



# Operators

---

Robert Bohne

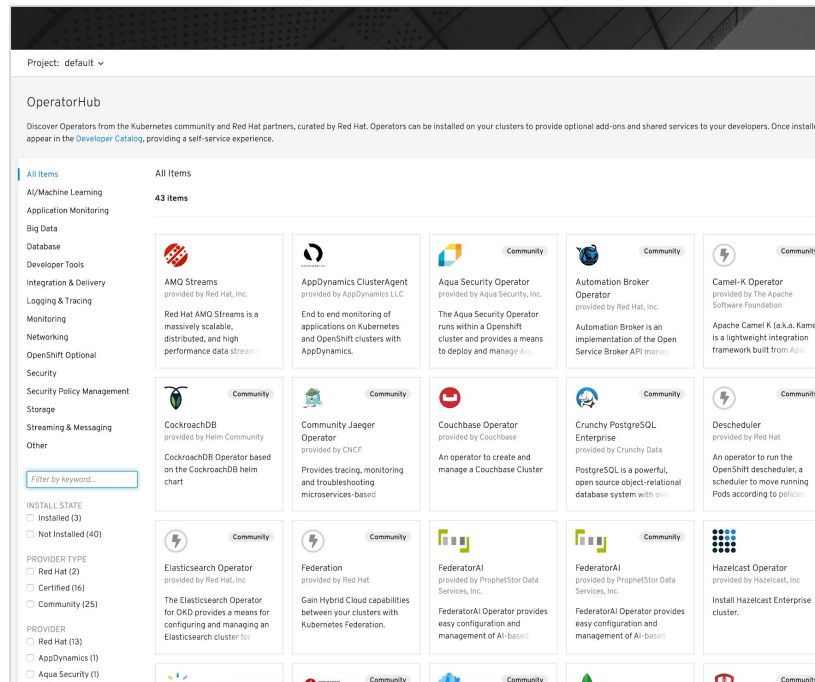
SR. SPECIALIST SOLUTION ARCHITECT | OPENSIFT

Twitter: @RobertBohne

Operators are **automated software managers** that deal with installation and lifecycle of Kubernetes applications

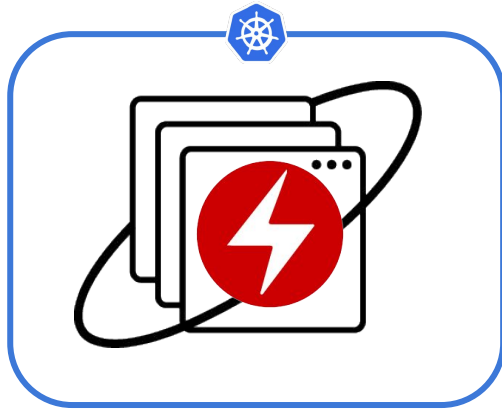
# OperatorHub in OpenShift

The embedded registry for  
Community and Certified  
Operators from Red Hat and  
Partners, tested and verified on  
OpenShift 4



1. Application-specific custom **controller**
2. Custom Resource Definition (CRD)

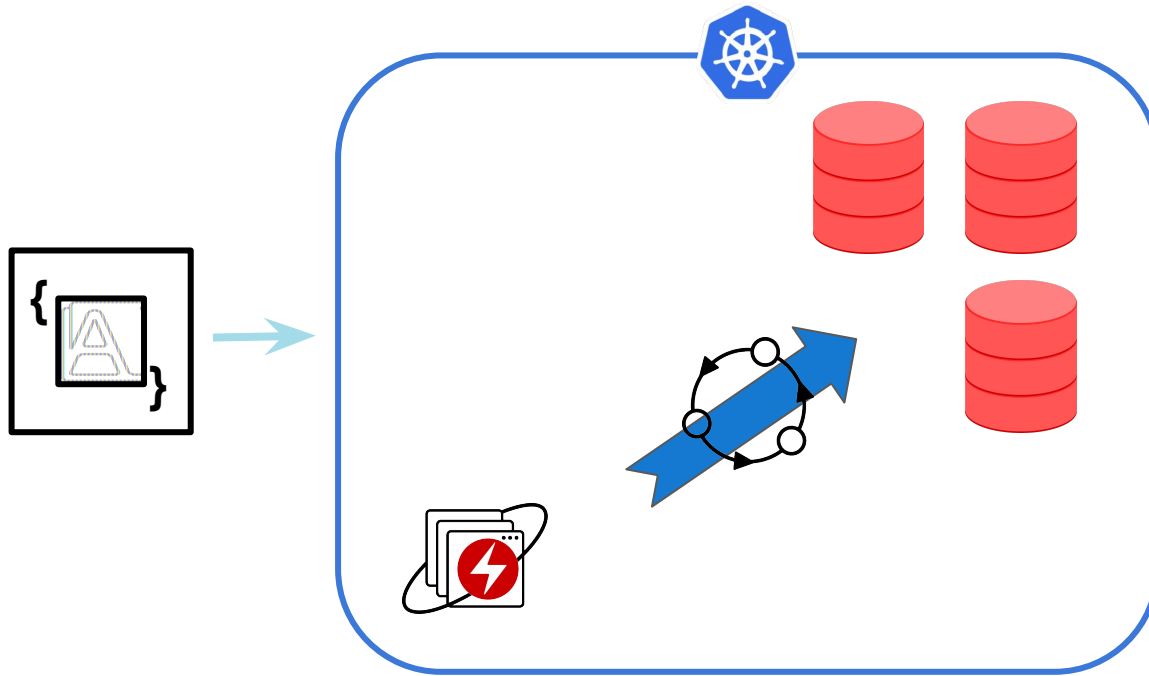
# Application-Specific Controller

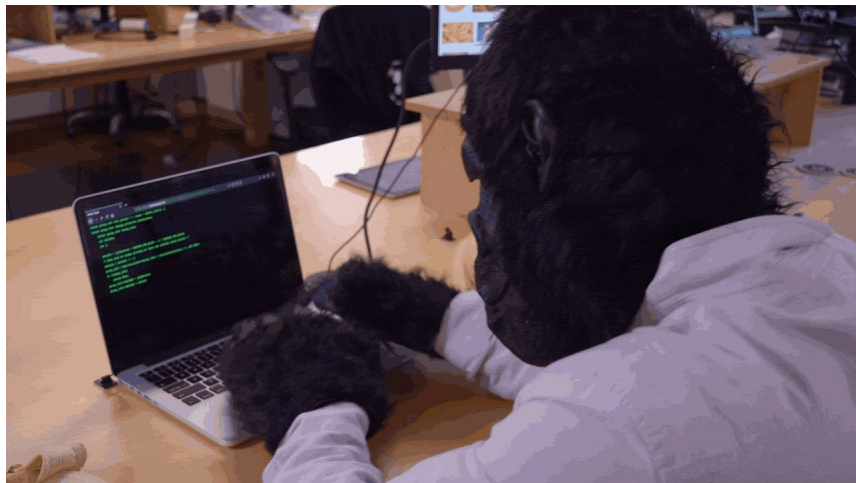


# Custom Resource Definition (CRD)

```
kind: ProductionReadyDatabase
apiVersion: database.example.com/v1alpha1
metadata:
  name: my-production-ready-database
spec:
  clusterSize: 3
  readReplicas: 2
  version: v4.0.1
[...]
```

# Custom Resource (CR)



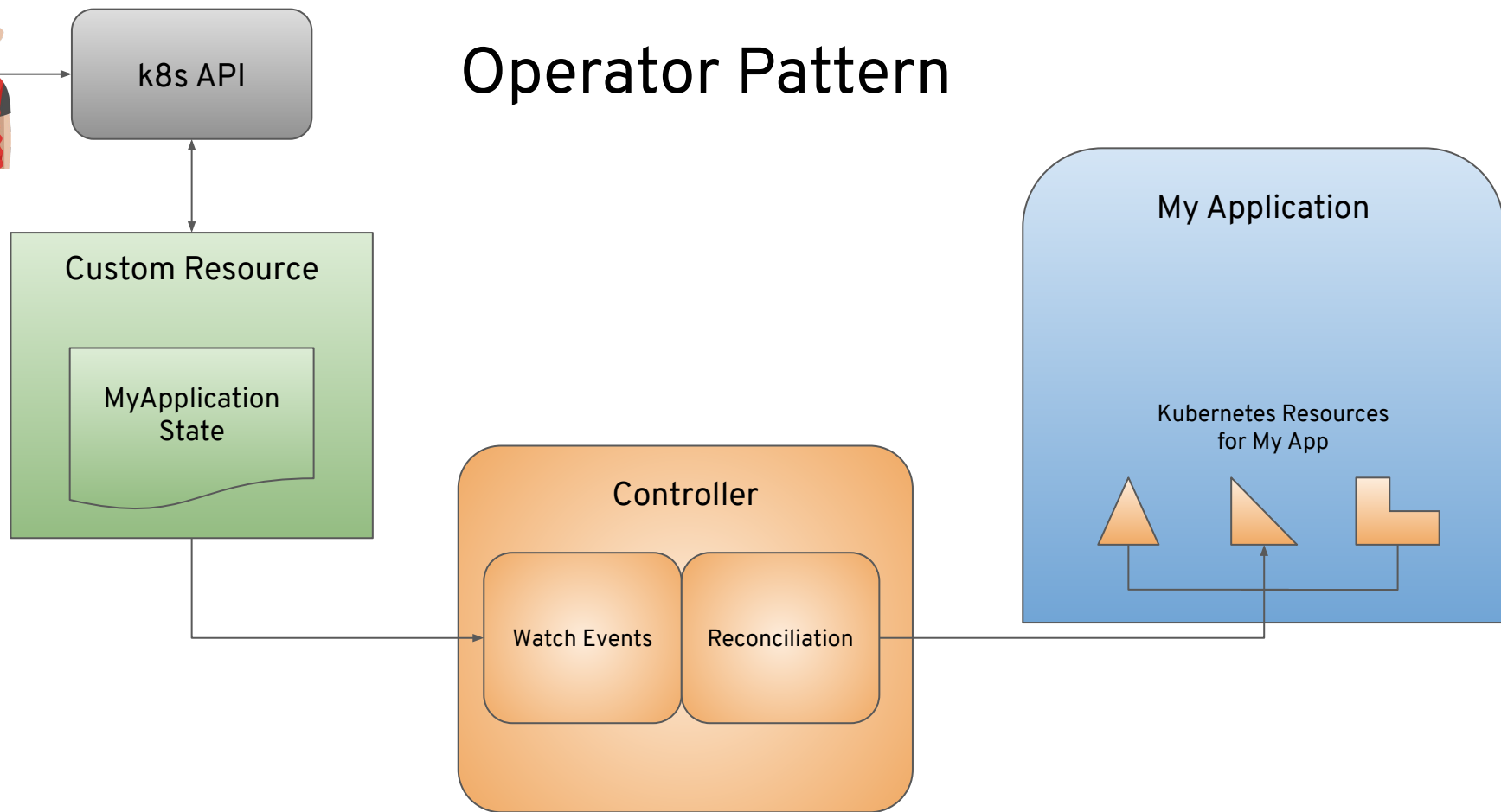


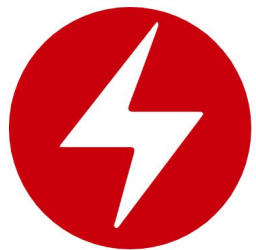
# Demo





# Operator Pattern





# OPERATOR FRAMEWORK

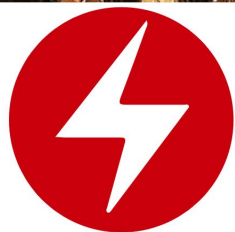
<https://github.com/operator-framework>



# OPERATOR LIFECYCLE MANAGER

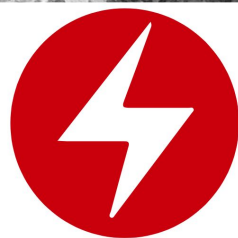






# OPERATOR METERING

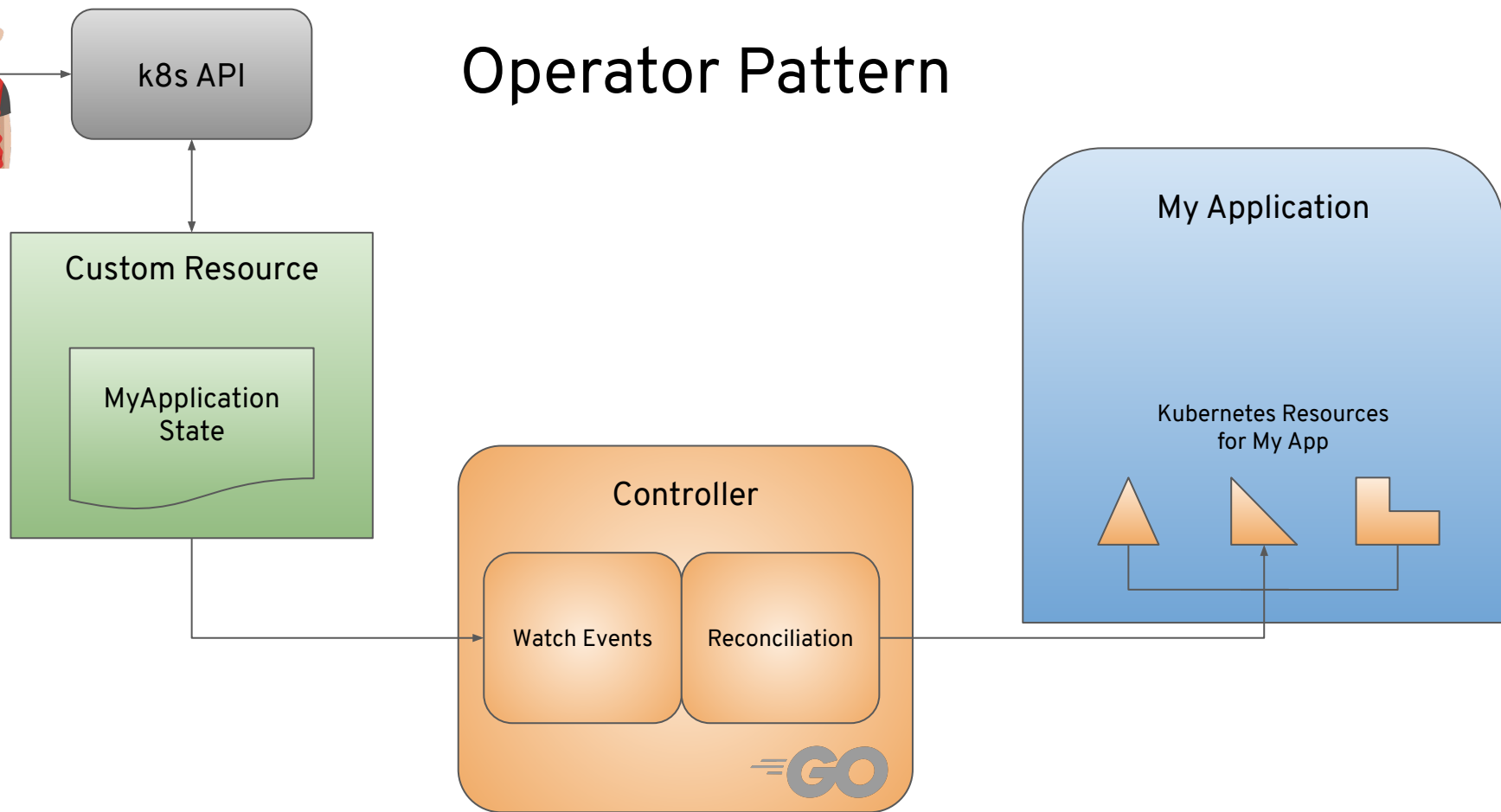





**OPERATOR  
SDK**



# Operator Pattern



You don't need to learn  to write  
an Operator!







ANSIBLE

# k8s YAML

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: foo
  namespace: default
data:
  color: red
```

# Ansible Task

```
---
- name: create foo configmap
  k8s:
    definition:
      apiVersion: v1
      kind: ConfigMap
      metadata:
        name: foo
        namespace: default
      data:
        color: "{{ color }}"
```

# k8s Module + Jinja2 Templates

```
---  
- name: create foo configmap  
  k8s:  
    definition: "{{ lookup('template', '/foo.yml') | from_yaml }}"
```



k8s API

# Ansible Operator

Custom Resource

MyApplication  
State

Ansible Operator

Operator-  
SDK



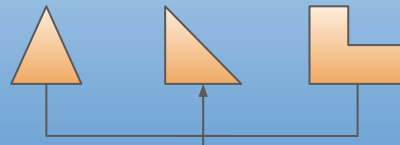
Ansible  
Role



File  
Mapping

My Application

Kubernetes Resources  
for My App



Ansible k8s modules will be  
used to create resources in  
kubernetes

# What you need to create an Ansible Operator

- A *CustomResourceDefinition* (CRD)
- An Ansible Playbook or Role
- A mapping from *CRD* to Ansible playbook / roles
- `operator-sdk`

# Create the Operator with the SDK

```
$ operator-sdk new memcached-operator \
  --api-version=cache.example.com/v1alpha1 \
  --kind=Memcached --type=ansible
```

Creates:

- Ansible Role
- Mapping File (watches.yaml)
- *Custom Resource Definition*
- Deploy manifest for the new Operator

# Custom Resource (CR)

```
apiVersion:
<Group/Version>
kind: <kind>
metadata:
  name: <name>
spec:
  <key>: <value>
  ...
status:
  <key>: <value>
  ...
```

## Ansible Operator

Spec values will be translated to Ansible extra vars.

Status will be a generic status defined by the operator. This will use ansible runner output to generate meaningful output for the user.

# Ansible Role

```
memcached/
├── defaults
│   └── main.yml
├── files
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── README.md
├── tasks
│   └── main.yml
├── templates
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    └── main.yml
```

Create a Role that deploys and manages your application



# Mapping between *CRDs* and Ansible

Maps a Group Version Kind (GVK) to a role or playbook.

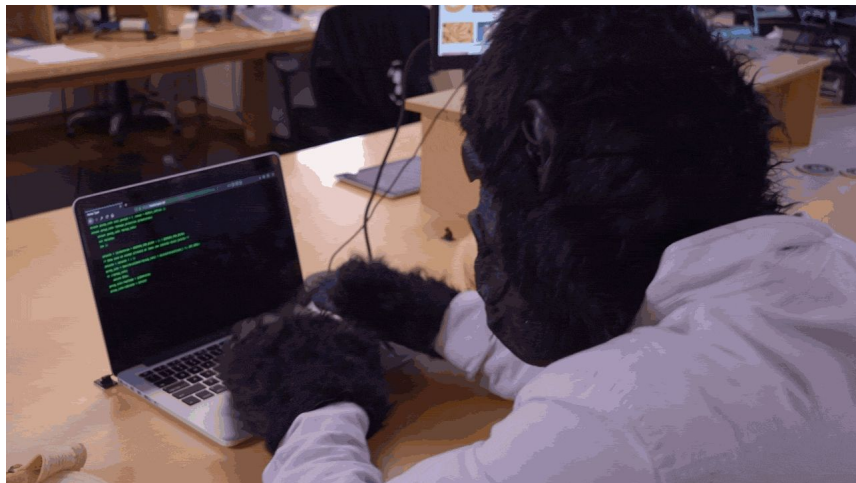
```
# watches.yaml
---
- version: v1alpha1
  - group: cache.example.com
    kind: Memcached
    playbook: /path/to/playbook
```

# Build the Operator with the SDK

```
$ operator-sdk build memcached-operator:v0.0.1
```

Creates:

- A Dockerfile that creates the Operator
- Builds the container on top of ansible-runner image



# Demo

# Try it yourself!

[learn.openshift.com/ansibleop](https://learn.openshift.com/ansibleop)

Ansible Refresher

START SCENARIO

Ansible Kubernetes  
Modules

START SCENARIO

Ansible Operator  
Overview

START SCENARIO

Operator SDK  
Playground

START SCENARIO

# Resource

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

<https://coreos.com/operators/>

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

<https://coreos.com/blog/introducing-operator-framework>

<https://learn.openshift.com/operatorframework/>



Danke!