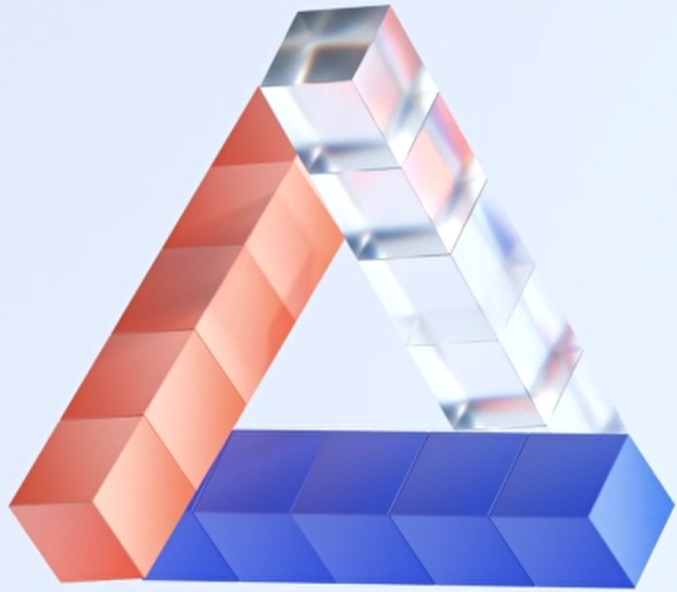




쿠버네티스가 가능하게 한 GitOps

이재상, SK telecom





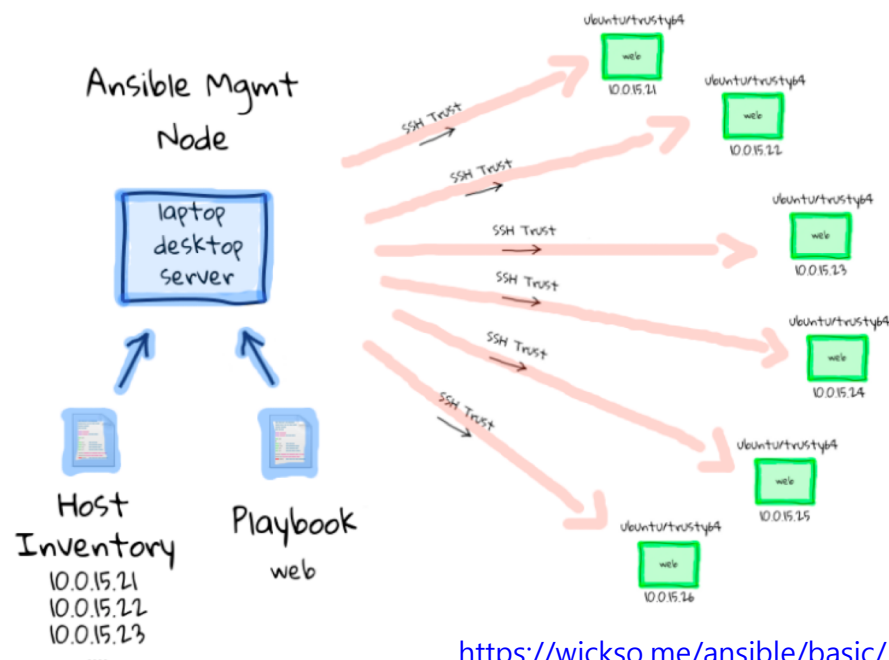
Agenda

- GitOps 첫 추억
- OpenStack과 Kubernetes
- 다시 만난 GitOps
- GitOps 장점
- GitOps 원칙
- 쿠버네티스와 GitOps
- Pull 전략
- GitOps Tools
- GitOps in TACO



2015년 H사의 H*** OpenStack

- Ansible을 통해 OS 설치부터 OpenStack 배포까지 지원
- 노드 정보, OpenStack 설정을 Local Git Repo로 관리
- Ansible 실행 시 Git Repo 상태를 점검하여 non-commit 상태의 파일이 존재할 경우
• 에러 발생.
 - 변경내역 기록 가능: 누가, 무엇을, 언제 변경하였는가
- Ansible을 사용하면 Git Commit을 반드시 해야하기에 변경 내용의 저장을 강제



<https://wickso.me/ansible/basic/>



- https://github.com/hpe-helion-os/helion-ansible/blob/master/hp_hlm_ansible/roles/git-operations/tasks/check-repo-status.yml

```
17  # Check if a git repo has all changes committed.
18  # Parameters:
19  #   repo: ~/helion
20  ---
21  - name: git-operations | check-repo-status | Check for uncommitted changes
22    shell: >
23      git diff-index --quiet HEAD &&
24      [ -z "$(git ls-files --others --exclude-standard)" ]
25    args:
26      chdir: "{{ repo }}"
27    register: git_result
28    ignore_errors: true
29
30  - name: git-operations | check-repo-status | Report uncommitted changes
31    fail:
32      msg: >
33          You have uncommitted changes in {{ repo }}. You should review these
34          (with a "git status" command) and resolve them before proceeding.
35    when: git_result|failed
```



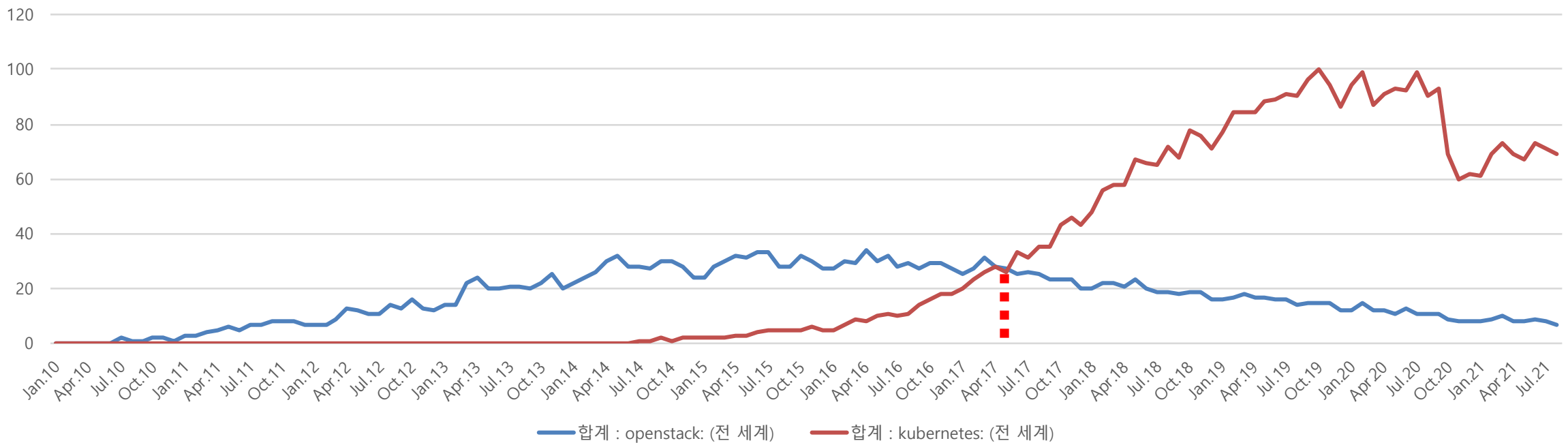

- 그러나 Ansible을 사용하지 않는 경우는 막을 수가 없음
 - Git repo를 변경안하고, 직접 서버에 접속하여 환경파일 변경
 - Git repo의 소스와 서버에 있는 설정파일 사이에 차이가 생기기 시작
 - 지금 배포된 소스가 무엇이지??
- 알 수 없는 Git commit의 증가
 - 트러블슈팅을 하면서 무의미한 커밋 메시지 축적
 - Test
 - Test2...
 - 히스토리 파악 불가

GitOps 첫 추억 그 이후...



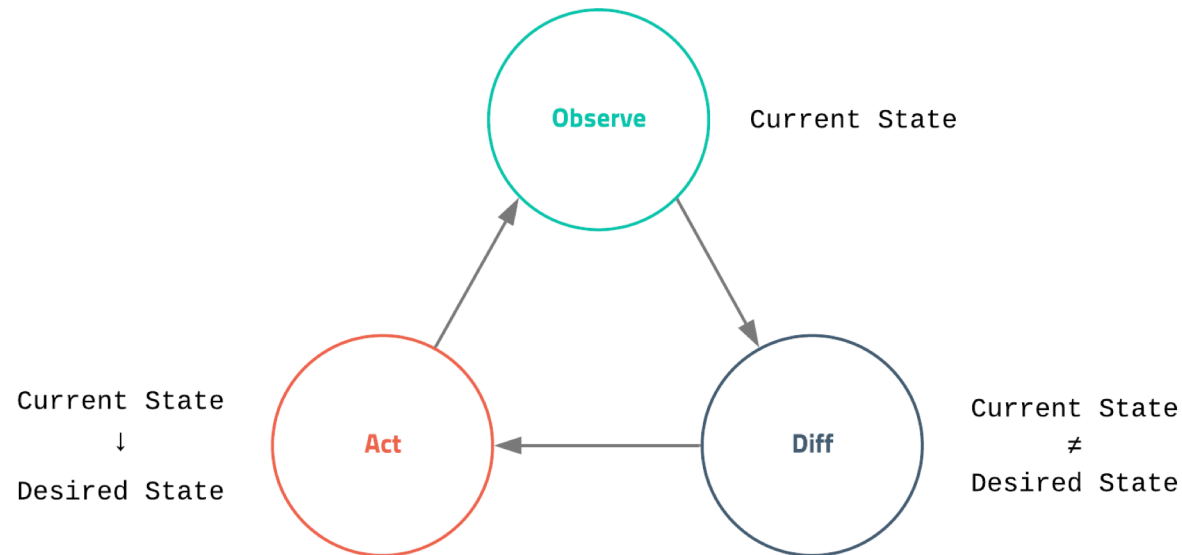
- Cloud 리소스, 트렌드가 VM에서 Container로 이동
- OpenStack -> Docker, Kubernetes

Google Trends: OpenStack vs Kubernetes





- Container App의 배포, 확장, 관리 시스템
- Container Scheduling, Service Scaling, Self-healing ...
- 모든 리소스를 선언적(Declarative) 방법으로 관리할 수 있다.

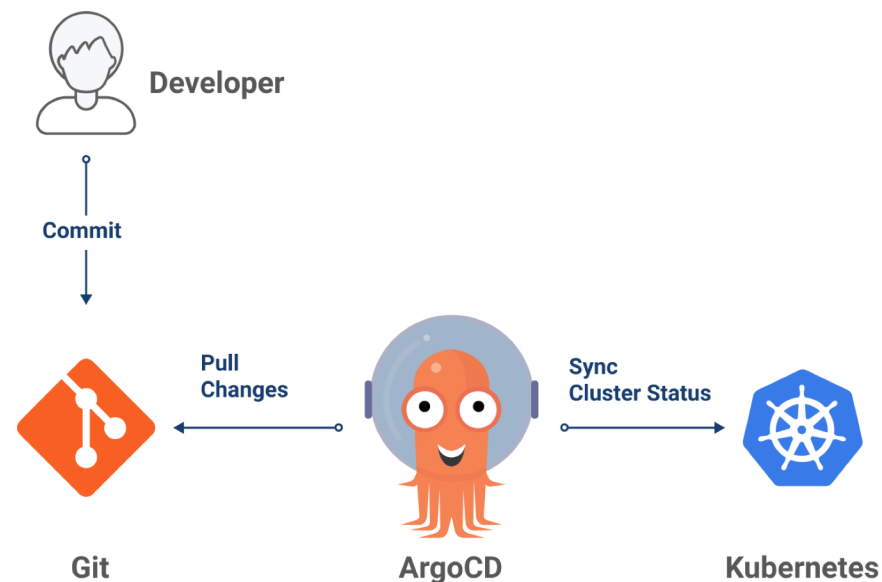


<https://subicura.com/2019/05/19/kubernetes-basic-1.html>

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 80
```



- ArgoCD
 - Kubernetes의 리소스를 Git에 저장하면, Git을 읽어와 Kubernetes에 배포
- ArgoCD는 Kubernetes 리소스와 Git의 저장된 리소스를 비교하여 내용이 다를 경우 이를 Git의 저장된 리소스로 Sync
 - 항상 같은 내용을 보장
- Github의 PR을 통해 변경내용에 대한 사전 리뷰/승인 가능
- 배포된 내용이 잘못 되었을 경우 Git revert를 통해 원복가능

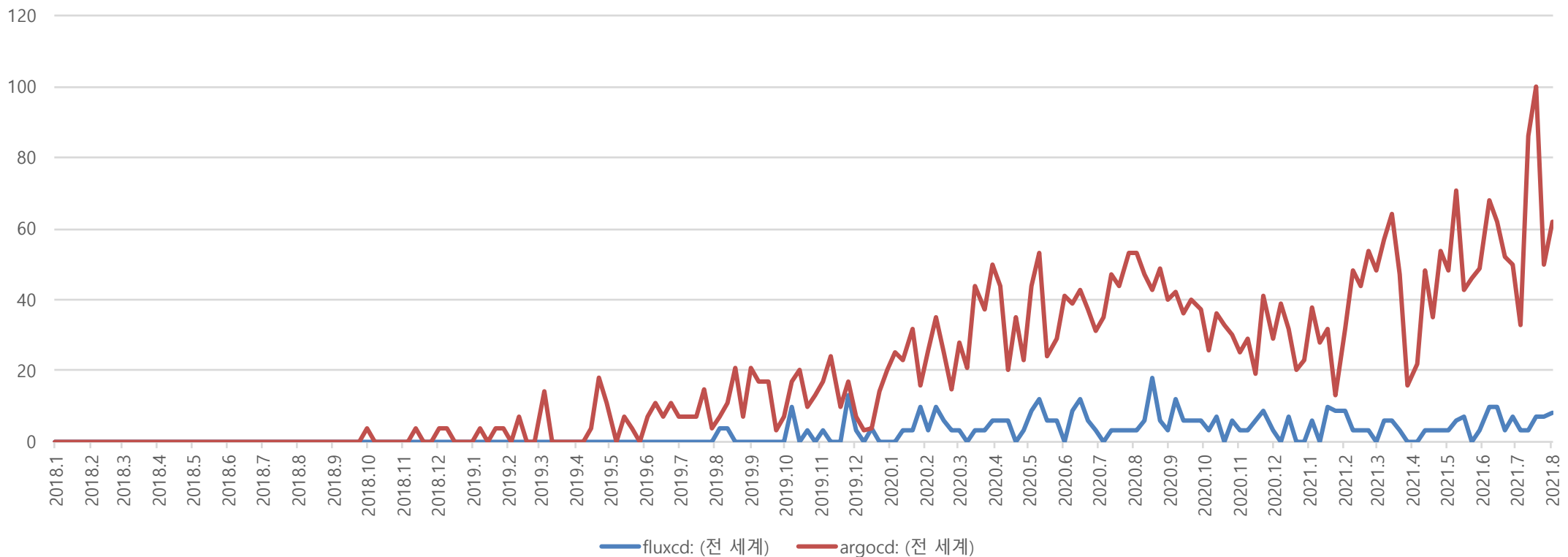


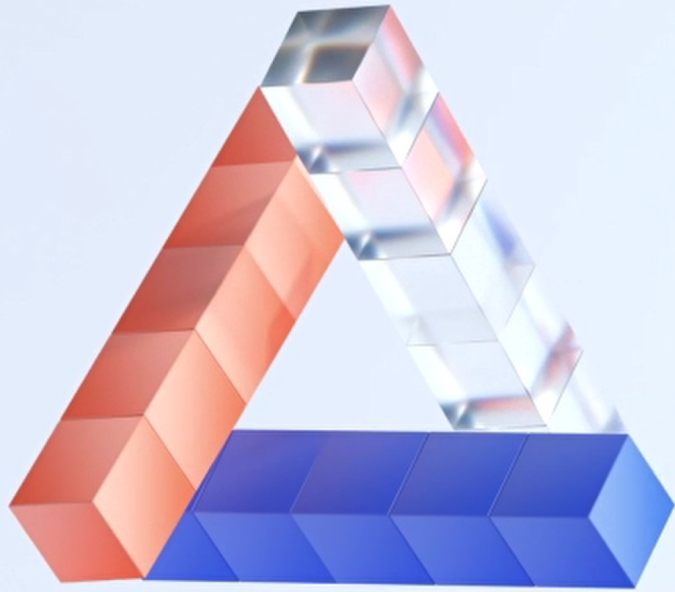
다시 만난 GitOps



- 이전의 GitOps에서는 타겟과 소스가 다를 경우 이를 조정해주는 컨트롤 서비스가 부재하였으나 Kubernetes에서 이 역할을 하는 서비스가 등장
 - ArgoCD와 Flux

Google Trends: Fluxcd vs ArgoCD

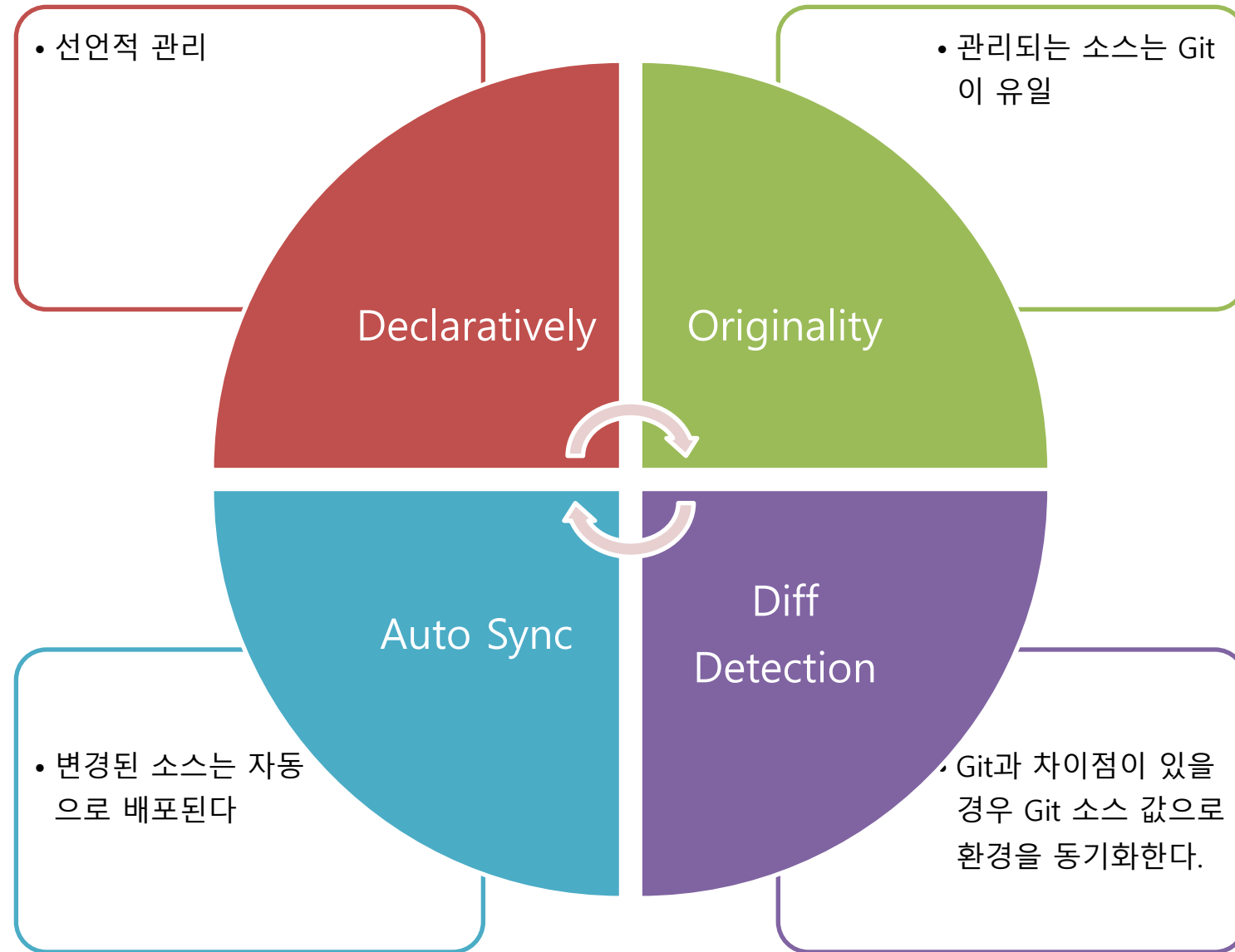




GitOps Quickstart

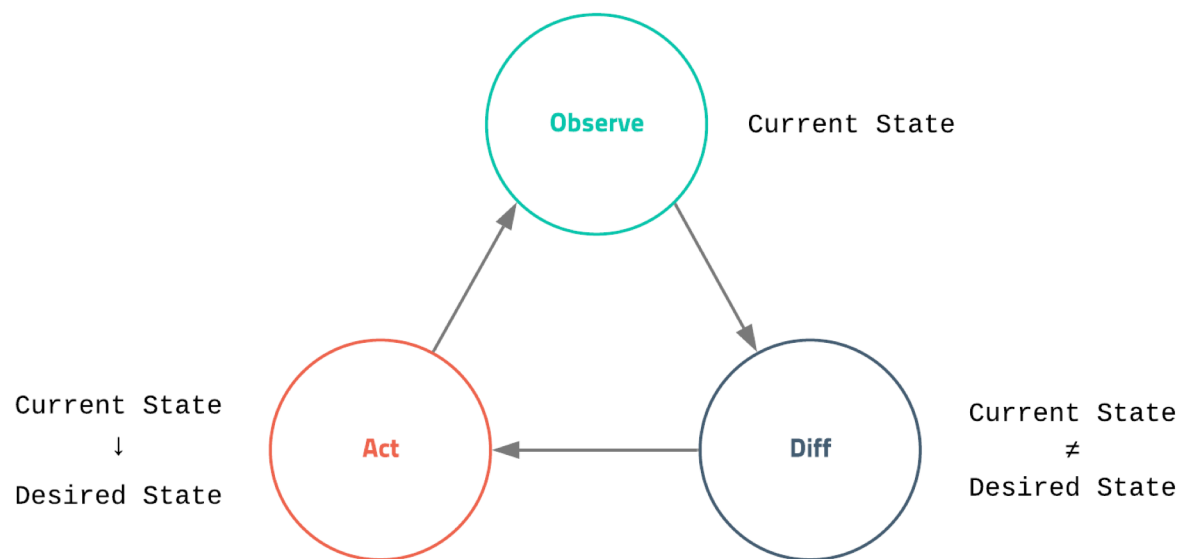


- 개발자 친화적
 - 소스코드 개발/빌드 시 사용하는 Git을 인프라/서비스 배포에도 사용
- 코드 리뷰
 - 테스트 환경에 선배포하거나 운영 환경 배포 전 리뷰 가능
- 타겟과 소스의 분리
 - 타겟에 접근할 필요 없이 Git repo를 통해 타겟 변경
- 타겟의 내용은 Git과 항상 같다
 - 형상에 대해 의심하지 않아도 된다.





- 쿠버네티스의 모든 리소스는 선언적 리소스를 바탕으로 컨트롤러, 오퍼레이터가 관리한다.
- Git에 저장된 이러한 선언적 리소스를 쿠버네티스에 적용하면 배포는 알아서 진행된다.



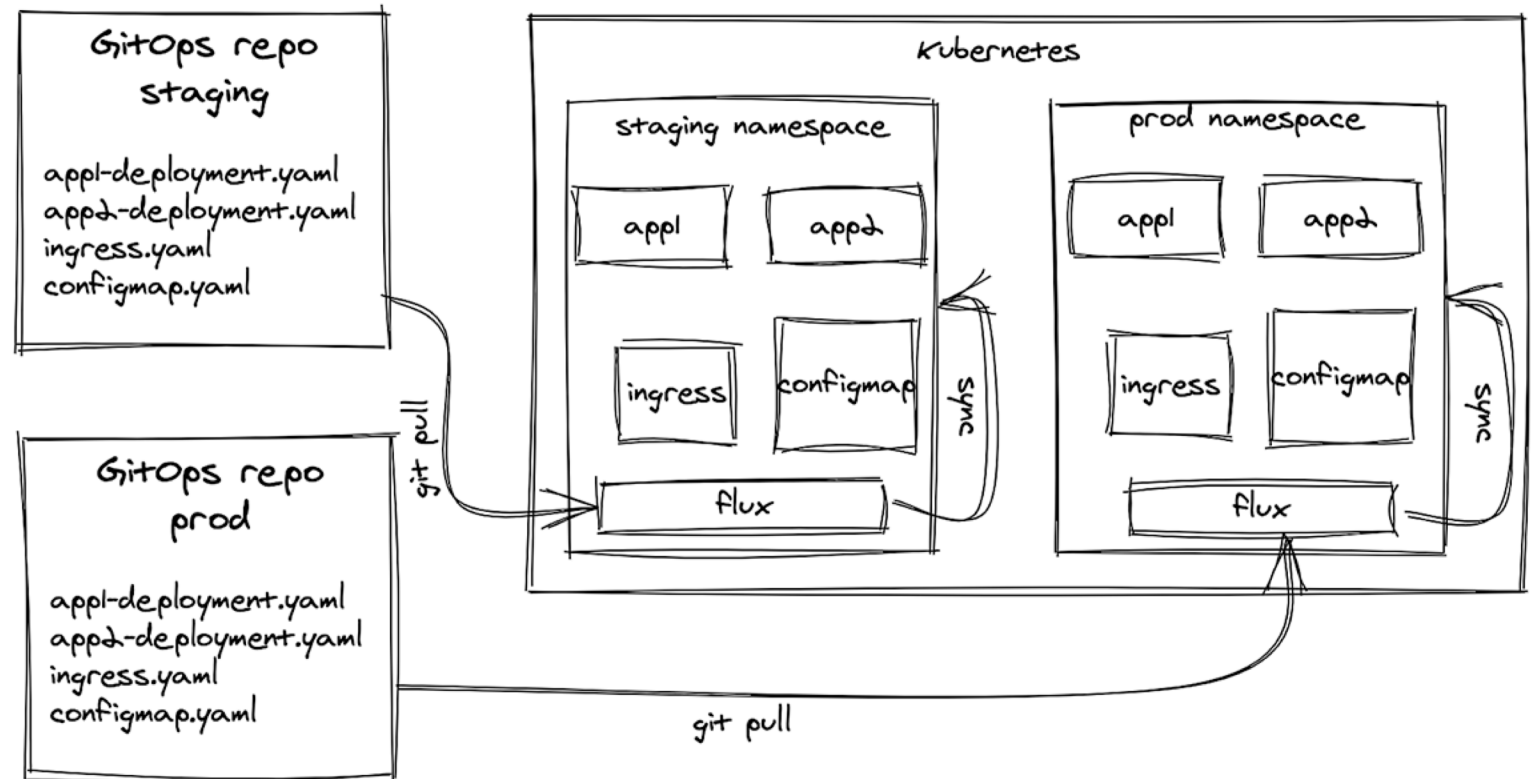
<https://subicura.com/2019/05/19/kubernetes-basic-1.html>



- 배포를 위해 Jenkins에서 Job을 수행하는 것은 Push
- Code repo에 commit을 하면 컨트롤러에 의해 자동으로 적용되는 것은 Pull
- Push방식은 클러스터의 외부에서 클러스터의 접근 권한이 노출될 수 있는 잠재적 위험 존재
- Pull방식은 클러스터 내부에서 소스를 받아 처리하기 때문에 이러한 보안 리스크를 줄임



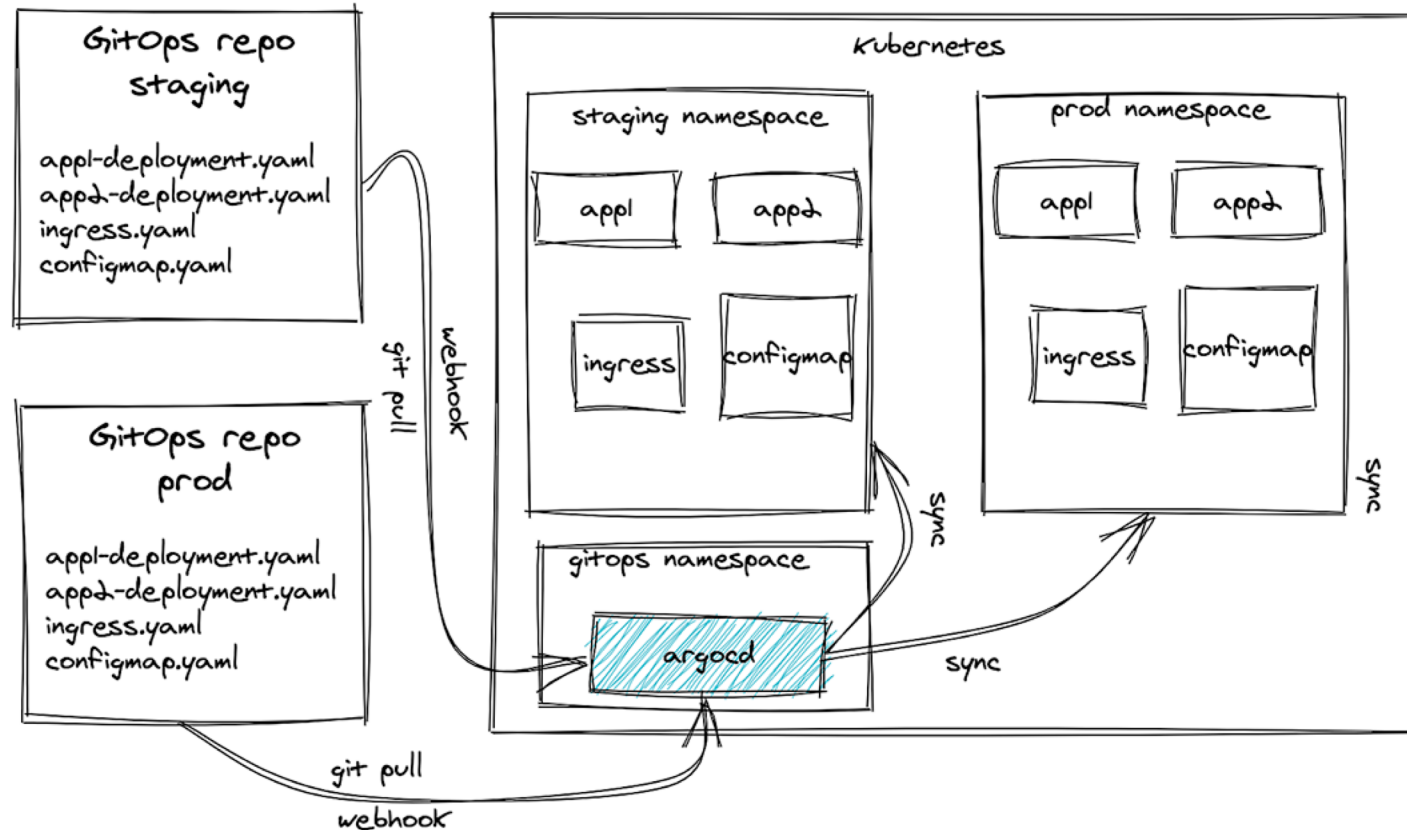
- Flux
 - <https://fluxcd.io/>



<https://blog.container-solutions.com/fluxcd-argocd-jenkins-x-gitops-tools>



- ArgoCD
 - <https://argoproj.github.io/argo-cd/>

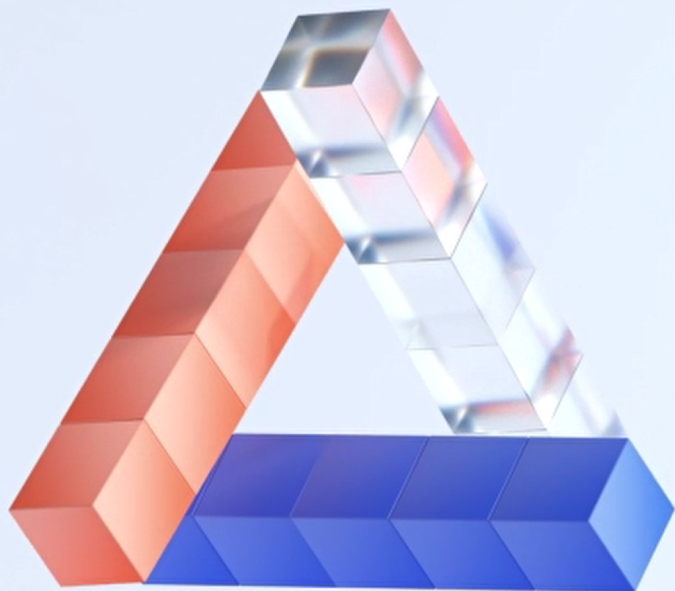


<https://blog.container-solutions.com/fluxcd-argocd-jenkins-x-gitops-tools>



- ArgoCD

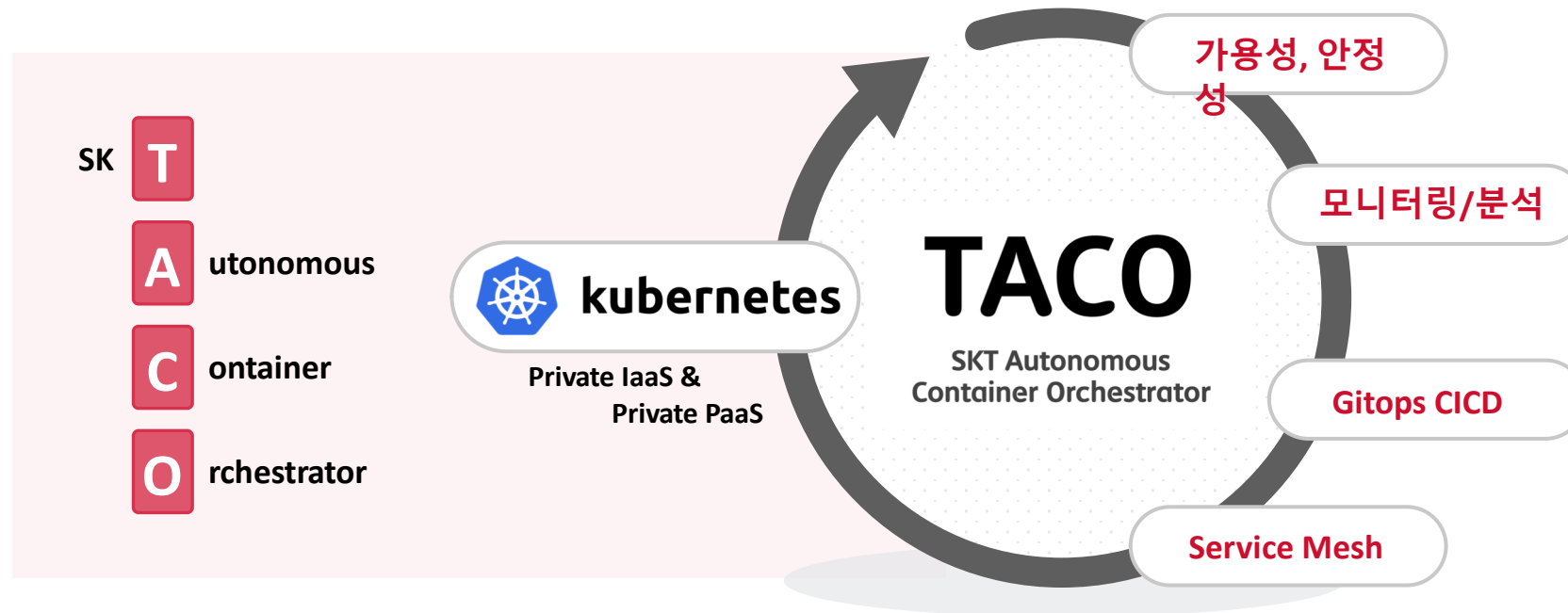




GitOps in TACO



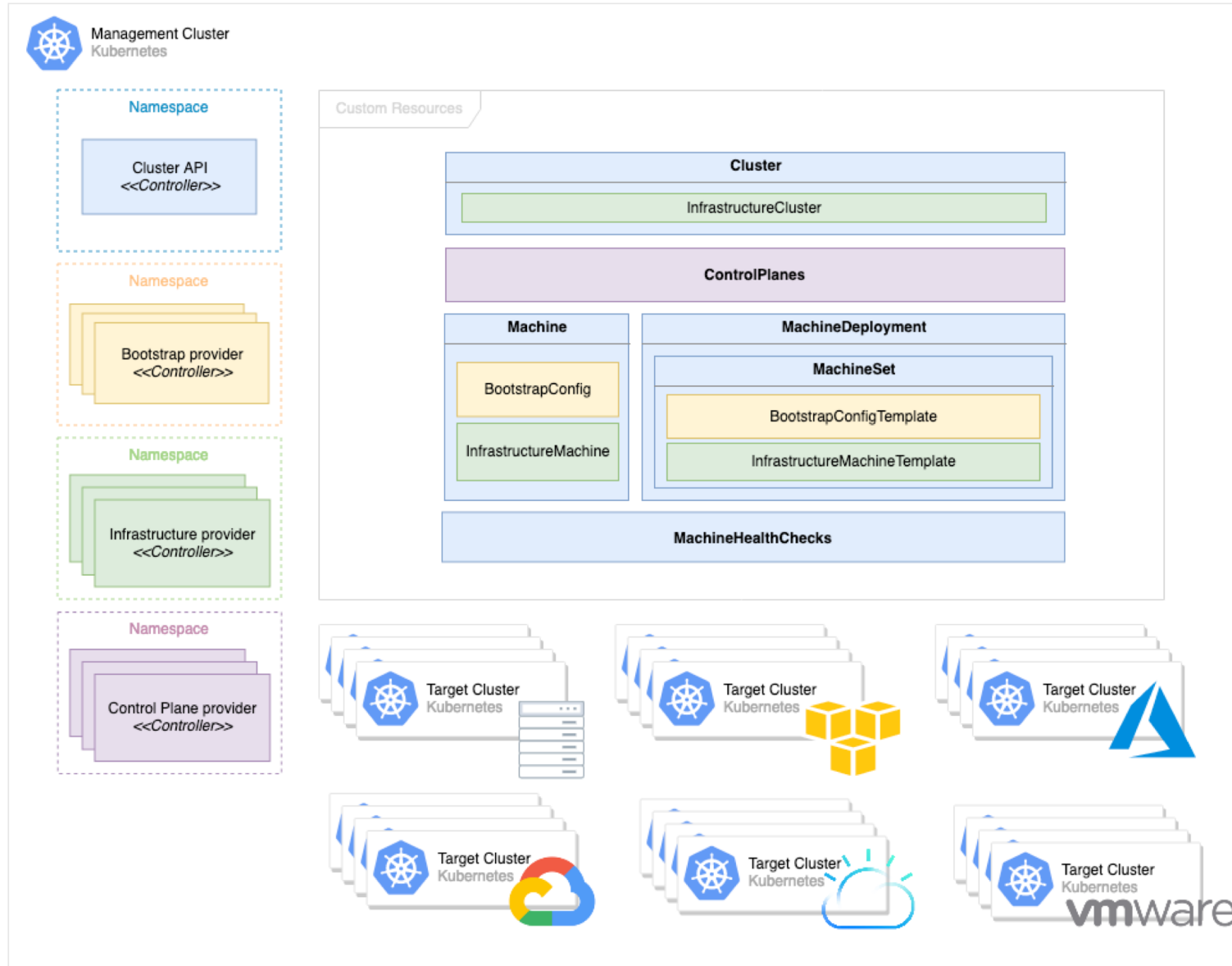
- TACO (SKT **A**utonomous **C**loud **O**rchestrator)
 - 쿠버네티스 기반으로 컨테이너화된 클라우드 환경을 제공하고, MSA(Micro Service Architecture) 형태의 App 개발을 위한 다양한 PaaS 기능 제공





- TACO (SKT **A**utonomous **C**loud **O**rchestrator)
 - 쿠버네티스 배포를 Ansible에서 Cluster-API로 변경
 - Application 배포/관리를 위해 Decapod 프로젝트 시작

GitOps in TACO

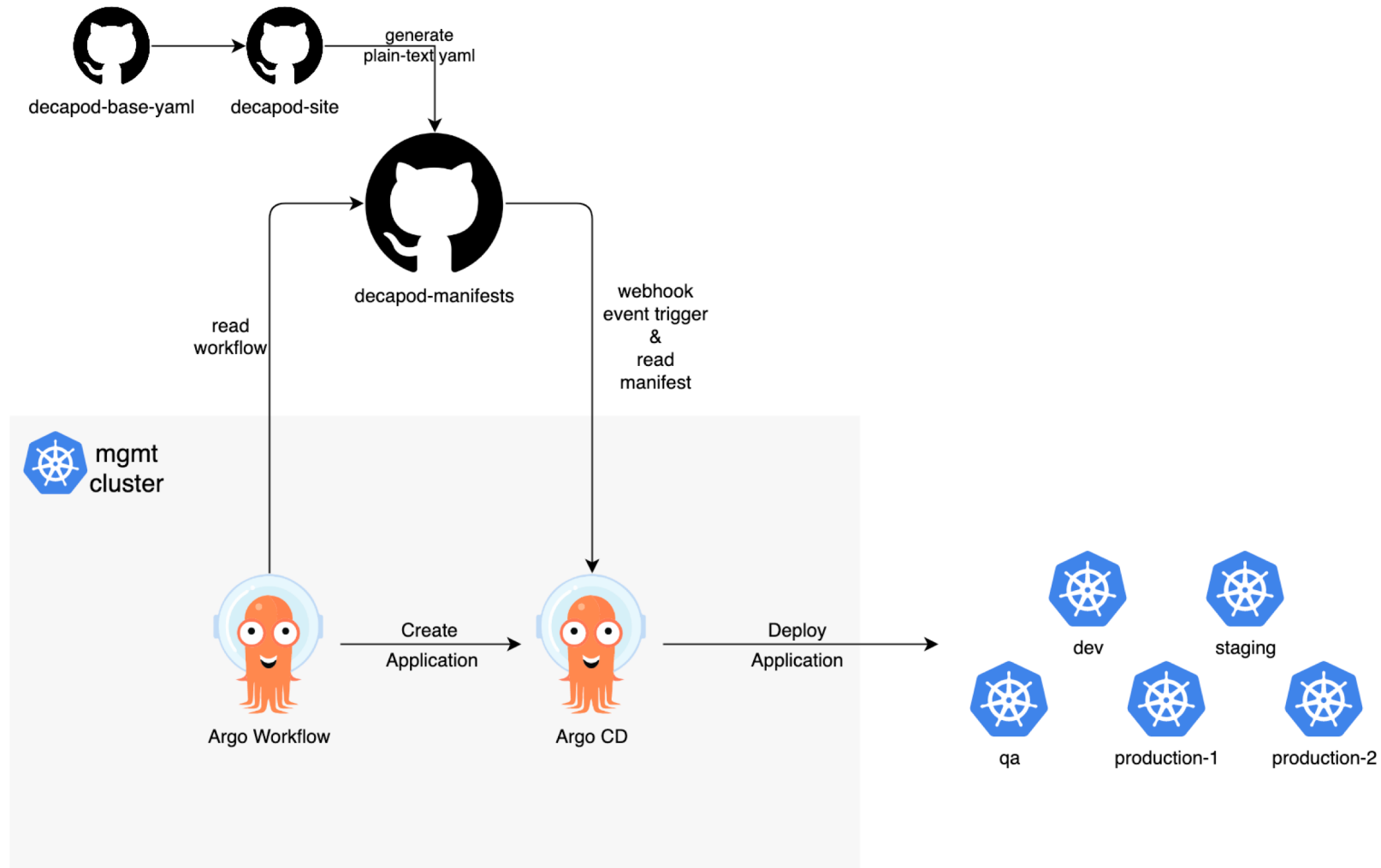




- Decapod
 - <https://openinfra.dev/github.io/decapod-docs/>
 - Decapod은 여러 Kubernetes Cluster 환경에 비슷한 Application Group 들을 배포해야할 때,
 - 효과적인 Yaml Documentation과 Deployment Pipeline 제공
 - TACO에서는 쿠버네티스와 App들의 대한 모든 정보를 Decapod으로 관리



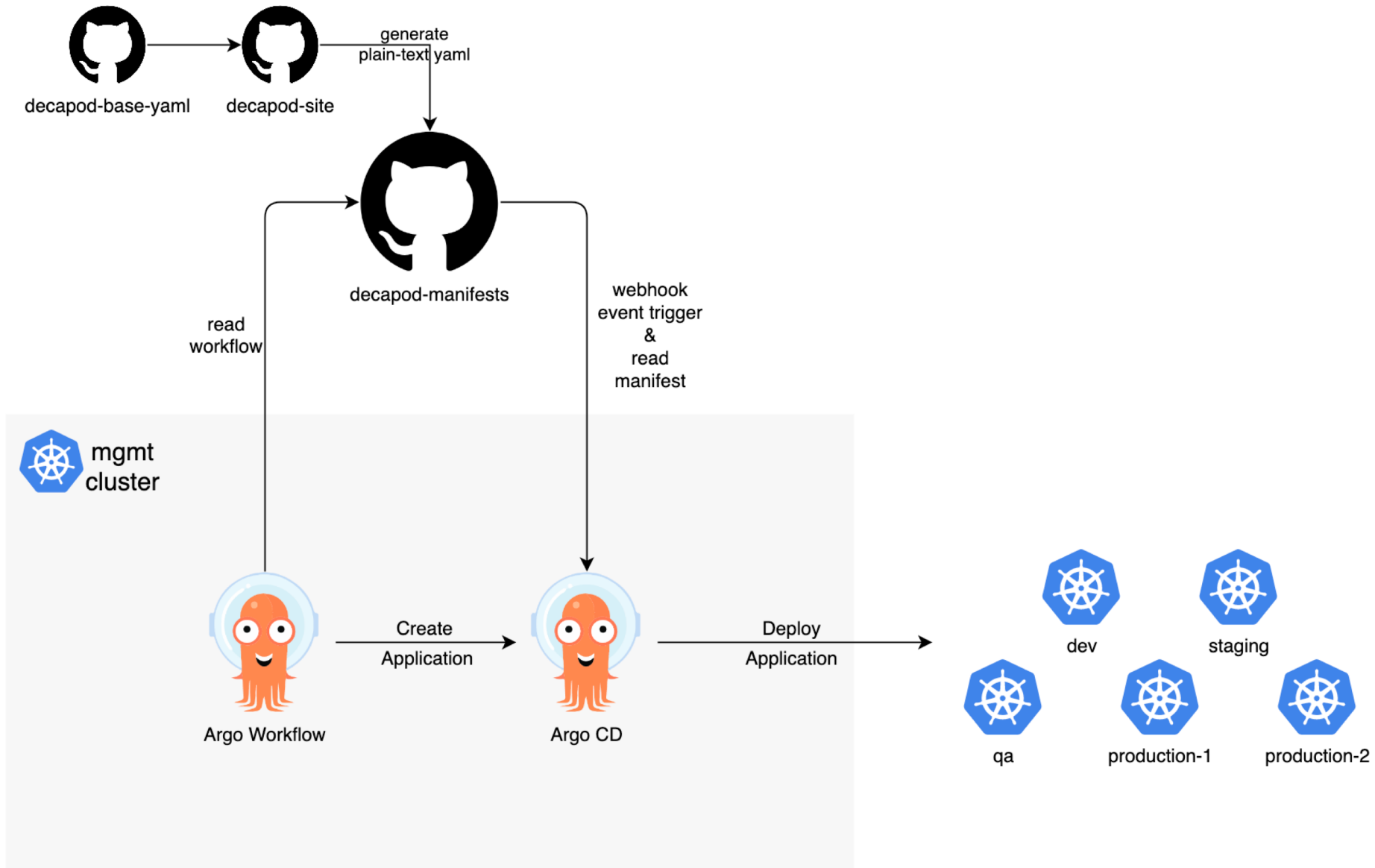
- Decapod





- ArgoWorkflow와 ArgoCD
 - Decapod으로 작성된 쿠버네티스 리소스 배포
 - 쿠버네티스 리소스 yaml을 Git에 저장하고, Kubernetes 클러스터와 App의 대한 설정을 ArgoCD가 관리
 - ArgoCD에 App을 최초 등록하기 위해 ArgoWorkflow 사용

Decapod Demo





- Decapod Official Site
 - <https://openinfra.dev.github.io/decapod-docs/>
- Presentation
 - Kubernetes Community Days (Sep, 9)
 - <http://kcd-korea.net/>
 - OpenInfra Days Asia 2021 (Sep, 11)
 - <https://2021.openinfra.asia/>