

Kubernetes Operators

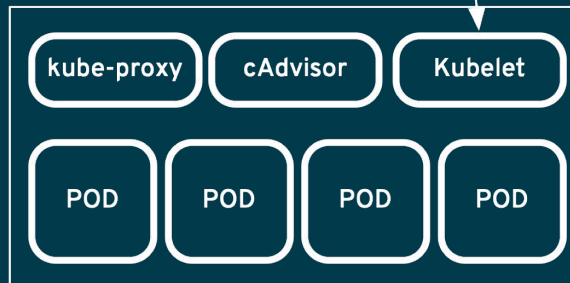
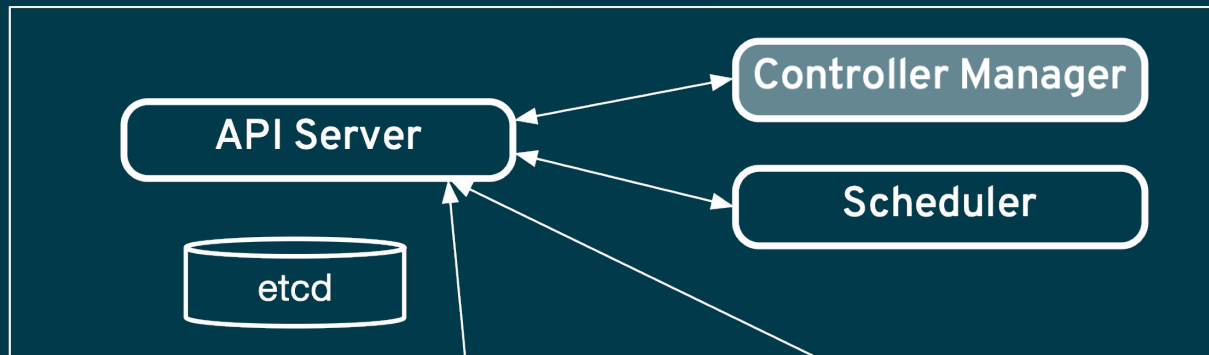
Dr. Roland Huß
Principal Software Engineer
@ro14nd
<https://k8spatterns.io>



Kubernetes



Master



Node



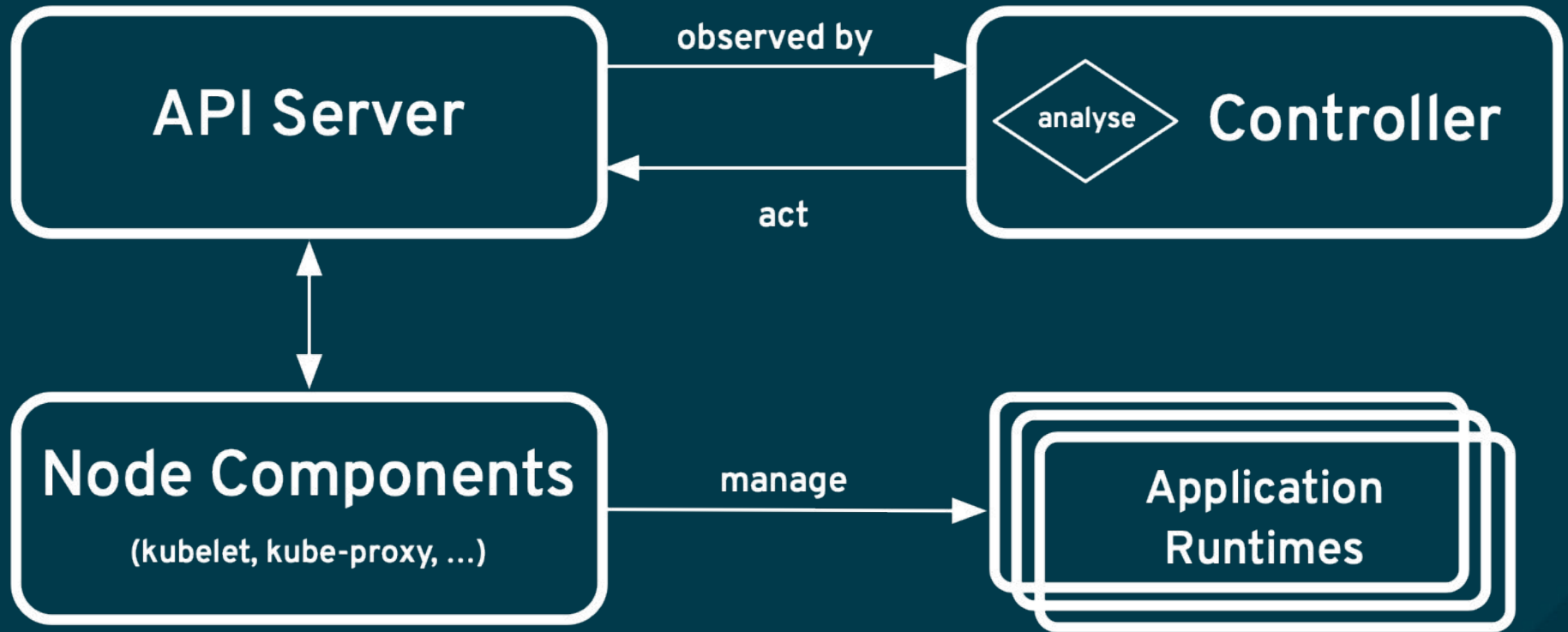
Node

Kubernetes Controllers

- Kubernetes: Declarative orchestration platform
- Based on resource objects for declaring target state
- Reconciliation:
 - Observe **current** state
 - Analyse and compare against **declared** state
 - Bring current state closer to declared state

Observe - Analyse - Act

Observe - Analyze - Act



Custom Controller

- Watches Kubernetes Resources
- Enhances platform behaviour or introduces new features
- Regular Kubernetes application (Deployments, Pod)
- Running permanently in the background
- Common Custom Controller triggers:
 - Labels
 - Annotations
 - ConfigMaps

Operators

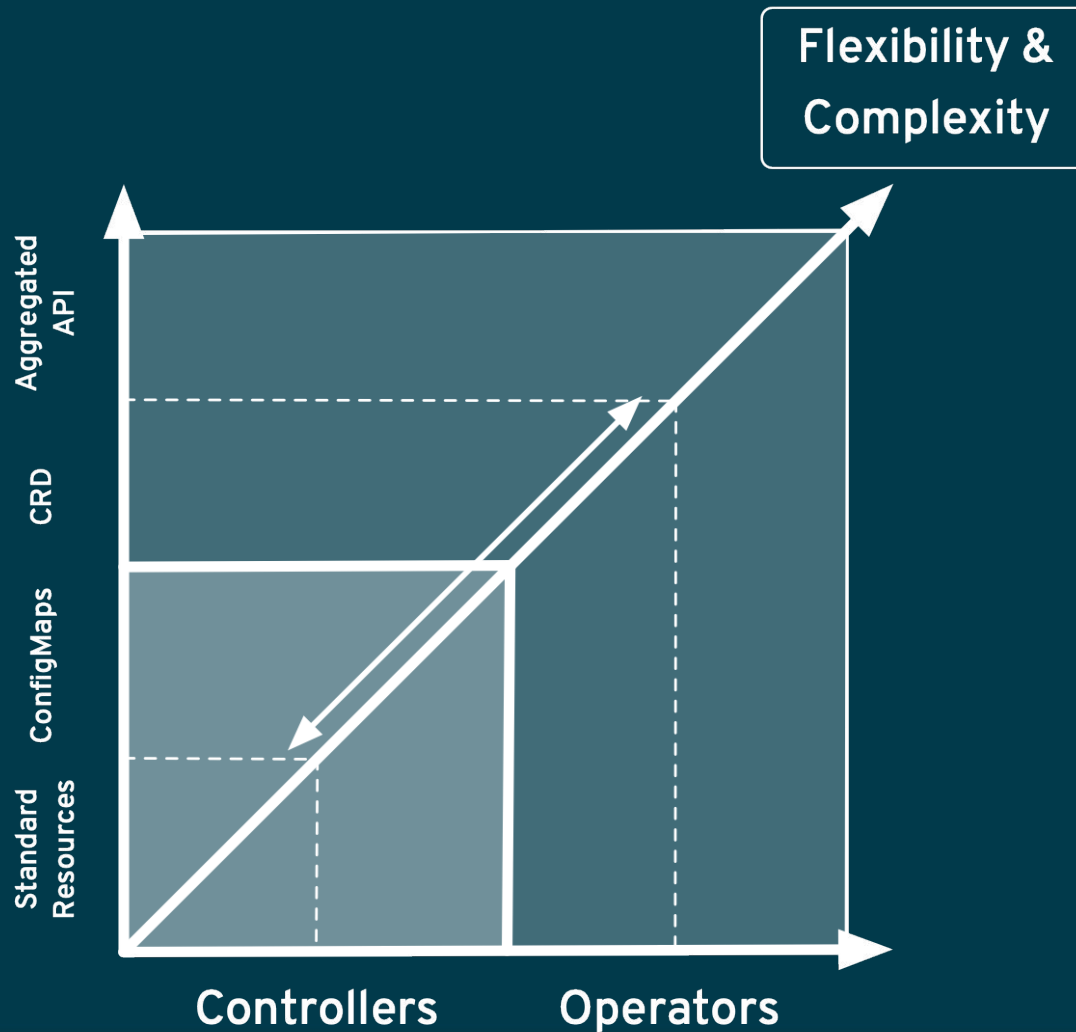
An **Operator** is a Kubernetes **Controller** that understands **two domains**: Kubernetes and something else. By combining knowledge of both areas, it can **automate tasks** that usually require a human operator that understands both domains.

– *Jimmy Zelinskie*

<https://github.com/kubeflow/tf-operator/issues/300#issuecomment-357527937>

- Operator **IS-A** Controller
- Custom domain modelled as **Custom Resource Definition**
- Semantically: Operator automates operational tasks and is codified knowledge
- Technically: Operator = Controller + CRD
- CoreOS pioneered the Operator movement

Controller Operator Spectrum



Custom Resource Definition

```
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
metadata:
  name: prometheuses.monitoring.coreos.com
spec:
  group: monitoring.coreos.com
  names:
    kind: Prometheus
    plural: prometheuses
  scope: Namespaced
  version: v1
  validation:
    openAPIV3Schema: ....
```

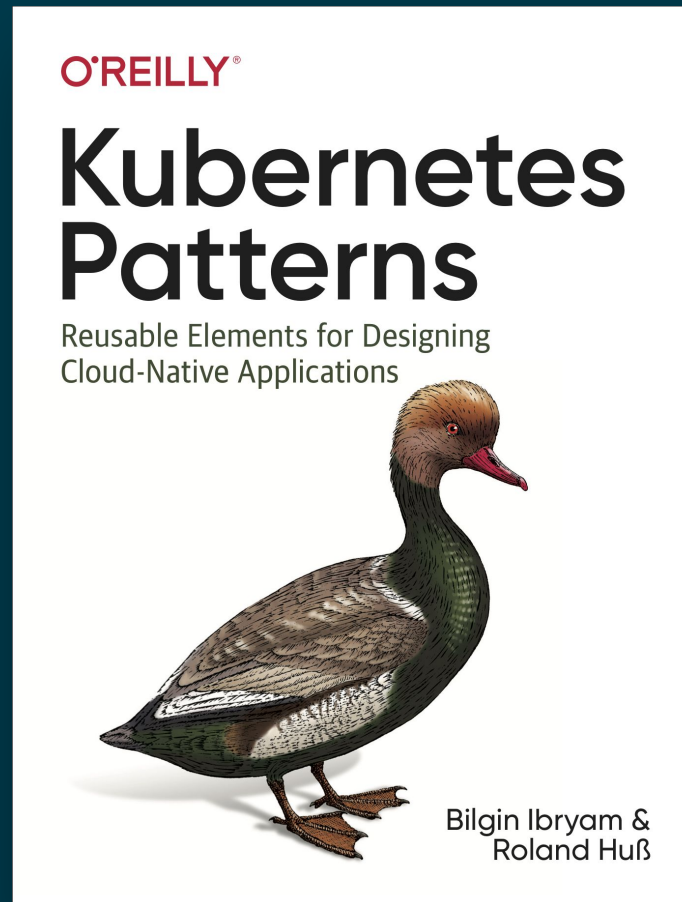

Custom Resource

```
apiVersion: monitoring.coreos.com/v1
kind: Prometheus
metadata:
  name: prometheus
spec:
  serviceMonitorSelector:
    matchLabels:
      team: frontend
  resources:
    requests:
      memory: 400Mi
status:
  phase: Installed
```

CRD Classification

- **Installation CRDs**
 - Installing and operating applications
 - Backup and Restore
 - Monitoring and self-healing
 - Example: Prometheus for installing Prometheus & components
- **Application CRDs**
 - Application specific domain concepts
 - Example: ServiceMonitor for registering Kubernetes service to be scraped by Prometheus

KUBERNETES PATTERNS



<https://k8spatterns.io>

https://github.com/operator-framework/awesome-operators

README.md		
Awesome Operators in the Wild		
<p>Operators are Kubernetes native applications. We define native as being both managed using the Kubernetes APIs via kubectl and ran on Kubernetes as containers. Operators take advantage of Kubernetes's extensibility to deliver the automation advantages of cloud services like provisioning, scaling, and backup/restore while being able to run anywhere that Kubernetes can run.</p> <p>This list is built by the community. Have you built or are you using an Operator that is not listed? Please send a pull request and we will add that Operator to the list.</p> <p>If you want to start building an Operator, you should definitely look into the Operator SDK.</p>		
App Name	Github	Description
Aerospike	traveldance/aerospike-operator	Aerospike is a NoSQL distributed database. This Operator manages Aerospike clusters atop Kubernetes, automating their creation and administration.
Airflow	GoogleCloudPlatform/airflow-operator	A Kubernetes operator to manage Apache Airflow.
Android SDK	aerogear/android-sdk-operator	A Kubernetes operator to manage android sdk packages synchronization in a persistent volume.
ArangoDB	arangodb/kube-arangodb	ArangoDB Kubernetes Operator - Start ArangoDB on Kubernetes in 5min.
Velero	heptio/velero	Velero (formerly Ark) is a utility for managing disaster recovery, this operator manages the backup and restoration of cluster components (pv,pvc,deployments, etc.) to aid in disaster recovery.
AWS	giantswarm/aws-operator	Manages Kubernetes clusters running on Amazon Web Services
AWS Services	aws-labs/aws-service-operator	Manages AWS services that are used by your applications running in Kubernetes.
Camel-k	apache/camel-k	Lightweight integration framework built from Apache Camel that runs natively on Kubernetes
Cassandra #1	instaclustr/cassandra-operator	Kubernetes operator for Apache Cassandra.
Cassandra #2	vgkowsky/cassandra-operator	Kubernetes operator for cassandra clusters automation.

OperatorHub.io | The registry f x +

https://www.operatorhub.io/?provider=%5B"Red+Hat"%2C"Jaeger"%2C"MongoDB"%5D

OperatorHub.io

Search OperatorHub...

Contribute v

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.


PROVIDER 5 ITEMS VIEW SORT A-Z v

- ☐ Amazon Web Services (1)
- ☐ CNCF (1)
- ☐ Couchbase (1)
- ☐ Crunchy Data Solutions (1)
- ☐ Dynatrace (1)
- ☒ Jaeger (1)
- ☒ MongoDB (1)
- ☐ Percona (1)
- ☐ PlanetScale (1)
- ☒ Red Hat (3)
- ☐ Redis Labs (1)

[Show less](#)


CAPABILITY LEVEL

- ☐ Basic Install (4)
- ☐ Seamless Upgrades (3)
- ☐ Full Lifecycle (6)




Jaeger Tracing
provided by Jaeger

Provides tracing, monitoring and troubleshooting microservices-based




Kubernetes Federation
provided by Red Hat

Gain Hybrid Cloud capabilities between your clusters with Kubernetes Federation.




MongoDB
provided by MongoDB, Inc

The MongoDB Enterprise Kubernetes Operator enables easy deploys of MongoDB



Prometheus Operator
provided by Red Hat

The Prometheus Operator for Kubernetes provides easy monitoring definitions for



Strimzi Kafka
provided by Red Hat

Run an Apache Kafka cluster, including Kafka Connect, ZooKeeper and more.

OperatorHub.io | The registry

+

← → ↺


https://www.operatorhub.io/operator/strimzi-cluster-operator.v0.11.1

🔍 ☆ ⓘ 🌐 👤 ⋮

OperatorHub.io

🔍 Search OperatorHub...

Contribute ▾



Strimzi Kafka

Run an Apache Kafka cluster, including Kafka Connect, ZooKeeper and more.

[Home](#) > [Strimzi Kafka](#)

Strimzi Kafka

Strimzi provides a way to run an [Apache Kafka](#) cluster on [Kubernetes](#) or [OpenShift](#) in various deployment configurations. See our [website](#) for more details about the project.

Supported Features

- **Manages the Kafka Cluster** - Deploys and manages all of the components of this complex application, including dependencies like ZooKeeper that are traditionally hard to administer.
- **Includes Kafka Connect** - Allows for configuration of common data sources and sinks to move data into and out of the Kafka cluster.
- **Topic Management** - Creates and manages Kafka Topics within the cluster.
- **User Management** - Creates and manages Kafka Users within the cluster.

Upgrading your Clusters

The Strimzi Operator understands how to run and upgrade between a set of Kafka versions. When specifying a new version in your config, check to make sure you aren't using any features that may have been removed. See [the upgrade guide](#) for more information.

Install

OPERATOR VERSION

0.11.1 (latest) ▾

CAPABILITY LEVEL

☒ Basic Install

☒ Seamless Upgrades

☐ Full Lifecycle

☐ Deep Insights

☐ Auto Pilot

PROVIDER

Red Hat

LINKS

OperatorHub - Red Hat OpenShift

Not Secure | https://console-openshift-console.apps.rhuss-dev.devcluster.openshift.com/operatorhub/ns/default?category=Database

Red Hat OpenShift Container Platform

kube:admin

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

Project: default

OperatorHub

Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. Operators can be installed on your clusters to provide optional add-ons and shared services to your developers. Once installed, the capabilities provided by the Operator appear in the [Developer Catalog](#), providing a self-service experience.

All Items Database

AI/Machine Learning 18 items

Application Monitoring

Big Data

Cloud Provider

Database

Developer Tools

Integration & Delivery

Logging & Tracing

Monitoring

Networking

OpenShift Optional

Security

Storage

Streaming & Messaging

Other

CockroachDB
provided by Helm Community

CockroachDB Operator based on the CockroachDB helm chart

Couchbase Operator
provided by Couchbase

An operator to create and manage a Couchbase Cluster

Crunchy Postgres Cluster
provided by CrunchyData.com

A Postgres Operator from Crunchydata.com

Crunchy PostgreSQL Enterprise
provided by Crunchy Data

etcd
provided by CNCF

Hazelcast Operator
provided by Hazelcast, Inc

Operator Development

- Operator can be implemented in any language
- Frameworks:
 - Operator Framework (Golang, Helm, Ansible)
 - <https://github.com/operator-framework>
 - Kubebuilder (Golang)
 - <https://github.com/kubernetes-sigs/kubebuilder>
 - Metacontroller (Language agnostic)
 - <https://metacontroller.app/>
 - jvm-operators (Java, Groovy, Kotlin,)
 - <https://github.com/jvm-operators>
 - Base on fabric8io/kubernetes-client

Operator Framework



**OPERATOR
SDK**

Build Operators



**OPERATOR
LIFECYCLE MANAGER**

Install, update, and
manage
Operators



**OPERATOR
METERING**

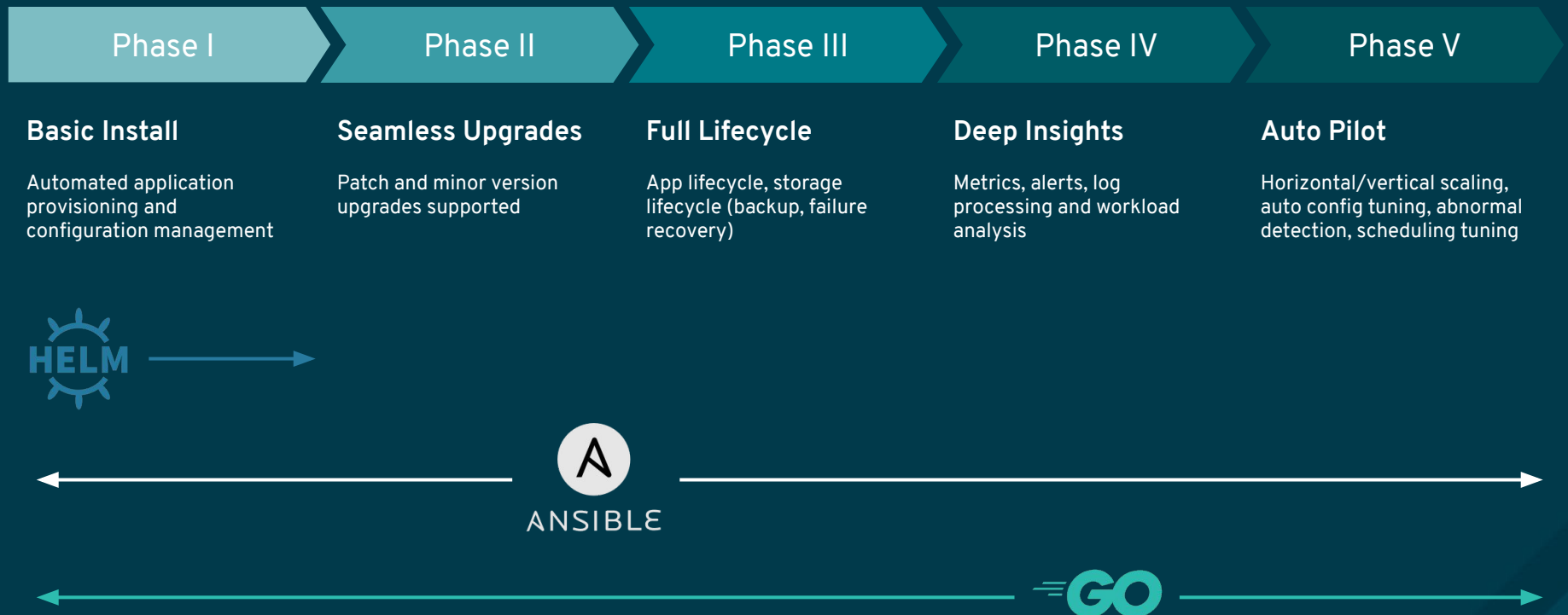
Operator usage
reporting

<https://github.com/operator-framework>

Operator SDK

- Framework and Toolkit for creating Operators
- Scaffolding of a project skeleton
- Scorecard check
- Uses controller-runtime under the hood
 - <https://github.com/kubernetes-sigs/controller-runtime>
- Modes
 - Golang
 - Ansible
 - Helm

Operator Maturity Model



DEMO

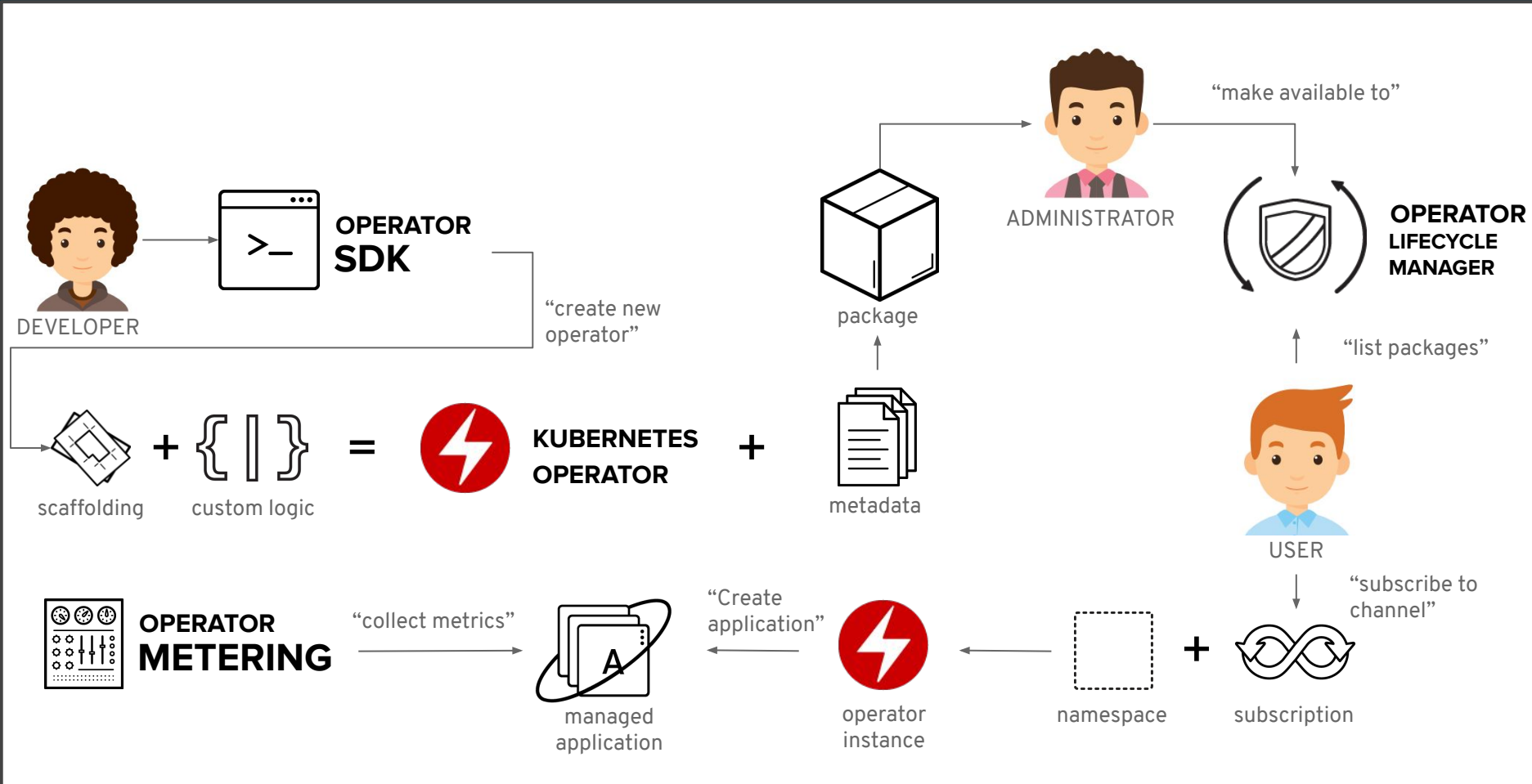
Operation Lifecycle Manager (OLM)

- OLM: Operators for managing lifecycle or Operators
- “Operator Operator”
- Main components:
 - **olm-operator**: Managing operator deployments
 - **catalog-operator**: Managing subscriptions to channels
 - OperatorGroups for implementing multi-tenancy of operators
- Installs packages from OperatorHub.io
- Included OOTB in OpenShift 4
- Main CRD: `ClusterServiceVersion`

ClusterServiceVersion

- Metadata (name, description, version, links, labels, icon, ...)
- Operator Installation
 - Type: Deployment
 - Set of service accounts / required permissions
 - Set of deployments
- CRDs
 - Type
 - Owned - managed by this service
 - Required - must exist in the cluster for this service to run
 - Resources - a list of resources that the Operator interacts with
 - Descriptors - annotate CRD spec and status fields to provide semantic information

Operator Framework Summary



Thank you



<https://k8spatterns.io>



@ro14nd



@k8spatterns