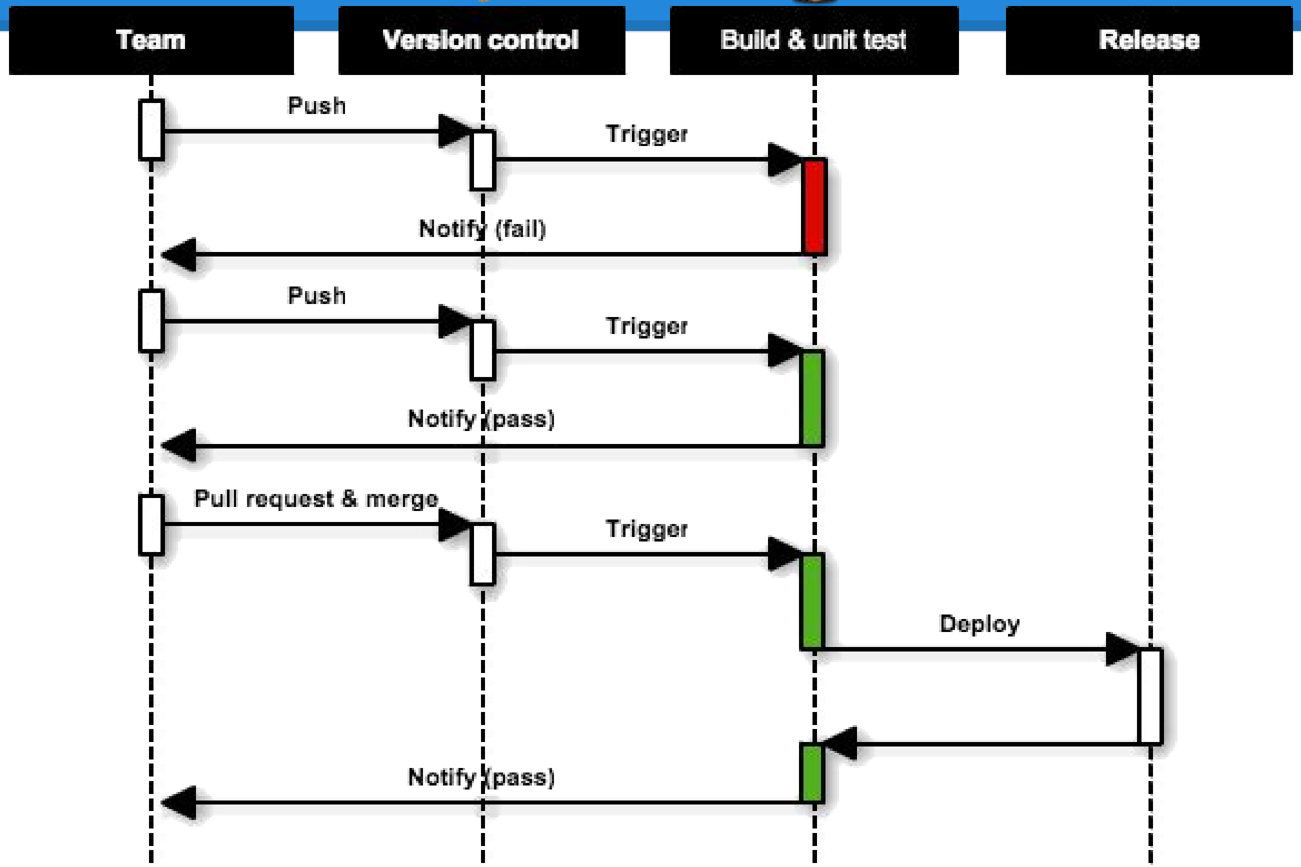


the curves indicate that the command on the right is usually executed after the command on the left. This gives an idea of the flow of commands someone usually does with Git.

of HEAD

容器開發流程


Code
Dockerfile
Tests
Source Repository
CI Server
Deploy (docker push)
Docker Swarm
Kubernestes
Build
Test
Package
(docker build)
Service
Tests
GitLab-Runner
GitLab-Runner
GitLab-Runner



Team

Version control

Build & unit test

Release

Push

Trigger

Notify (fail)

Push

Trigger

Notify (pass)

Pull request & merge

Trigger

Deploy

Notify (pass)

GitLab CI YAML

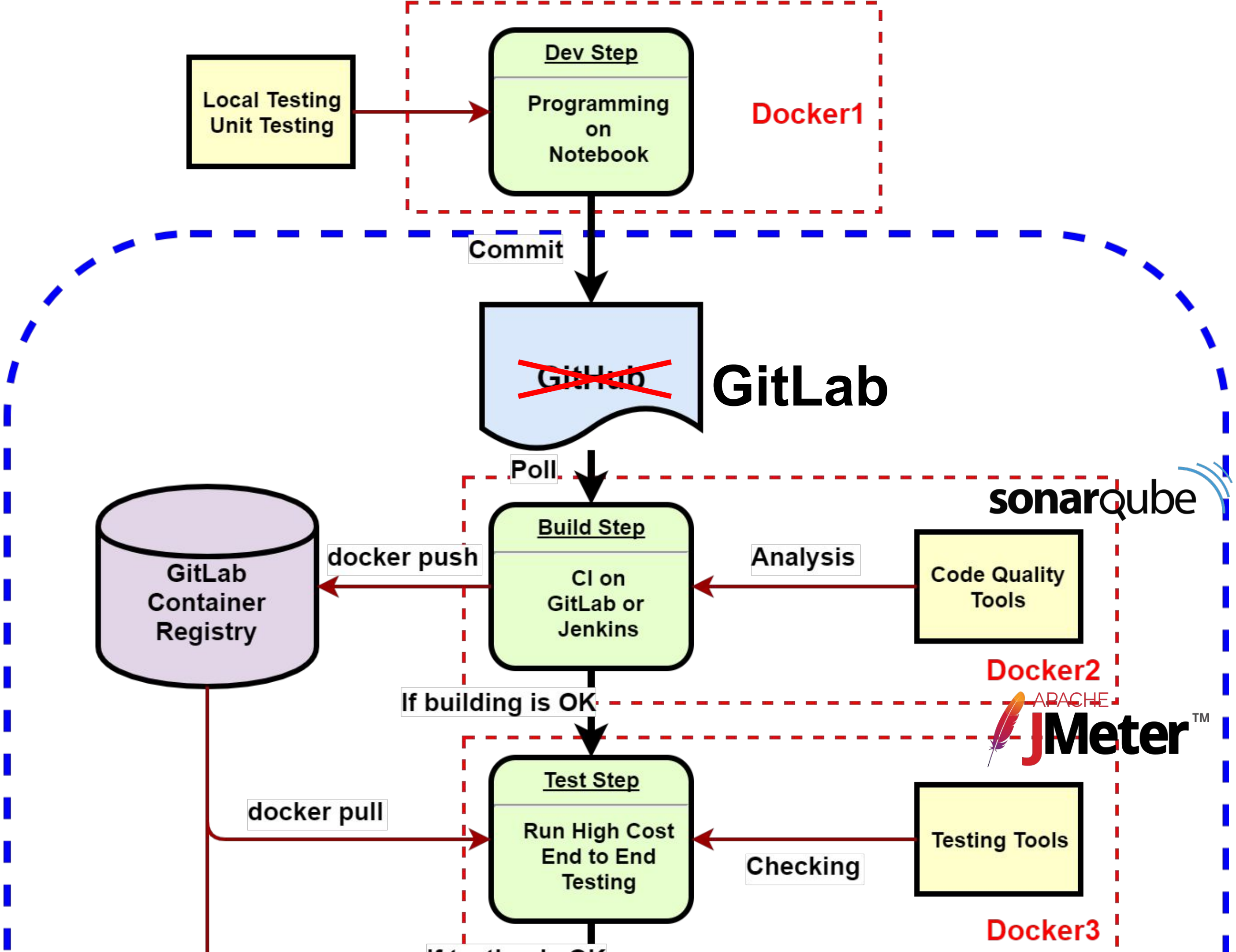
```yaml
image: ruby:2.1
services:
  - postgres
before_script:
  - bundle install
after_script:
  - rm secrets
```

```yaml
stages:
  - build
  - test
  - deploy
job1:
  stage: build
  script:
    - execute-script-for-job1
  only:
    - master
```

GitLab CI Examples, Docker

用 GitLab CI 进行持续集成

3. GitLab CI 與雲端 Azure PaaS 整合



Local Testing
Unit Testing

Dev Step

Programming
on
Notebook

Docker1

Commit

~~GitHub~~

GitLab

Poll

sonarqube

Build Step

CI on
GitLab or
Jenkins

Analysis

Code Quality
Tools

Docker2

docker push

GitLab
Container
Registry

If building is OK

APACHE JMeter™

Test Step

Run High Cost
End to End
Testing

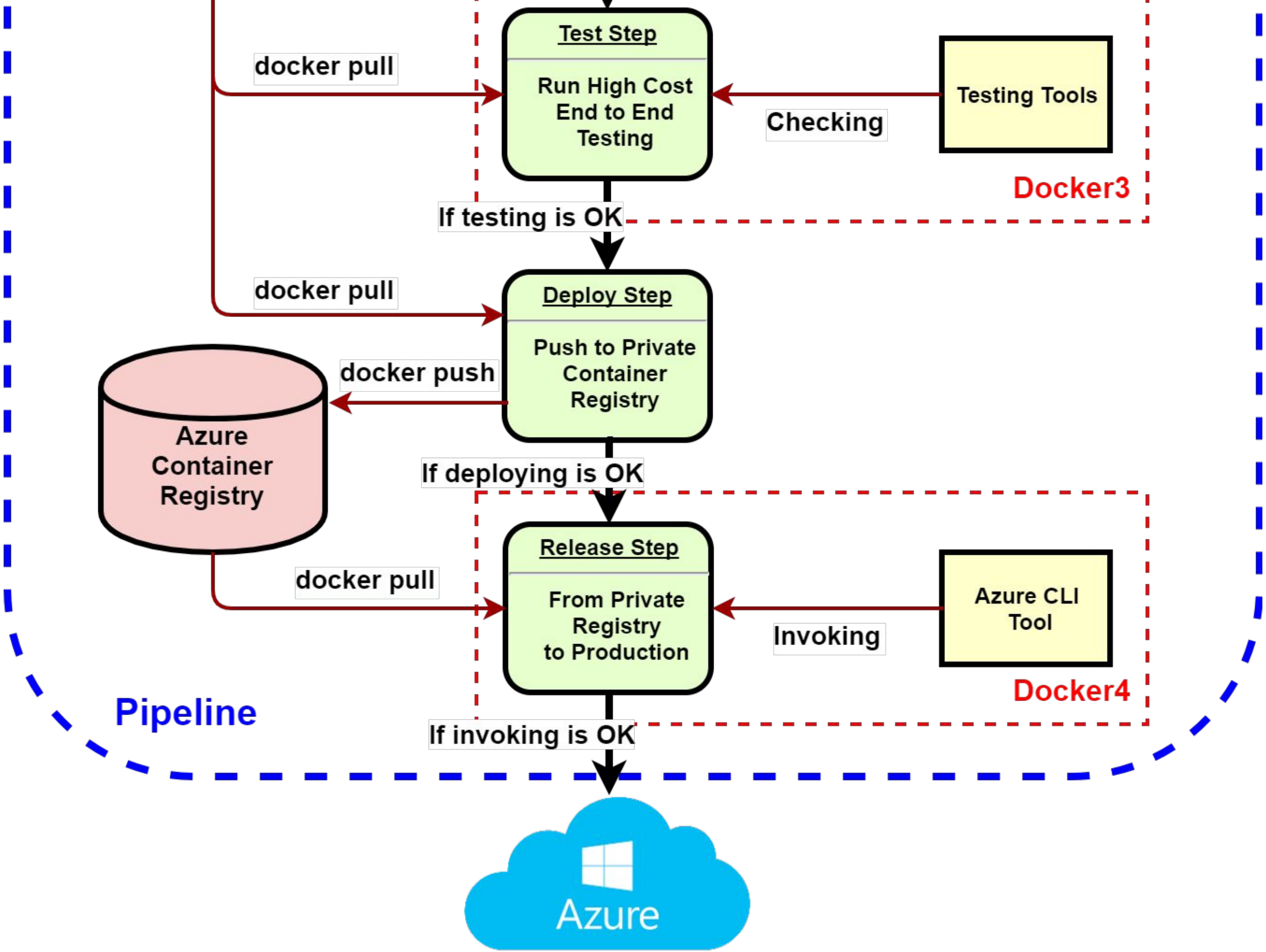
docker pull

Checking

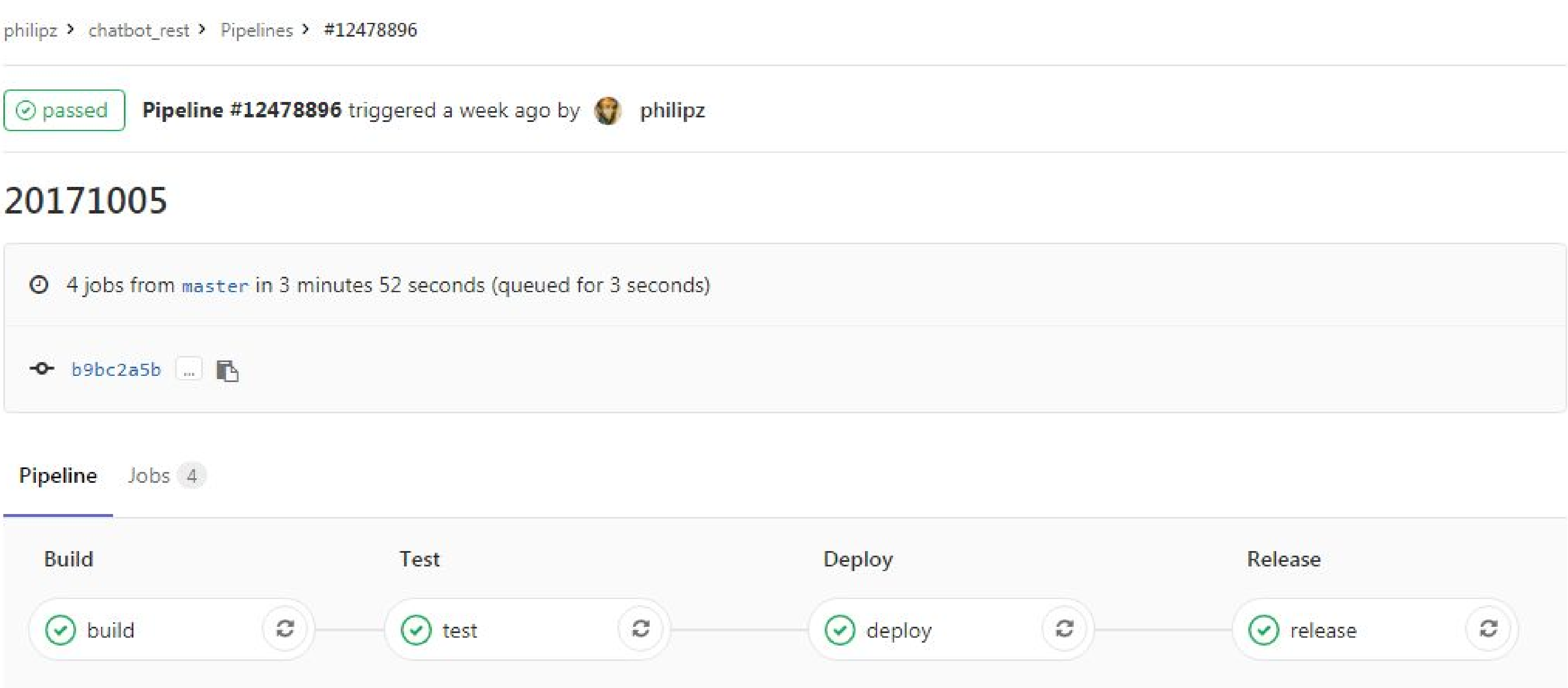
Testing Tools

Docker3

If testing is OK


Test Step
Run High Cost End to End Testing
docker pull
Testing Tools
Checking
Docker3
If testing is OK
docker pull
Deploy Step
Push to Private Container Registry
docker push
Azure Container Registry
If deploying is OK
Release Step
From Private Registry to Production
docker pull
Azure CLI Tool
Invoking
Docker4
Pipeline
If invoking is OK
Azure

Azure Pipeline



philipz › chatbot_rest › Pipelines › #12478896

✓ passed **Pipeline #12478896** triggered a week ago by philipz

20171005

4 jobs from `master` in 3 minutes 52 seconds (queued for 3 seconds)

`b9bc2a5b` ...

.gitlab-ci.yml (build)

```
image: docker:latest
services:
  - docker:dind
stage:
  - build
  - test
  - deploy
  - release
```

```
before_script:
  - docker info
build:
  stage: build
  script:
   - docker build . -t registry.gitlab.com
   - docker login -u gitlab-ci-token -p $T
   - docker push registry.gitlab.com/$IMG
```

.gitlab-ci.yml (test)

```
test:
  stage: test
  script:
   - docker login -u gitlab-ci-token -p $CI_Token registry.gitlab.com
   - docker pull registry.gitlab.com/$USER/$IMG:$TAG
   - docker network create wrktest
   - docker run -d -p 1337:1337 --name app --network wrktest $IMG
   - sleep 5
   - docker run --name wrk --network wrktest --rm williamyeh/wrk -t2
-c5 -d5s --timeout 2s http://app:1337/ > test/result.txt
   - apk add --update bash bc
   - cd test && cat result.txt && ./test.sh
```

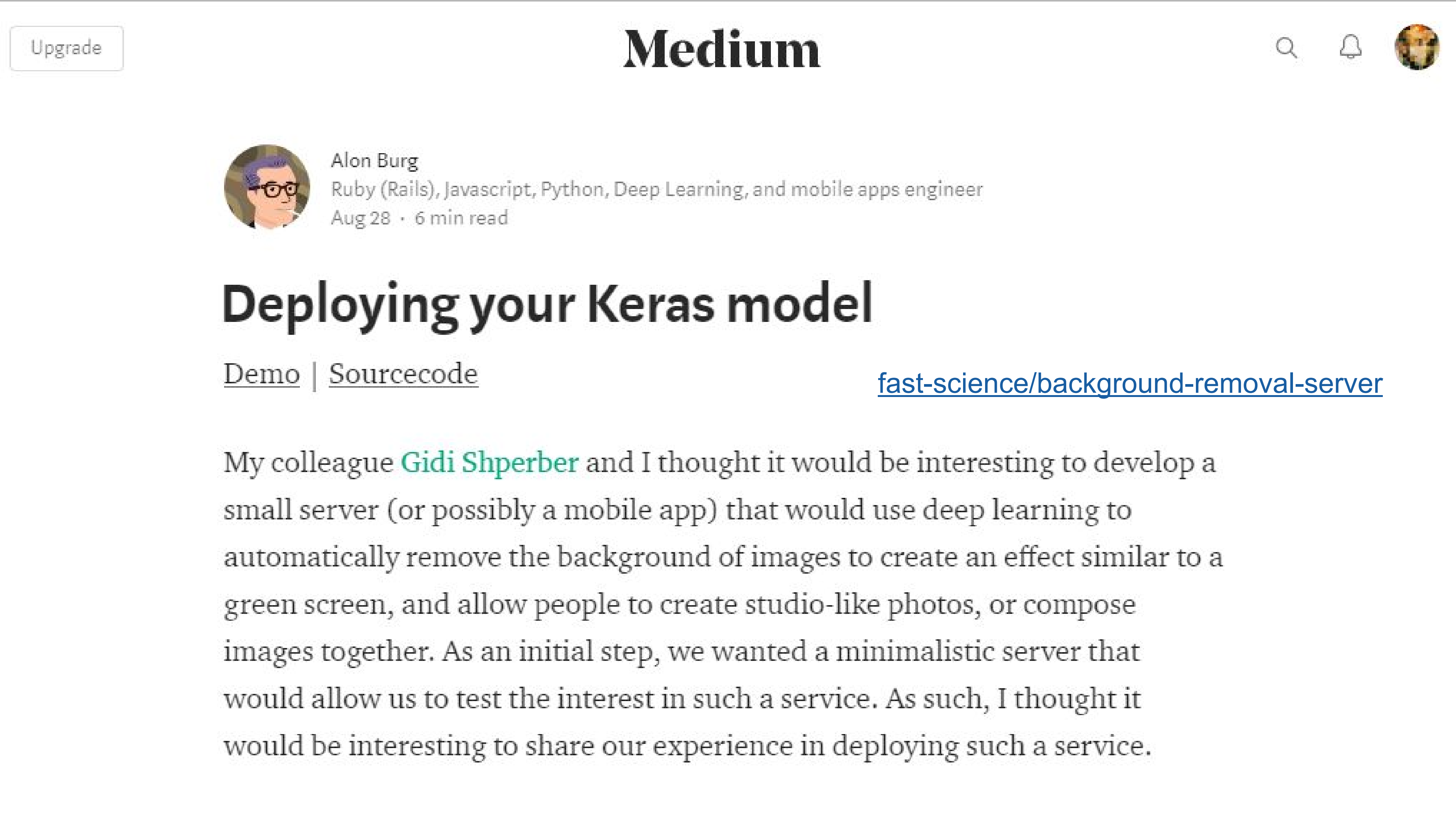
.gitlab-ci.yml (deploy)

```yaml
deploy:
 stage: deploy
 script:
  - docker login -u gitlab-ci-token -p $CI_Token registry.gitlab.com
  - docker pull registry.gitlab.com/$USER/$IMG:$TAG
  - docker login -u azure -p $AZURE_DOCKER_PASS dockware.azurecr.io
  - docker tag registry.gitlab.com/$USR/$IMG dockware.azurecr.io/$IMG
  - docker push dockware.azurecr.io/$IMG
```

.gitlab-ci.yml (release)

```
release:
  stage: release
  script:
   - docker pull azuresdk/azure-cli-python:0.2.8
   - docker run -t --rm -v $(pwd)/release.sh:/release.sh -e
AZURE_LOGIN_USER=$AZURE_LOGIN_USER -e AZURE_PASSWORD=$AZURE_PASSWORD
-e AZURE_TENANT=$AZURE_TENANT -e DOCKER_IMAGE_NAME=$DOCKER_IMAGE_NAME
-e DOCKER_IMAGE_TAG=$DOCKER_IMAGE_TAG -e
AZURE_REG_PASSWORD=$AZURE_DOCKER_PASS azuresdk/azure-cli-python:0.2.8
/release.sh
```

4. GitLab CI 與資料科學



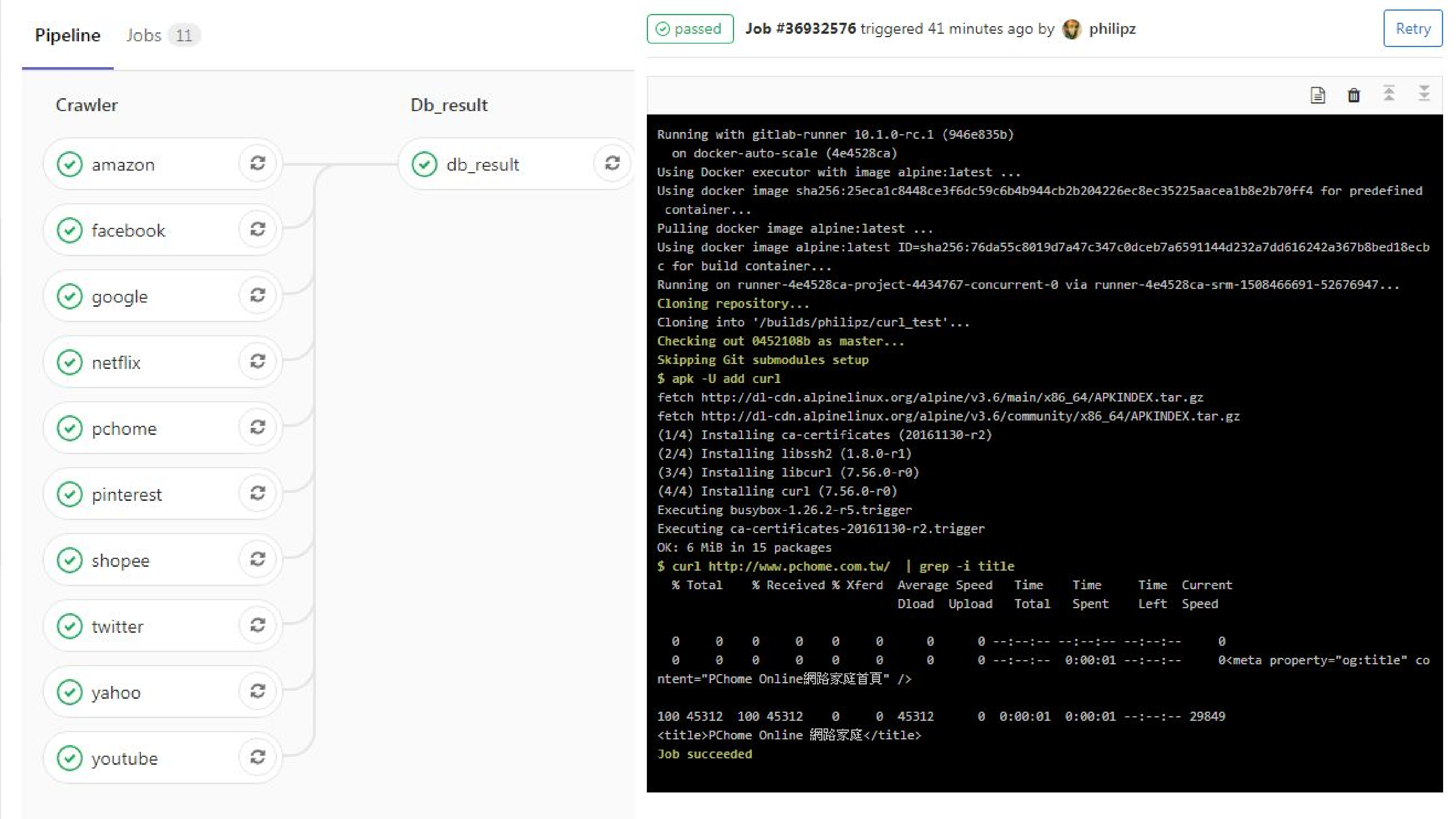
Alon Burg
Ruby (Rails), Javascript, Python, Deep Learning, and mobile apps engineer
Aug 28 · 6 min read

Deploying your Keras model

Demo | Sourcecode

fast-science/background-removal-server

My colleague Gidi Shperber and I thought it would be interesting to develop a small server (or possibly a mobile app) that would use deep learning to automatically remove the background of images to create an effect similar to a green screen, and allow people to create studio-like photos, or compose images together. As an initial step, we wanted a minimalistic server that would allow us to test the interest in such a service. As such, I thought it would be interesting to share our experience in deploying such a service.



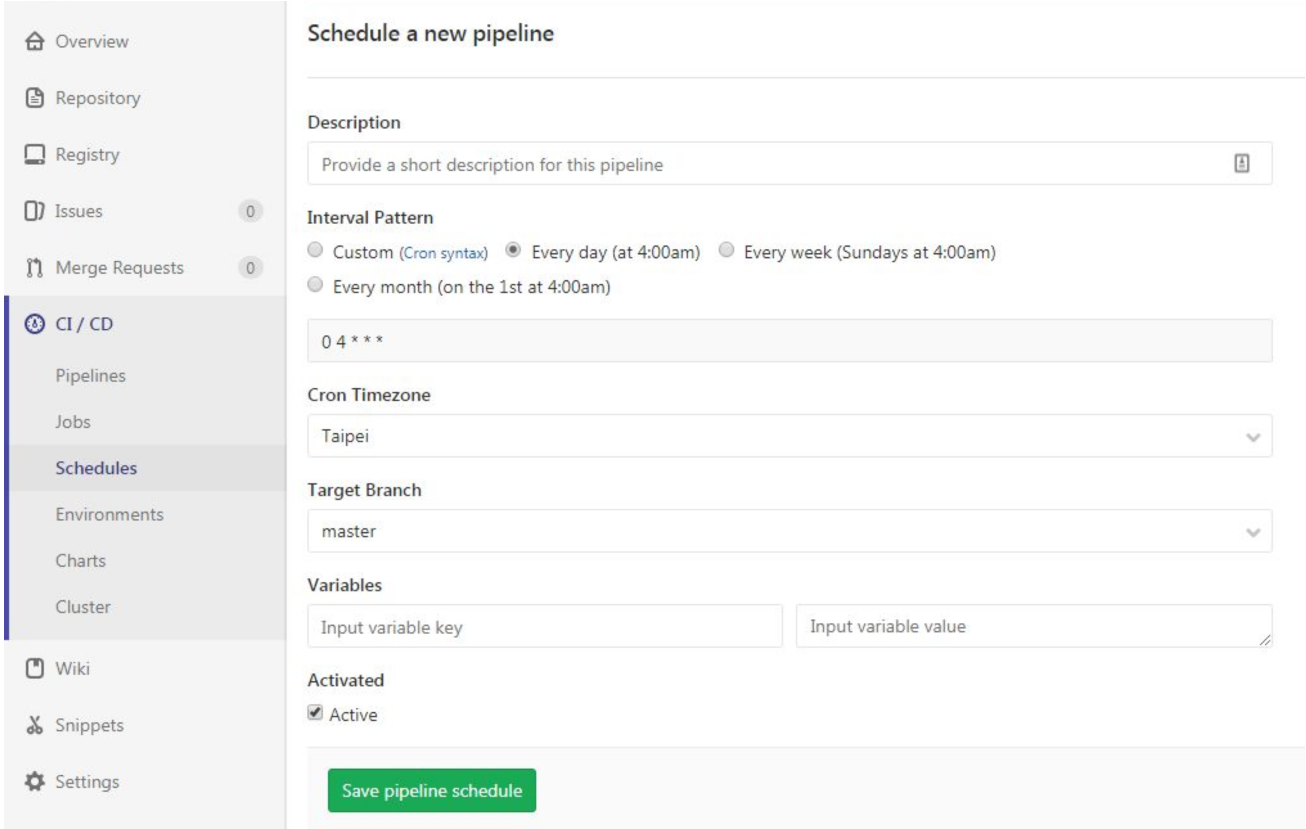
Job #36932576 triggered 41 minutes ago by philipz

```
Running with gitlab-runner 10.1.0-rc.1 (946e835b)
  on docker-auto-scale (4e4528ca)
Using Docker executor with image alpine:latest ...
Using docker image sha256:25eca1c8448ce3f6dc59c6b4b944cb2b204226ec8ec35225aacea1b8e2b70ff4 for predefined
 container...
Pulling docker image alpine:latest ...
Using docker image alpine:latest ID=sha256:76da55c8019d7a47c347c0dceb7a6591144d232a7dd616242a367b8bed18ecb
c for build container...
Running on runner-4e4528ca-project-4434767-concurrent-0 via runner-4e4528ca-srm-1508466691-52676947...
Cloning repository...
Cloning into '/builds/philipz/curl_test'...
Checking out 0452108b as master...
Skipping Git submodules setup
$ apk -U add curl
fetch http://dl-cdn.alpinelinux.org/alpine/v3.6/main/x86_64/APKINDEX.tar.gz
fetch http://dl-cdn.alpinelinux.org/alpine/v3.6/community/x86_64/APKINDEX.tar.gz
(1/4) Installing ca-certificates (20161130-r2)
(2/4) Installing libssh2 (1.8.0-r1)
(3/4) Installing libcurl (7.56.0-r0)
(4/4) Installing curl (7.56.0-r0)
Executing busybox-1.26.2-r5.trigger
Executing ca-certificates-20161130-r2.trigger
OK: 6 MiB in 15 packages
$ curl http://www.pchome.com.tw/  | grep -i title
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed

  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0     0    0     0    0     0      0      0 --:--:--  0:00:01 --:--:--     0<meta property="og:title" co
ntent="PChome Online網路家庭首頁" />

100 45312  100 45312    0     0  45312      0  0:00:01  0:00:01 --:--:-- 29849
<title>PChome Online 網路家庭</title>
Job succeeded
```

Pipeline Schedules



Schedule a new pipeline

Description

Provide a short description for this pipeline

Interval Pattern

○ Custom (Cron syntax) ● Every day (at 4:00am) ○ Every week (Sundays at 4:00am)
○ Every month (on the 1st at 4:00am)

0 4 * * *

Cron Timezone

Taipei

Target Branch

master

Variables

Input variable key | Input variable value

Activated

☑ Active

Save pipeline schedule

5. GitLab CI 與 IoT 整合

Docker Autobuild

Public Docker Hub

Private Docker Registry, Distribution

Building ARM containers on any x86 machine, even DockerHub

GitHub source code

Resin.io - IoT DevOps platform

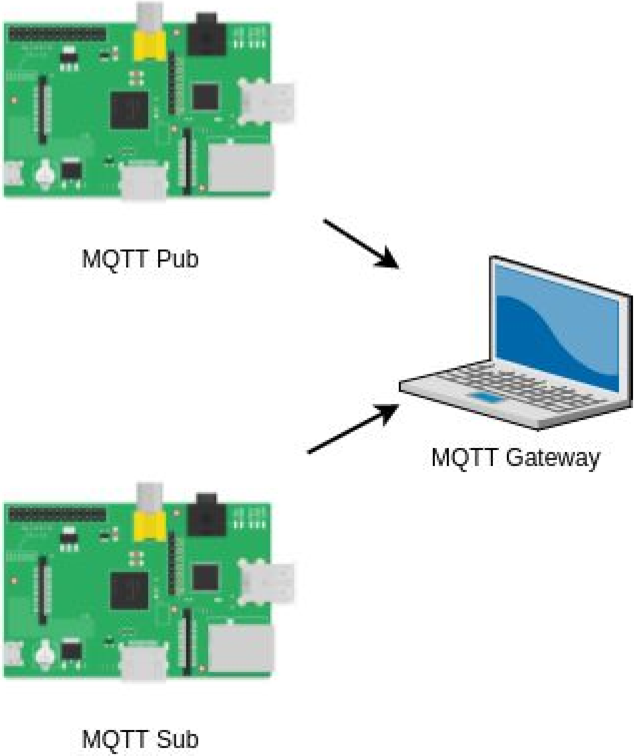
resin.io

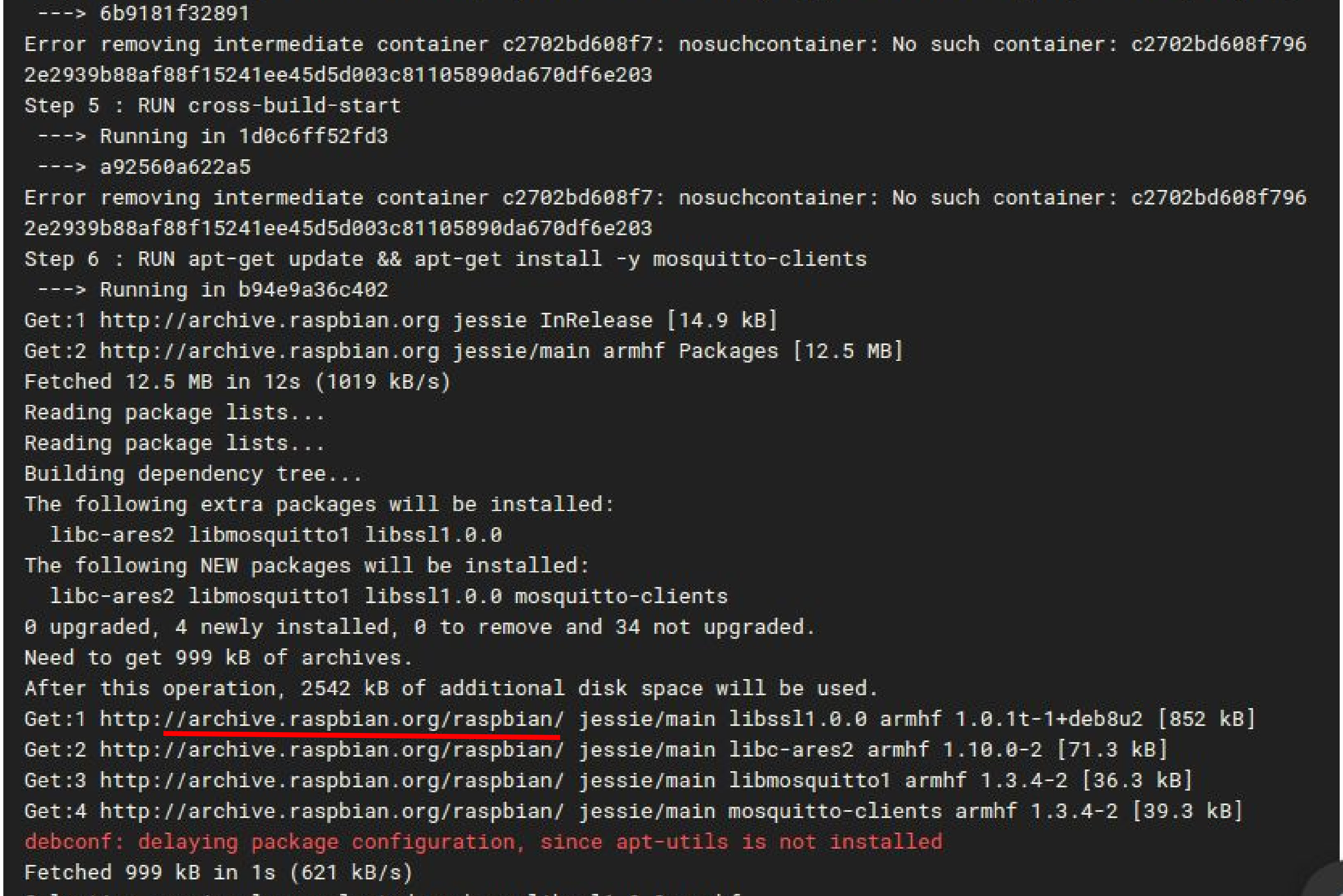
Only One Command

docker-compose.yml

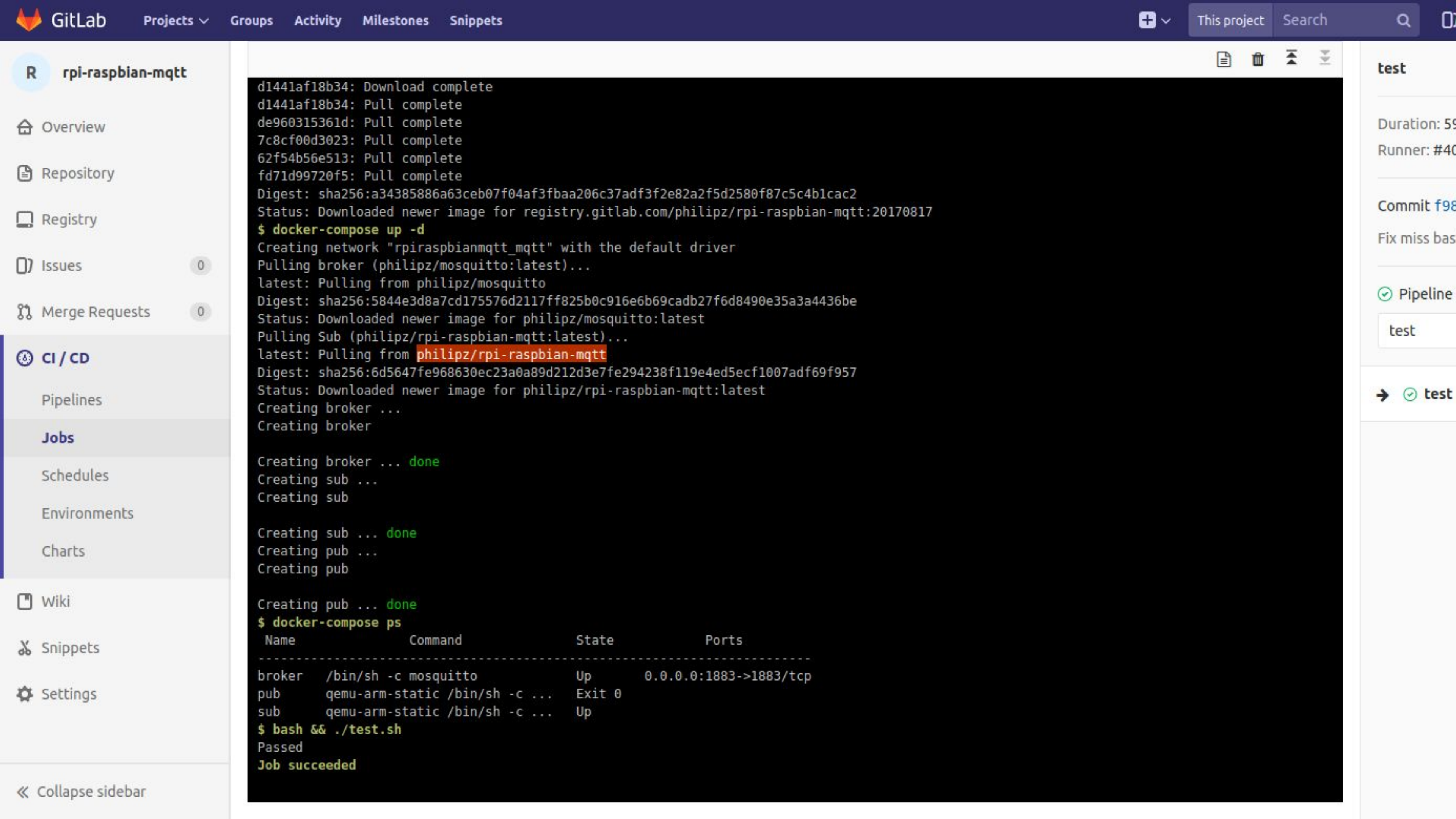
&

docker-compose up





```
 ---> 6b9181f32891
Error removing intermediate container c2702bd608f7: nosuchcontainer: No such container: c2702bd608f796
2e2939b88af88f15241ee45d5d003c81105890da670df6e203
Step 5 : RUN cross-build-start
 ---> Running in 1d0c6ff52fd3
 ---> a92560a622a5
Error removing intermediate container c2702bd608f7: nosuchcontainer: No such container: c2702bd608f796
2e2939b88af88f15241ee45d5d003c81105890da670df6e203
Step 6 : RUN apt-get update && apt-get install -y mosquitto-clients
 ---> Running in b94e9a36c402
Get:1 http://archive.raspbian.org jessie InRelease [14.9 kB]
Get:2 http://archive.raspbian.org jessie/main armhf Packages [12.5 MB]
Fetched 12.5 MB in 12s (1019 kB/s)
Reading package lists...
Reading package lists...
Building dependency tree...
The following extra packages will be installed:
  libc-ares2 libmosquitto1 libssl1.0.0
The following NEW packages will be installed:
  libc-ares2 libmosquitto1 libssl1.0.0 mosquitto-clients
0 upgraded, 4 newly installed, 0 to remove and 34 not upgraded.
Need to get 999 kB of archives.
After this operation, 2542 kB of additional disk space will be used.
Get:1 http://archive.raspbian.org/raspbian/ jessie/main libssl1.0.0 armhf 1.0.1t-1+deb8u2 [852 kB]
Get:2 http://archive.raspbian.org/raspbian/ jessie/main libc-ares2 armhf 1.10.0-2 [71.3 kB]
Get:3 http://archive.raspbian.org/raspbian/ jessie/main libmosquitto1 armhf 1.3.4-2 [36.3 kB]
Get:4 http://archive.raspbian.org/raspbian/ jessie/main mosquitto-clients armhf 1.3.4-2 [39.3 kB]
debconf: delaying package configuration, since apt-utils is not installed
Fetched 999 kB in 1s (621 kB/s)
```



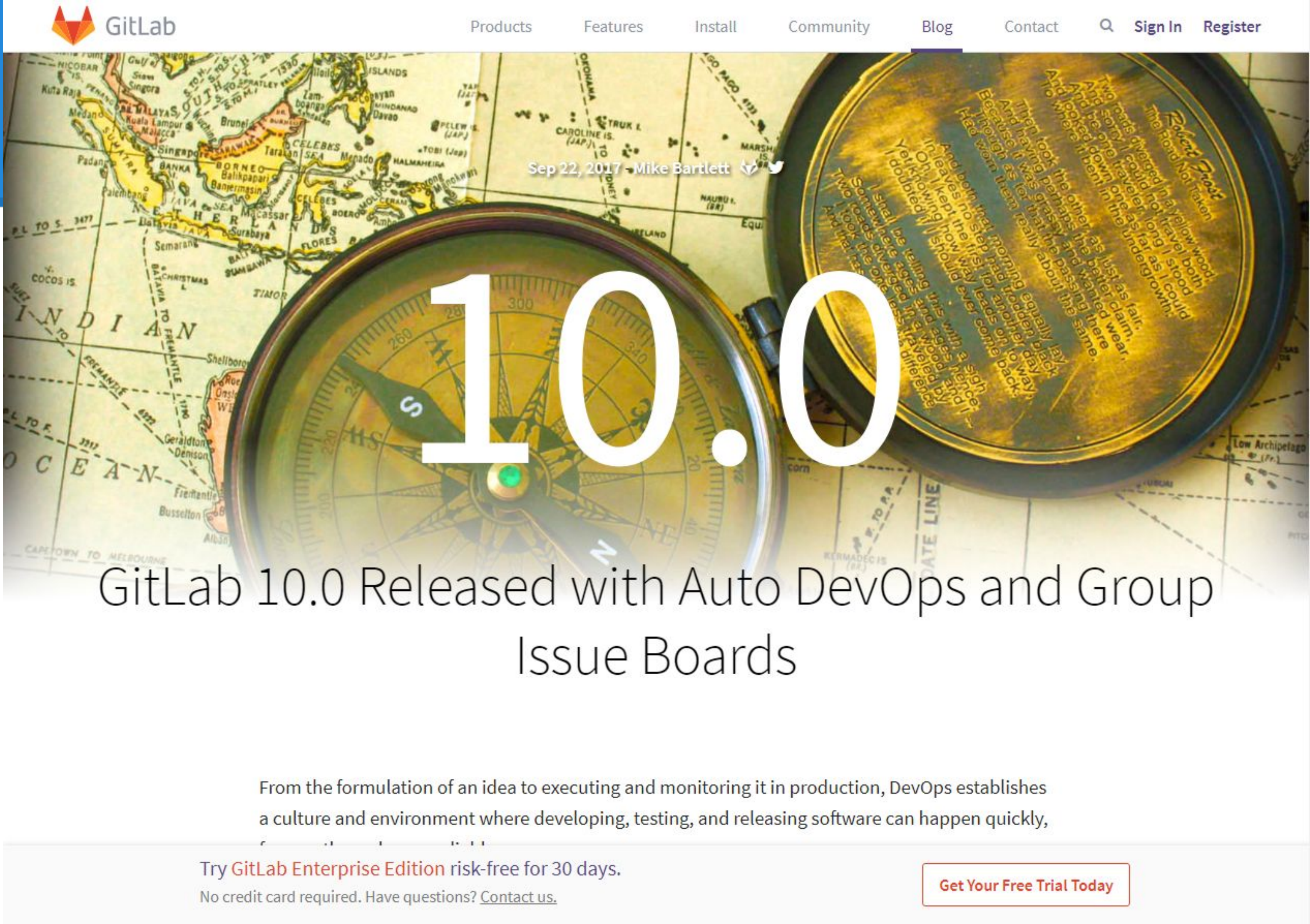
```
d1441af18b34: Download complete
d1441af18b34: Pull complete
de960315361d: Pull complete
7c8cf00d3023: Pull complete
62f54b56e513: Pull complete
fd71d99720f5: Pull complete
Digest: sha256:a34385886a63ceb07f04af3fbaa206c37adf3f2e82a2f5d2580f87c5c4b1cac2
Status: Downloaded newer image for registry.gitlab.com/philipz/rpi-raspbian-mqtt:20170817
$ docker-compose up -d
Creating network "rpiraspbianmqtt_mqtt" with the default driver
Pulling broker (philipz/mosquitto:latest)...
latest: Pulling from philipz/mosquitto
Digest: sha256:5844e3d8a7cd175576d2117ff825b0c916e6b69cadb27f6d8490e35a3a4436be
Status: Downloaded newer image for philipz/mosquitto:latest
Pulling Sub (philipz/rpi-raspbian-mqtt:latest)...
latest: Pulling from philipz/rpi-raspbian-mqtt
Digest: sha256:6d5647fe968630ec23a0a89d212d3e7fe294238f119e4ed5ecf1007adf69f957
Status: Downloaded newer image for philipz/rpi-raspbian-mqtt:latest
Creating broker ...
Creating broker

Creating broker ... done
Creating sub ...
Creating sub

Creating sub ... done
Creating pub ...
Creating pub

Creating pub ... done
$ docker-compose ps
 Name               Command               State            Ports
-------------------------------------------------------------------------
broker   /bin/sh -c mosquitto             Up      0.0.0.0:1883->1883/tcp
pub      qemu-arm-static /bin/sh -c ...   Exit 0
sub      qemu-arm-static /bin/sh -c ...   Up
$ bash && ./test.sh
Passed
Job succeeded
```

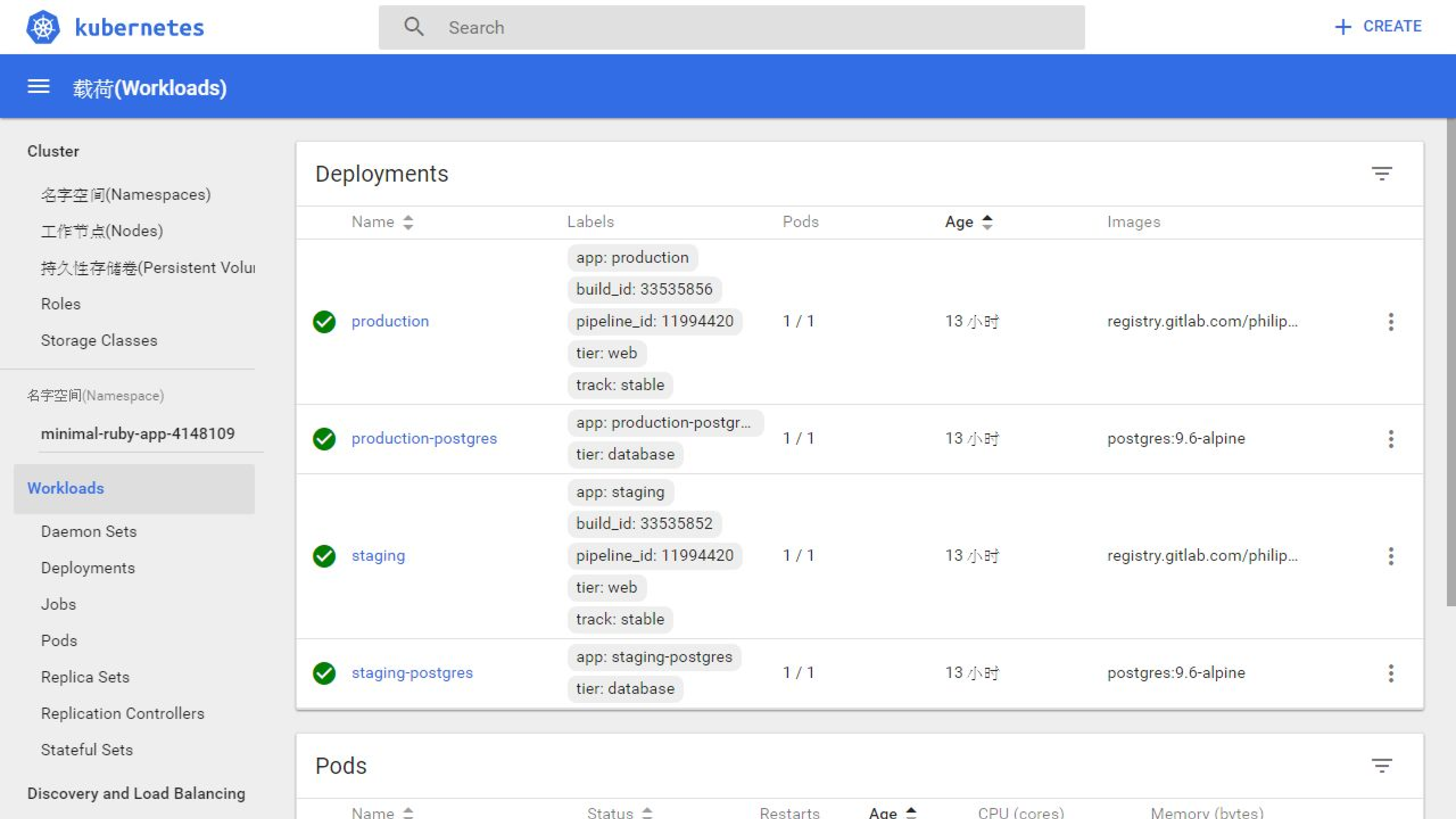
6. GitLab CI 與 K8S 整合



Sep 22, 2017 · Mike Bartlett

GitLab 10.0 Released with Auto DevOps and Group Issue Boards

From the formulation of an idea to executing and monitoring it in production, DevOps establishes a culture and environment where developing, testing, and releasing software can happen quickly,

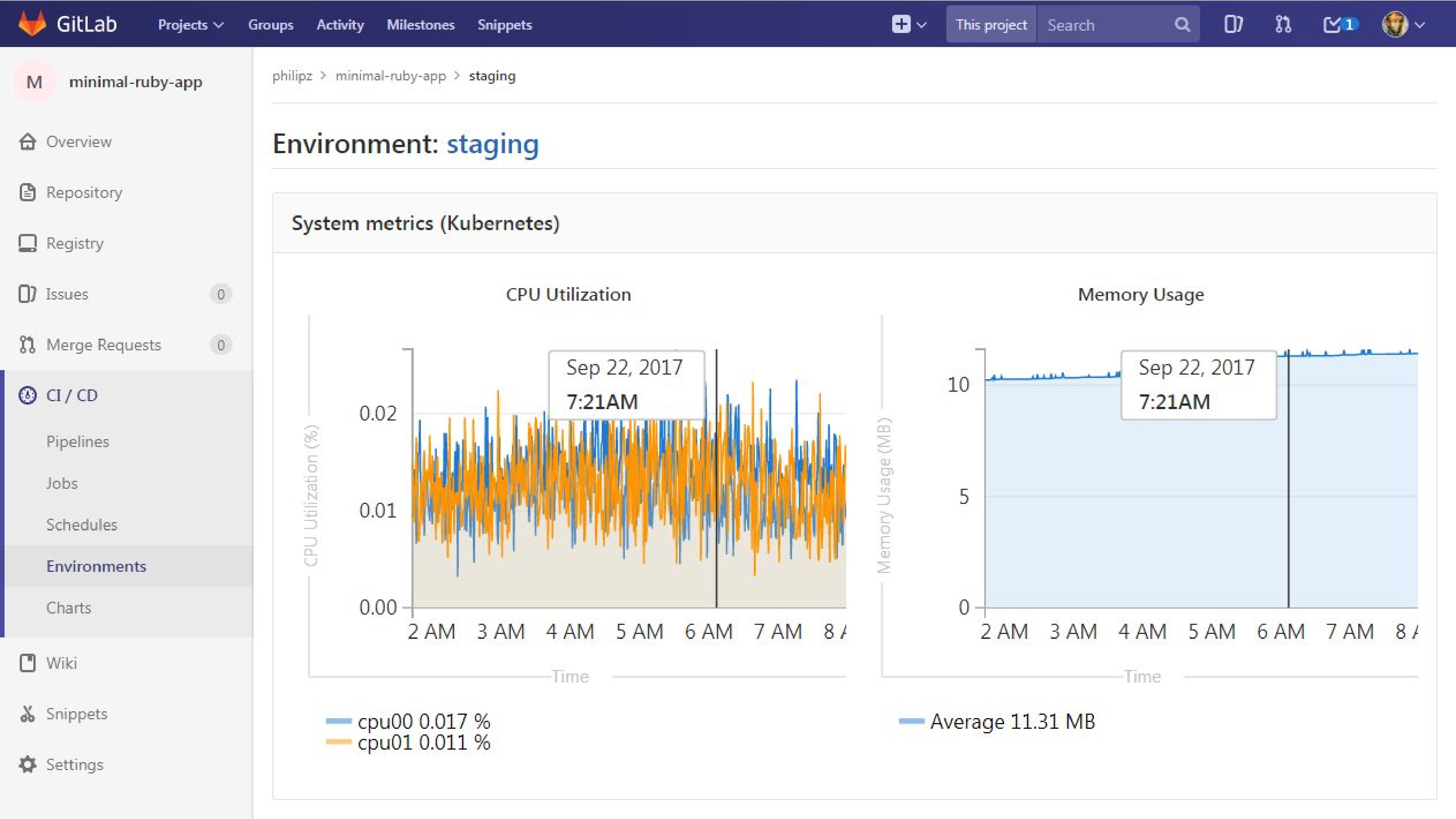


Deployments

	Name	Labels	Pods	Age	Images
✓	production	app: production build_id: 33535856 pipeline_id: 11994420 tier: web track: stable	1 / 1	13 小时	registry.gitlab.com/philip...
✓	production-postgres	app: production-postgr... tier: database	1 / 1	13 小时	postgres:9.6-alpine
✓	staging	app: staging build_id: 33535852 pipeline_id: 11994420 tier: web track: stable	1 / 1	13 小时	registry.gitlab.com/philip...
✓	staging-postgres	app: staging-postgres tier: database	1 / 1	13 小时	postgres:9.6-alpine

Pods

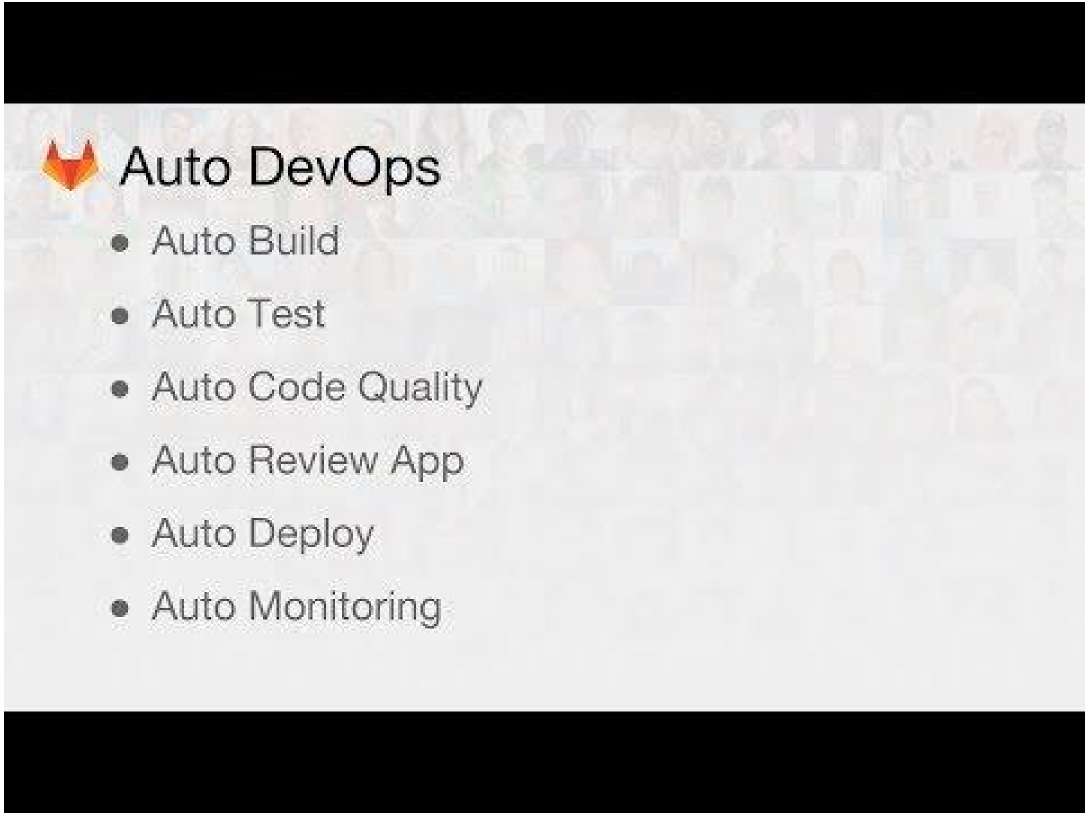
Name	Status	Restarts	Age	CPU (cores)	Memory (bytes)



Environment: staging

System metrics (Kubernetes)

GitLab Auto DevOps



Auto DevOps

- Auto Build
- Auto Test
- Auto Code Quality
- Auto Review App
- Auto Deploy
- Auto Monitoring

Canary Deployment

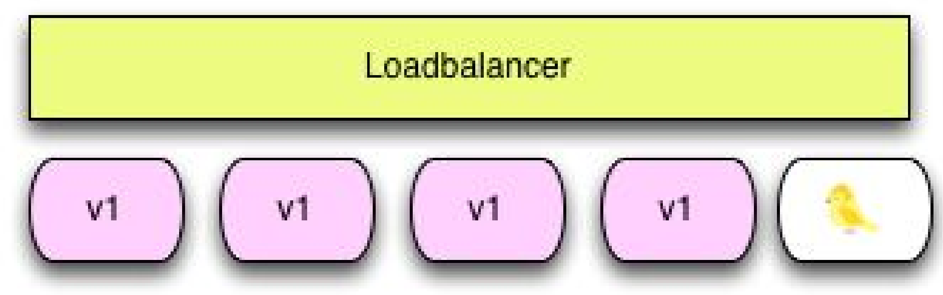
金絲雀部署

只部署部分實例

即早發現問題

藍綠部署

A/B Testing


Loadbalancer

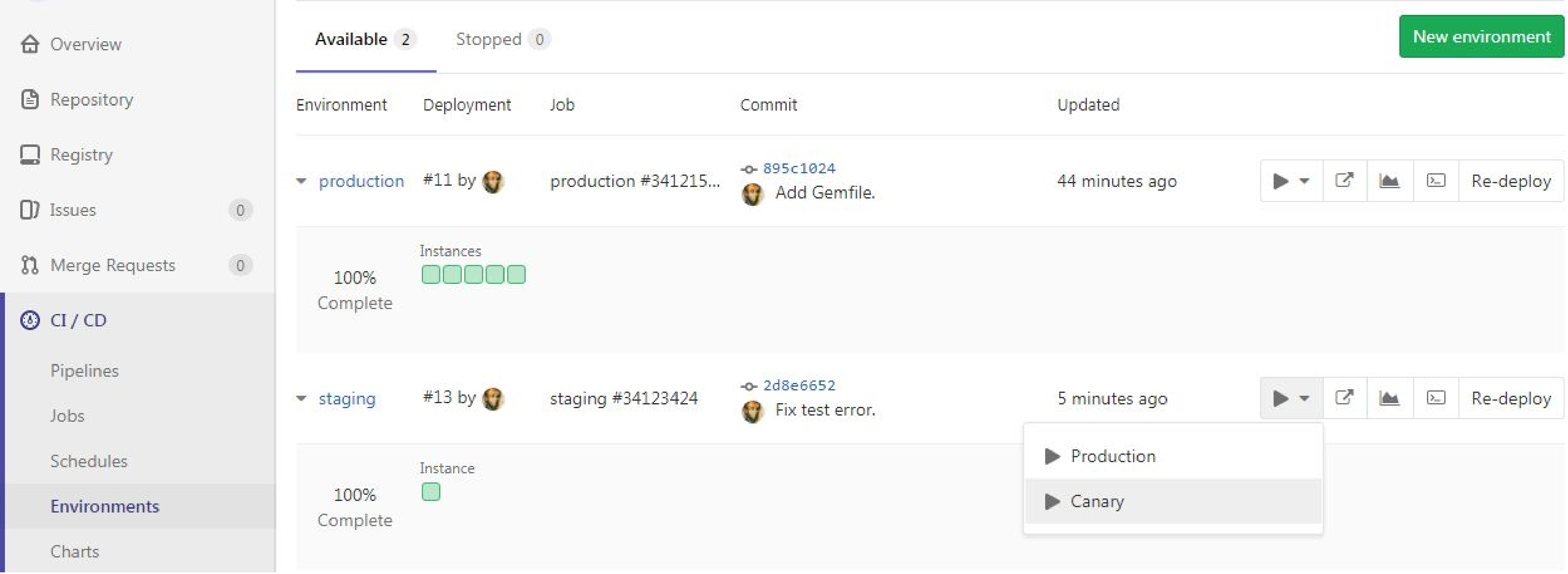
v1

v1

v1

v1

Environments

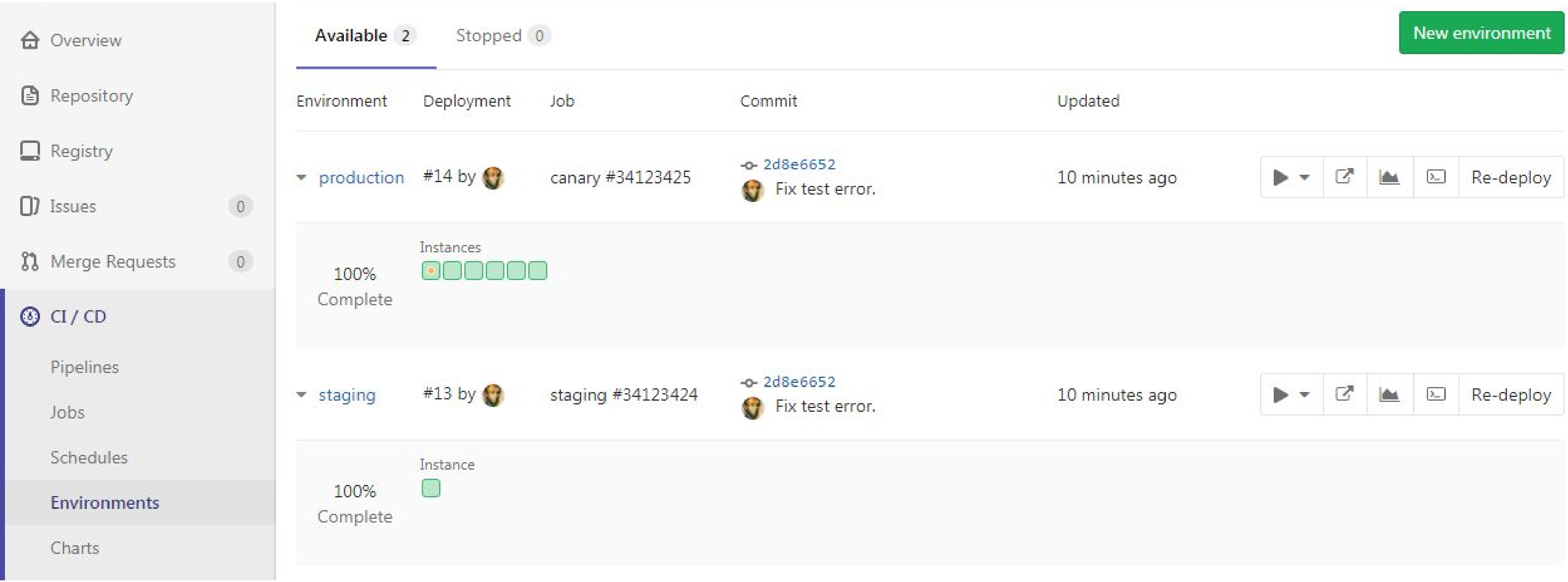


Environment	Deployment	Job	Commit	Updated
production	#11 by	production #341215...	895c1024 Add Gemfile.	44 minutes ago
staging	#13 by	staging #34123424	2d8e6652 Fix test error.	5 minutes ago

production: 100% Complete — Instances 🟩🟩🟩🟩🟩

staging: 100% Complete — Instance 🟩

Environments

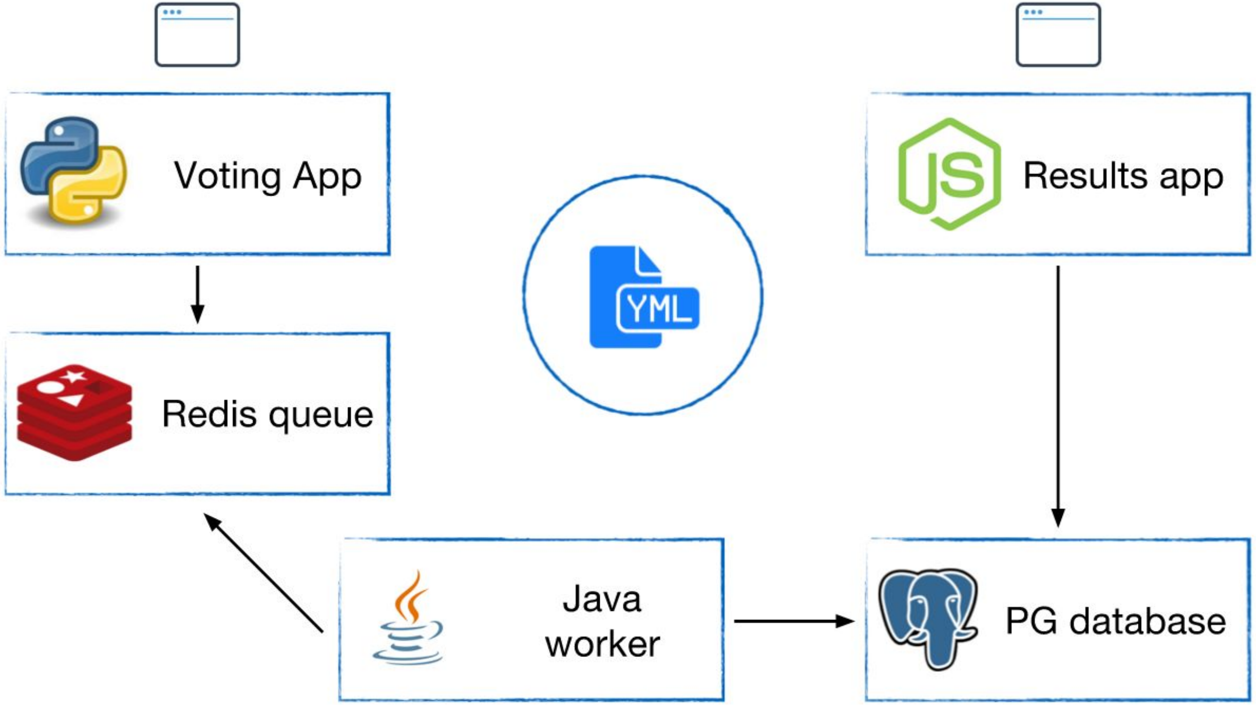


Environment	Deployment	Job	Commit	Updated
production	#14 by	canary #34123425	2d8e6652 Fix test error.	10 minutes ago
100% Complete	Instances 🟩🟩🟩🟩🟩🟩			
staging	#13 by	staging #34123424	2d8e6652 Fix test error.	10 minutes ago
100% Complete	Instance 🟩			

7. 結語

Microservices Java Worker

Docker Birthday #3 training



Still No Silver Bullet

容器只是其中一個關鍵，並非全部

DevOps pipeline 軟體開發流程

Microservices微服務，或其他架構

Infrastructure as Code

Business model

The Docker Stack

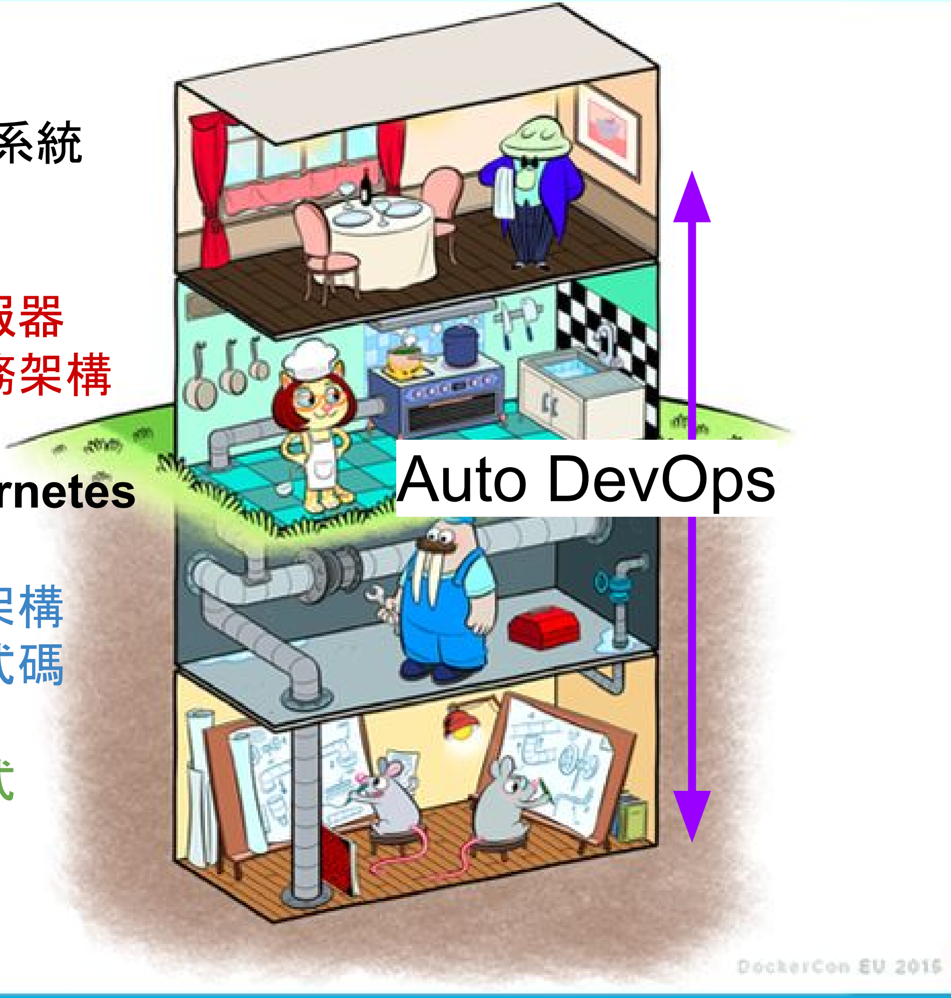
*業務系統

無伺服器
微服務架構

Kubernetes

基礎架構
即程式碼

容器式
設計



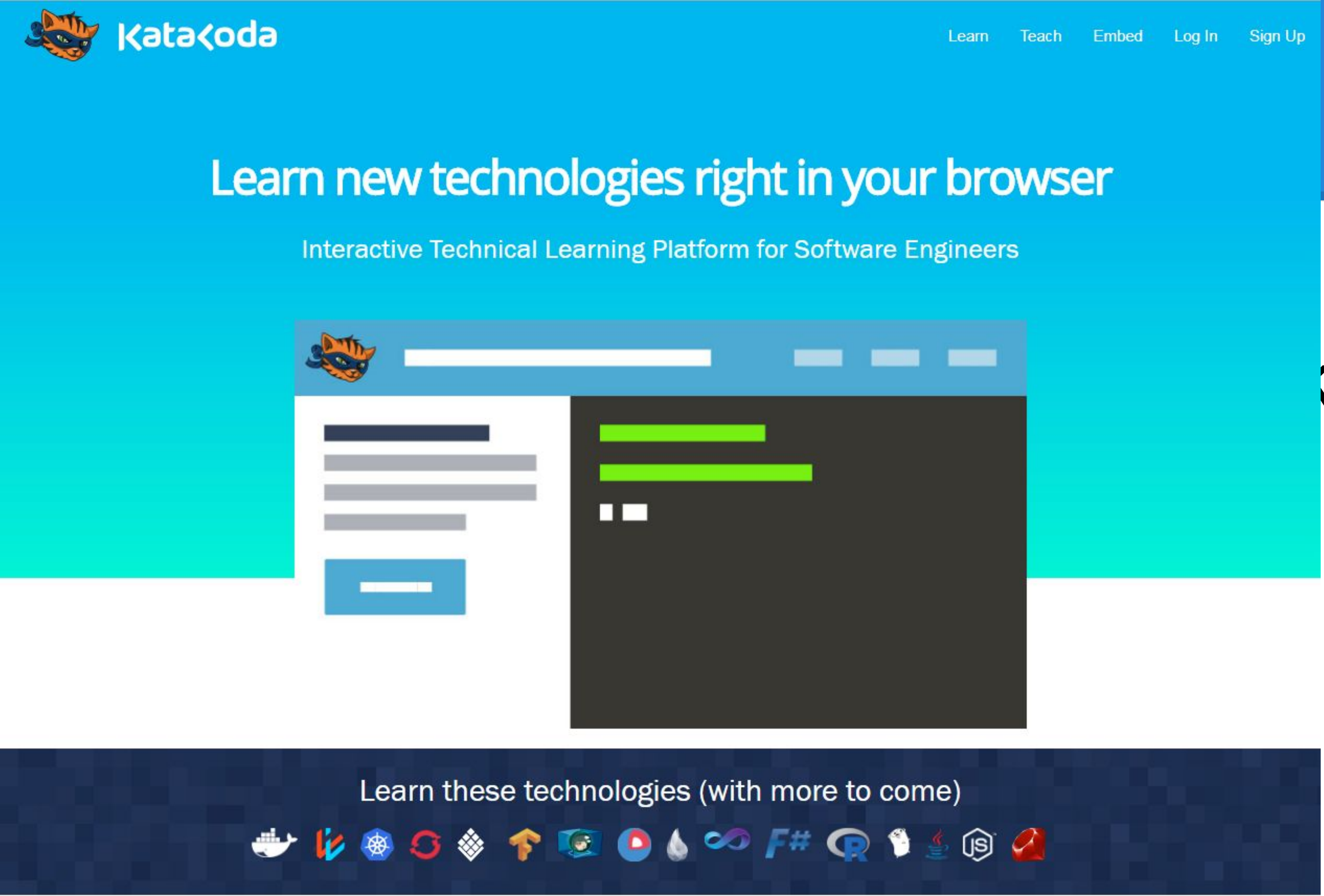
Auto DevOps

Philip Zheng
Aug 27 · 3 min read

GitLab 結合 Docker 容器開發測試

感謝各位學員參加廣宣學堂的 Docker 進階班，因時間限制，未能詳細說明 GitLab 和 Azure CLI 相關設置步驟。故附上此篇教學文件，希望能幫助各位自行學習和操作。

此文章主要講解 GitLab 基本使用，涉及 Git 版本控管，GitLab CI 持續整合 pipeline 和存放容器映像檔的 GitLab Registry ，如圖一，三位一體的整合功能，示範如何透過 Dockerfile 建置出映像檔，並存放到 GitLab 內建的 Registry ，再使用包裝成容器映像檔的測試工具作壓力測試，最後存放到 Docker Hub，並且透過 Azure CLI 上線到 Azure PaaS 網站，整個建置、測試、部署、上線，這四個階段都是使用容器。

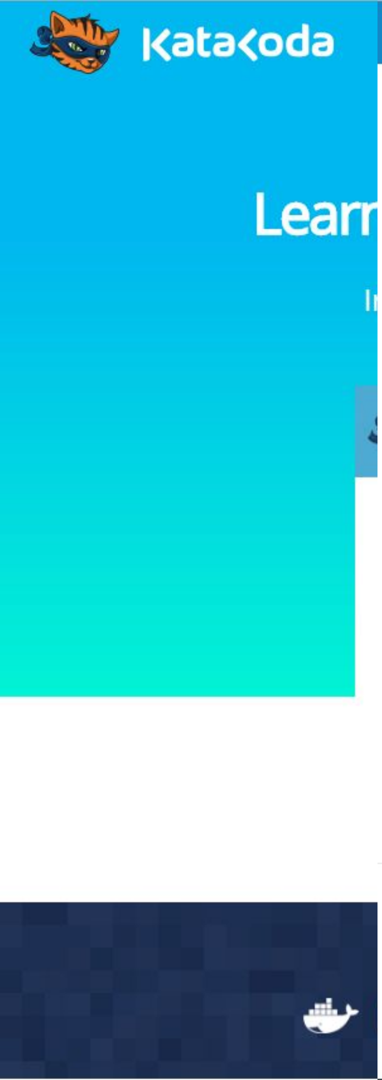


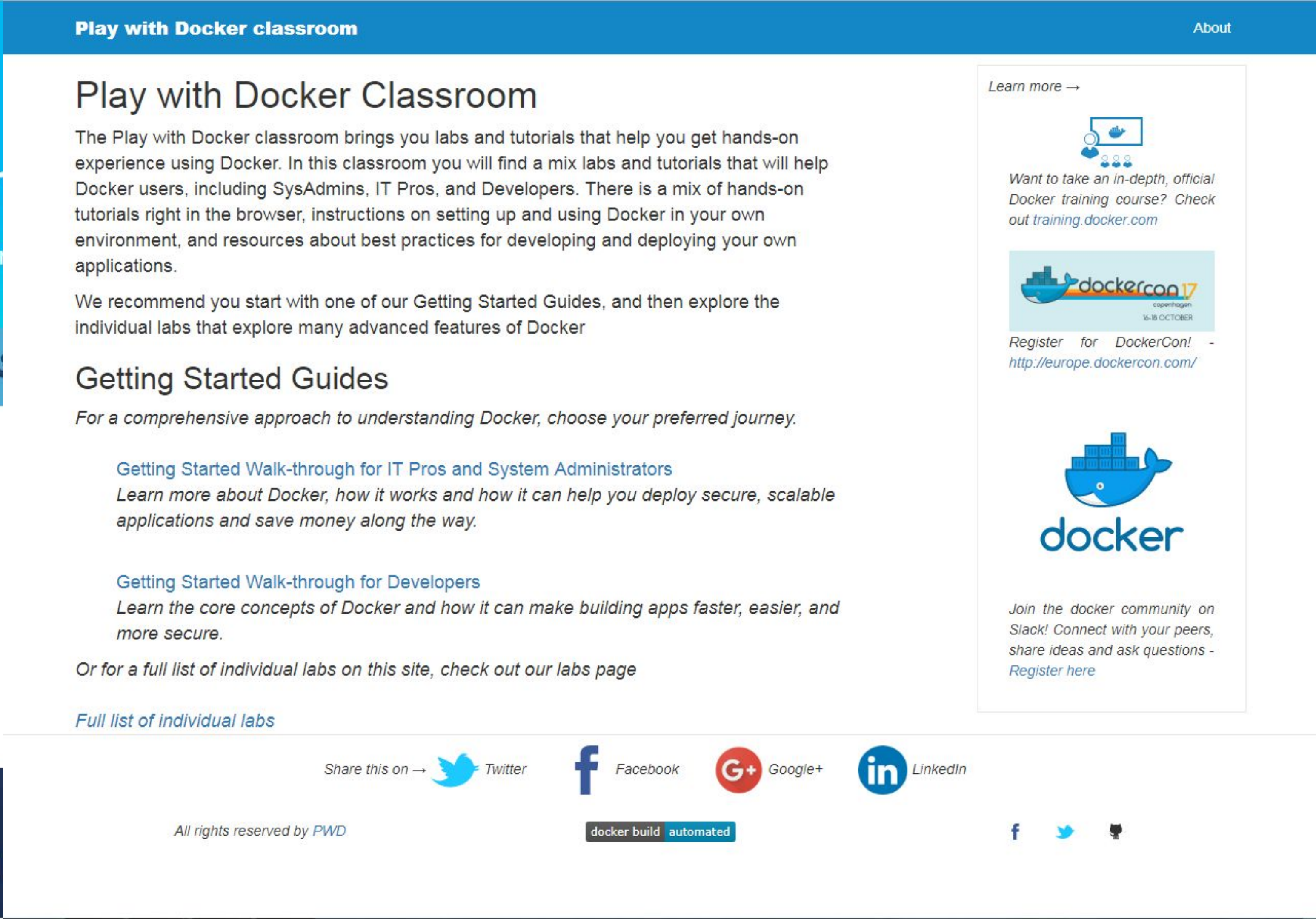
Learn new technologies right in your browser

Interactive Technical Learning Platform for Software Engineers

Learn these technologies (with more to come)

ker_p





Play with Docker Classroom

The Play with Docker classroom brings you labs and tutorials that help you get hands-on experience using Docker. In this classroom you will find a mix labs and tutorials that will help Docker users, including SysAdmins, IT Pros, and Developers. There is a mix of hands-on tutorials right in the browser, instructions on setting up and using Docker in your own environment, and resources about best practices for developing and deploying your own applications.

We recommend you start with one of our Getting Started Guides, and then explore the individual labs that explore many advanced features of Docker

Getting Started Guides

For a comprehensive approach to understanding Docker, choose your preferred journey.

Getting Started Walk-through for IT Pros and System Administrators
Learn more about Docker, how it works and how it can help you deploy secure, scalable applications and save money along the way.

Getting Started Walk-through for Developers
Learn the core concepts of Docker and how it can make building apps faster, easier, and more secure.

Or for a full list of individual labs on this site, check out our labs page

Full list of individual labs

Learn more →

Want to take an in-depth, official Docker training course? Check out training.docker.com

Register for DockerCon! - http://europe.dockercon.com/

Join the docker community on Slack! Connect with your peers, share ideas and ask questions - Register here

Share this on → Twitter Facebook Google+ LinkedIn

All rights reserved by PWD

docker build automated

Thank you

Docker可省下比金錢更寶貴的時間！