

GitLab과 Kubernetes를 통한 CI/CD 구축

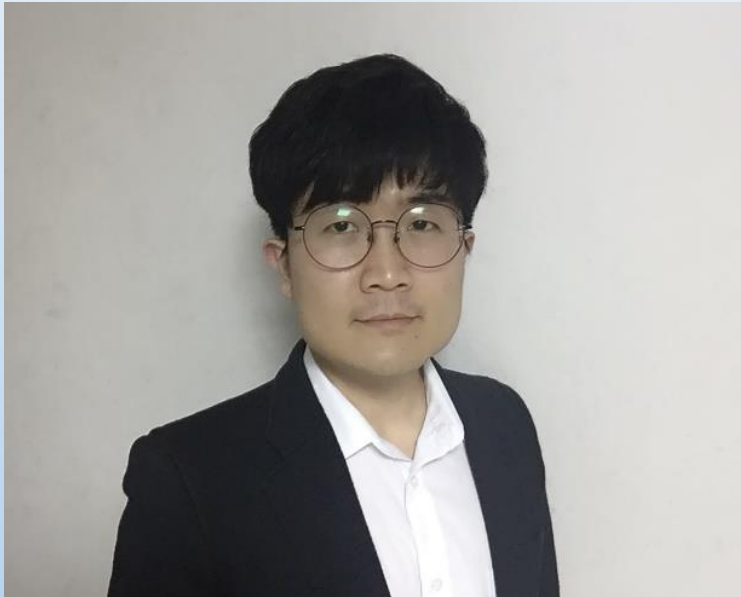
2018.6

콘텐츠연합플랫폼

김철구



Who am I?



콘텐츠연합플랫폼 DevOps Engineer

- Cloud System 운영
- Cloud 이전 및 MSA 설계
- CI/CD process 도입 및 운영

Email: cgkim@captv.co.kr

주간 인기 프로그램

VOD 홈

인기

최신

요일

장르

다시보고싶은

에디터Pick

전체 드라마 예능 시사교양



1 나 혼자 산다
247회 ◦ 2018-06-08(금)
MBC



2 하트시그널 시즌2
12회 ◦ 2018-06-08(금)
채널A



3 아는 형님
131회 ◦ 2018-06-09(토)
JTBC



4 백종원의 골목식당
21회 ◦ 2018-06-08(금)
SBS



5 런닝맨
536회 ◦ 2018-06-10(일)
SBS

NEW!
화제의
신작업데이트

전체보기 >



CI/CD란 무엇인가?



CI(Continuous Integration)



지속적인 통합(CI) : software 개발 방식

- 개발팀 구성원들이 각자 일단위로 code를 integration.
→ 하루에 다수의 integration이 발생.
- integration은 자동화된 build와 test로 검증.
→ integration error는 가능한 한 빠르게 감지.

위 방식의 도입으로 다수의 팀에서 만들어지는 integration 문제를 획기적으로 줄임으로써 빠르게 software를 개발할 수 있다.

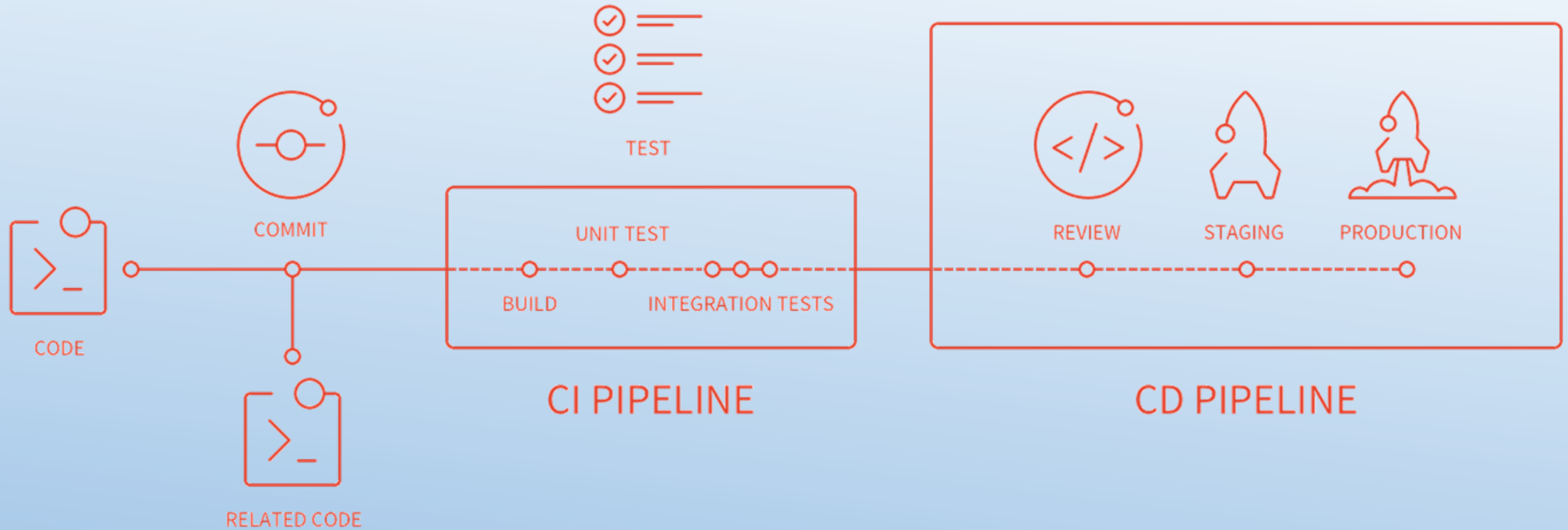
2006... Martin Fowler

CD(Continuous Delivery)

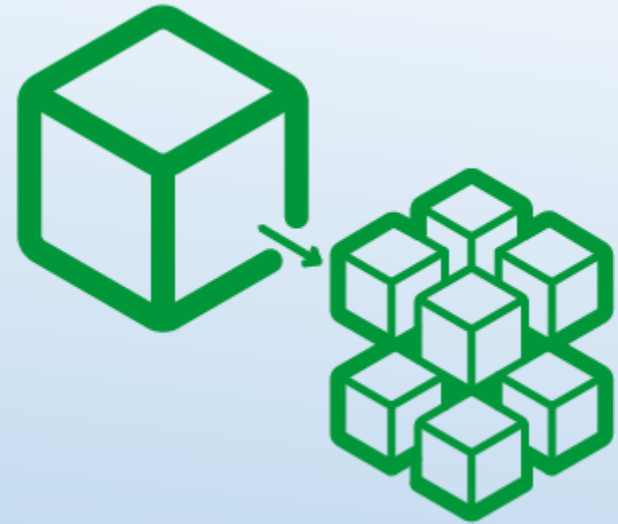


- 지속적인 통합을 확장.
 - CI 프로세스를 통과한 변경사항을 Test/Production 환경에 즉시 배포.
- 지속적인 전달(CD).
 - dev/staging/production 환경에 대한 application 배치 자동화.
 - 여러 application에 대한 변경 사항, 동시 배치 관리의 Pipeline.
 - production 업데이트에 대한 승인 프로세스 제공.

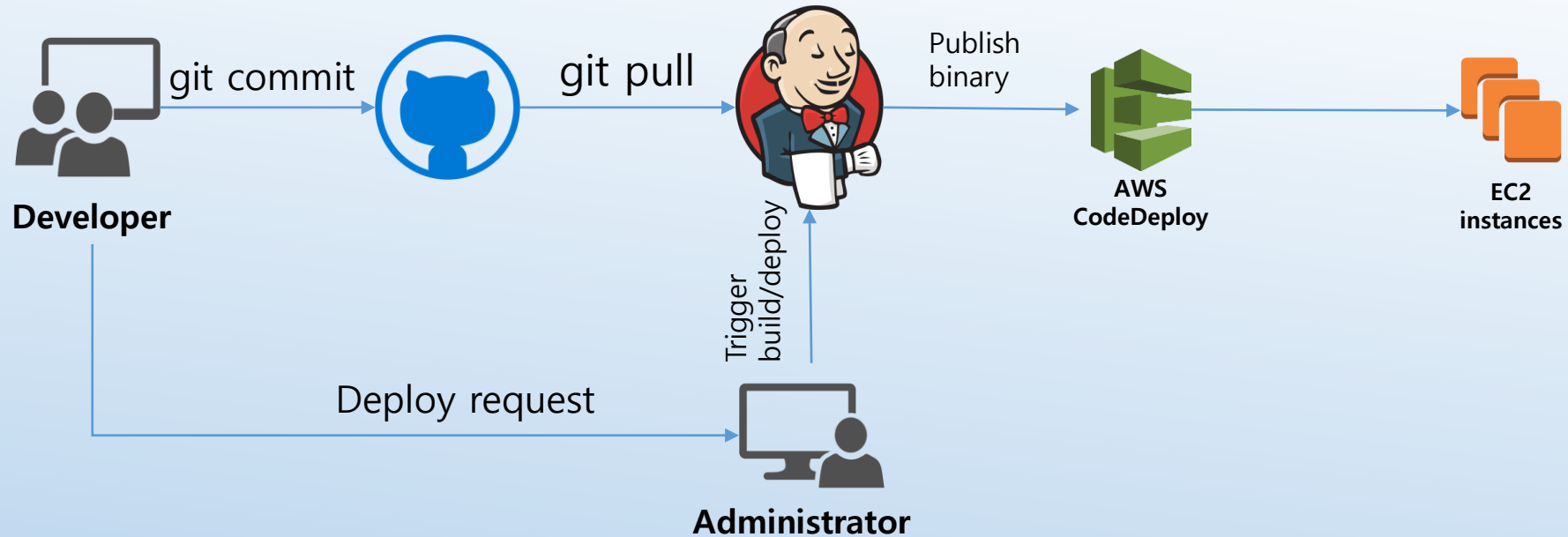
CI/CD workflow



MSA에서 CI/CD를
어떻게 구현할 것인가?



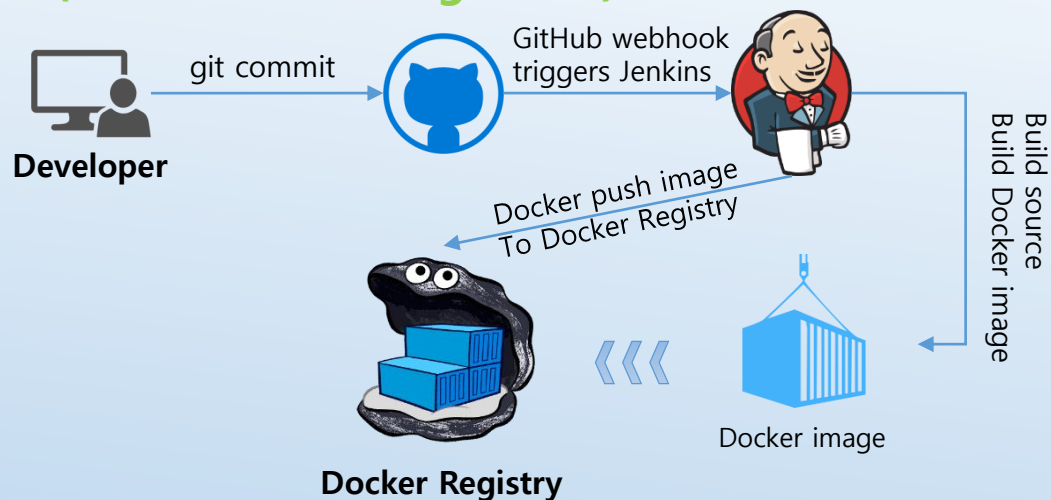
pooq 2.x CI/CD process



1. 개발자 - git branch에 source code commit 후 Github repository에 merge.
2. 배포 요청 → 운영자 - Jenkins build trigger를 수행.
3. Jenkins에서 Build된 binary는 codedeploy plugin을 배포 trigger.
4. AWS CodeDeploy에서 지정된 target EC2 instance들로 배포.

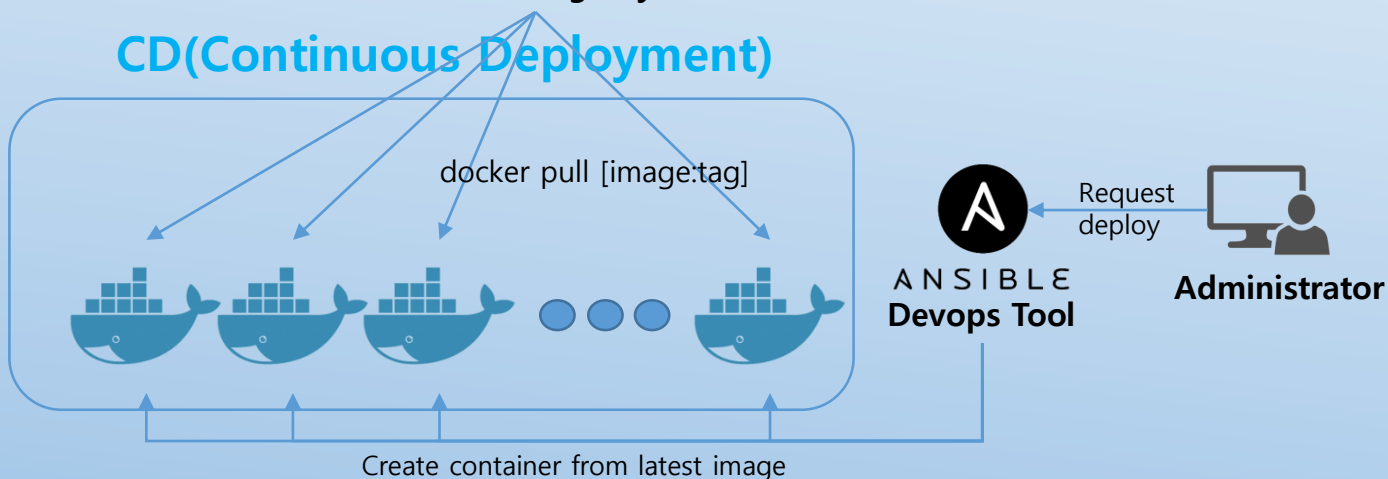
Docker 기반 CI/CD process 구성안

CI(Continuous Integration)



1. 개발자 - git branch에 source code commit 후 Github repository에 merge.
2. Github webhook으로 Jenkins build 실행 – docker image build 후 private registr로 push.
3. 운영자 – push된 image를 Devops tool(ex.Ansible)로 Docker instance group에 배포.

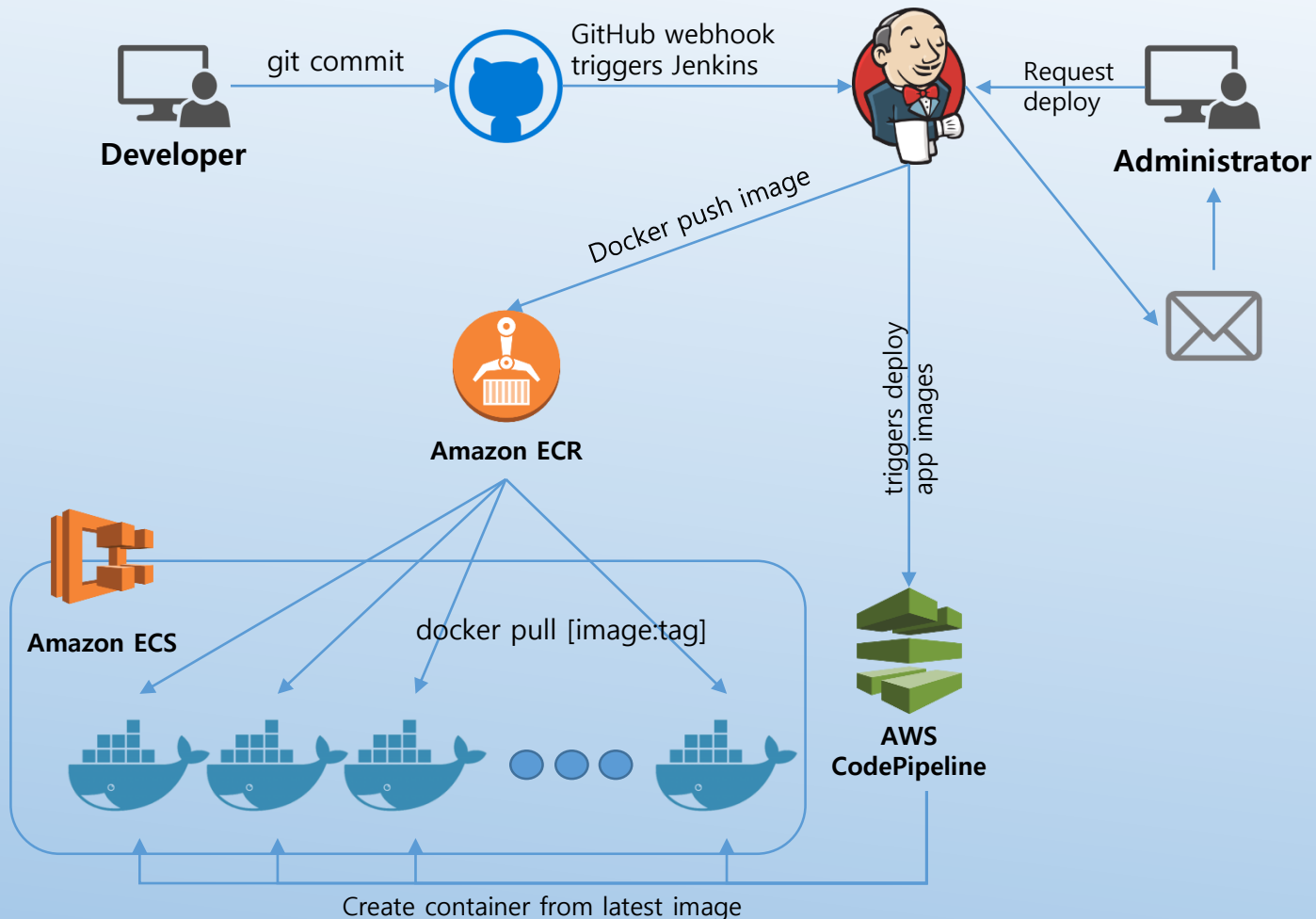
CD(Continuous Deployment)



Container 기반 MSA에서 CI/CD 고려사항

- **VCS(Version Control System)** – Git 기반 repository.
- **Container image registry** – 내부 project들의 이미지가 외부에 노출되지 않도록 private registry 필요.
- **Container Orchestration** – 다수의 Docker instance들을 clustering하여 scheduling, network setup, 고가용성, 확장성을 관리.
- **CI(Continuous Integration)** – Polyglot(multi-language) programming을 지원, 다수의 MicroService CI Pipeline을 동시에 실행.
- **CD(Continuous Delivery)** – Container Orchestration과 통합되어 최신의 Container image를 Roll out/back할 수 있는 기능과 인터페이스 제공.

CI/CD architecture 초안



- CI/CD는 Jenkins를 통해 관리.
- AWS ECR을 private image registr로 사용.
- Container Orchestration platform으로 AWS ECS 사용.
- Jenkins에서는 AWS CodePipeline plug-in을 통해 deploy trigger.
- deploy step에서 issue 발생 시 email로 통지.

CI/CD architecture 초안 문제 사항



Jenkins

- plug-in 관리의 복잡성.
- CI/CD Pipeline 작성의 어려움.
- 계정관리의 복잡성.



Amazon ECR



Amazon ECS



AWS
CodePipeline

AWS Container Service

- Service discovery 미지원.(별도 solution 구축 필요.)
- Korea region 미지원(당시).
- AWS 서비스 의존성 100%.
- AWS 구성요소들에 대한 설정 복잡성 증가.



Administrator

Administrator

- 모든 요소들에 대한 설정 및 제어.
- 배워야 하는 기술들의 증가.
- Microservice Project가 수 백개가 증가하면 작업량도 비례하여 증가.

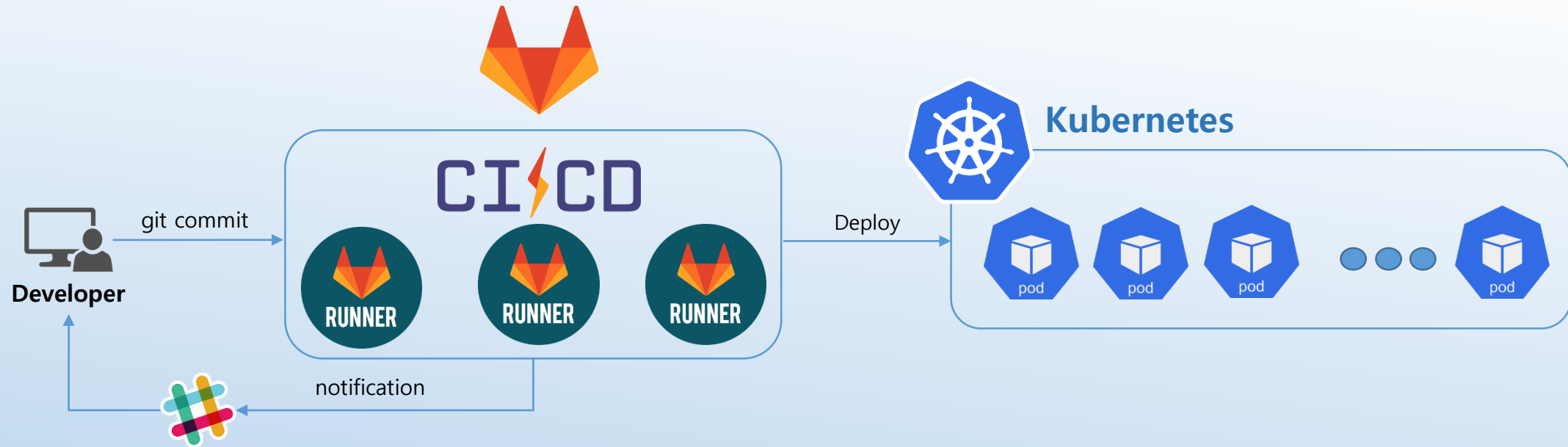
Solution은?



GitLab

- ✓ Open source
- ✓ Git 기반 version 관리 서비스
- ✓ 통합된 CI 제공.
- ✓ Private Container Registry 통합.
- ✓ Docker 기반으로 build 및 Test, 등록, 배포 자동화.
- ✓ 다수의 GitLab Runner 서버를 통해 동시작업을 병렬로 수행.
- ✓ YAML 기반의 CI workflow 정의로 code화 template화 가능.
- ✓ CLI script job executor로 Container Orchestration(Kubernetes)에 배포가능.

GitLab CI/CD process

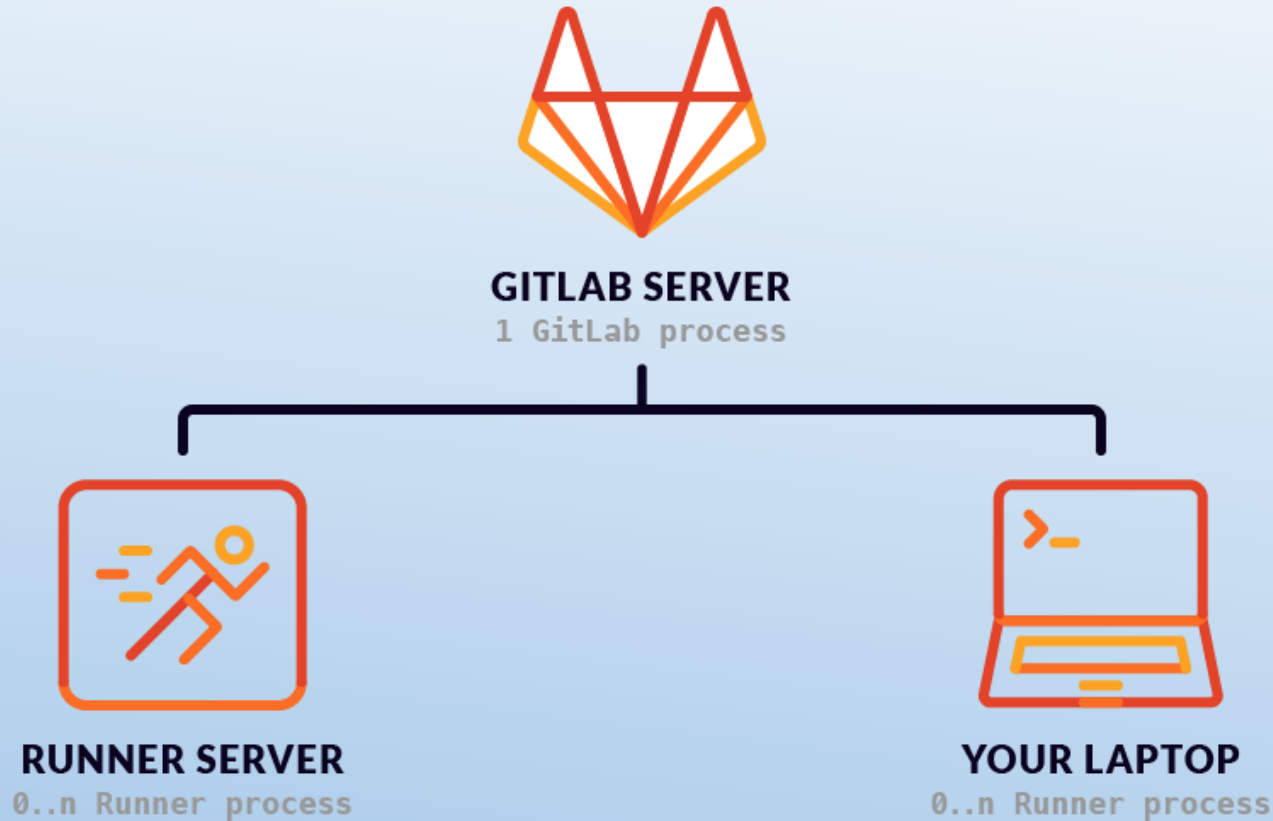


- 개발자 – microservice project에 code를 commit.
- CI/CD pipeline은 .gitlab-ci.yml에 YAML code를 통해 정의.
- GitLab은 GitLab Runner들을 통해 병렬로 target에 deploy를 수행.
- pipeline 실행과정에서 발생한 issue는 slack에 통합되어 통지.

GitLab CI/CD



GitLab CI/CD Architecture



- **GitLab CI/CD** – GitLab의 일부분으로 web application과 state를 DB에 저장하는 API로 동작한다. project들과 build들을 관리하며 사용자 친화적인 UI를 제공한다.
- **GitLab Runner** – build들을 처리하는 application으로 별도로 배치되어 GitLab CI/CD와 API를 통해 동작된다. 각각의 Runner는 공유되지만 project에 dedicate될 수 있다.

GitLab CI/CD Pipeline 기초

- **.gitlab-ci.yml**
 - CI/CD pipeline을 정의하는 file로 YAML로 기술.
 - git repository의 root director에 위치.

before_script:

- apt-get update -qq && apt-get install -y -qq sqlite3 libsqlite3-dev nodejs
- ruby -v
- which ruby
- gem install bundler --no-ri --no-rdoc
- bundle install --jobs \$(nproc) "\${FLAGS[@]}"

rspec:

script:

- bundle exec rspec

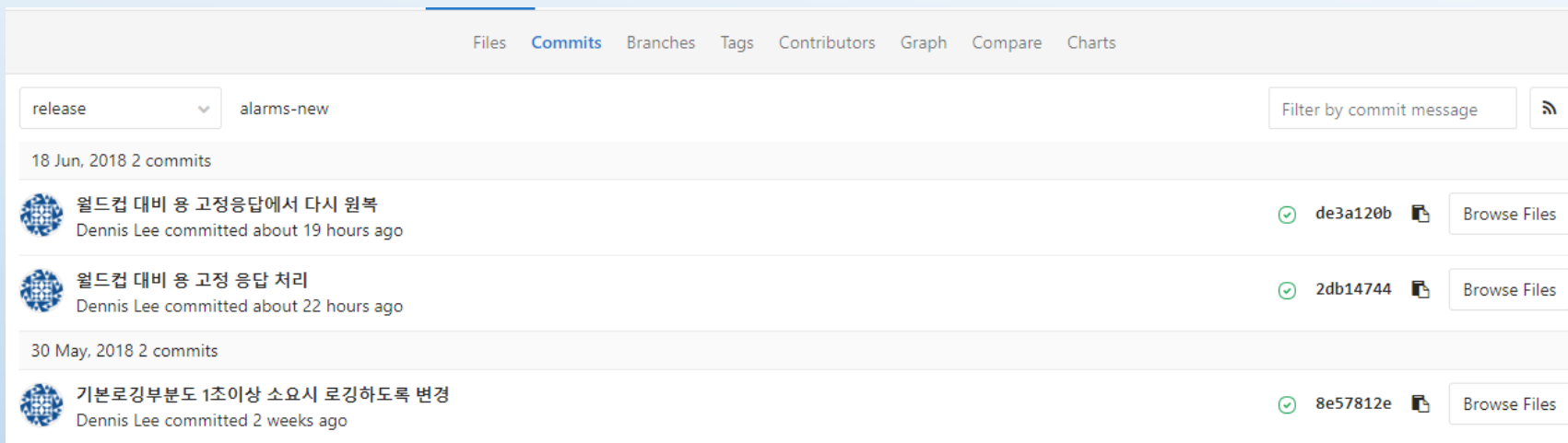
rubocop:

script:

- bundle exec rubocop

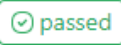






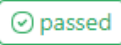








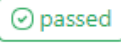








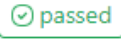








GitLab CI/CD Pipeline 기초

- Commit으로 CI를 Triggering



GitLab CI/CD Pipeline 기초

- Pipeline Tab에서 commit history 별로 배포 상황을 확인.

All 33 Pending 0 Running 0 Finished 33 Branches Tags CI Lint				
Status	Pipeline	Commit	Stages	
	#12931 by  latest	P stg-hfix -> de3a120b  월드컵 대비 용 고정응답에서 다시 원복	 	 00:00:49  about 19 hours ago
	#12930 by  latest	P release -> de3a120b  월드컵 대비 용 고정응답에서 다시 원복	  	 00:01:05  about 19 hours ago 
	#12919 by 	P release -> 2db14744  월드컵 대비 용 고정 응답 처리	  	 00:01:20  about 21 hours ago 
	#12689 by 	P release -> 8e57812e  기본로깅부분도 1초이상 소요시 로깅...	  	 00:01:06  2 weeks ago 

GitLab CI/CD Pipeline 기초

- 상세 log는 StdOut, StdErr으로 출력되어 UI에서 확인

passed

Job #31371 triggered about 19 hours ago by 이승준

```
Running with gitlab-ci-multi-runner 9.4.2 (6d06f2e)
on bh-gitlab-runner-prd-05 (deb63e39)
Using Docker executor with image gitlab-registry.pooq.co.kr/pooq-system/pooq3-ms-deploy:latest ...
Using docker image sha256:0debb63004cd435fef8eb8dd4a5d35277bd941c9a01f334321a37663ba0a032e for predefined container...
Pulling docker image gitlab-registry.pooq.co.kr/pooq-system/pooq3-ms-deploy:latest ...
Using docker image gitlab-registry.pooq.co.kr/pooq-system/pooq3-ms-deploy:latest ID=sha256:b8fdeb96e828170cad989d88cad1f1364a5377f0d3353b28fcd8ec82bf08f4d7 for build container...
Running on runner-deb63e39-project-1467-concurrent-0 via bh-gitlab-runner-prd-05...
Fetching changes...
HEAD is now at de3a120 월드컵 대비 용 고정응답에서 다시 원복
Checking out de3a120b as release...
Skipping Git submodules setup
$ /deployapp/pooq3_ms_deploy.py -e $CI_COMMIT_REF_NAME -n $CI_PROJECT_NAME -f pooq-app-info.json -i $CI_REGISTRY_IMAGE -v $CI_COMMIT_SHA
2018-06-18T14:00:05.976333 - app_name:alarms-new, commit_tag:de3a120b5f050fd4c7cf94bd4a4b29a7adda290f
NAME      DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
alarms-new-dpm  5        5        5           5          76d
deployment.apps "alarms-new-dpm" image updated
Waiting for rollout to finish: 0 out of 5 new replicas have been updated...
Waiting for rollout to finish: 0 out of 5 new replicas have been updated...
Waiting for rollout to finish: 2 out of 5 new replicas have been updated...
Waiting for rollout to finish: 3 out of 5 new replicas have been updated...
Waiting for rollout to finish: 3 out of 5 new replicas have been updated...
Waiting for rollout to finish: 3 out of 5 new replicas have been updated...
Waiting for rollout to finish: 4 out of 5 new replicas have been updated...
Waiting for rollout to finish: 4 out of 5 new replicas have been updated...
Waiting for rollout to finish: 4 out of 5 new replicas have been updated...
Waiting for rollout to finish: 2 old replicas are pending termination...
Waiting for rollout to finish: 2 old replicas are pending termination...
Waiting for rollout to finish: 2 old replicas are pending termination...
Waiting for rollout to finish: 1 old replicas are pending termination...
Waiting for rollout to finish: 1 old replicas are pending termination...
Waiting for rollout to finish: 1 old replicas are pending termination...
Waiting for rollout to finish: 4 of 5 updated replicas are available...
deployment "alarms-new-dpm" successfully rolled out
2018-06-18T14:00:13.125450 - Complete deploying app to Kubernetes.
2018-06-18T14:00:13.129910 - domains:apis.pooq.co.kr, uris:/alarms-new, methods:GET, api_name:alarms-new, api_timeout:5000
2018-06-18T14:00:13.192762 - alarms-new updated at Kong API G/W.
2018-06-18T14:00:13.192792 - Complete registering app to Kong API G/W.
Job succeeded
```

deploy-to-k8s-prd

Duration: 16 seconds
Runner: #10

Commit [de3a120b](#)

월드컵 대비 용 고정응답에서 다시 원복

✓ Pipeline #12930 from release

deploy

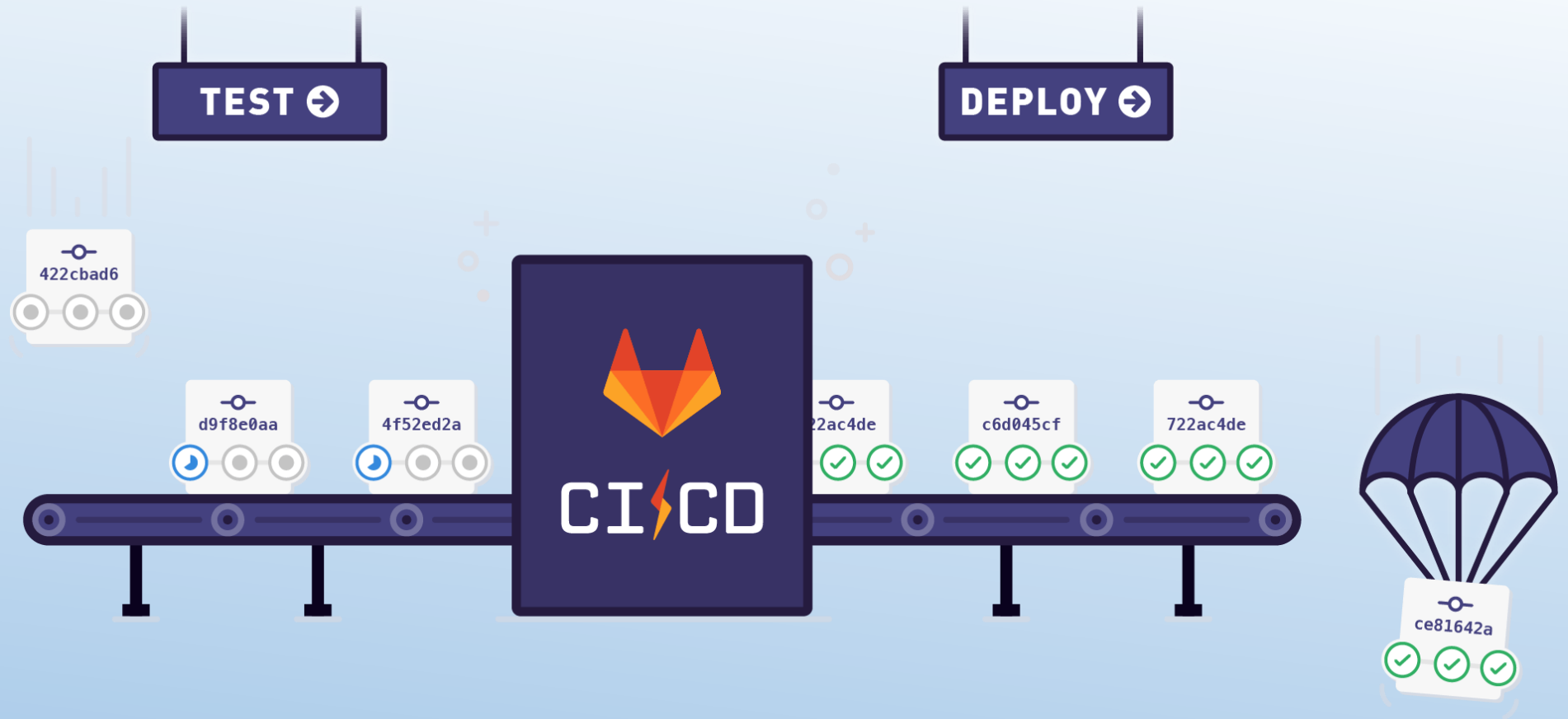
→ ✓ deploy-to-k8s-prd

.gitlab-ci.yml Job Example

Simple Job Example	job1: script: ls -l
Multiple Script Commands	job1: script: <ul style="list-style-type: none">- cd myproject- vender/bin/phpunit
Multi Stages	stages: <ul style="list-style-type: none">- test- deploy run_tests: stage: test script: php run-some-tests.php deploy_to_server: stage: deploy script: php mageDeploy2.php

※ .gitlab-ci.yml Guide: <https://docs.gitlab.com/ee/ci/yaml/README.html>

시연



Q&A

감사합니다.