파스-타 Sidecar 활용하기

2022.10.27.

개방형 클라우드 플랫폼 센터 **남동윤**

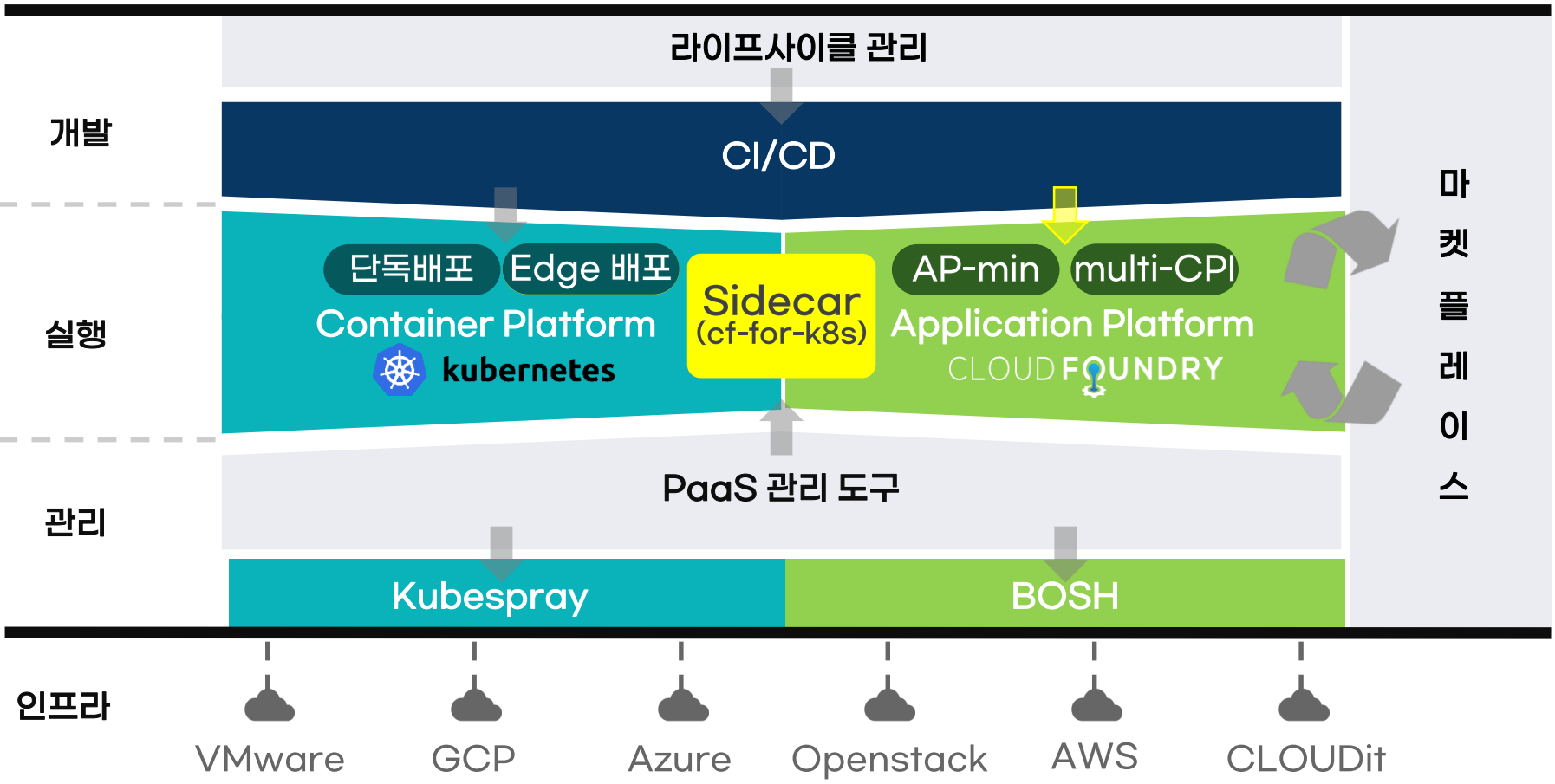
NIA 한국지능정보사회진흥원

PaaS-TA

- 파스-타 Sidecar 소개
- Paketo Buildpack 소개
- Sidecar 앱 배포 과정
- 배포 사례 - Portal Application

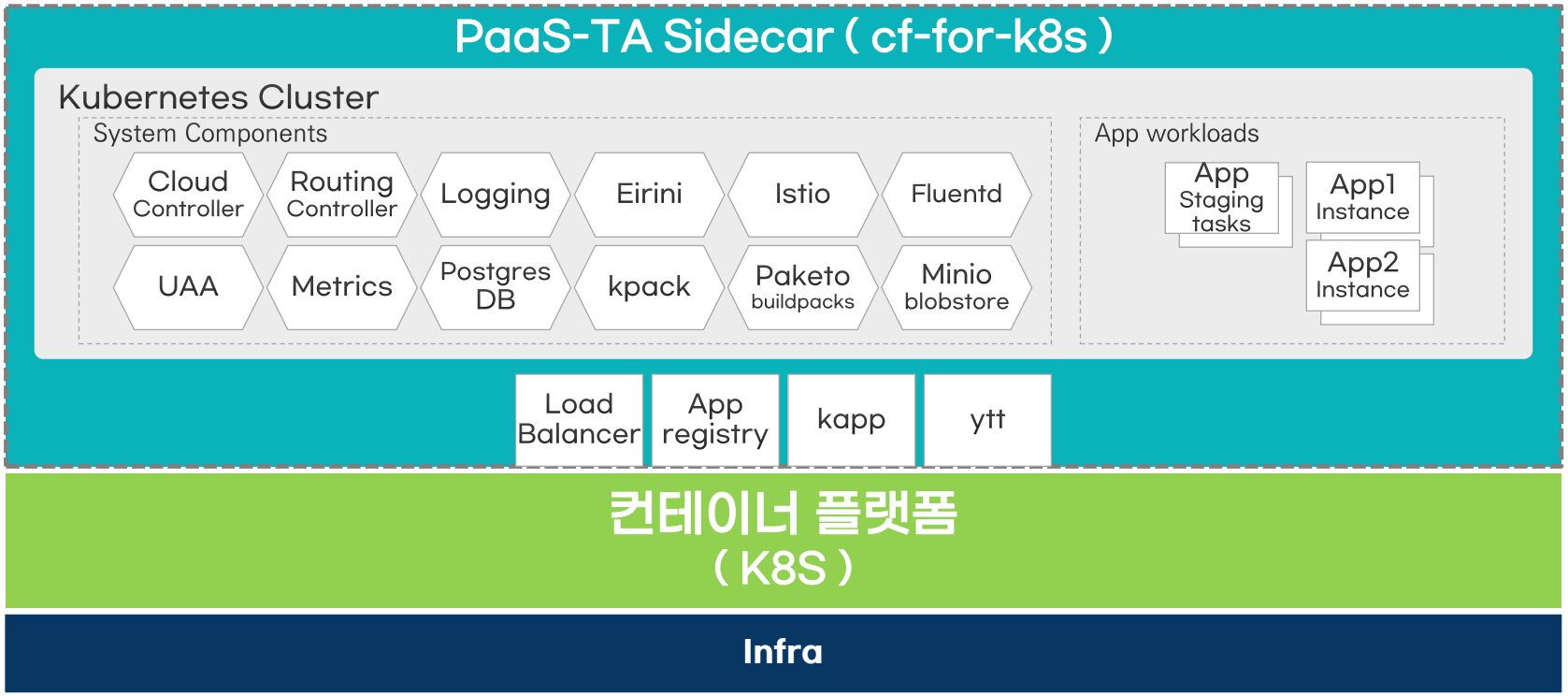
파스-타 Sidecar 소개

파스-타 구성



파스-타 Sidecar 소개

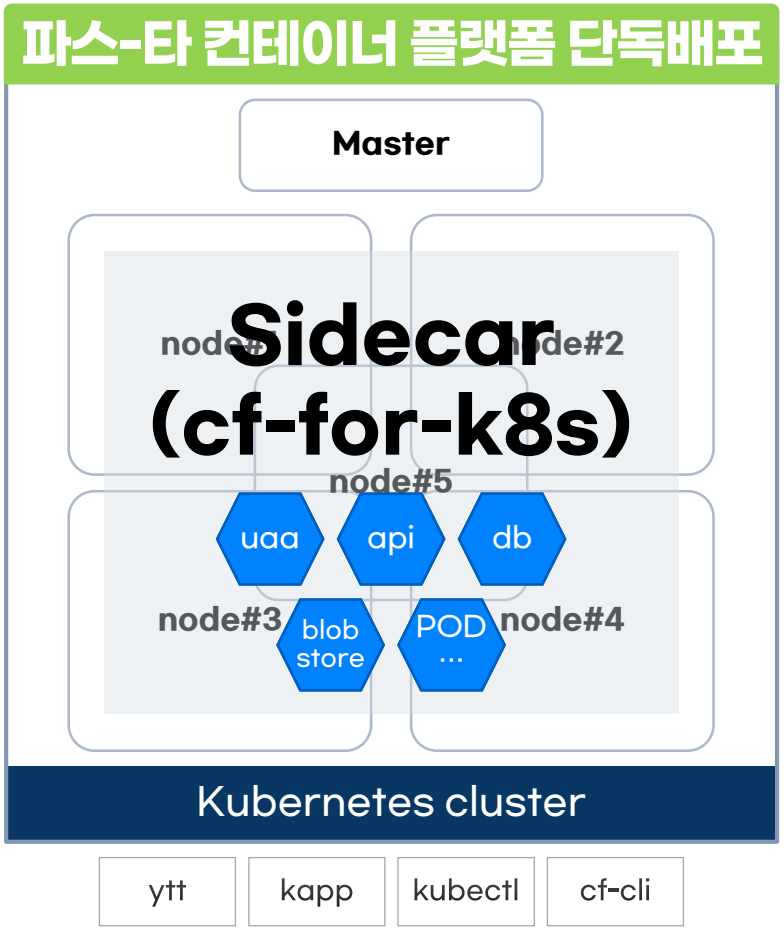
파스-타 Sidecar 컴포넌트



- 파스-타 Application Platform과 동일한 Cloud Controller, UAA 등의 컴포넌트를 사용
- Kubernetes에 컴포넌트가 POD 형태로 실행

https://github.com/cloudfoundry/cf-for-k8s

파스-타 Sidecar 배포 전제 조건



필요 도구

- ytt
- kapp
- kubectl
- cf cli v7+

클러스터 요구사항

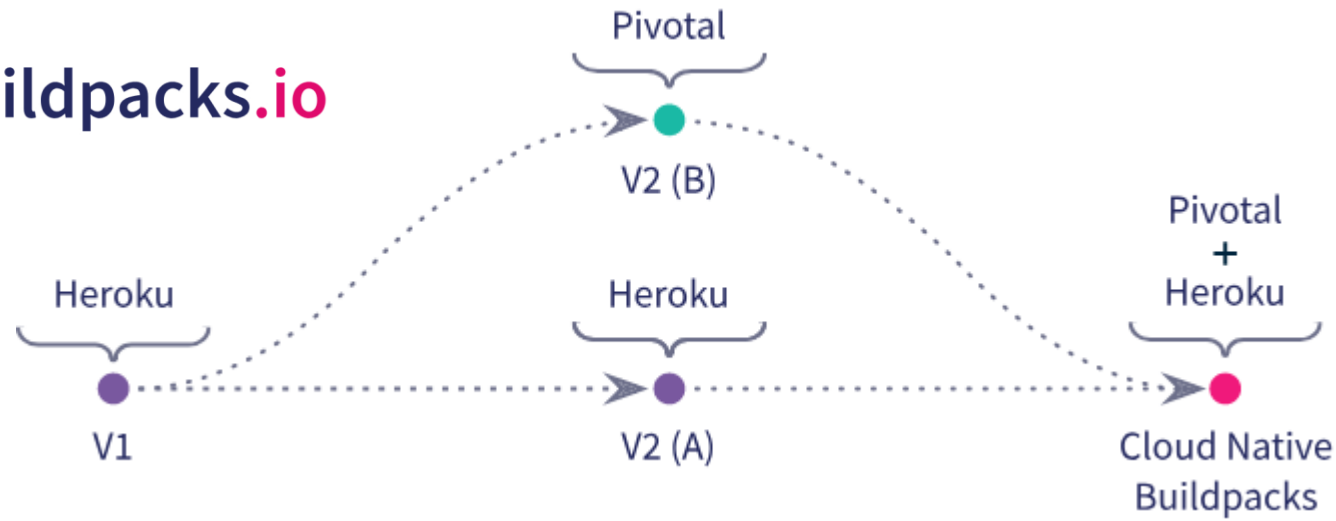
- Kubernetes v1.19.x ~ v1.24.x
- 최소 5개 노드 필요
- 노드 당 최소 4 CPU, 15GB memory
- CNI plugin, LoadBlancer service, StorageClass 등

ContainerRegistry

- hub.docker.com 계정
- Google/Azure ConatinerRegistry
- Private Registry (harbor….)

Cloud Native Buildpacks

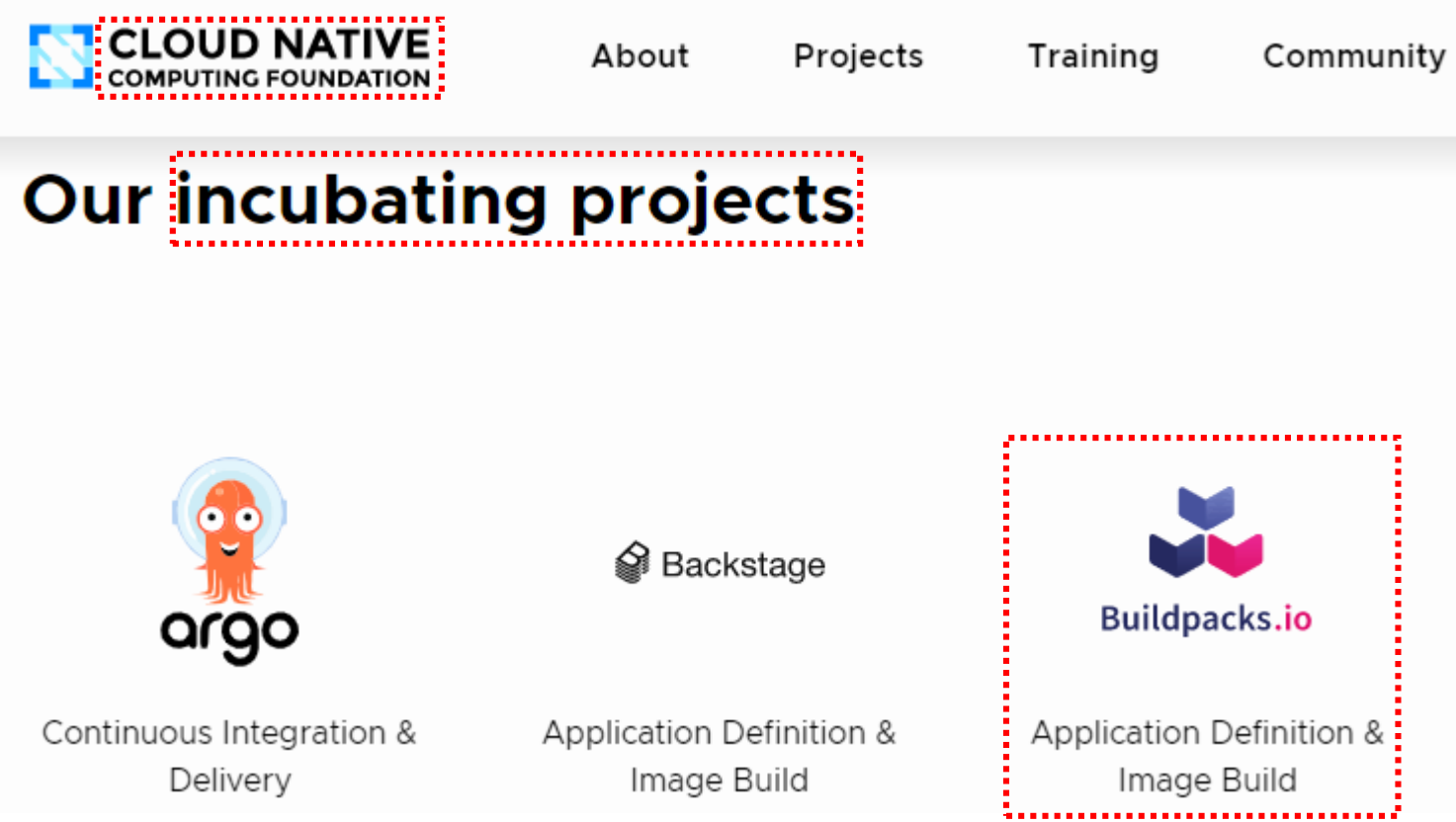
Buildpacks.io



- 빌드팩은 Heroku에 의해 처음 고안되었으며 이후 다양한 PaaS에 채택됨
- 이후 Pivotal과 Heroku에서 사용되는 빌드팩 생태계를 통합을 목표
- Cloud Native Buildpack은 OCI 이미지 형식과 같은 현대적인 컨테이너 표준을 수용

https://buildpacks.io/

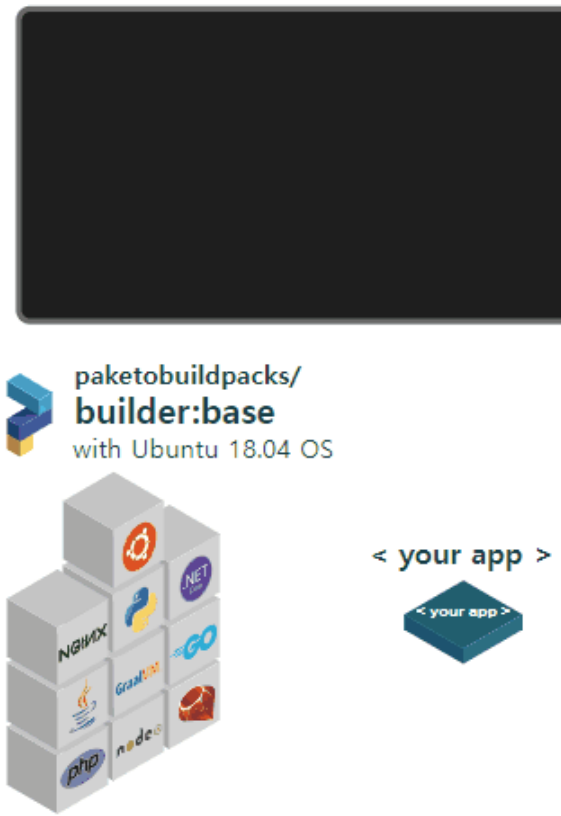
Cloud Native Buildpacks



- Cloud Native Buildpacks는 CNCF의 Incubating 단계에 포함된 프로젝트

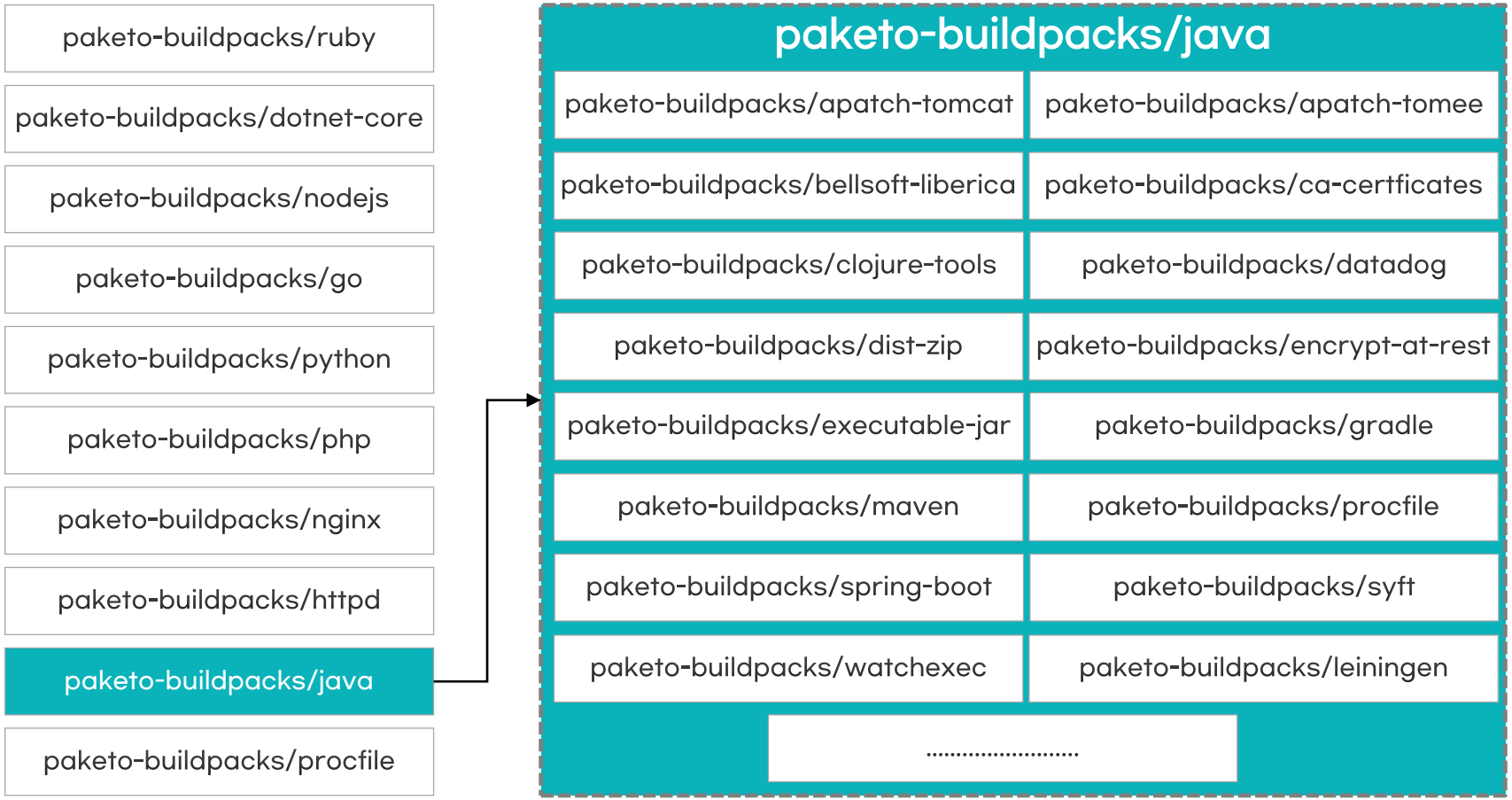
https://www.cncf.io/projects/

Paketo Buildpack 소개

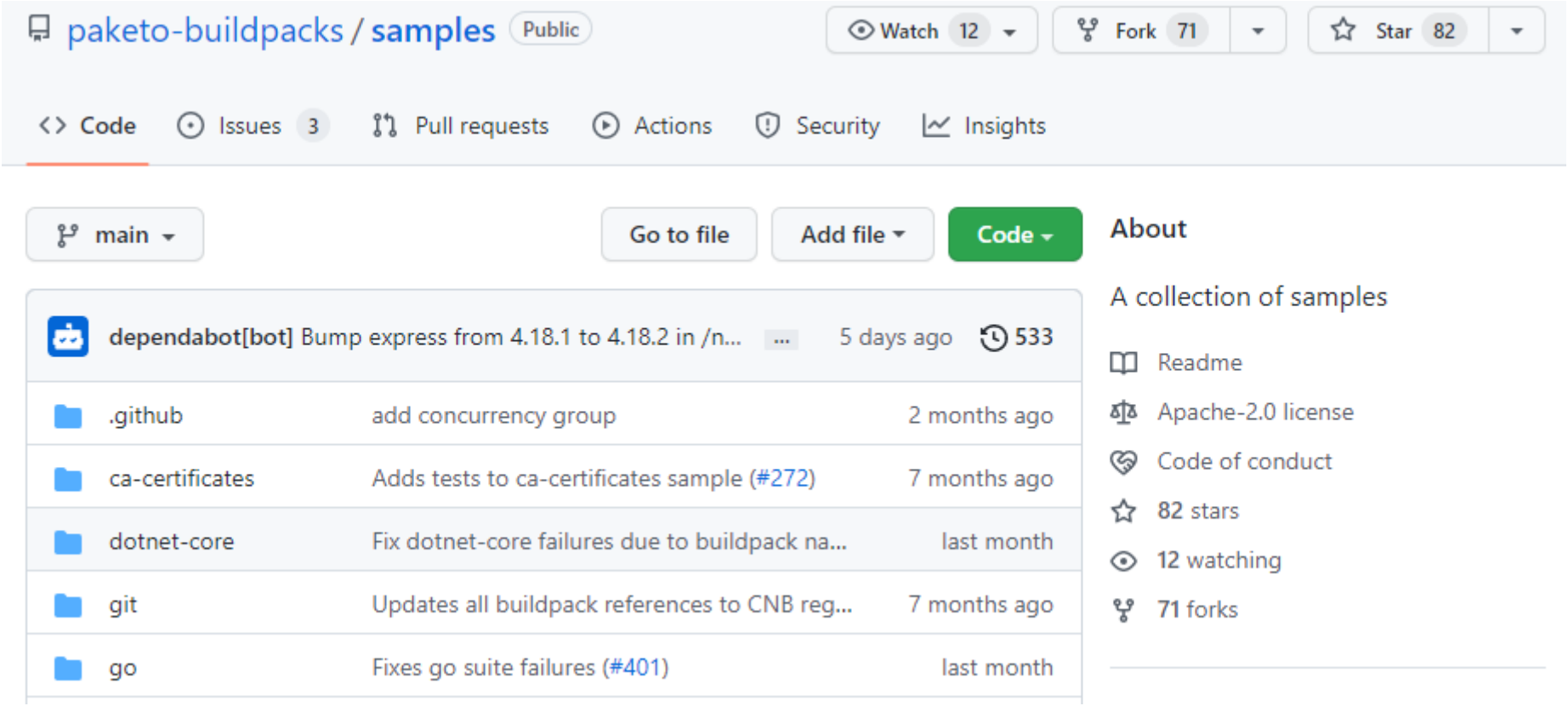


- Paketo Buildpack은 Cloud Native Buildpacks의 규격에 맞춘 빌드팩 중 하나
- Paketo Buildpack은 가장 인기 있는 언어와 프레임워크을 제공
- Paketo Buildpack은 Docker 및 Kubernetes를 포함하는 컨테이너 이미지를 지원하는 모든 플랫폼에서 실행 가능
- 다양한 언어를 감지하고 필요한 Dependency를 수집하여 이미지로 빌드

Paketo Buildpack 종류



Paketo Buildpack Github



- Paketo Buildpack은 테스트 배포가 가능한 Sample을 지원함

https://github.com/paketo-buildpacks/samples

Paketo Buildpack Github

main samples / java / jar / README.md Go to file ...

ekcasey Adds java native image samples ... Latest commit 35a4f06 on 25 Sep 2020 History

19 lines (13 sloc) | 255 Bytes Raw Blame

Pre-compiled Java Sample Application

Building

```
pack build applications/jar
```

- 빌드팩 별 상세 폴더에는 실행 방법을 확인 가능

https://github.com/paketo-buildpacks/samples

Paketo Buildpack 소개

Paketo Buildpack Github

- 언어 빌드팩 Repository 릴리즈 노트 & README.md 파일에서 해당 빌드팩에 포함된 서브 빌드팩 목록을 확인 가능

https://github.com/paketo-buildpacks/java

Paketo Buildpack ENV

main spring-boot / README.md Go to file ...

53 lines (40 sloc) | 4.15 KB Raw Blame

Configuration

Environment Variable	Description
`$BP_SPRING_CLOUD_BINDINGS_DISABLED`	Whether to contribute Spring Cloud Bindings support to the image at build time. Defaults to false.
`$BPL_SPRING_CLOUD_BINDINGS_DISABLED`	Whether to auto-configure Spring Boot environment properties from bindings at runtime. This requires Spring Cloud Bindings to have been installed at build time or it will do nothing. Defaults to false.
`$BPL_SPRING_CLOUD_BINDINGS_ENABLED`	Deprecated in favour of `$BPL_SPRING_CLOUD_BINDINGS_DISABLED`. Whether to auto-configure Spring Boot environment properties from bindings at runtime. This requires Spring Cloud Bindings to have been

https://github.com/paketo-buildpacks/spring-boot

Paketo Buildpack ENV

Configuration

Environment Variable
$BP_SPRING_CLOUD_BINDINGS_DISABLED
$BPL_SPRING_CLOUD_BINDINGS_DISABLED
$BPL_SPRING_CLOUD_BINDINGS_ENABLED

<Spring Cloud Buildpack>

```yaml
---
applications:
  - name: portal-api
    memory: 4G
    disk: 2G
    instances: 1
    buildpacks:
    - paketo-buildpacks/java
    path: ./paas-ta-portal-api.jar
    env:
      server_port: 8080

      BP_JVM_VERSION: 8
      BP_SPRING_CLOUD_BINDINGS_DISABLED: true
```

<Portal API Manifest>

- 파스-타 AP의 경우 Java - Spring Boot에 server.port의 값을 변경하고 싶은 경우 manifest.yml의 env에 server_port 값을 정의
- 파스-타 Sidecar에서도 동일하게 정의가 가능하며, Paketo Buildpack의 환경 변수도 정의 가능

Sidecar 앱 배포 과정

Cloud Controller Process

<AP - Cloud Controller>

```
Process 'cloud_controller_ng'          running
Process 'ccng_monit_http_healthcheck' running
Process 'cloud_controller_worker_local_1' running
Process 'cloud_controller_worker_local_2' running
Process 'nginx_cc'                     running
Process 'route_registrar'              running
Process 'statsd_injector'              running
Process 'file_server'                  running
Process 'routing-api'                  running
Process 'policy-server'                running
Process 'policy-server-internal'       running
Process 'policy-server-asg-syncer'     running
Process 'cc_uploader'                  running
Process 'loggr-udp-forwarder'          running
Process 'loggregator_agent'            running
Process 'loggr-forwarder-agent'        running
Process 'loggr-syslog-agent'           running
Process 'prom_scraper'                 running
Process 'metrics-discovery-registrar' running
Process 'metrics-agent'                running
Process 'bosh-dns'                     running
Process 'bosh-dns-resolvconf'          running
Process 'bosh-dns-healthcheck'         running
```

<Sidecar - Cloud Controller>

```
NAME                                                READY
ccdb-migrate-w6bfh                                  0/2
cf-api-clock-758498f949-pjt9t                       2/2
cf-api-controllers-864f4d4c4b-4gvdk                 3/3
cf-api-deployment-updater-84f544f6f5-6xl64          2/2
cf-api-server-c894c4cf6-s4pkn                       6/6
cf-api-worker-bb7d95957-j64zw                       3/3
eirini-api-7fdf99c999-gjwqj                         2/2
eirini-app-migration-9jcgd                          0/1
eirini-event-reporter-56fd848bb6-jhblv              2/2
eirini-event-reporter-56fd848bb6-xlfwr              2/2
eirini-instance-index-env-injector-76c9c6cf65-78r65 1/1
eirini-task-reporter-5456449fc9-cjgqd               2/2
eirini-task-reporter-5456449fc9-qt24g               2/2
fluentd-5cdvr                                       2/2
fluentd-7hkzn                                       2/2
fluentd-9xbll                                       2/2
fluentd-gmrcn                                       2/2
fluentd-qfjmt                                       2/2
log-cache-backend-85b8dc4fc8-bsfkh                  3/3
log-cache-frontend-6fb76684df-lkxqc                 3/3
metric-proxy-779bf7fd59-sx7vh                       2/2
routecontroller-fcfd89646-hrv5t                     2/2
uaa-687744767d-wzzmp                                3/3
```

- CF API v3에 대한 상세 정보는 CF API Docs에서 확인 가능
- AP와 Sidecar는 Cloud Controller 라는 동일한 모듈을 통해 CF API v3로 동작

https://v3-apidocs.cloudfoundry.org/

Sidecar 앱 배포 과정

앱 배포 시 호출되는 CF API v3 리스트

HTTP Method	Request	설명
POST	/v3/spaces/:guid/manifest_diff	Space의 Manifest diff를 생성합니다.
POST	/v3/spaces/:guid/actions/apply_manifest	Space의 Manifest를 적용합니다.
POST	/v3/resource_matches	Resource Match를 생성합니다
POST	/v3/packages	Package를 생성합니다
POST	/v3/packages/:guid/upload	Package에 파일을 업로드 합니다
POST	/v3/builds	Build를 생성합니다
PATCH	/v3/apps/:guid/relationships/current_droplet	Current Droplet을 설정합니다
POST	/v3/apps/:guid/actions/restart	App을 재실행합니다

https://v3-apidocs.cloudfoundry.org/

Sidecar 앱 배포 과정

App Lifecycle

<Lifecycle Type - buildpack>

Example Buildpack Lifecycle

```
{
  "type": "buildpack",
  "data": {
    "buildpacks": ["java_buildpack"],
    "stack": "cflinuxfs3"
  }
}
```

<Lifecycle Type - Kpack>

Example Kpack Lifecycle

```
{
  "type": "kpack",
  "data": {
    "buildpacks": ["paketo-buildpacks/java"]
  }
}
```

- Lifecycle은 플랫폼에 어떻게 Droplet을 만들고 실행하는 방법을 안내
- CF API에서 Lifecycle 객체의 Type은 buildpack, docker, Kpack 3종류
- buildpack Type은 AP의 기본 라이프 사이클
- Kpack Type은 Sidecar의 기본 라이프 사이클

https://v3-apidocs.cloudfoundry.org/

Sidecar 앱 배포 과정

소스의 이미지 생성 과정

```
Containers(cf-workloads-staging/066b34a6-c238-4e62-a3c7-dccafbbc36cc-build-1-build-pod)
NAME↑       IMAGE
analyze                          /sidecar/cf-default-builder@sha256:d924e1846fe3432a3ba074c1c96aff6b527
build                            /sidecar/cf-default-builder@sha256:d924e1846fe3432a3ba074c1c96aff6b527
completion  gcr.io/cf-build-service-public/kpack/completion@sha256:be91b1535d09ef73319470b9aab6ba17d
detect                           /sidecar/cf-default-builder@sha256:d924e1846fe3432a3ba074c1c96aff6b527
export                           /sidecar/cf-default-builder@sha256:d924e1846fe3432a3ba074c1c96aff6b527
prepare     gcr.io/cf-build-service-public/kpack/build-init@sha256:358f6c68cdff4df532f81e377f51eaf50
restore                          /sidecar/cf-default-builder@sha256:d924e1846fe3432a3ba074c1c96aff6b527
```

<소스 코드를 이미지로 생성하는 POD의 Container 목록>

```
5 of 11 buildpacks participating
paketo-buildpacks/ca-certificates 3.1.0
paketo-buildpacks/node-engine     0.12.3
paketo-buildpacks/npm-install     0.9.0
paketo-buildpacks/node-module-bom 0.2.4
paketo-buildpacks/node-start      0.8.0
```

<Detect 동작>

```
Paketo CA Certificates Buildpack 3.1.0
  https://github.com/paketo-buildpacks/c
  Launch Helper: Contributing to layer
    Creating /layers/paketo-buildpacks_c
Paketo Node Engine Buildpack 0.12.3
  Resolving Node Engine version
```

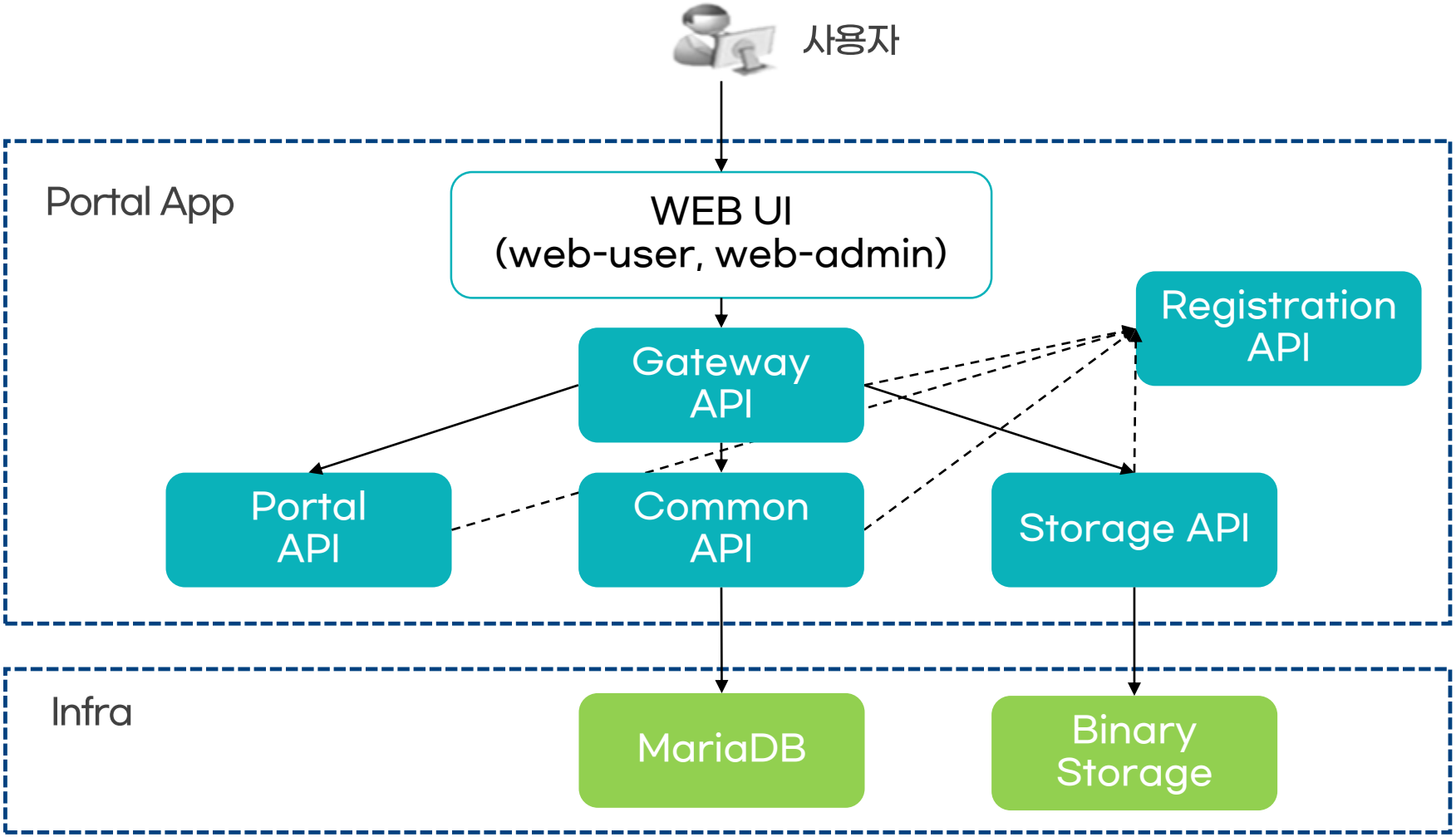
<Build 동작>

- Kpack은 Prepare, Analyze, Detect, Restore, Build, Export, Completion의 과정을 거쳐 이미지를 생성함

Sidecar 앱 배포 과정

Registry에 업로드 된 이미지 확인

파스-타 Portal Application Architecture



배포 사례 - Portal Application

사전 준비 - Infra 배포

<MariaDB Manifest>

```yaml
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: mariadb
  namespace: paasta
spec:
  serviceName: mariadb
  replicas: 1
  selector:
    matchLabels:
      app: mariadb
  template:
    metadata:
      labels:
        app: mariadb
    spec:
      initContainers:
      - name: mariadb-create-directory-structu
        image: busybox
        command:
          [
            "sh",
            "-c",
            "/bin/mkdir -p /bitnami/mariadb/da
          ]
        volumeMounts:
          - name: mariadb-data
            mountPath: /bitnami/mariadb
      containers:
      - image: docker.io/bitnami/mariadb:10.7
```

<Storage Manifest>

```yaml
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: openstack-swift-keystone-docker
  namespace: paasta
spec:
  serviceName: openstack-swift-keystone-docker
  replicas: 1
  selector:
    matchLabels:
      app: openstack-swift-keystone-docker
  template:
    metadata:
      labels:
        app: openstack-swift-keystone-docker
    spec:
      containers:
      - image: 3.36.165.47.nip.io/sidecar/saio
        name: openstack-swift-keystone-docker
        env:
        - name: IF_USE_SWIFT_EXTERNAL_MARIADB
          value: "true"
        - name: MARIADB_PORT
          value: "3306"
        - name: MARIADB_ADMIN_PASSWORD
          value: "Qca$Drtyf"
        - name: SWIFT_ADDRESS
          value: "openstack-swift-keystone-doc
        - name: KEYSTONE_PORT
          value: "55001"
```

배포 사례 - Portal Application

사전 준비 - Network

< Network Policy - Portal App ↔ CF DB >

```yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-cf-db-ingress-from-cf-workloads
  namespace: cf-db
spec:
  podSelector: {}
  policyTypes:
  - Ingress
  ingress:
  - from:
    - namespaceSelector:
        matchLabels:
          cf-for-k8s.cloudfoundry.org/cf-workloads
      podSelector:
        matchLabels:
          cloudfoundry.org/org_name: system
      podSelector:
        matchLabels:
          cloudfoundry.org/space_name: dev
```

<Sidecar - default (기본 배포 시 배포됨)>

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: Sidecar
metadata:
  name: default
  namespace: cf-workloads
spec:
  egress:
  - hosts:
    - cf-system/*
```

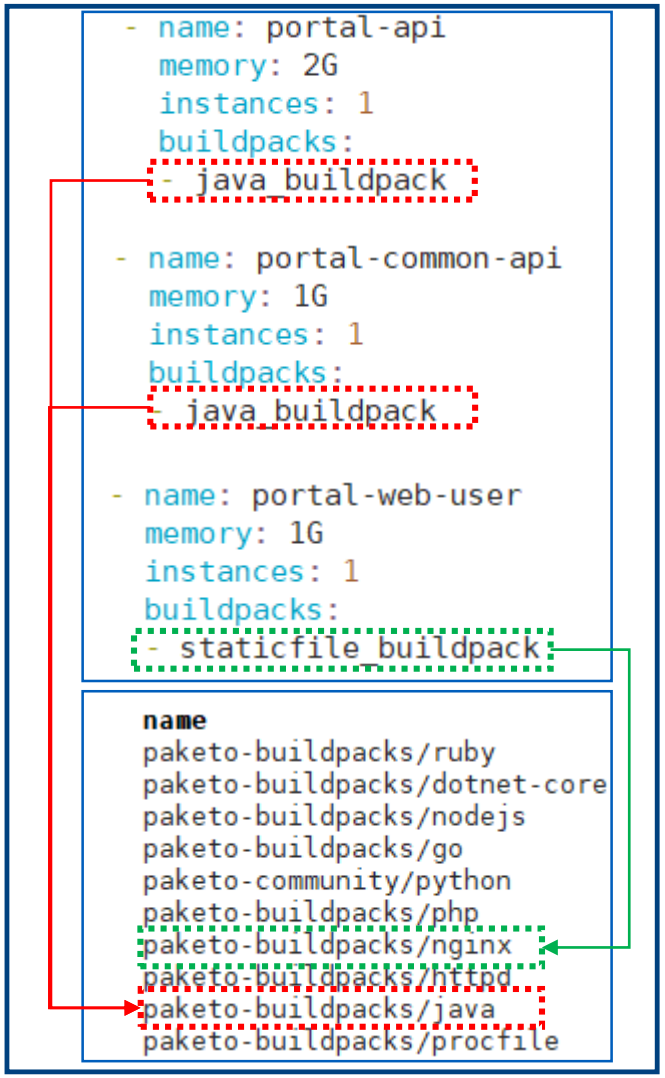
<Sidecar - Portal App ↔ CF DB >

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: Sidecar
metadata:
  name: portal-db
  namespace: cf-workloads
spec:
  workloadSelector:
    labels:
      cloudfoundry.org/org_name: system
      cloudfoundry.org/space_name: dev
  egress:
  - hosts:
    - cf-db/*
```

- 배포되는 Portal App은 CF DB와의 연결이 필요하나 default 상태로는 연결 불가
- CF DB와의 연결을 위하여 Network Policy와 Istio Sidecar 추가 Resource 배포 필요

배포 사례 - Portal Application

Portal Application Buildpack 현황



- 기존 파스-타 AP 기준의 Portal Application manifest.yml 현황
  - portal-web-user - staticfile buildpack (staticfile buildpack은 nginx 기반 동작)
  - 그외 모듈 - java buildpack

- 기존 staticfile buildpack은 paketo-buildpacks/nginx 로 사용
- 기존 java buildpack은 paketo-buildpacks/java 로 사용

배포 사례 - Portal Application

Portal Application 배포 - Nginx Buildpack

<AP - portal-web-user nginx.conf>

```
server {
  listen 8080;
  root /home/vcap/app/public;
  index index.html index.htm;
```

<AP - portal-web-user Directory>

```
3rdpartylicenses.txt
assets/
index.html
inline.318b50c57b4eba3d437b.bundle.js
main.57d9daba2e83f3b74d0d.bundle.js
nginx.conf
polyfills.de363ddcc2f14bec497c.bundle.js
styles.ac89bfdd6de82636b768.bundle.css
```

<Sidecar - portal-web-user nginx.conf>

```
server {
  listen {{port}};
  root /workspace;
  index index.html index.htm;
```

<Sidecar - portal-web-user Directory>

```
3rdpartylicenses.txt
assets/
index.html
inline.318b50c57b4eba3d437b.bundle.js
main.1a53ad96a4061d399c1c.bundle.js
mime.types
nginx.conf
polyfills.de363ddcc2f14bec497c.bundle.js
styles.ac89bfdd6de82636b768.bundle.css
```

- staticfile buildpack을 nginx buildpack으로 변환할 때 nginx.conf 포트의 변수화 필요
- AP와 Sidecar는 root가 다르므로 root Directory 수정 필요
- staticfile buildpack의 경우 mime.types를 자동 생성하나 nginx buildpack은 파일 필요

Portal Application 배포 - Java Buildpack

```
buildscript {
    ext {
        springBootVersion = '1.5.10.RELEASE'
    }
```

<Portal API - build.gradle>

```
applications:
  - name: portal-api
    memory: 2G
    disk: 1G
    instances: 1
    buildpacks:
    - paketo-buildpacks/java
    env:
      server_port: 8080
      BP_JVM_VERSION: 8
      BP_SPRING_CLOUD_BINDINGS_DISABLED: true
```

<Sidecar Portal API - manifest.yml>

- Spring Boot 2.3 미만의 버전을 사용할 경우 Spring Cloud Bindings 옵션 비활성화 필요 (BP_SPRING_CLOUD_BINDING_DISABLE)
- Sidecar의 경우 포트 1024 이상 사용 필요 (server_port)
- 필요에 따라 JVM 버전 변경 필요 (BP_JVM_VERSION)
- 기존 AP Java Buildpack의 경우 jar, war 파일만 가능 하나 Sidecar는 소스 레벨에서 배포 가능 (gradle, maven buildpack)

https://github.com/paketo-buildpacks/spring-boot

https://github.com/paketo-buildpacks/bellsoft-liberica

배포 사례 - Portal Application

Portal Application 배포 결과

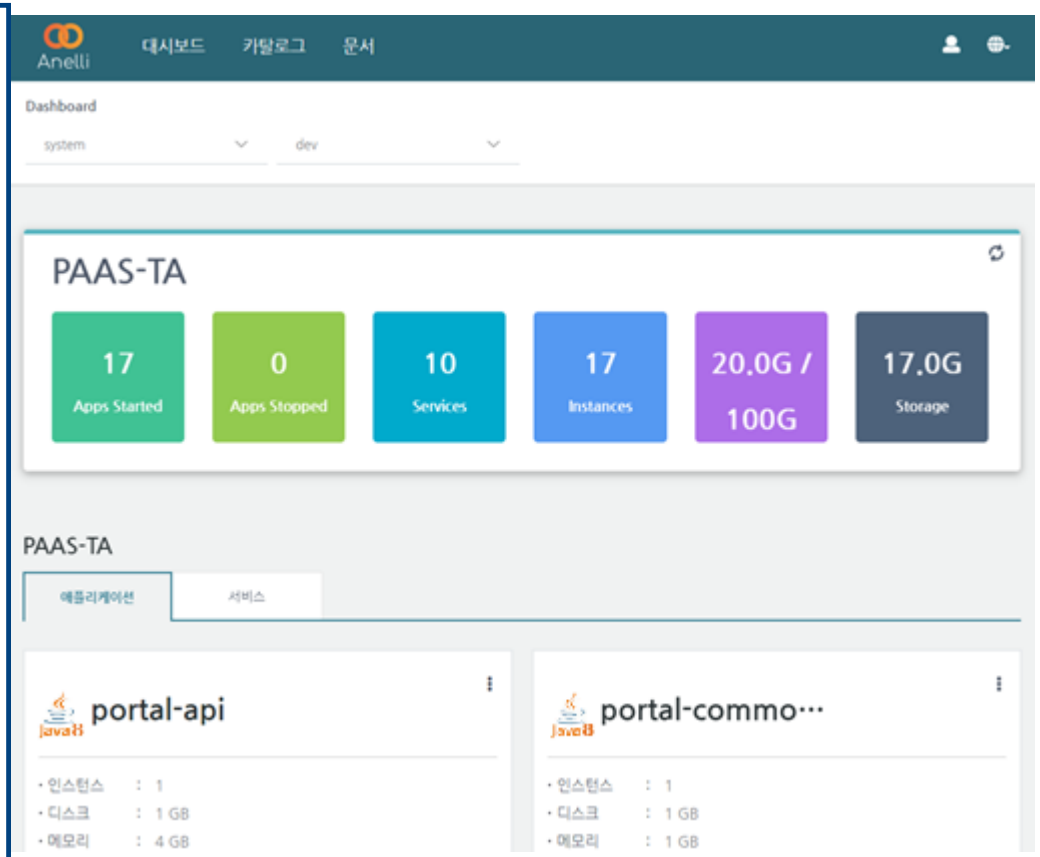
```
Pushing app portal-common-api to org system
Applying manifest file /home/ubuntu/test/por

Updating with these attributes...
  ---
  applications:
  - name: portal-common-api
    instances: 1
    memory: 1G
    default-route: true
    buildpacks:
      paketo-buildpacks/java
```

```
Waiting for app portal-common-api to start...

Instances starting...
Instances starting...
```

```
name                 requested state   processes
portal-api           started           web:1/1
portal-common-api    started           web:1/1
portal-gateway       started           web:1/1
portal-registration  started           web:1/1
portal-storage-api   started           web:1/1
portal-web-admin     started           web:1/1
portal-web-user      started           web:1/1
```

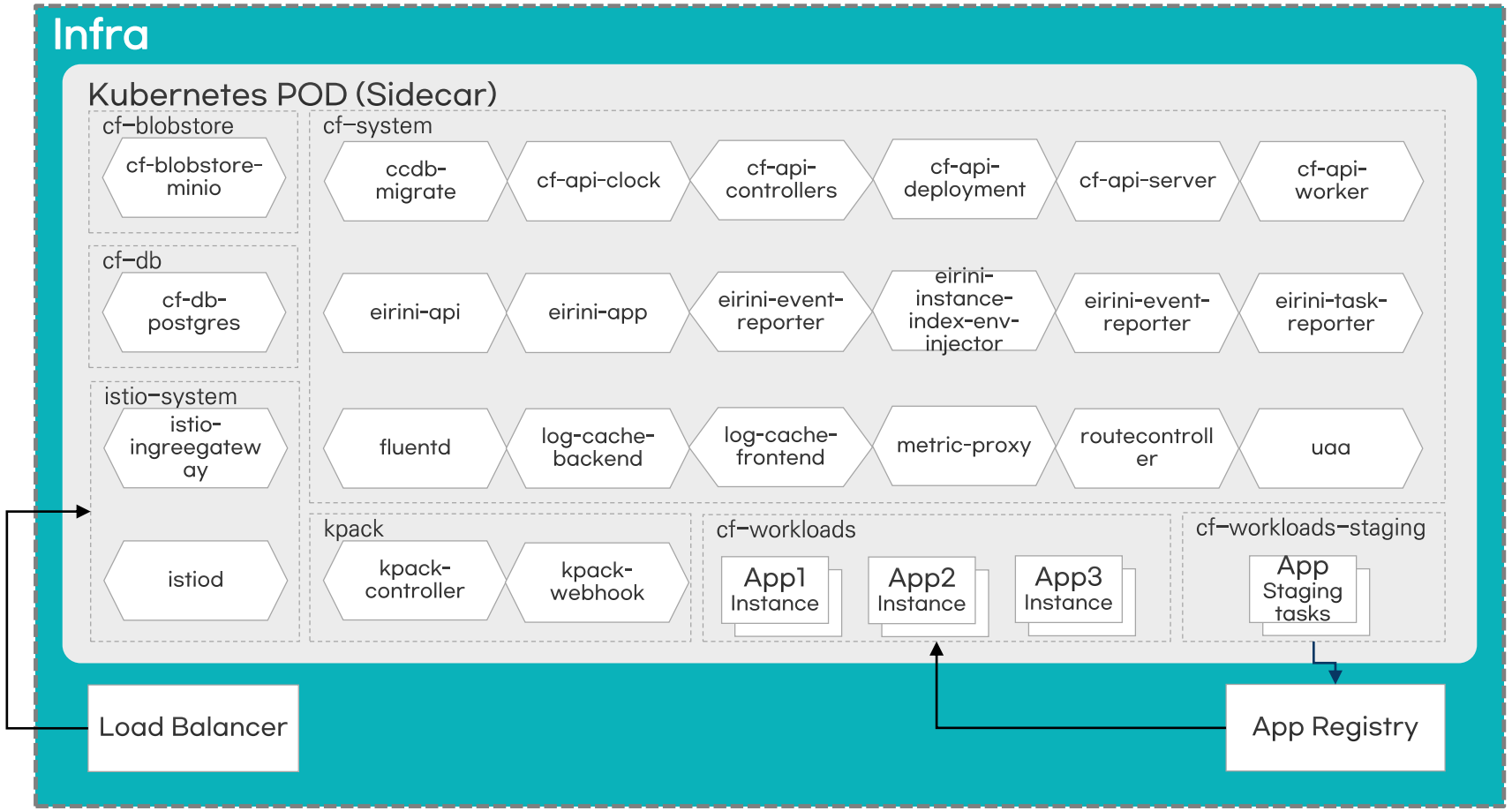


< 배포된 portal-web-user 화면>

배포 사례 - Portal Application

NIA 한국지능정보사회진흥원

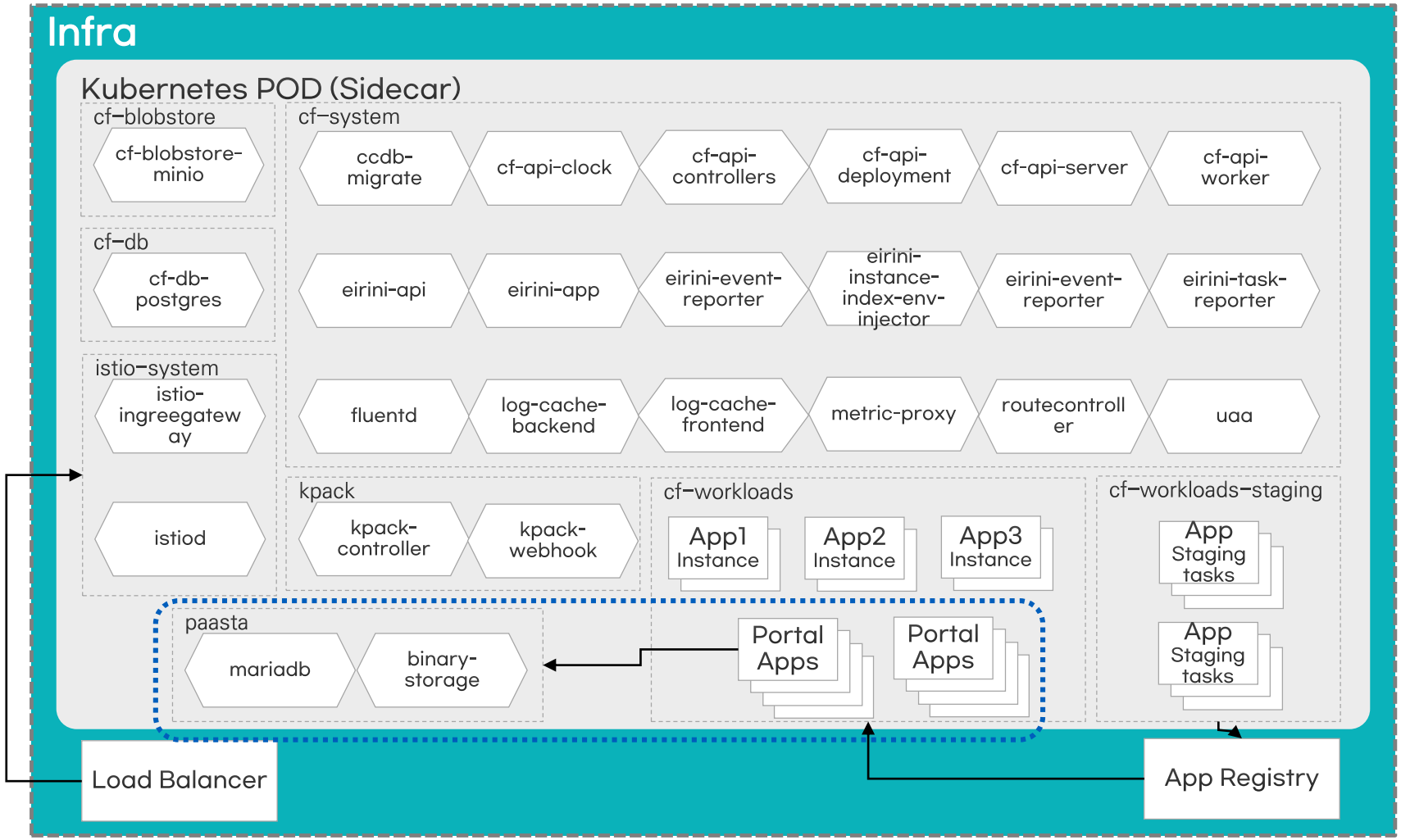
파스-타 Sidecar 컴포넌트



배포 사례 - Portal Application

NIA 한국지능정보사회진흥원

파스-타 Sidecar 컴포넌트 - Portal App 배포



감사합니다.

NIA 한국지능정보사회진흥원