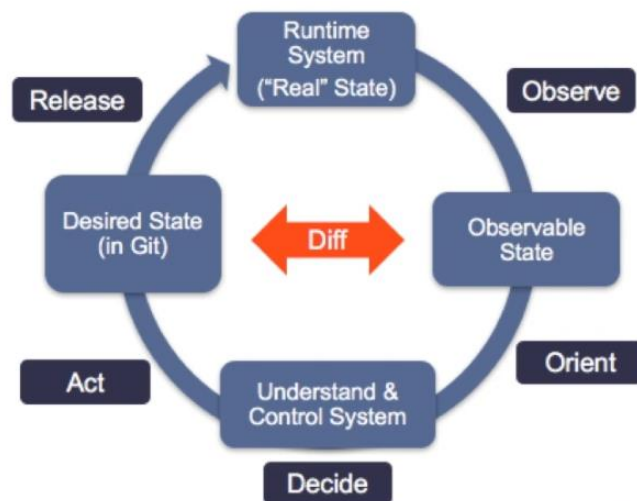


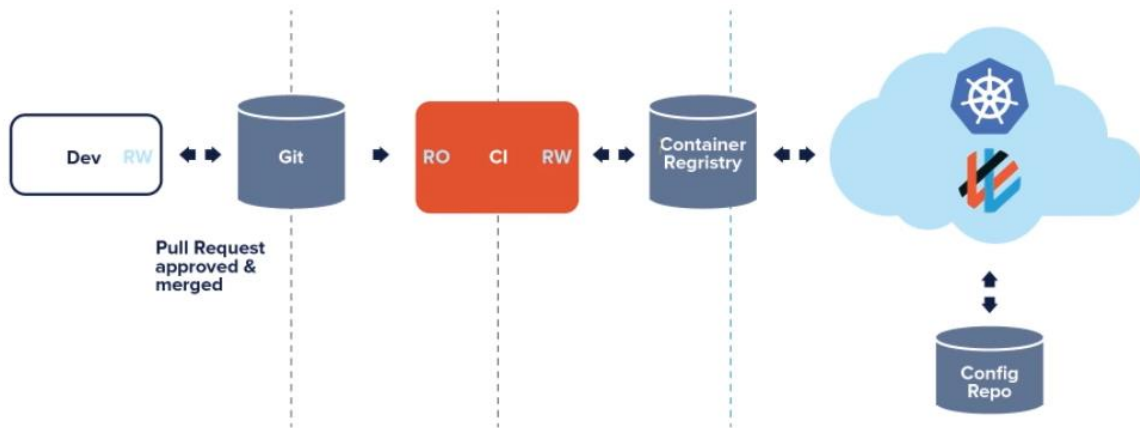
<ArgoCD>

1. Gitops

- Weaveworks에서 개발
- Code로 관리하는 것(git)을 운영(Operation) 하는 것이다. Git에 올려진 Desired State를 쿠버네티스에 떠있는 워크로드를 지속적으로 비교를 해서 git에 올려진 state에 맞게 지속적으로 싱크를 맞춘다. (Git(code) + Ops)
- 쿠버네티스에 어떠한 식으로 워크로드가 뜰지 명시되어 있는 git으로 관리가 되는 Config repo를 현재 Current state를 지속적으로 맞춰준다.



GitOps Deployment Workflow



2. ArgoCD 란?

쿠버네티스의 GitOps CD tool 이다. 자동으로 target 환경에 desired state 를 배포해주는 역할을 한다. Git 에 있는 상태를 읽어서 쿠버네티스에 배포해주는 역할을 한다.

1) Component

APIserver

gRPC/REST 형태로 명령어를 날린다. UI, CLI 를 통해서 명령어를 날릴 수 있다.

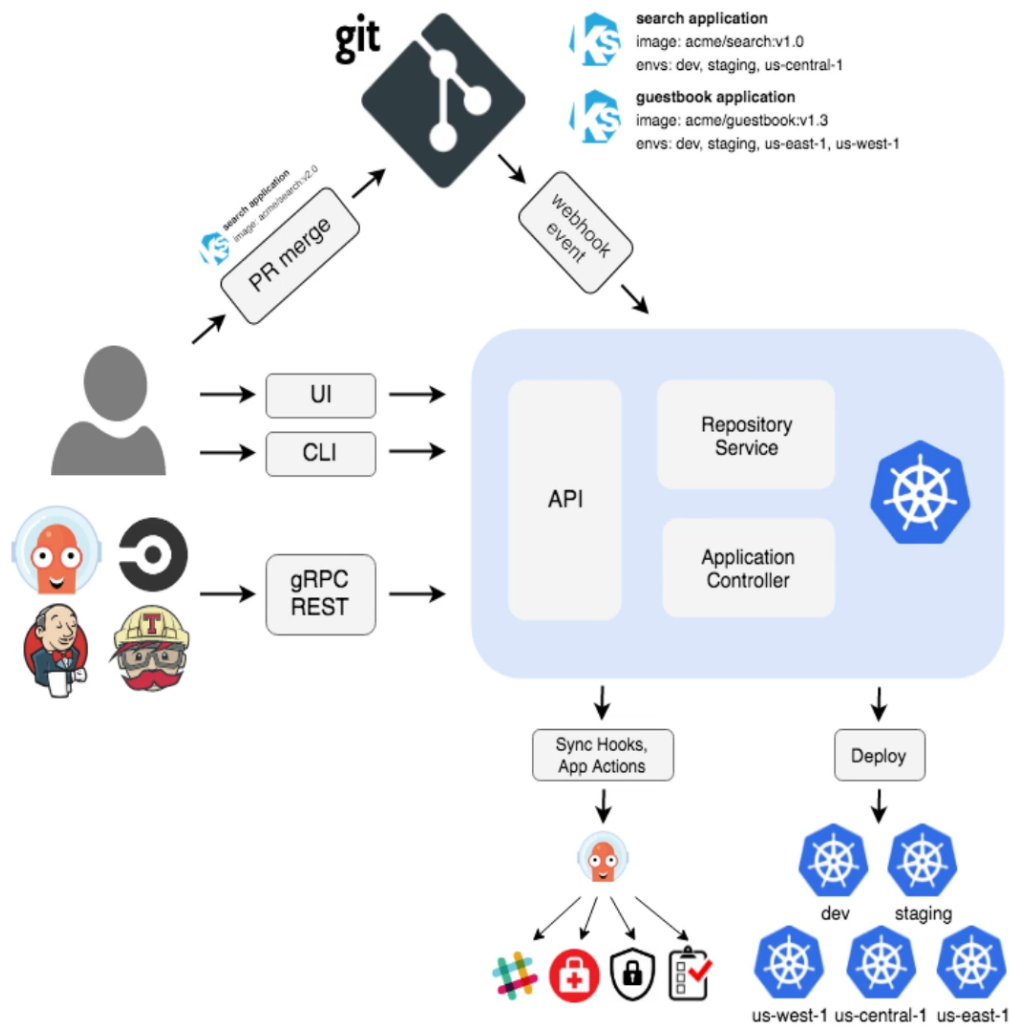
Repository server

깃에 있는 형상을 쿠버네티스에 맞춰주는 역할을 한다. Git repo 에 캐싱을 하고 있는 역할을 한다.

쿠버네티스가 이해할 수 있는 manifest 파일로 변경해주는 역할도 한다.

Application controller

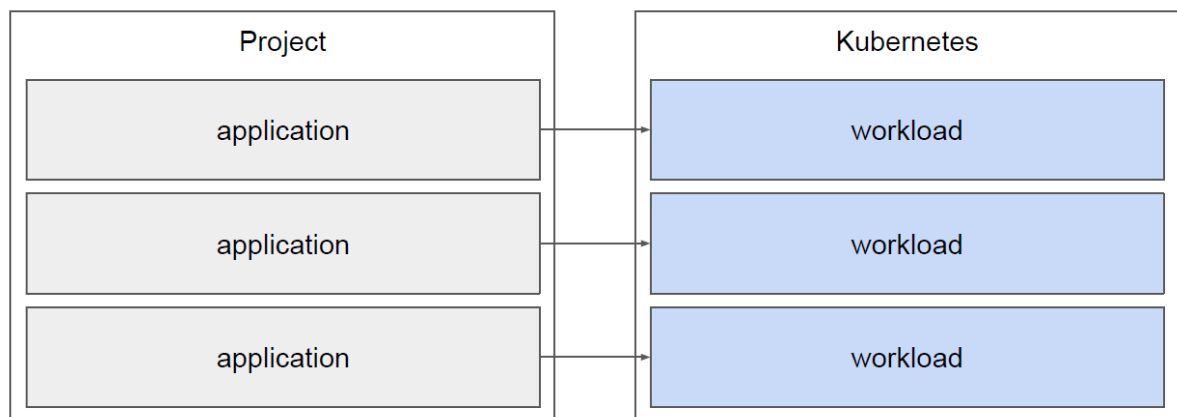
쿠버네티스에 컨트롤러 역할을 하는 컴포넌트로 current state 와 desired state 를 비교하는 역할을 한다.

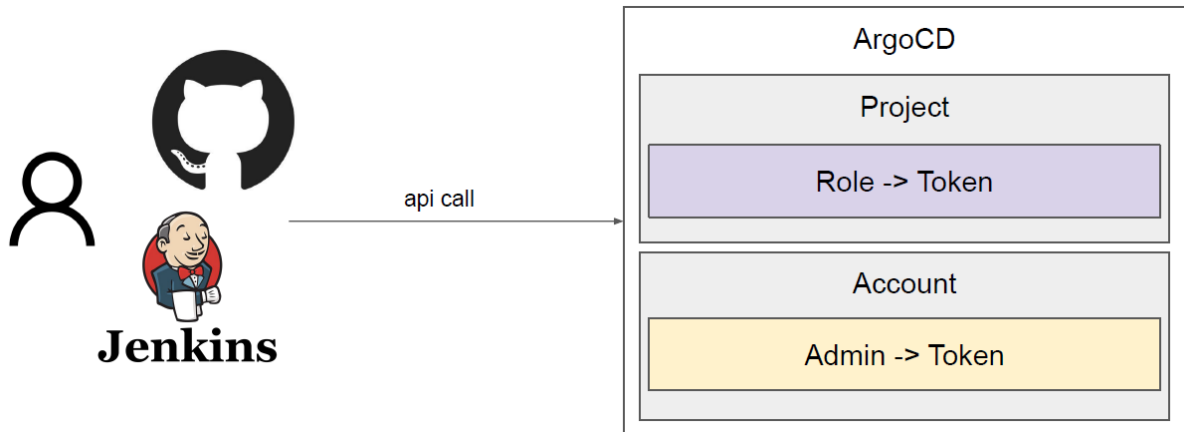


Project

프로젝트라는 단위 쿠버네티스의 namespace와 같은 개념이다. 여러 개의 어플리케이션을 하나로 묶는 역할을 한다.

어플리케이션은 쿠버네티스의 하나의 워크로드와 매핑이 된다.





Role

프로젝트에 Role을 생성하고 권한(특정 Application을 read, create...) 을 주어 Token을 발급한다. Token을 통해 api call을 날린다. 특정 프로젝트에 국한된 권한을 준다.

Account

Account 레벨의 권한을 생성한다. (예. Admin, visitor...계정 생성) 여러 개의 프로젝트에 권한을 효율적으로 생성 가능하다.

3. 배포하기

1) Repository 연동

ArgoCD와 Git을 연동하는 방식에는 SSH방식, HTTPS방식, Github App 방식이 있지만, HTTPS 방식으로 연동해보겠다.

타입은 Git이고, Repository URL에 주소를 넣어주면 된다.

CONNECT
SAVE AS CREDENTIALS TEMPLATE
CANCEL

Connect repo using HTTPS

Type




git

Project

default

Repository URL

https://github.com/limes22/study

TYPE	NAME	REPOSITORY	CONNECTION STATUS
	git	https://github.com/limes22/study	 Successful 

2) 애플리케이션 배포하기

ArgoCD에 Applications 탭에서 NEW APP 을 클릭한다.

GENERAL
EDIT AS YAML

Application Name

suejin-ui

Project

default

SYNC POLICY

Manual

- Application Name: 어플리케이션 이름을 지정해준다.
- Project: ArgoCD에서 애플리케이션을 구분하고 관리할 논리적인 그룹을 지정해준다.
- SYNC POLICY: Manual과 Automatic 중에 선택하며, Git의 환경과 비교하는 것을 수동으로 할지 자동으로 할지 선택한다.
- Prune Resources: 변경사항에 따라 리소스를 업데이트 할 때 기존의 리소스를 삭

제하고 새로운 리소스 생성

- Self Heal: ArgoCD가 지속적으로 Git 레포지토리의 설정값과 싱크를 맞추려고함.

SOURCE

Repository URL

<https://github.com/limes22/study.git>

GIT ▼

Revision

HEAD

Branches ▼



Path

CICD

- Repository URL: 사용할 레포지토리 URL
- Revision: Git의 어떤 Revision (HEAD, Master branch 등)을 바라볼지 결정
- Path: Git의 어떤 저장소의 어떤 디렉토리를 바라볼지 결정

DESTINATION

Cluster URL

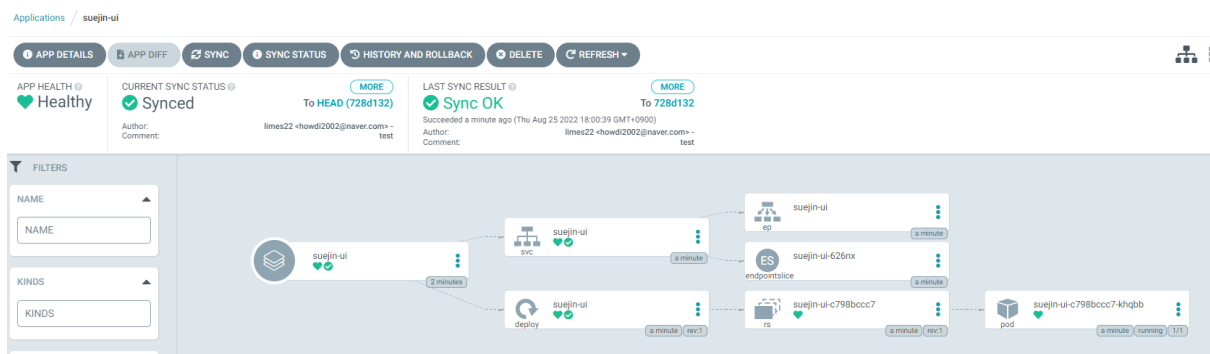
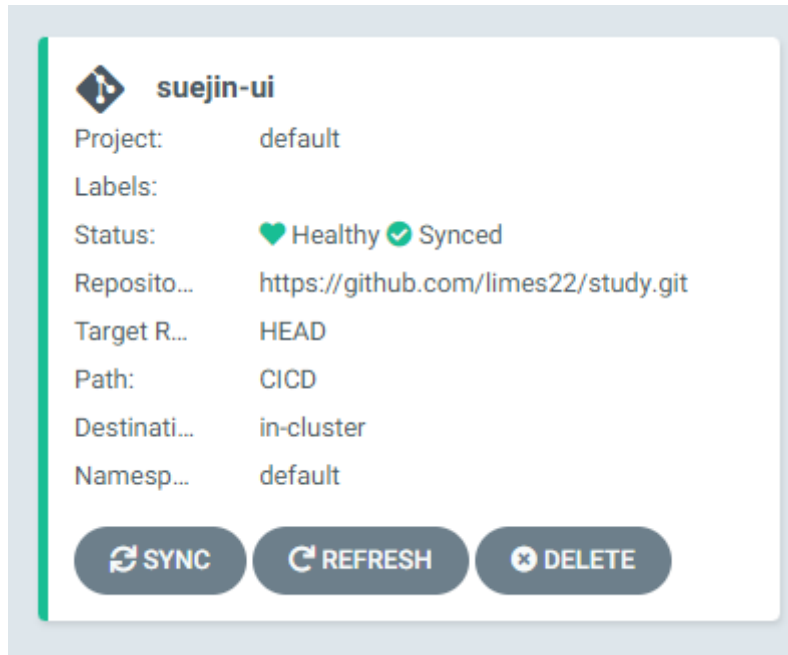
<https://kubernetes.default.svc>

URL ▼

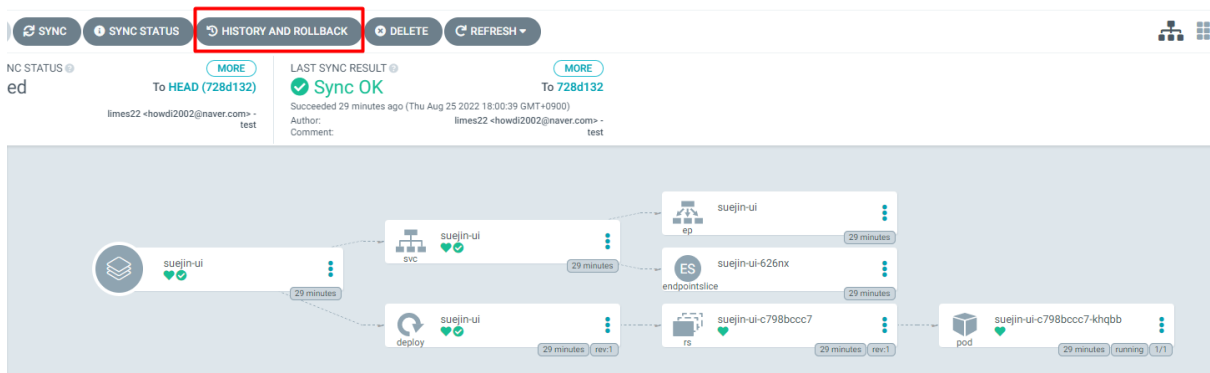
Namespace

default

- ClusterURL: 쿠버네티스의 어느 클러스터에 배포할지 결정
- Namespace: 쿠버네티스 클러스터의 어느 Namespace에 배포할지 결정



3) RollBack



App배포하면 git commit을 history에 revision으로 기록한다. 이후 히스토리에 있는 revision으로 rollback 가능하다. Rollback 시에도 히스토리에 기록한다.

<Reference>

1. gitops - <https://www.weave.works/technologies/gitops/>
2. argocd - <https://argo-cd.readthedocs.io/en/stable/>
3. argocd - <https://github.com/argoproj/argocd-example-apps/tree/master/guestbook>
4. <https://daddyprogrammer.org/post/14102/argocd-kubernetes-cluster-deploy/>
5. <https://waspro.tistory.com/758>