

Kubernetes vs Openshift

Nikita Chursin

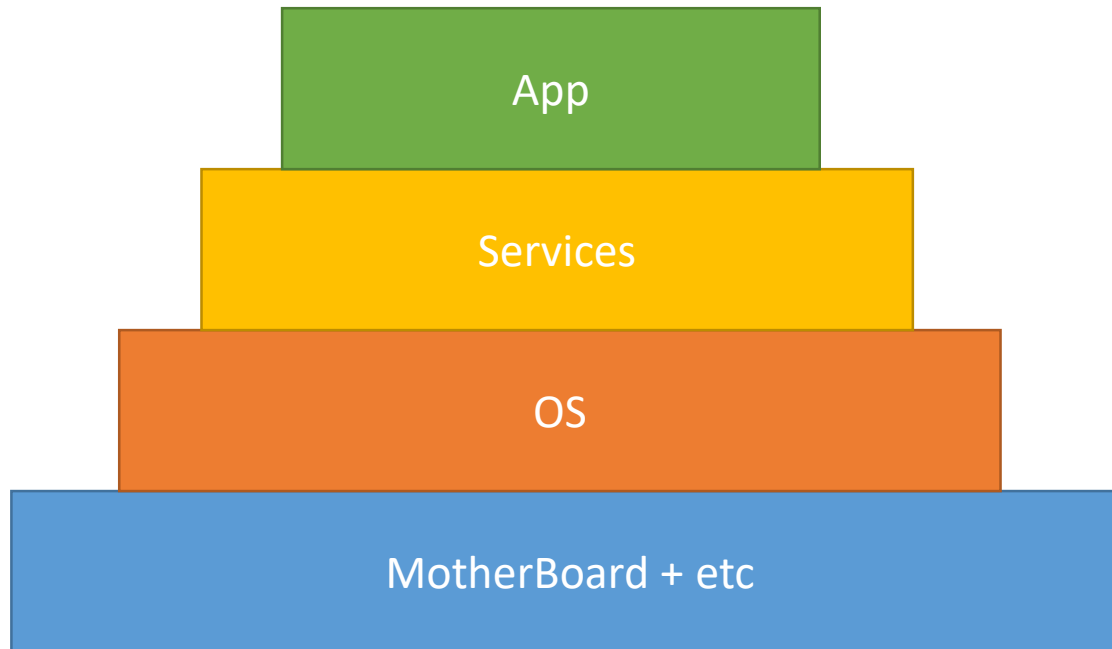
@nikitachrs

Nikita.Chursin@t-systems.com

<https://gitlab.t-systems.ru/nchursin/courses>



Simple Server App

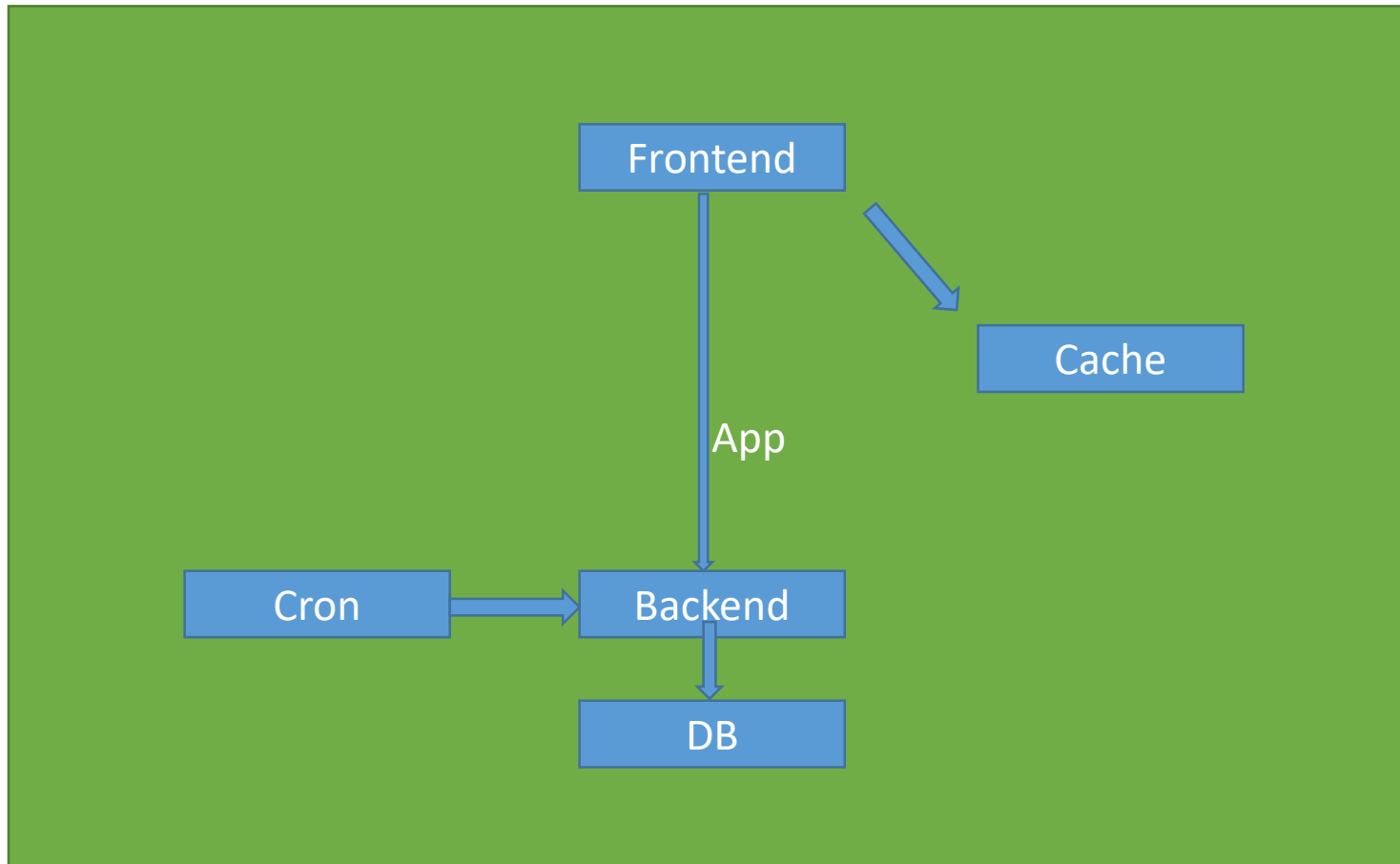


EDB

T..

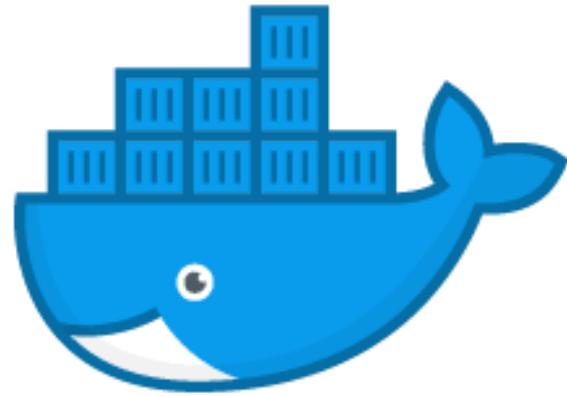
Simple Server App

EDB



T...

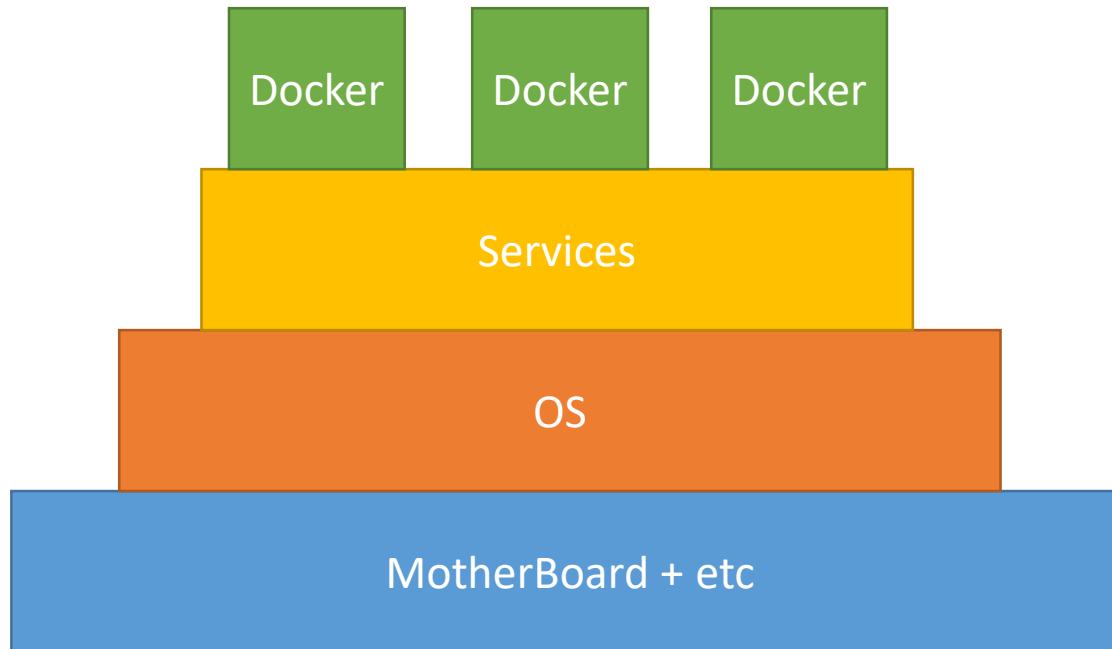
EDB



docker

T...

Simple Server App v2

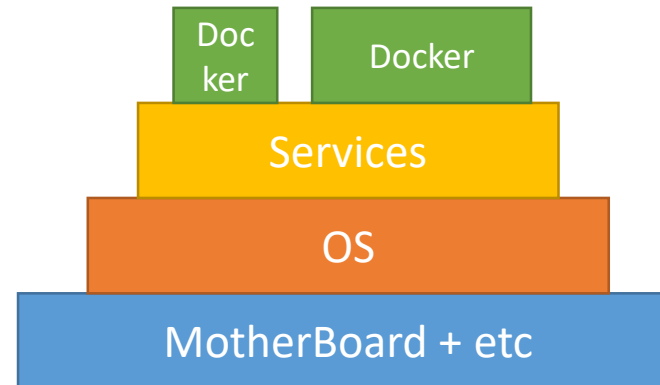
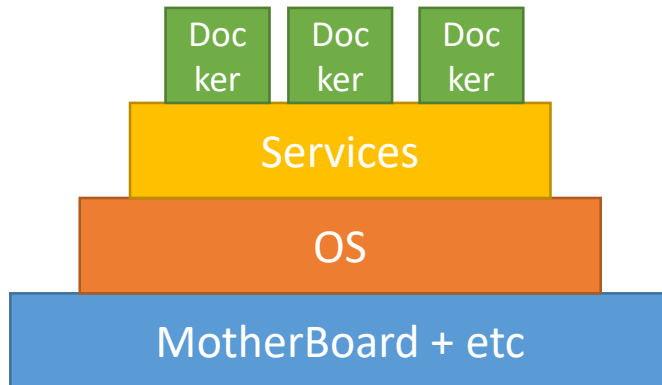


EDB

T...

Multiple Server App v2

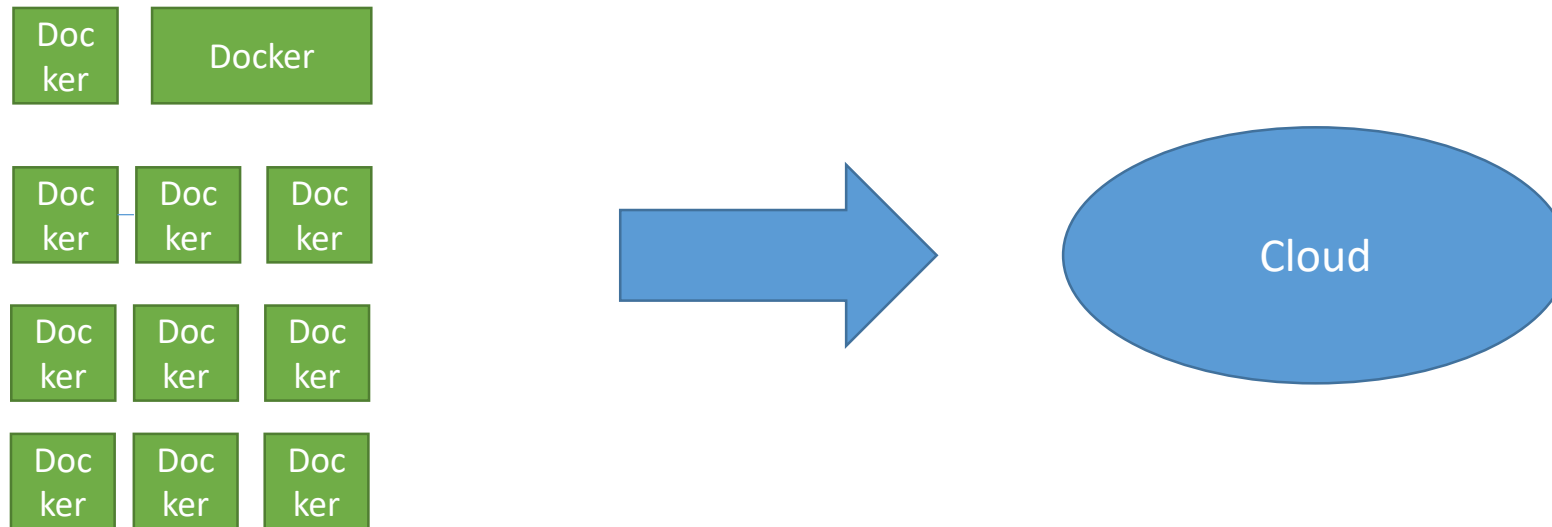
EDB



T . .

Cloud Server App

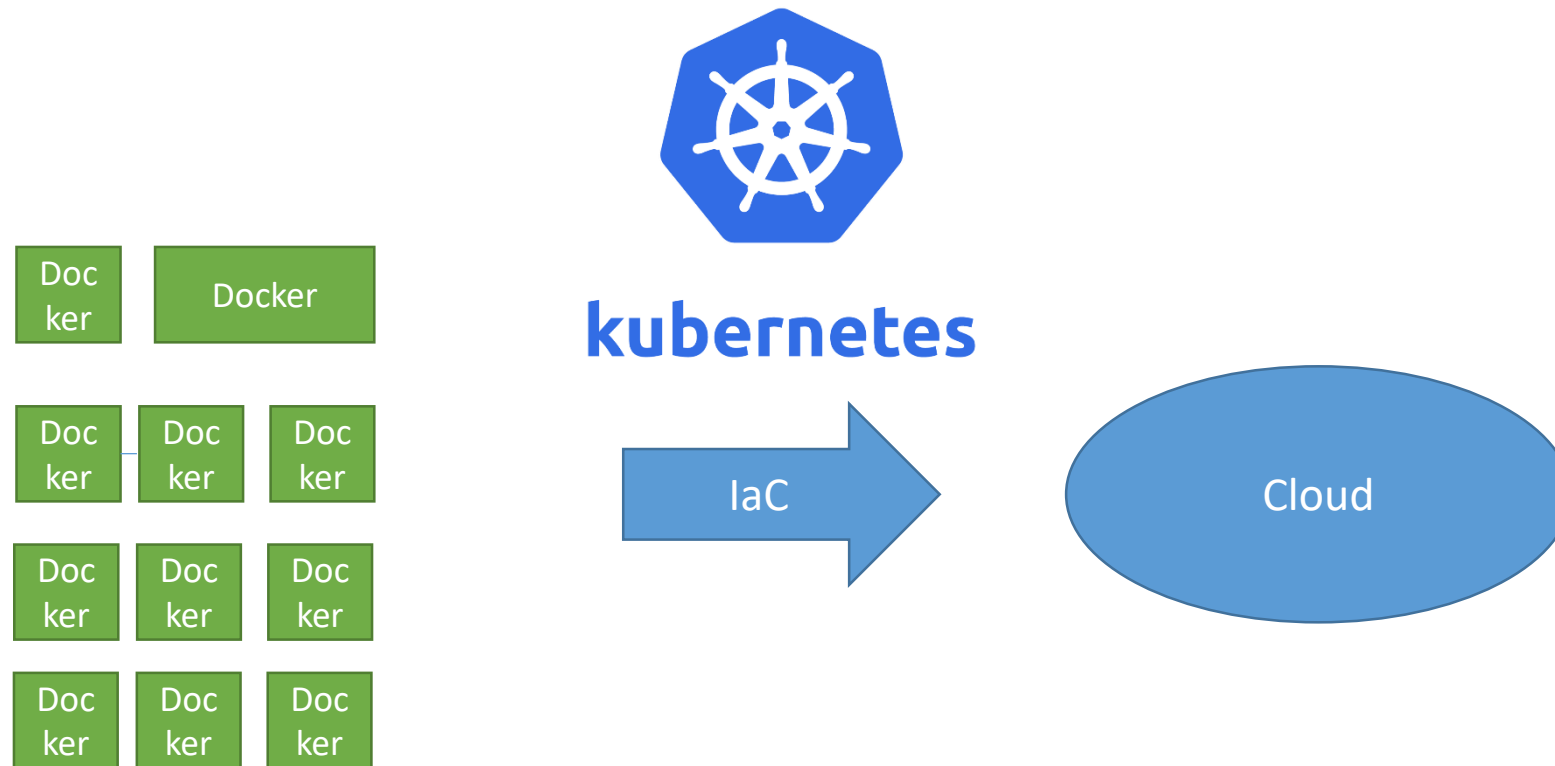
EDB



T . .

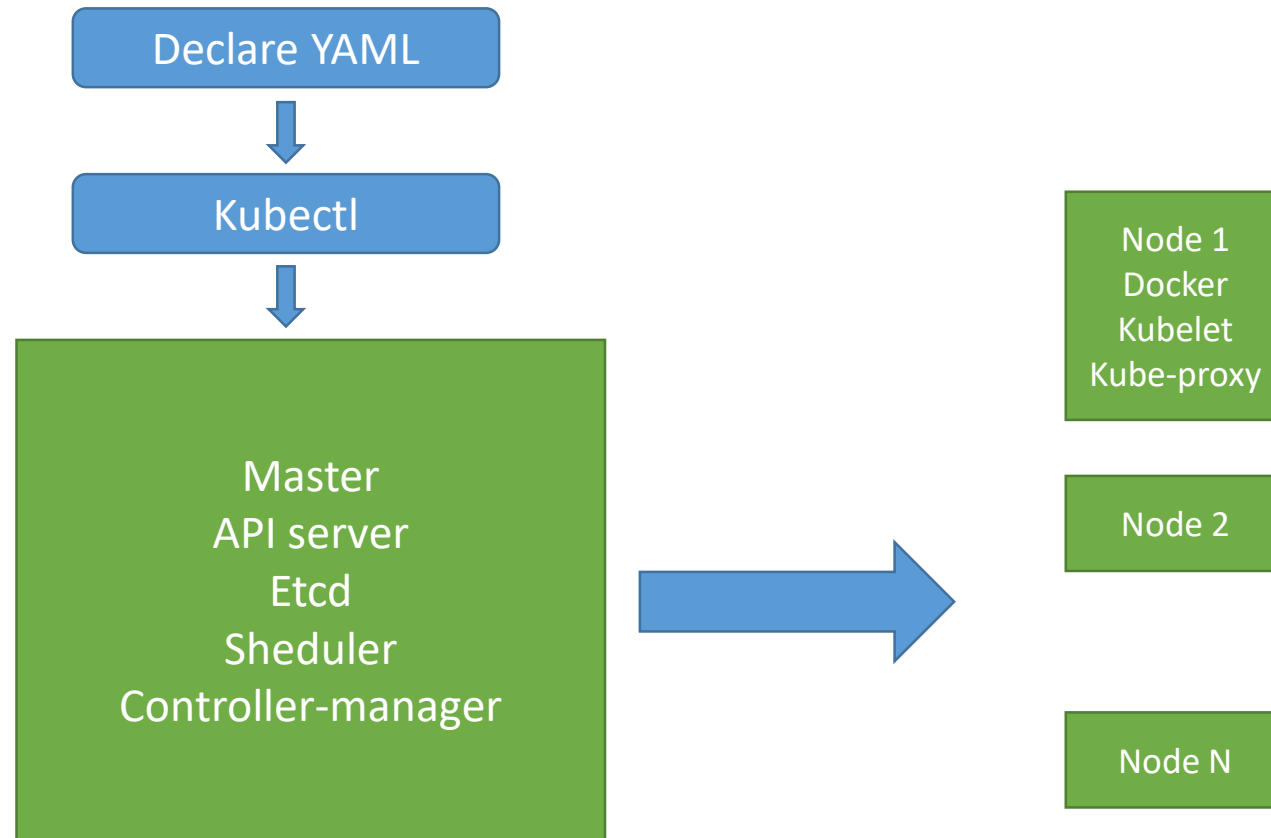
Cloud Server App

EDB



T . .

Kubernetes



Container

EDB

Docker container + CMD

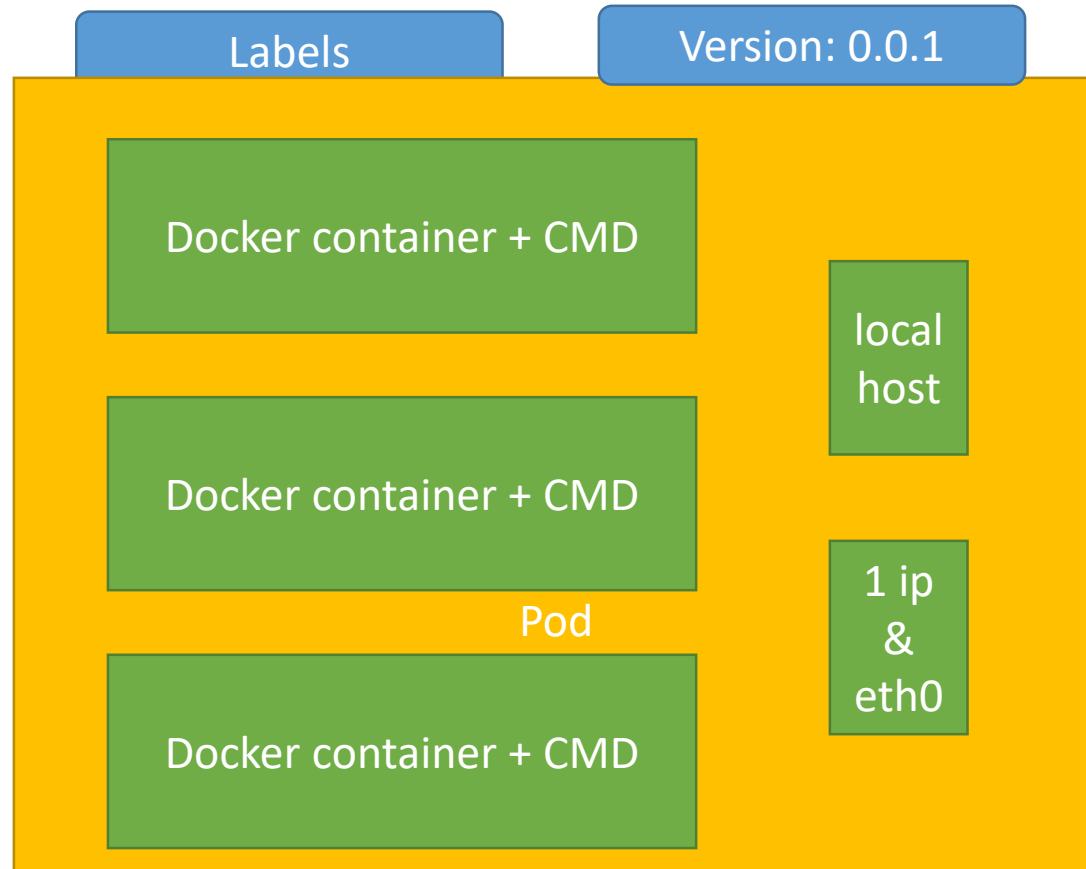
Logging

Supervision

Can't create container in Kubernetes

T . .

Pod



Infrastructure
automation

Supervision

Service discovery

EDB

T . .

Labels & selectors



Labels

foo:bar

version: 0.0.1

Selectors

app in (foo,bar)

version == 0.0.1



Pod yaml



```
apiVersion: v1
kind: Pod
metadata:
  name: manual-bash
spec:
  containers:
  - name: bash
    image: ubuntu:16.04
    command: bash
    args: [-c, "while true; do sleep 1; date; done"]
```



Pod yaml

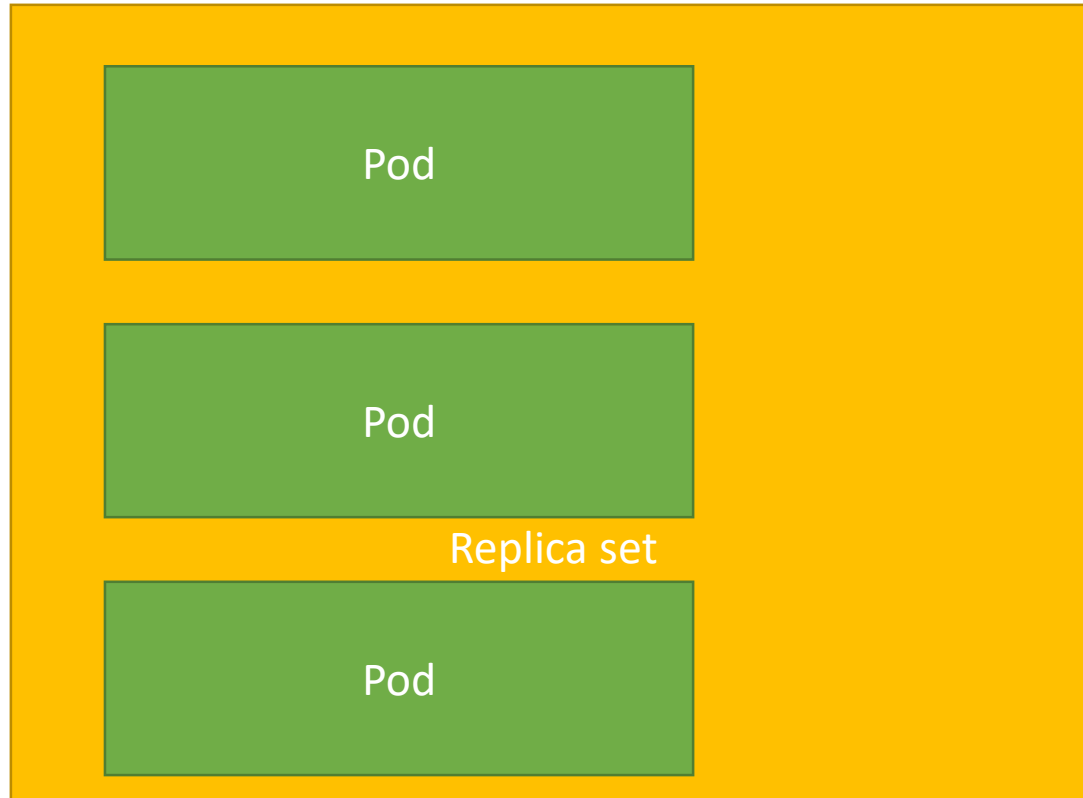


```
apiVersion: v1
kind: Pod
metadata:
  name: manual-backend
  labels:
    tier: backend
spec:
  containers:
  - name: web
    image: myregistry.local/backend:0.15.9
    command: /usr/bin/php-fpm
```



Replica set

EDB



Supervision

Auto Scaling

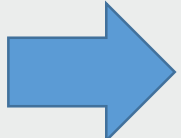
Don't create 1 pod in Kubernetes, use ReplicaSet

T . .

ReplicaSet yaml



```
apiVersion: extensions/v1beta1
kind: ReplicaSet
metadata:
  name: backend
spec:
  replicas: 3
  selector:
    matchLabels:
      tier: backend
  template:
    metadata:
      labels:
        tier: backend
    spec:
      containers:
        - name: fpm
          image: myregistry.local/backend:0.15.7
          command: php-fpm
```



Pod yml

Can't update image version in replica set



Deployment



CI/CD



History



Update deploy

EDB



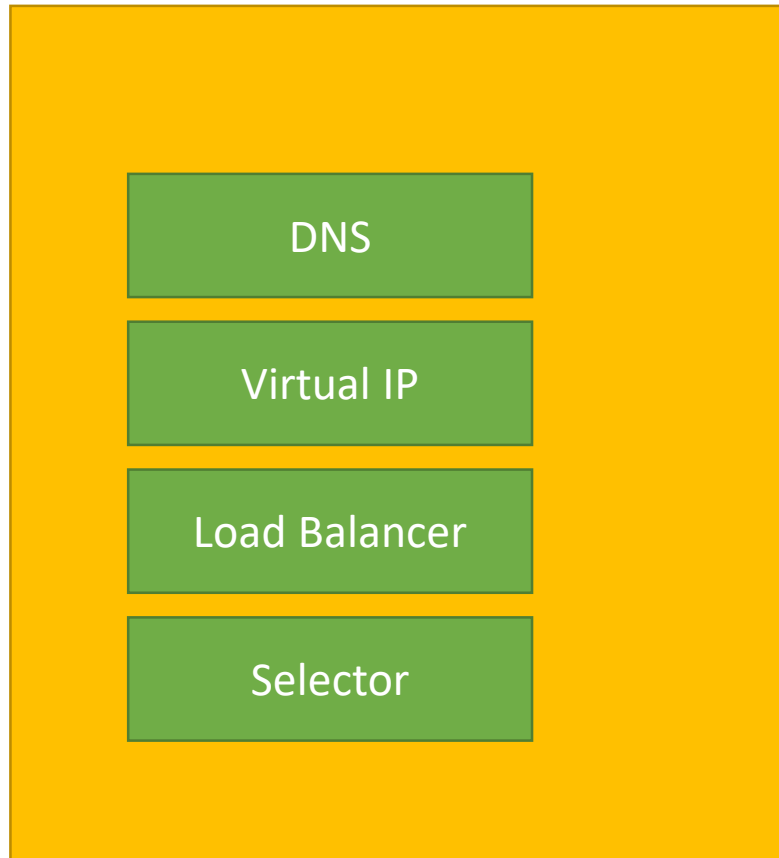
Deployment yaml



```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: backend
spec:
  replicas: 3
  selector:
    matchLabels:
      tier: backend
  template:
    metadata:
      labels:
        tier: backend
    spec:
      containers:
      - name: fpm
        image: myregistry.local/backend:0.15.9
        command: php-fpm
```



Service



CI/CD

EDB

T...

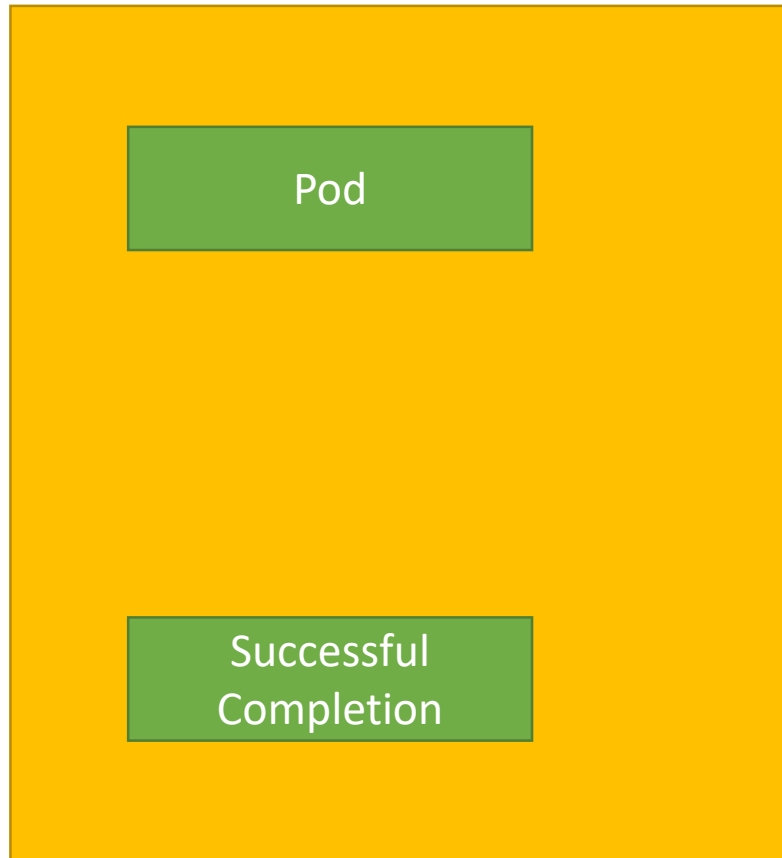
Service yaml



```
kind: Service
apiVersion: v1
metadata:
  name: backend
spec:
  selector:
    tier: backend
  ports:
    - protocol: TCP
      port: 80
      targetPort: 8080
```



job



EDDB

T...

job yaml



```
apiVersion: batch/v1
kind: Job
metadata:
  name: migrate
spec:
  template:
    metadata:
      name: migrate
    spec:
      containers:
      - name: migrate
        image: myregistry.local/backend:0.15.9
        command: [yiic, migrate]
      restartPolicy: Never
```



cronjob



EDDB

T...

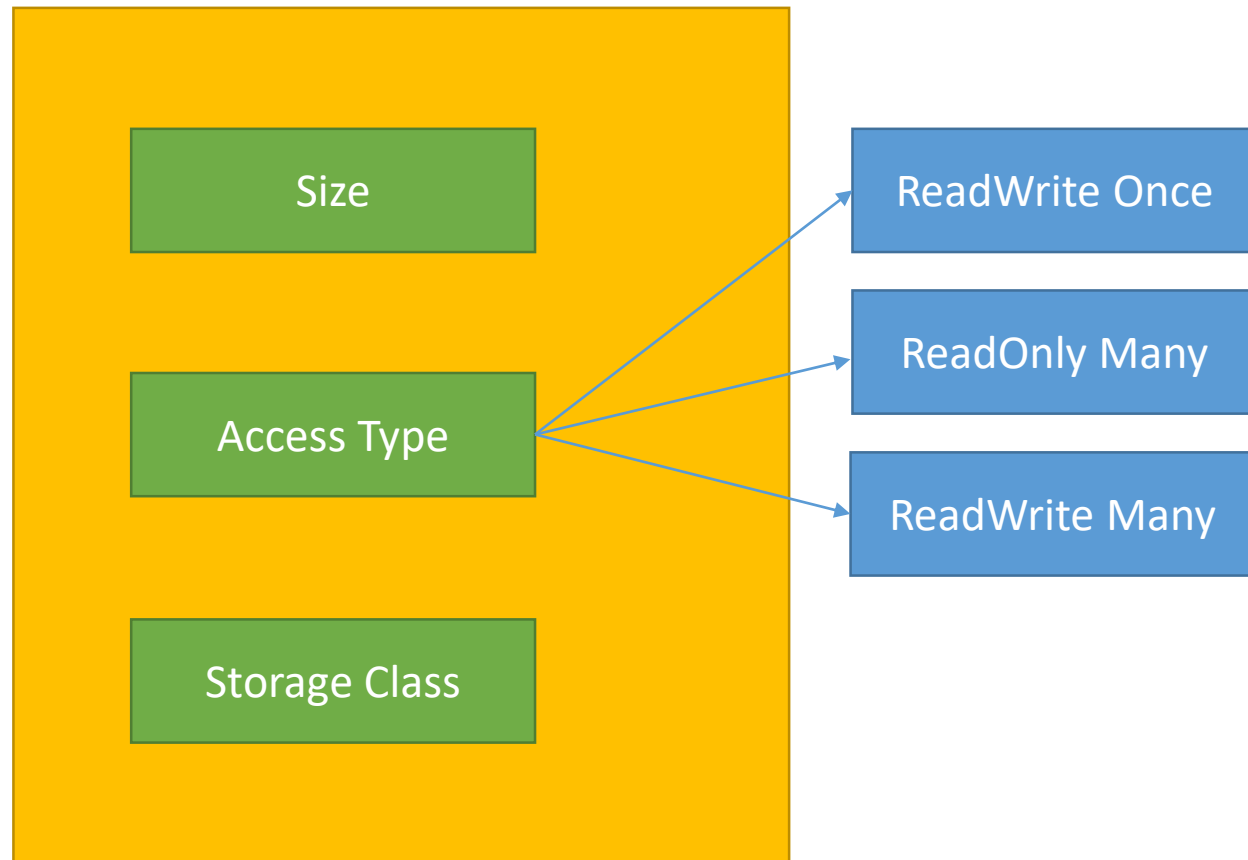
cronjob yaml



```
apiVersion: batch/v2alpha1
kind: CronJob
metadata:
  name: cron-foo
spec:
  schedule: "*/1 * * * *"
  jobTemplate:
    spec:
      template:
        spec:
          containers:
            - name: main
              image: myregistry.local/backend:0.15.9
              args: [yiic, cron-foo]
          restartPolicy: OnFailure
```



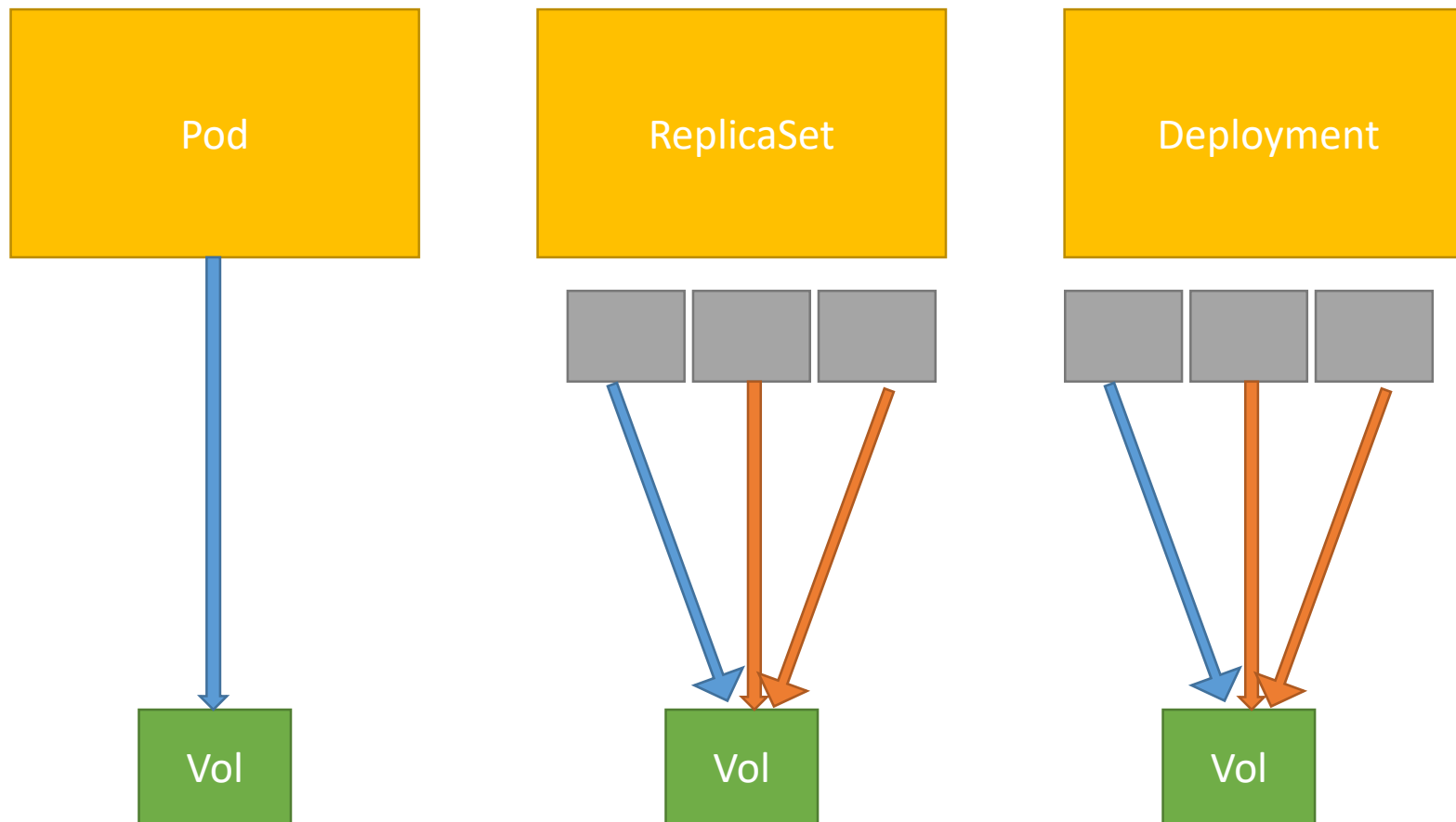
volume



EDB

T...

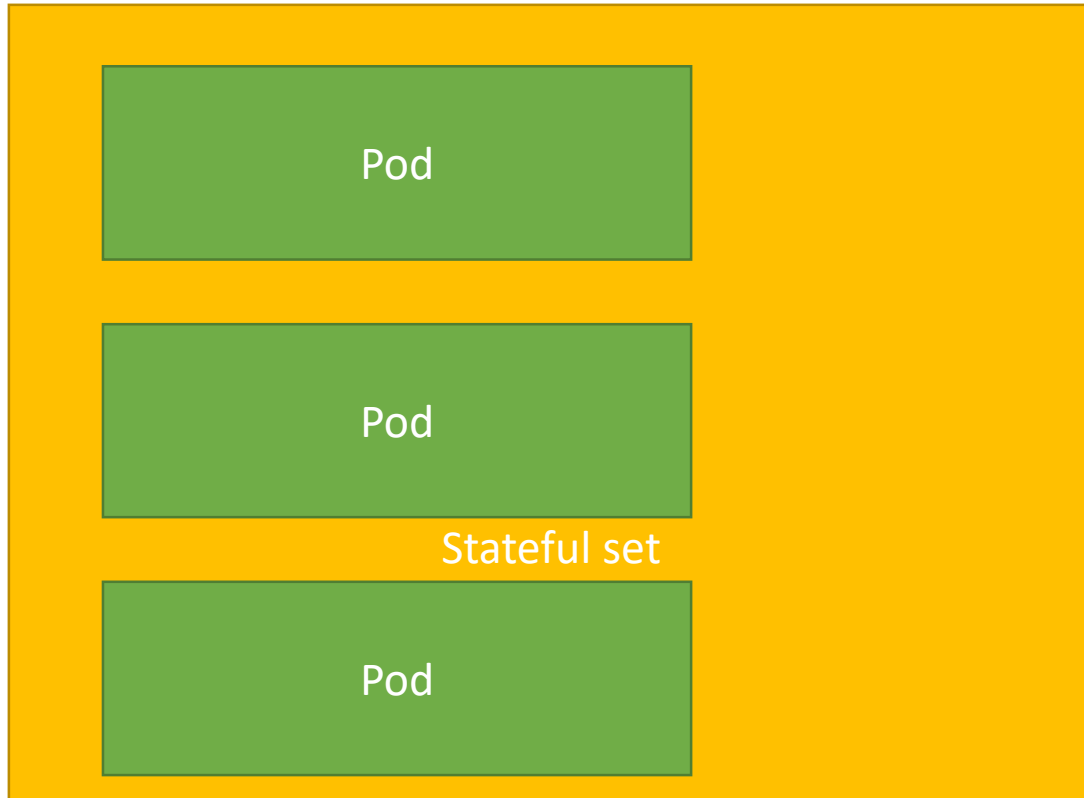
volume



ReadWriteOnce used only for 1 pod!

Stateful set

EDB



Statefulset create Pod names with numbers [0..N-1]

T ..

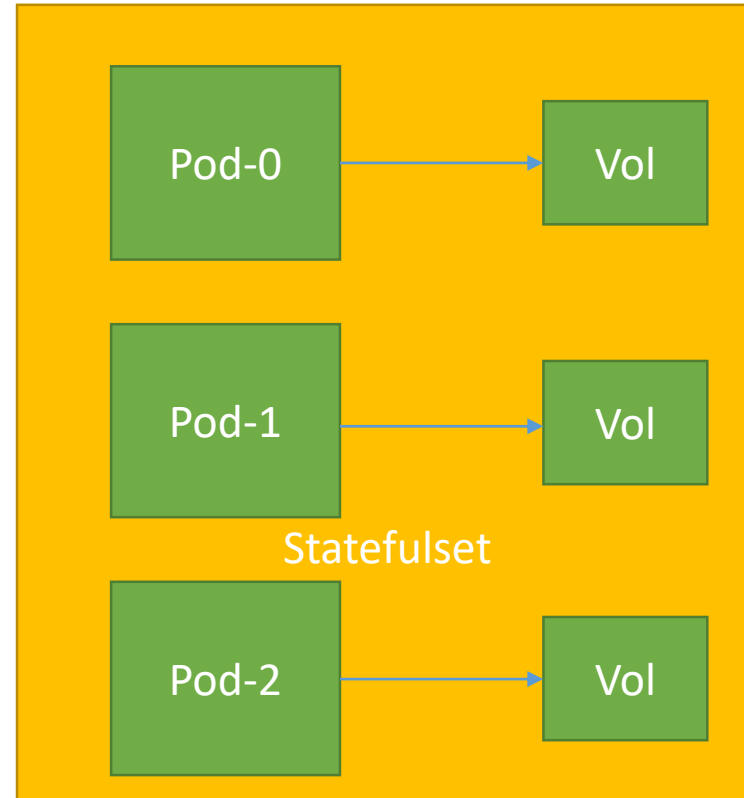
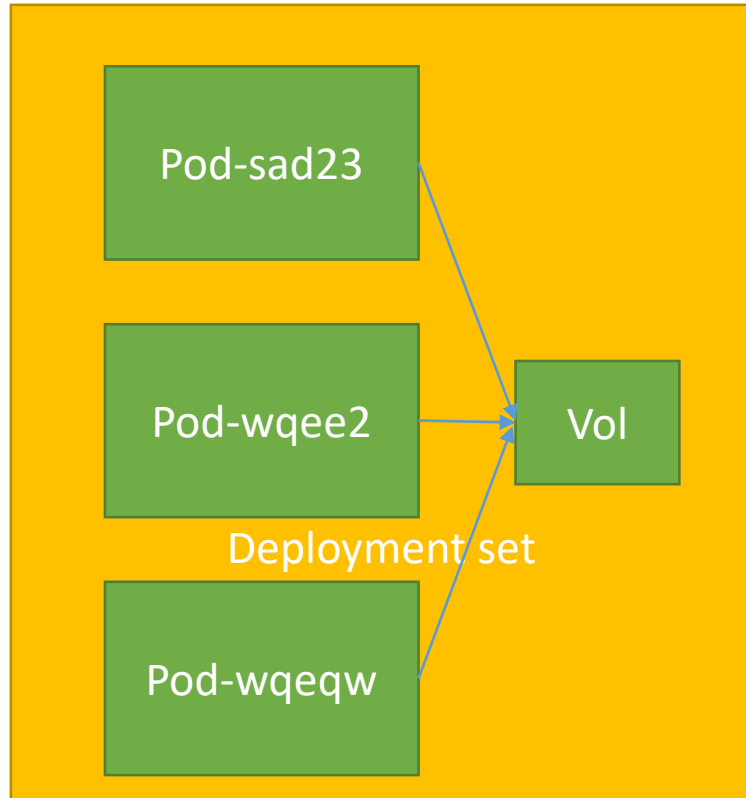
Statefulset vs Deployment

EDB

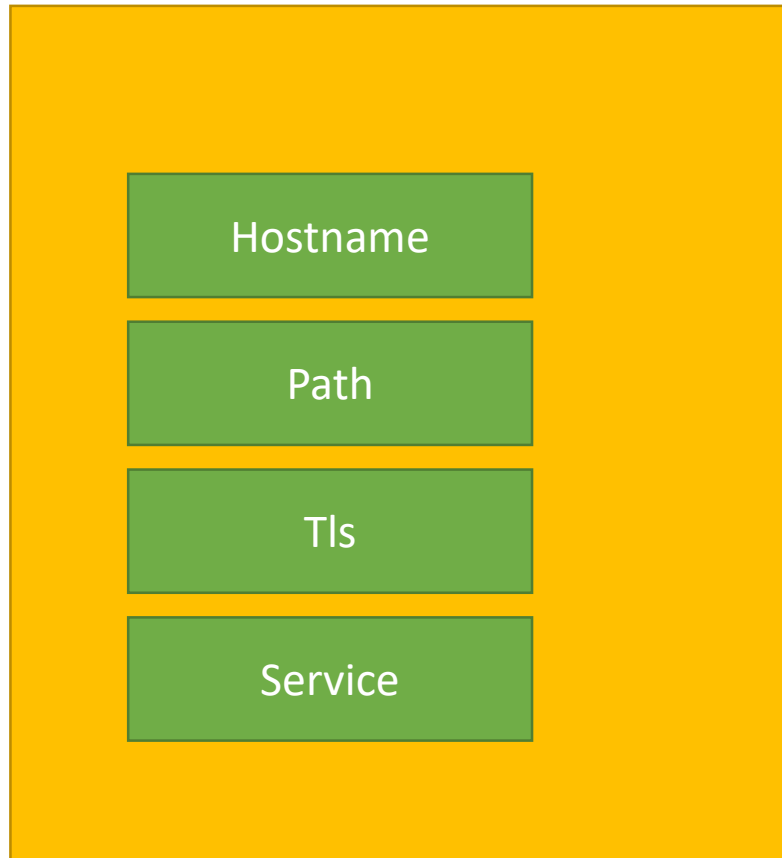


T . .

Statefulset vs Deployment



Ingress



CI/CD

EDDB

T...

Ingress yaml



```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: main
spec:
  rules:
  - host: foo.bar.com
    http:
      paths:
      - path: /assets
        backend:
          serviceName: frontend
          servicePort: 80
      - path: /
        backend:
          serviceName: backend
          servicePort: 80
```

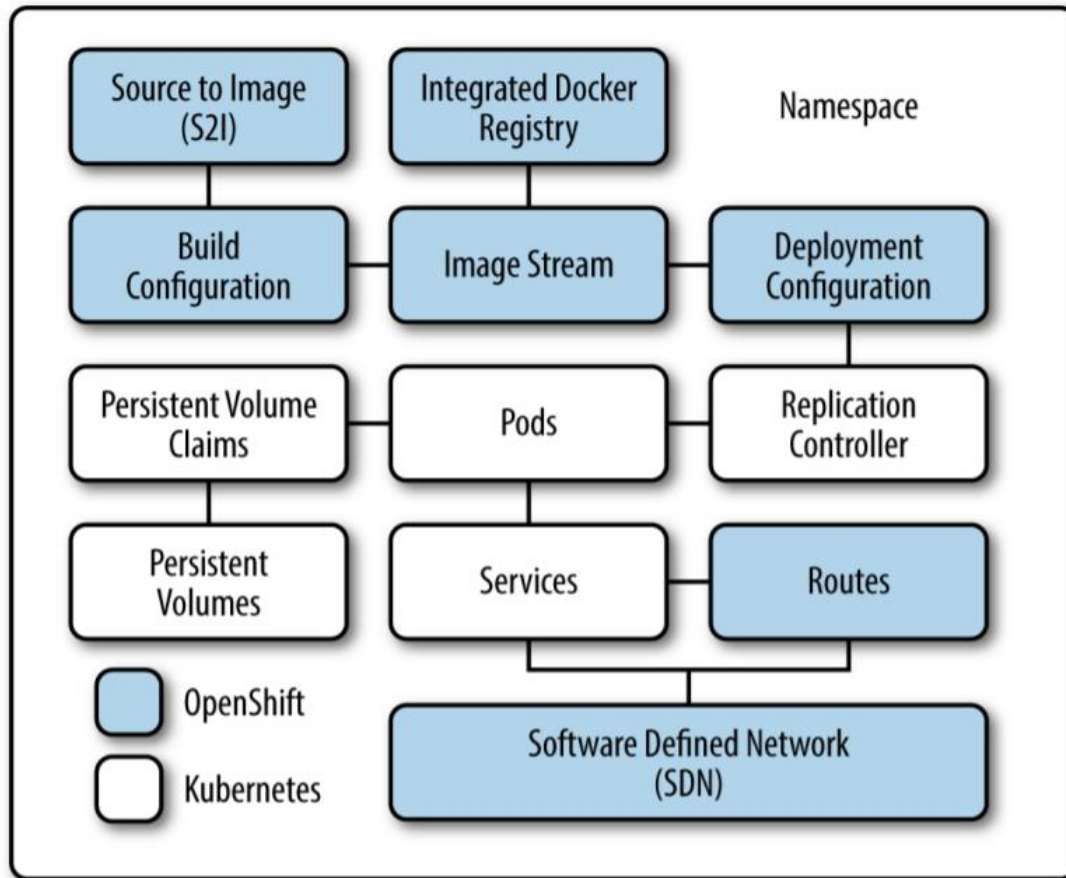


Openshift features

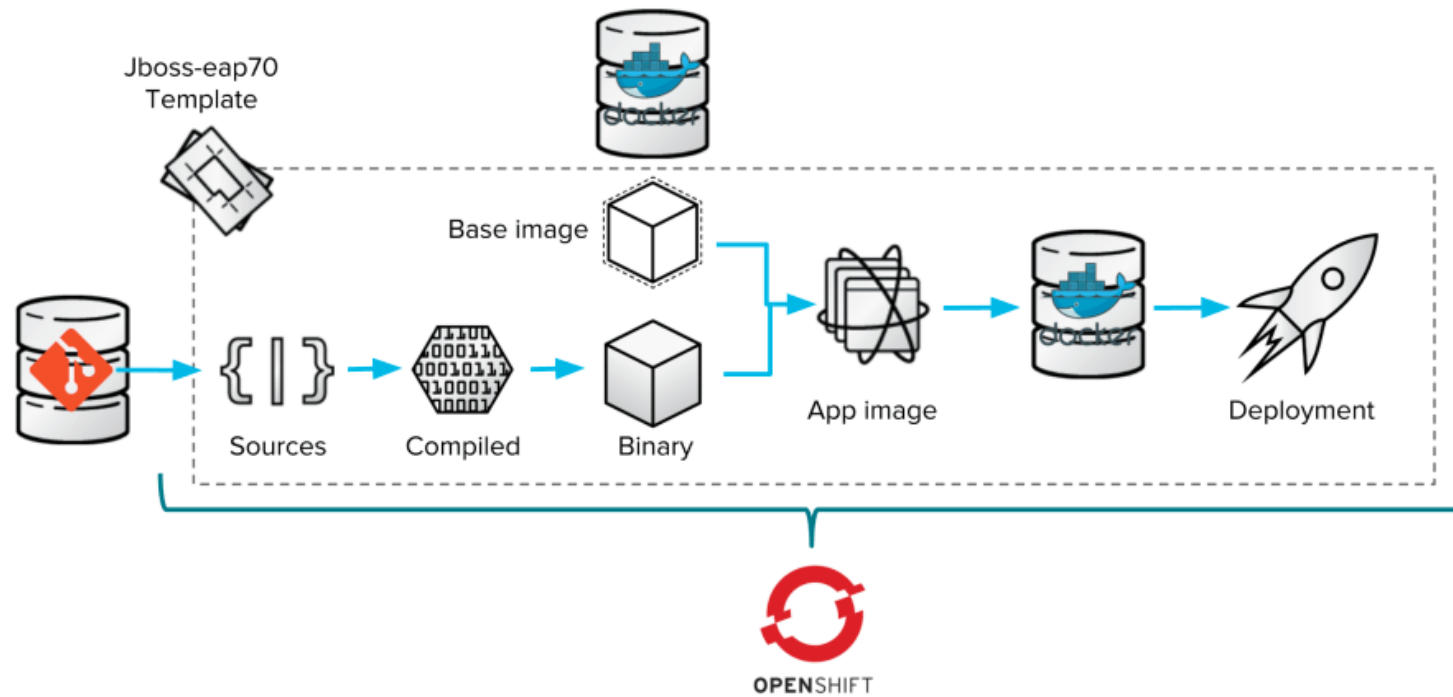
EDB



Openshift project



S2I



EDB

T..

Build Config yaml



BuildConfig Object Definition

```
kind: "BuildConfig"
apiVersion: "v1"
metadata:
  name: "ruby-sample-build" 1
spec:
  runPolicy: "Serial" 2
  triggers: 3
  -
    type: "GitHub"
    github:
      secret: "secret101"
  - type: "Generic"
    generic:
      secret: "secret101"
  -
    type: "ImageChange"
  source: 4
  git:
    uri: "https://github.com/openshift/ruby-hello-world"
  strategy: 5
  sourceStrategy:
    from:
      kind: "ImageStreamTag"
      name: "ruby-20-centos7:latest"
  output: 6
  to:
    kind: "ImageStreamTag"
    name: "origin-ruby-sample:latest"
  postCommit: 7
  script: "bundle exec rake test"
```



ImageStream



```
{
  "kind": "ImageStream",
  "apiVersion": "v1",
  "metadata": {
    "name": "origin-ruby-sample",
    "namespace": "p1",
    "selfLink": "/osapi/v1/namespaces/p1/imageStreams/origin-ruby-sample",
    "uid": "480dfe73-f340-11e4-97b5-001c422dcd49",
    "resourceVersion": "293",
    "creationTimestamp": "2015-05-05T16:03:34Z",
    "labels": {
      "template": "application-template-stibuild"
    }
  },
  "spec": {},
  "status": {
    "dockerImageRepository": "172.30.30.129:5000/p1/origin-ruby-sample",
    "tags": [
      {
        "tag": "latest",
        "items": [
          {
            "created": "2015-05-05T16:05:47Z",
            "dockerImageReference": "172.30.30.129:5000/p1/origin-ruby-sample@sha256:4d3a646b58685449179a0c61ad4baa19a8df8ba668e0f0704b9ad16f5e16e642",
            "image": "sha256:4d3a646b58685449179a0c61ad4baa19a8df8ba668e0f0704b9ad16f5e16e642"
          }
        ]
      }
    ]
  }
}
```



Route TLS



```
apiVersion: v1
kind: Route
metadata:
  name: route-edge-secured 1
spec:
  host: www.example.com
  to:
    kind: Service
    name: service-name 1
  tls:
    termination: edge 2
    key: |- 3
      -----BEGIN PRIVATE KEY-----
      [...]
      -----END PRIVATE KEY-----
    certificate: |- 4
      -----BEGIN CERTIFICATE-----
      [...]
      -----END CERTIFICATE-----
    caCertificate: |- 5
      -----BEGIN CERTIFICATE-----
      [...]
      -----END CERTIFICATE-----
```



Route LB



```
apiVersion: v1
kind: Route
metadata:
  name: route-alternate-service
  annotations:
    haproxy.router.openshift.io/balance: roundrobin 1
spec:
  host: www.example.com
  to:
    kind: Service
    name: service-name 2
    weight: 20 4
  alternateBackends:
  - kind: Service
    name: service-name2 3
    weight: 10 4
  - kind: Service
    name: service-name3 3
    weight: 10 4
```

