



# PERCONA

Databases run better with Percona

PERCONA



# Kubernetes Operators

Expanding Automation in Containerized  
Applications

# Edith Puclla (Edi)



- Technology Evangelist at Percona
- The UK Global Talent Visa
- CNCF Ambassador
  - KCD, Lima Peru
- Docker Captain
  - Meetups in Ayacucho
- Open Source Contributor:
  - Apache Airflow, Kubernetes Website



Edith Puclla



@edithpuclla

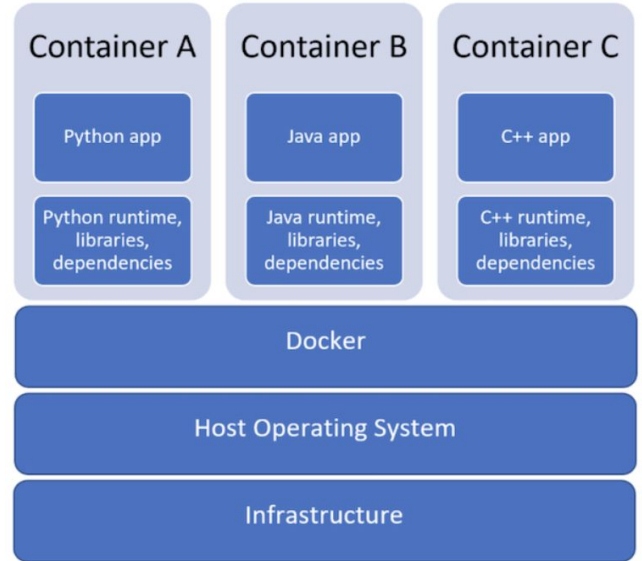
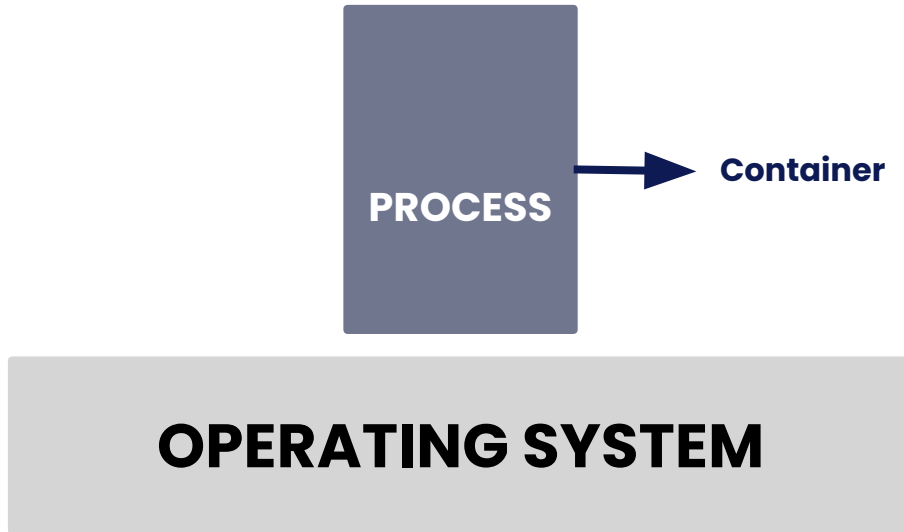
# Agenda

- Kubernetes
  - Deploying an Application
  - Default Resources
  - Limitations
- Kubernetes Operators
  - Components
    - CRD, CR, OLM, Controllers
  - Operator framework, Operator Hub, Capability Models

# Kubernetes

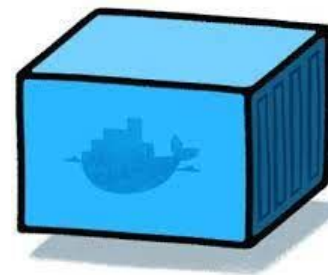
## k8s

# Contenedores



# Challenges with containers at scale

- Orchestration
- Security
- Monitoring and registration
- Scalability
- Data Storage and Persistence



# Kubernetes Advantages

- Deployment automation
- Scaling based on demand
- Application portability
- Self-healing
- Good option for microservices
- Active community and wide adoption



**kubernetes**



# Kubernetes Terminology

## Pods

- Containers
  - Red y storage

## Deployments

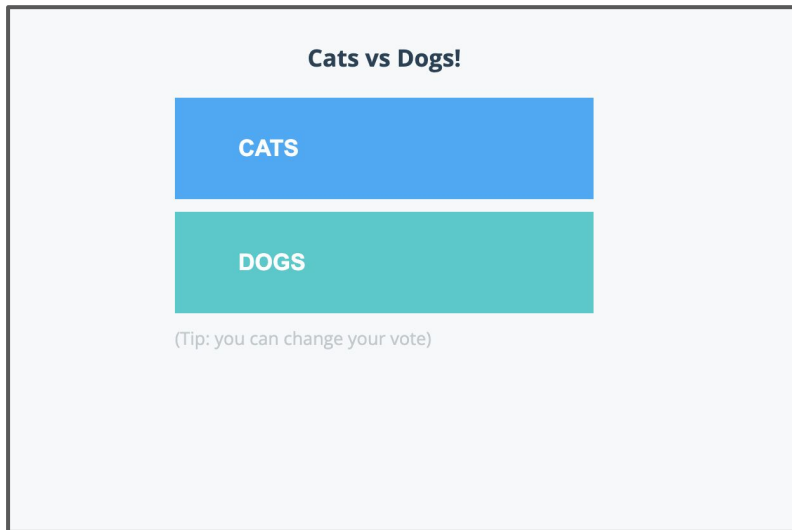
- Application Deployment
- Desired state
- Replicas

## Services

- Pod Access

# Example: Voting Application

voting-app

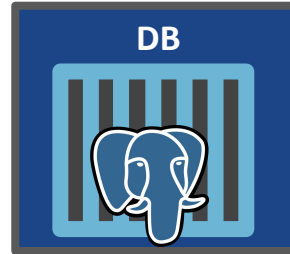
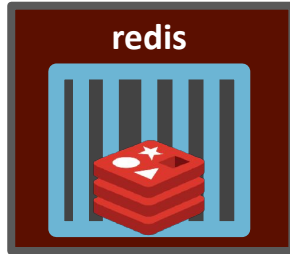
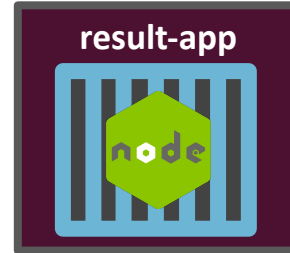


result-app



**KodeKloud:** [www.youtube.com/watch?v=XuSQU5Grvlg](https://www.youtube.com/watch?v=XuSQU5Grvlg)

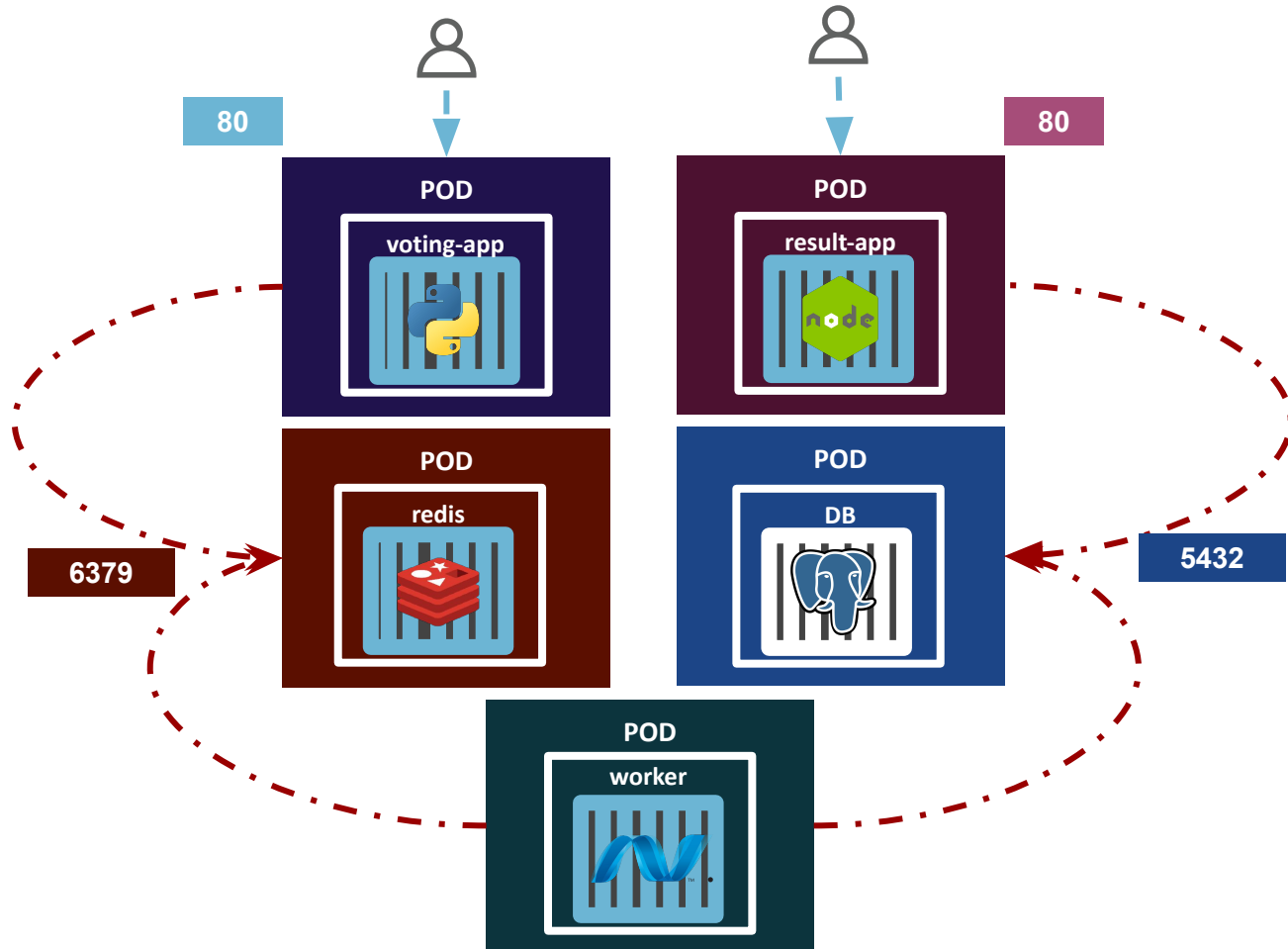
# Containers



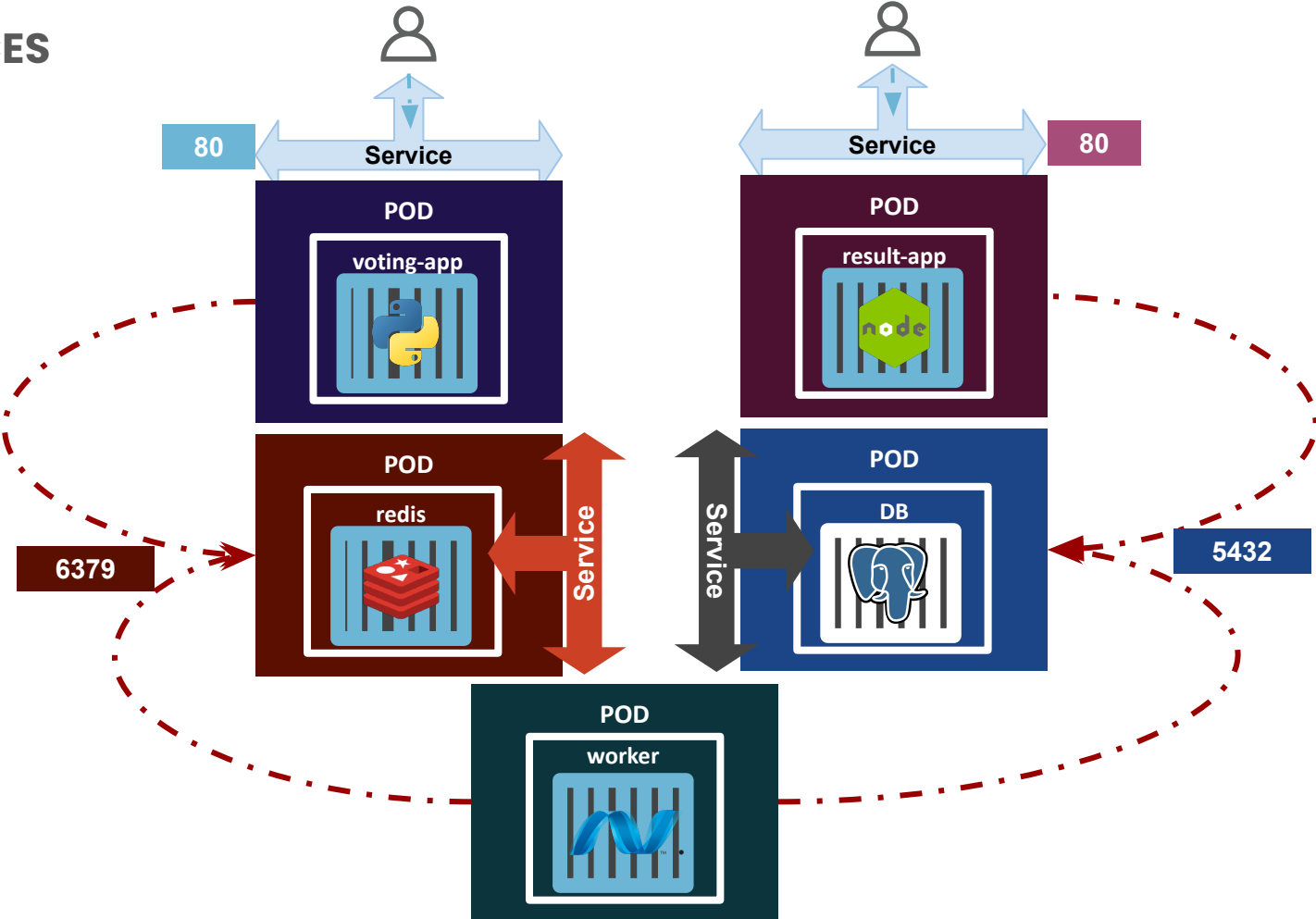
11

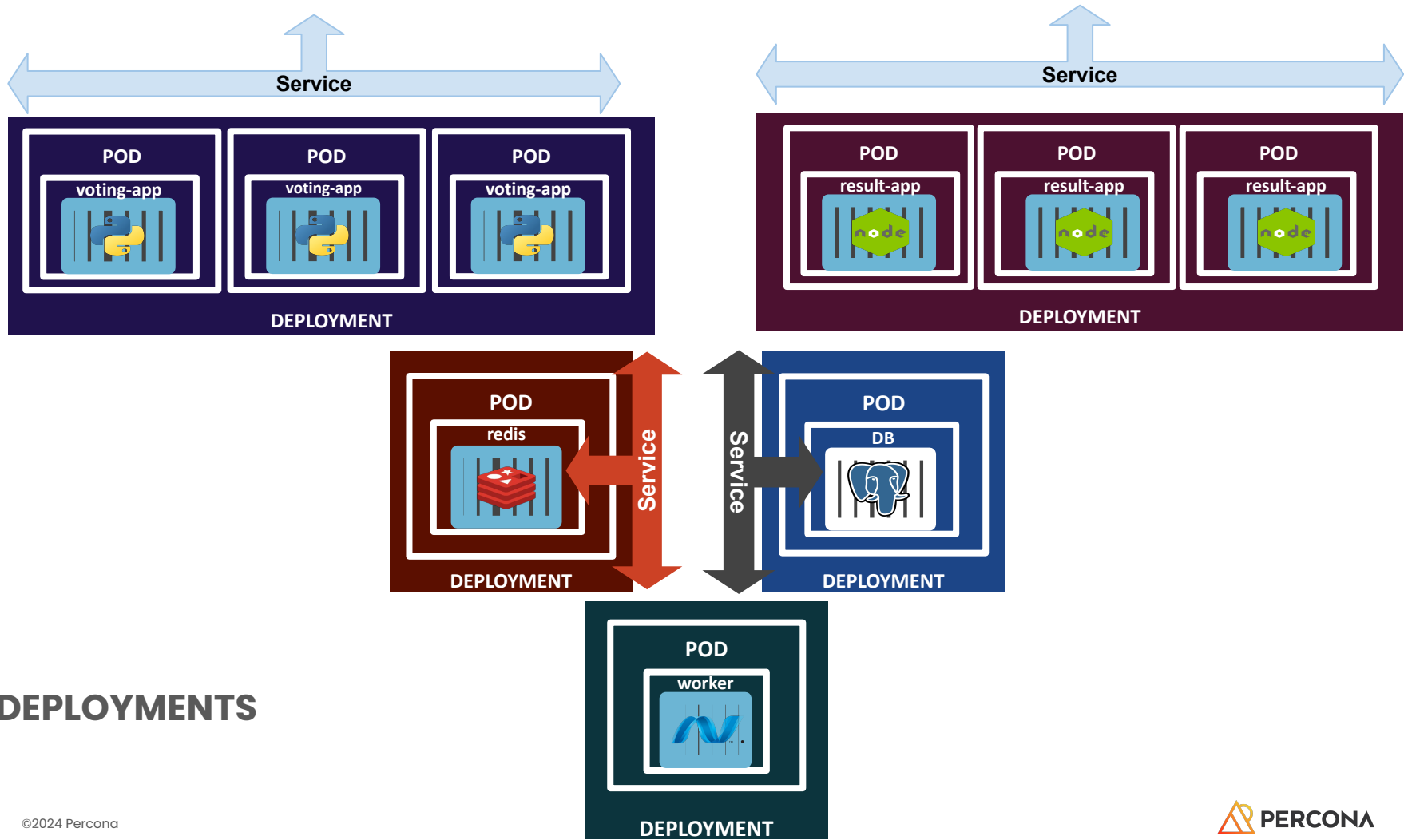


# PODS



# SERVICES





## DEPLOYMENTS

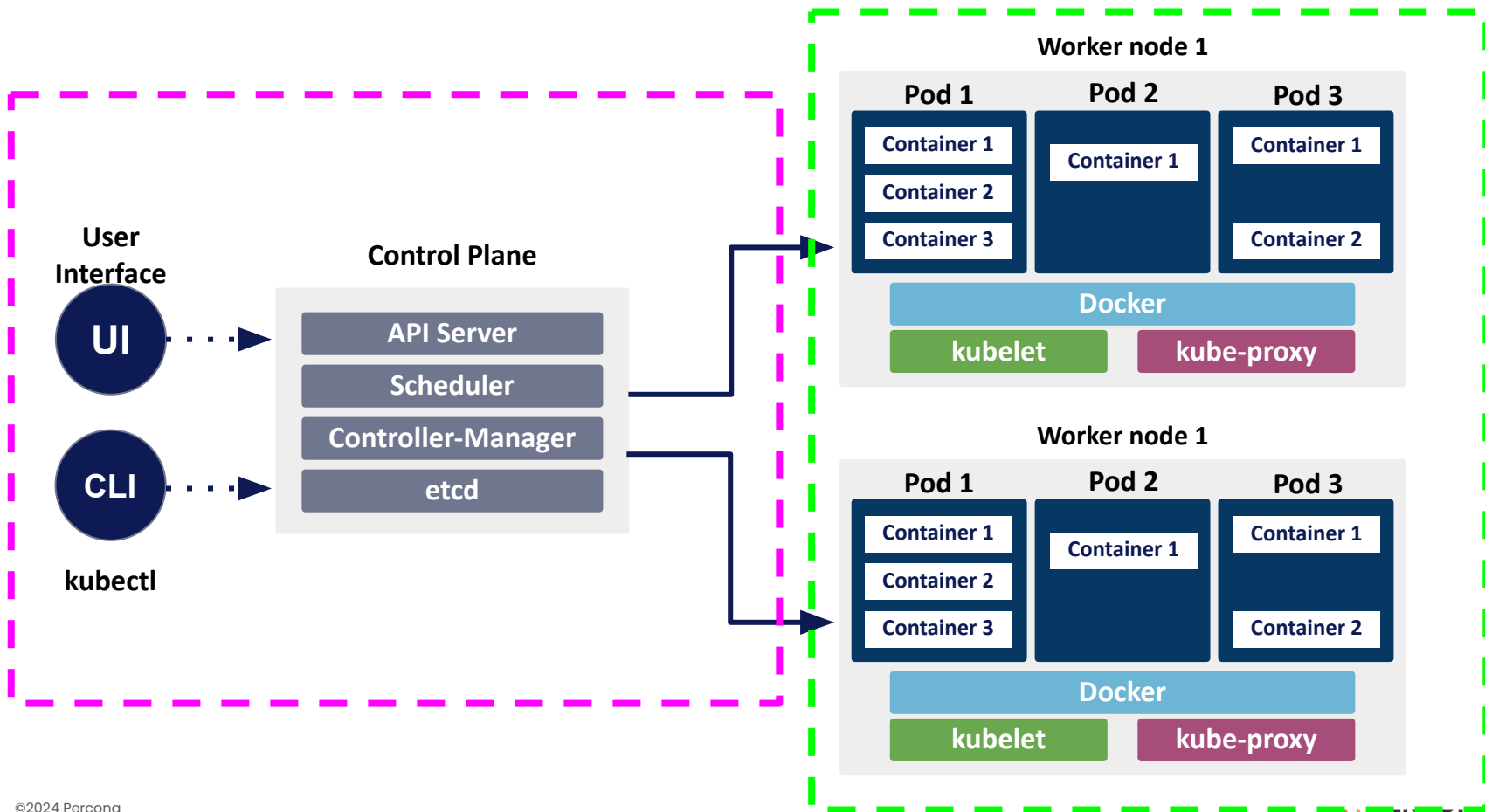
# YAML

File: `voting-app-deploy.yaml`

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: voting-app-deploy
  labels:
    name: voting-app-deploy
    app: demo-voting-app
spec:
  replicas: 1
  selector:
    matchLabels:
      name: voting-app-pod
      app: demo-voting-app
  template:
    metadata:
      name: voting-app-pod
      labels:
        name: voting-app-pod
        app: demo-voting-app
    spec:
      containers:
        - name: voting-app
          image: kodekloud/examplevotingapp_vote:v1
          ports:
            - containerPort: 80
```

15

# Kubernetes architecture





# Kubernetes Operators

# Stateless Application Scaling: **Easy**

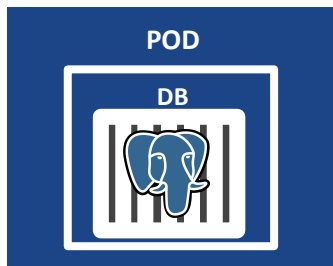
```
kubectl scale deploy/staticweb  
--replicas=4
```

19

What about applications that  
store data?

20

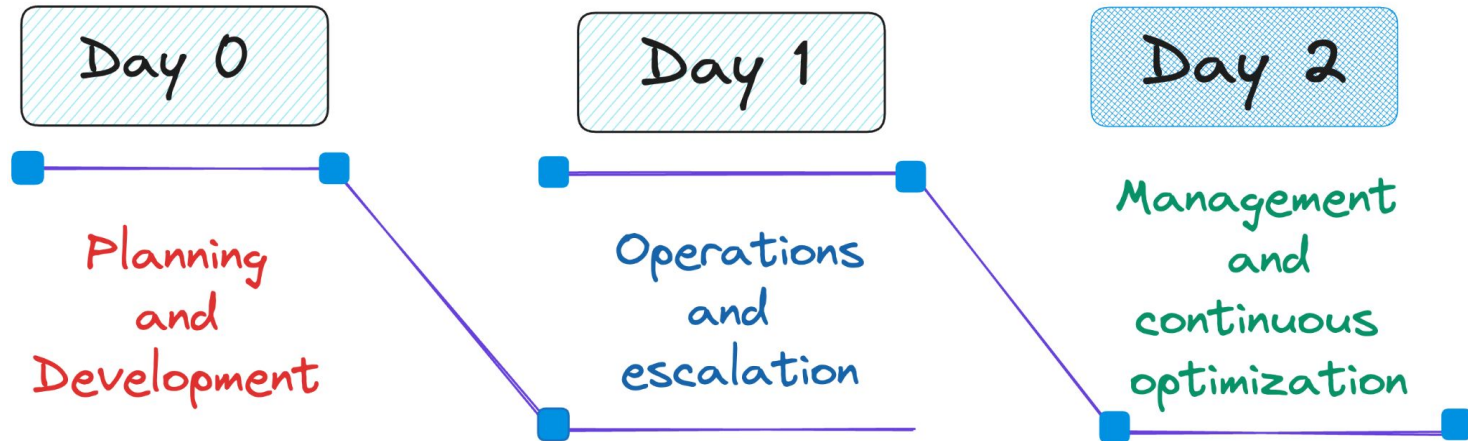
“Deploy” a database: **easy**



Running a database **over time** is the hardest



# Kubernetes Application Lifecycle



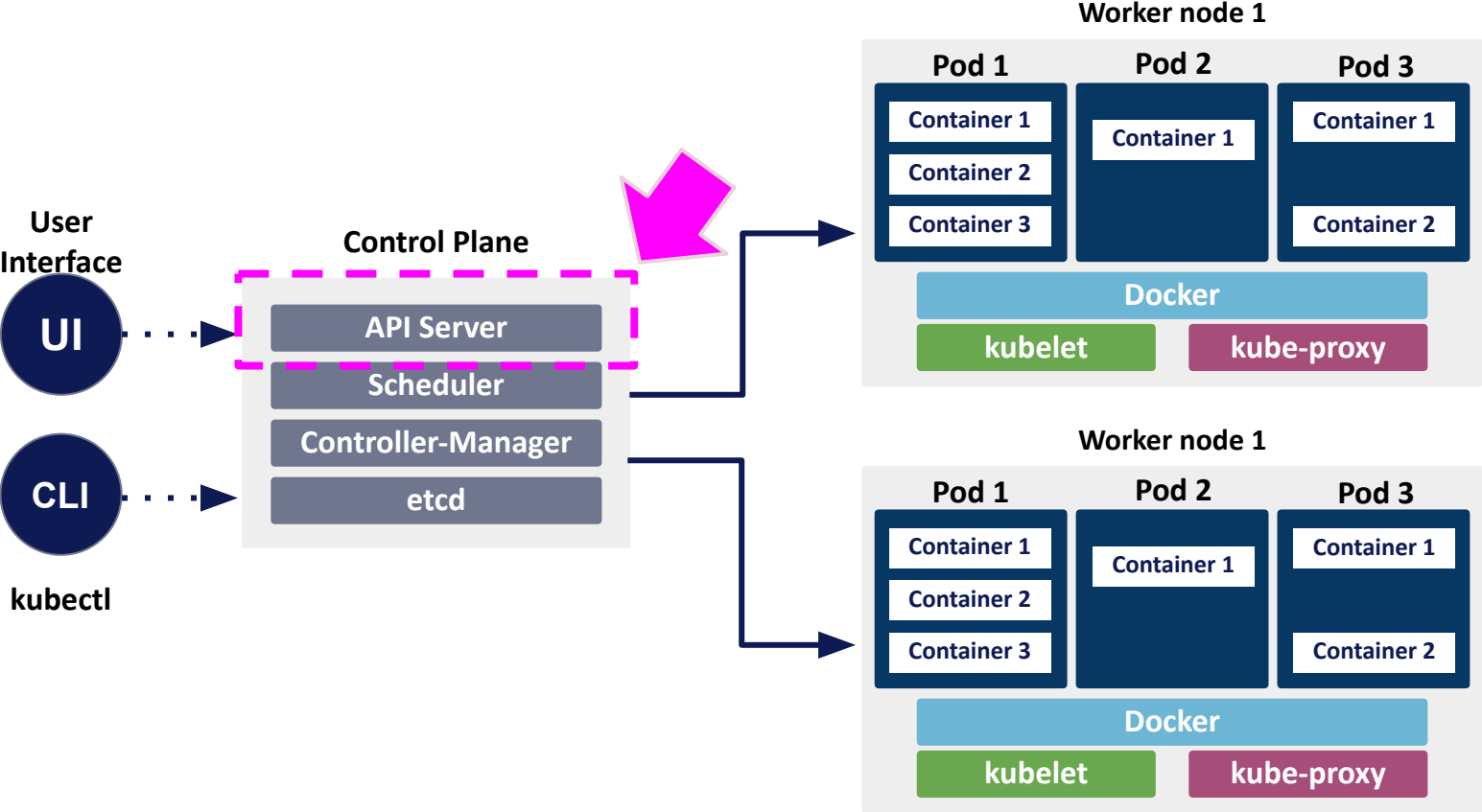
# Kubernetes Operators:

Extend the Kubernetes **API**

24



# Kubernetes architecture



## How the API is Extended:

- Custom Resource Definitions (CRD)
- Custom controllers for specific applications

# Custom Resource Definitions (CRD)

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  # name must match the spec fields below, and be in the form: <plural>.<group>
  name: crontabs.stable.example.com
spec:
  # group name to use for REST API: /apis/<group>/<version>
  group: stable.example.com
  # list of versions supported by this CustomResourceDefinition
  versions:
    - name: v1
      # Each version can be enabled/disabled by Served flag.
      served: true
      # One and only one version must be marked as the storage version.
      storage: true
      schema:
        openAPIV3Schema:
          type: object
          properties:
            spec:
              type: object
              properties:
                cronSpec:
                  type: string
                image:
                  type: string
                replicas:
                  type: integer
  # either Namespaced or Cluster
  scope: Namespaced
  names:
    # plural name to be used in the URL: /apis/<group>/<version>/<plural>
    plural: crontabs
    # singular name to be used as an alias on the CLI and for display
    singular: crontab
    # kind is normally the camelCased singular type. Your resource manifests use this.
    kind: CronTab
  # shortNames allow shorter string to match your resource on the CLI
  shortNames:
    - ct
```

**kind: CronTab**

## [CRD example](#)

# Custom Resource Definitions (CRD)

my-crontab.yaml

```
apiVersion: "stable.example.com/v1"  
kind: CronTab  
metadata:  
  name: my-new-cron-object  
spec:  
  cronSpec: "* * * * */5"  
  image: my-awesome-cron-image
```

28



kubernetes

```
kubectl apply -f my-crontab.yaml
```

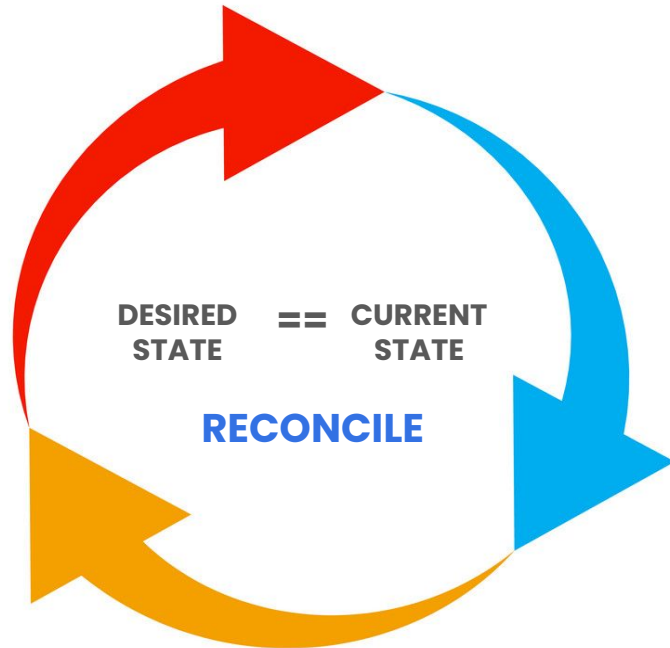
```
kubectl get crontab
```

NAME	AGE
my-new-cron-object	6s

9

# Custom Controller

Observes changes to resources defined in the **CRD** in the Kubernetes cluster, detects changes, and reacts to level them out.



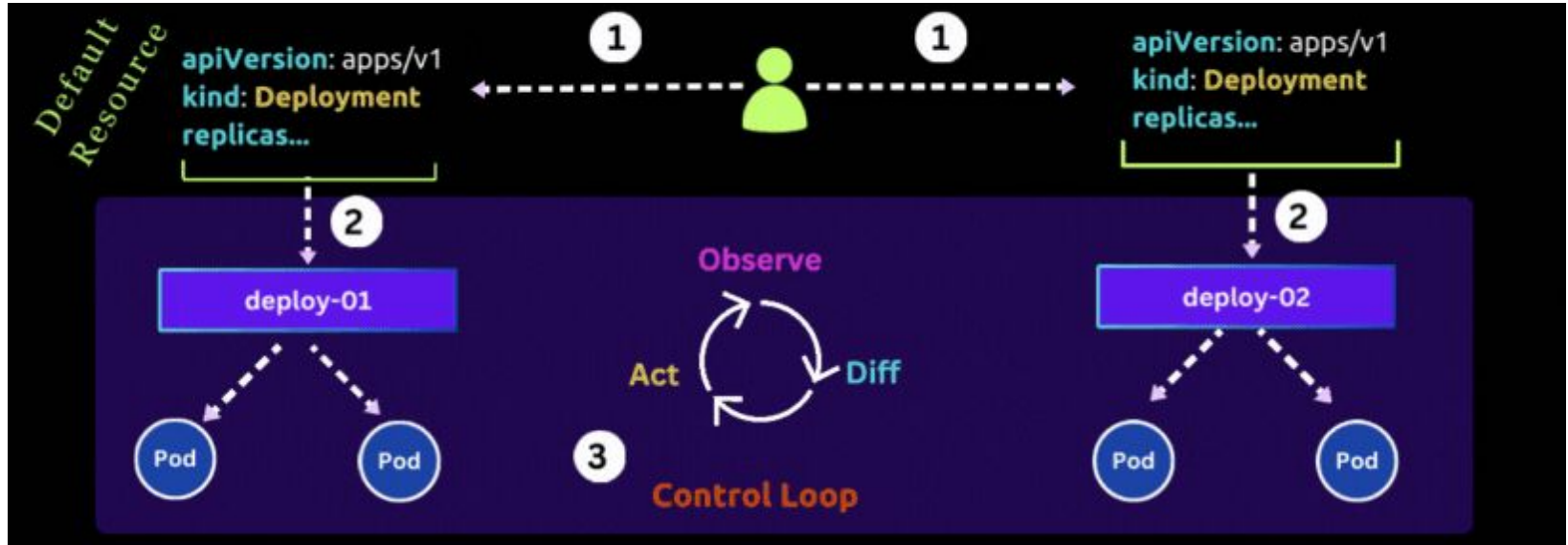
**Custom  
Resource**



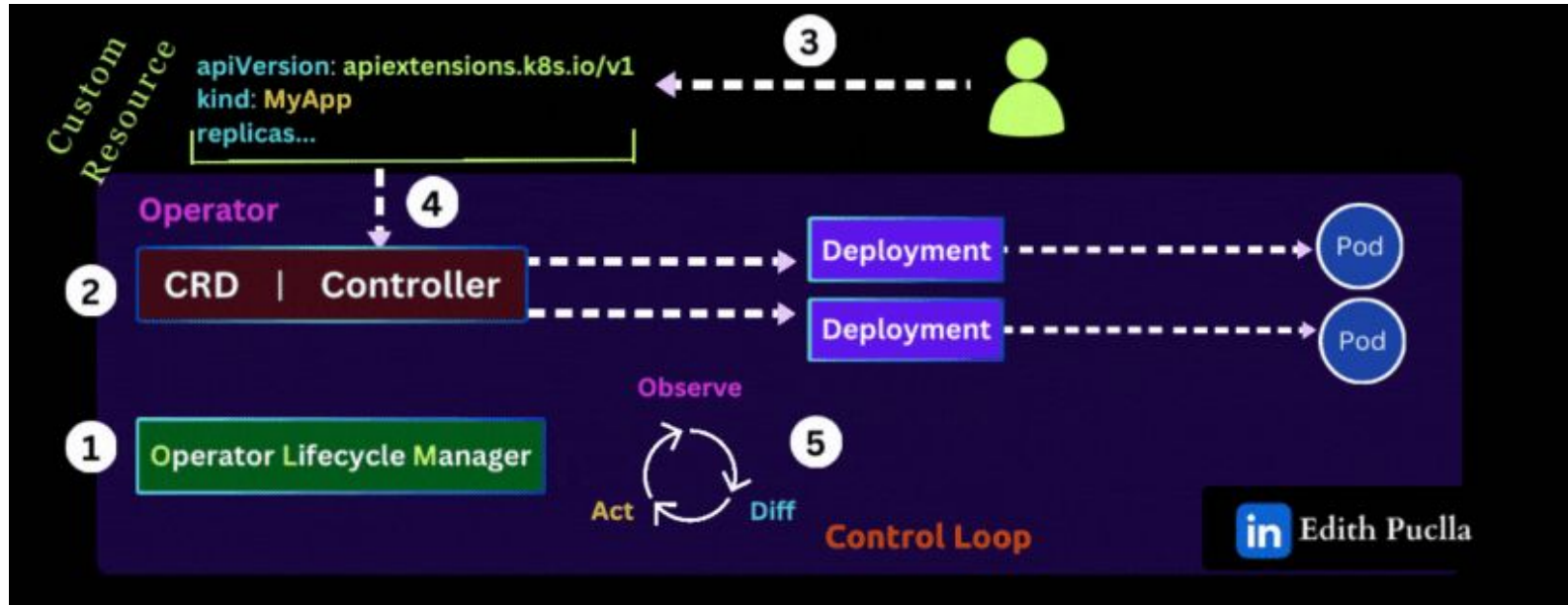
**Optionally  
Modify**

**Other K8s  
objects**

# Kubernetes without Operators

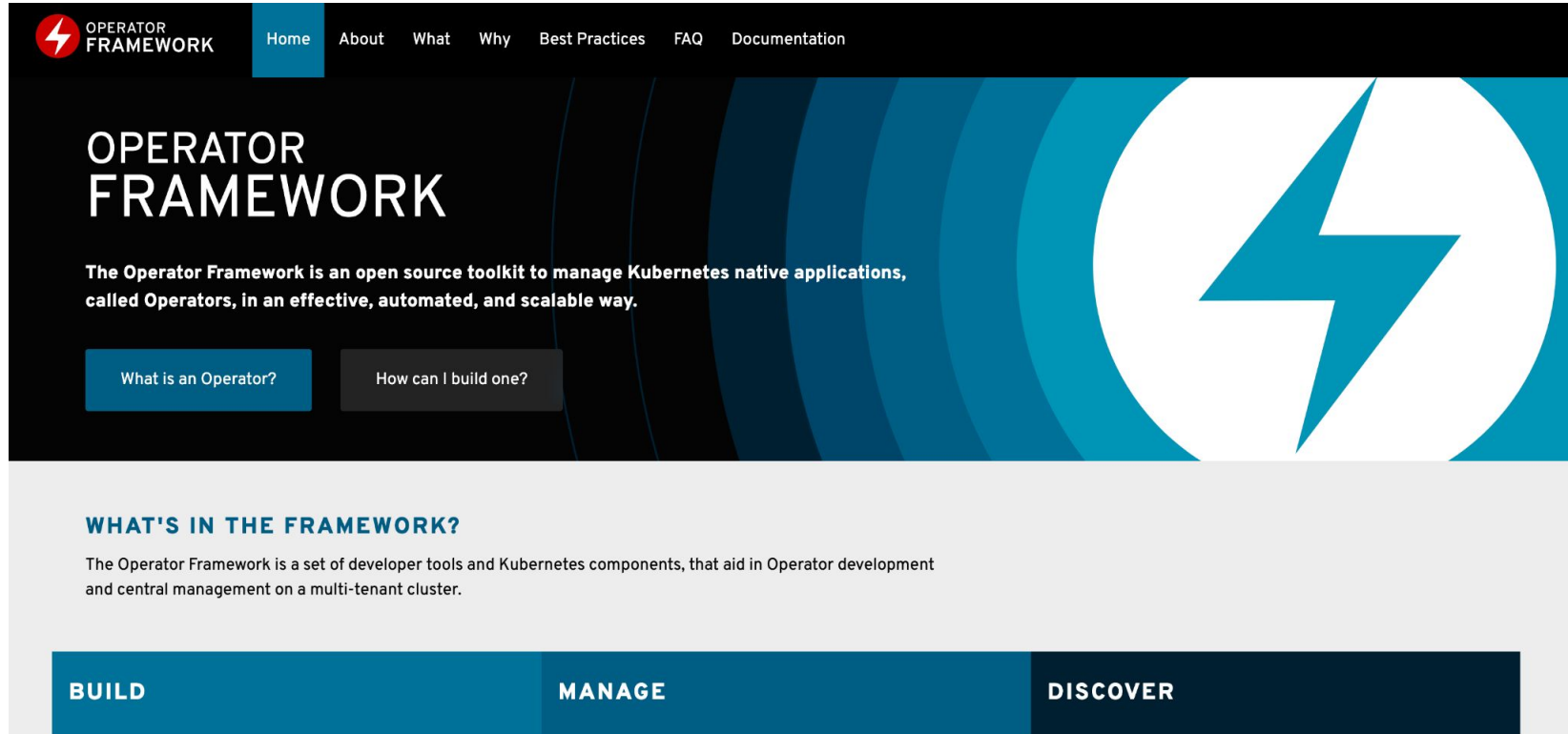


# Kubernetes with Operators





# How do I create the operators?



The screenshot shows the homepage of the Operator Framework website. At the top left is the logo, a red lightning bolt inside a circle, followed by the text "OPERATOR FRAMEWORK". To the right of the logo is a navigation menu with links: "Home", "About", "What", "Why", "Best Practices", "FAQ", and "Documentation". The "Home" link is highlighted with a blue background. Below the navigation is a large hero section with a dark background and a large blue lightning bolt icon on the right. The text in the hero section reads: "OPERATOR FRAMEWORK" in large white letters, followed by "The Operator Framework is an open source toolkit to manage Kubernetes native applications, called Operators, in an effective, automated, and scalable way." Below this text are two buttons: "What is an Operator?" (highlighted in blue) and "How can I build one?" (grey). Below the hero section is a light grey section titled "WHAT'S IN THE FRAMEWORK?" with the text: "The Operator Framework is a set of developer tools and Kubernetes components, that aid in Operator development and central management on a multi-tenant cluster." At the bottom of the page is a dark blue bar with three white text elements: "BUILD", "MANAGE", and "DISCOVER".

[Operator Framework](#)

# Welcome to OperatorHub.io

OperatorHub.io is a new home for the Kubernetes community to share Operators. Find an existing Operator or list your own today.

CATEGORIES

312 ITEMS

VIEW ▾ SORT A-Z ▾

- AI/Machine Learning
- Application Runtime
- Big Data
- Cloud Provider
- Database
- Developer Tools
- Drivers and plugins
- Integration & Delivery
- Logging & Tracing
- Modernization & Migration
- Monitoring
- Networking
- OpenShift Optional
- Security
- Storage
- Streaming & Messaging

PROVIDER

Aerospike (1)



**Aerospike Kubernetes Operator**  
provided by Aerospike

The Aerospike Kubernetes Operator automates the



**Aiven Operator**  
provided by aiven

Manage your https://aiven.io resources with Kubernetes.



**Akka Cluster Operator**  
provided by Lightbend, Inc.

Run Akka Cluster applications on Kubernetes.



**Altinity Operator for ClickHouse**  
provided by Altinity

ClickHouse Operator manages full lifecycle of ClickHouse.



**Alvearie Imaging Ingestion Operator**  
provided by Alvearie

The Alvearie Imaging Ingestion provides a collection of



**Anchore Engine Operator**  
provided by Anchore Inc.

Anchore Engine - container image scanning service for policy-based security, best



**Ansible Galaxy**  
provided by Galaxy Community

Ansible Galaxy is Ansible's official hub for sharing Ans



**Apache Spark Operator**  
provided by radanalytics.io

An operator for managing the Apache Spark clusters and intelligent applications that



**API Operator for Kubernetes**  
provided by WSO2

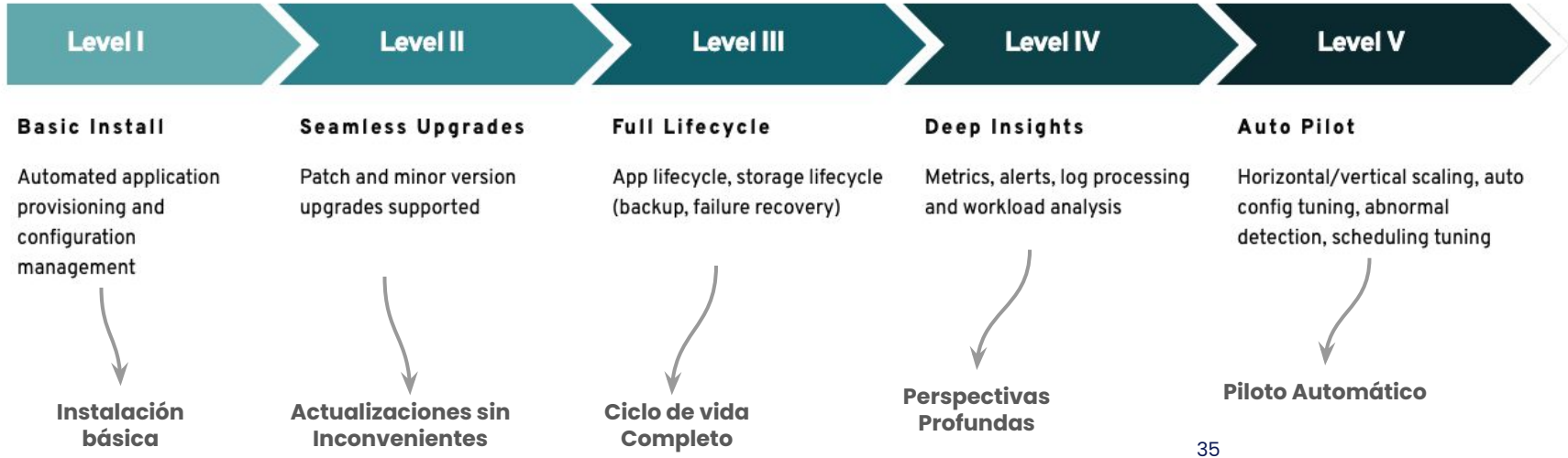
API Operator provides a fully automated experience for



**APIcast**  
provided by Red Hat

APIcast is an API gateway built on top of NGINX. It is part of the Red Hat 3scale API

# Capability Model



35



# Percona Operator for MySQL based on Percona XtraDB Cluster

Percona Operator for MySQL based on Percona XtraDB Cluster manages the lifecycle of Percona XtraDB cluster instances.

Home > Percona Operator for MySQL based on Percona XtraDB Cluster

## Percona Operator for MySQL based on Percona XtraDB Cluster

Install

### Percona is Cloud Native

Percona Operator for MySQL based on Percona XtraDB Cluster is an open-source drop in replacement for MySQL Enterprise with synchronous replication running on Kubernetes. It automates the deployment and management of the members in your Percona XtraDB Cluster environment. It can be used to instantiate a new Percona XtraDB Cluster, or to scale an existing environment.

Consult the [documentation](#) on the Percona Operator for MySQL based on Percona XtraDB Cluster for complete details on capabilities and options.

### Supported Features

- **Scale Your Cluster** change the `size` parameter to [add or remove members](#) of the cluster. Three is the minimum recommended size for a functioning cluster.
- **Manage Your Users** [add, remove, or change](#) the privileges of database users
- **Automate Your Backups** [configure cluster backups](#) to run on a scheduled basis. Backups can be stored on a persistent volume or S3-compatible storage. Leverage [Point-in-time recovery](#) to reduce RPO/RT0.
- **Proxy integration** choose HAProxy or ProxySQL as a proxy in front of the Percona XtraDB Cluster. Proxies are deployed and configured automatically with the

CHANNEL  
stable

VERSION  
1.12.0 (Current)

CAPABILITY LEVEL <sup>Ⓜ</sup>

- Basic Install
- Seamless Upgrades
- Full Lifecycle
- Deep Insights
- Auto Pilot

PROVIDER  
Percona

LINKS

# Percona Operators simplify setting up and maintaining robust, enterprise-grade MySQL, PostgreSQL, and MongoDB clusters on Kubernetes

Try Percona Operators:

- Percona Operator for MySQL
- Percona Operator for MongoDB
- Percona Operator for PostgreSQL

## Percona Everest open source, cloud-native database platform

- [docs.percona.com/everest](https://docs.percona.com/everest)

Ask questions and leave your feedback:

- [percona.community](https://percona.community)
- [forums.percona.com](https://forums.percona.com)
- [github.com/percona](https://github.com/percona)



**PERCONA**

Kubernetes  
Operators

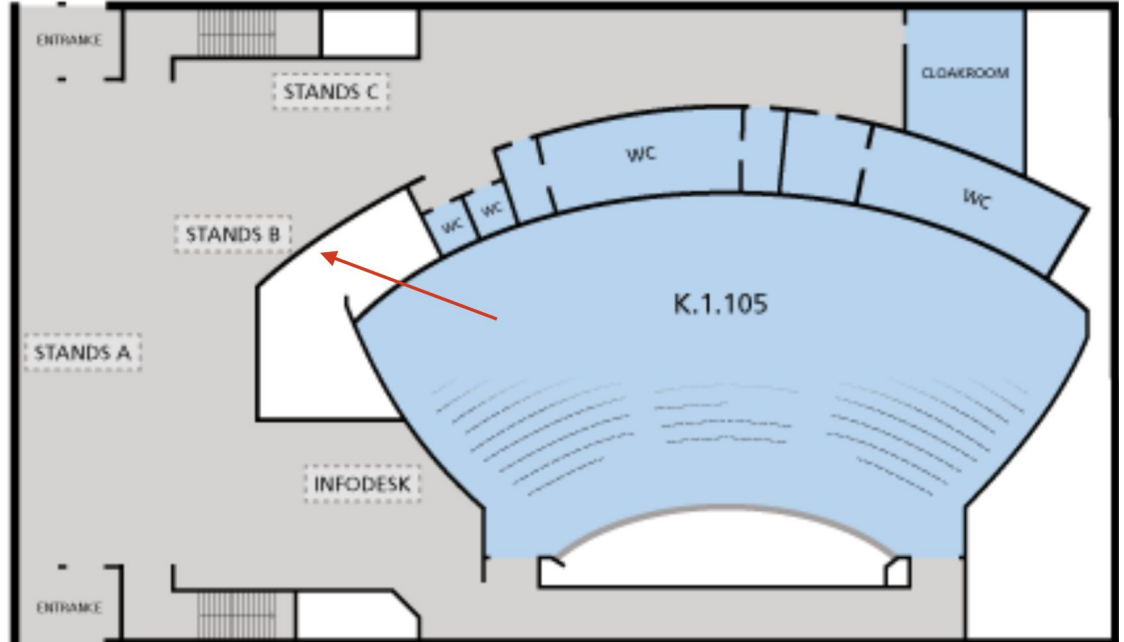


**PERCONA**  
Everest



PERCONA

# Building K level 1





**Thank You!**

[percona.com](https://percona.com)