What's New Within Operator Framework?

Jonathan Berkhahn, IBM
Varsha Prasad Narsing, Red Hat

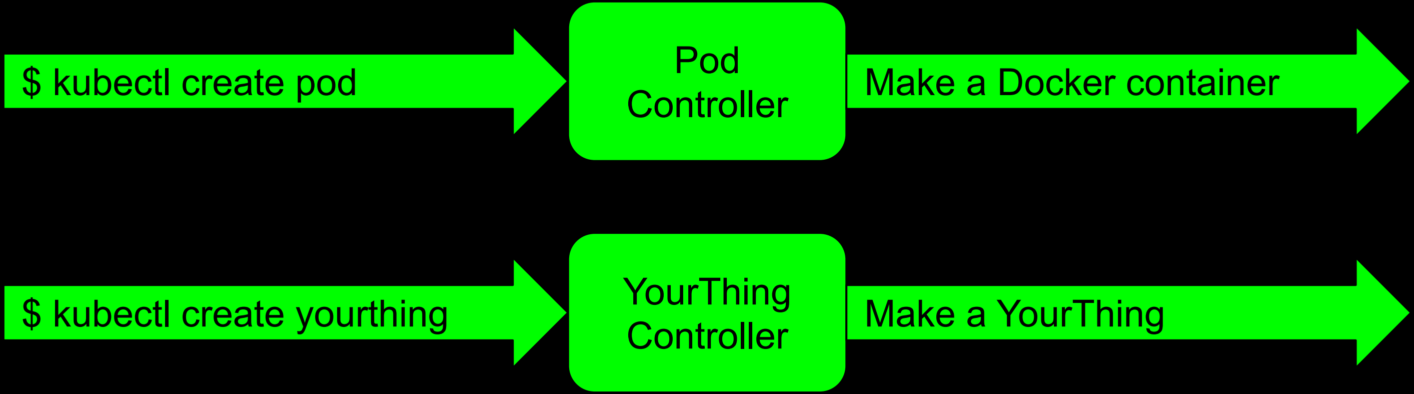
Speakers:

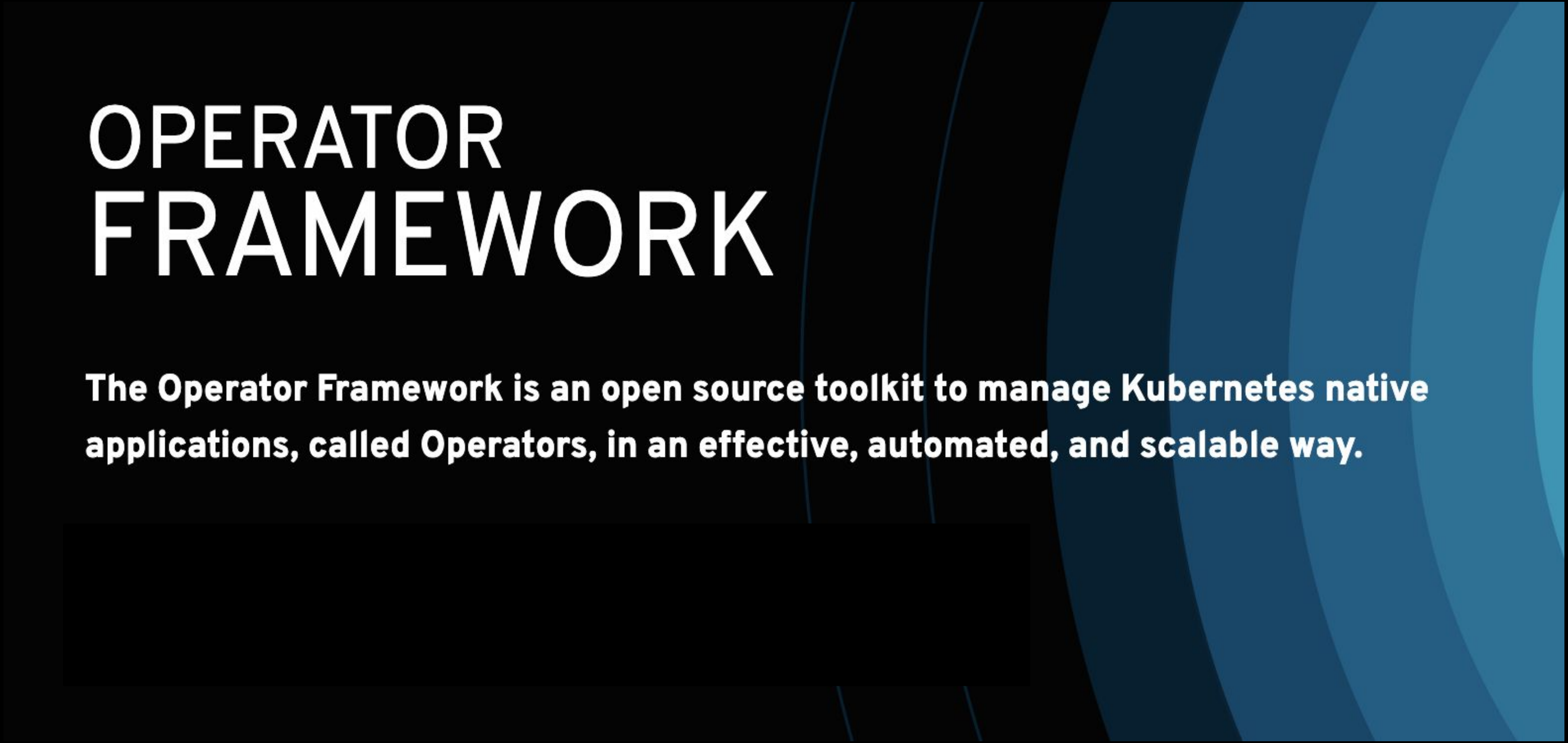
Who knows?

- Operators
- Operator Framework
- Operator SDK
- Operator Package Managers (OPM)
- Operator Lifecycle Manager (OLM)

What is an Operator

Operators are a design pattern that contain application specific logic that is usually handled by a human operator, hence the term “operator”.

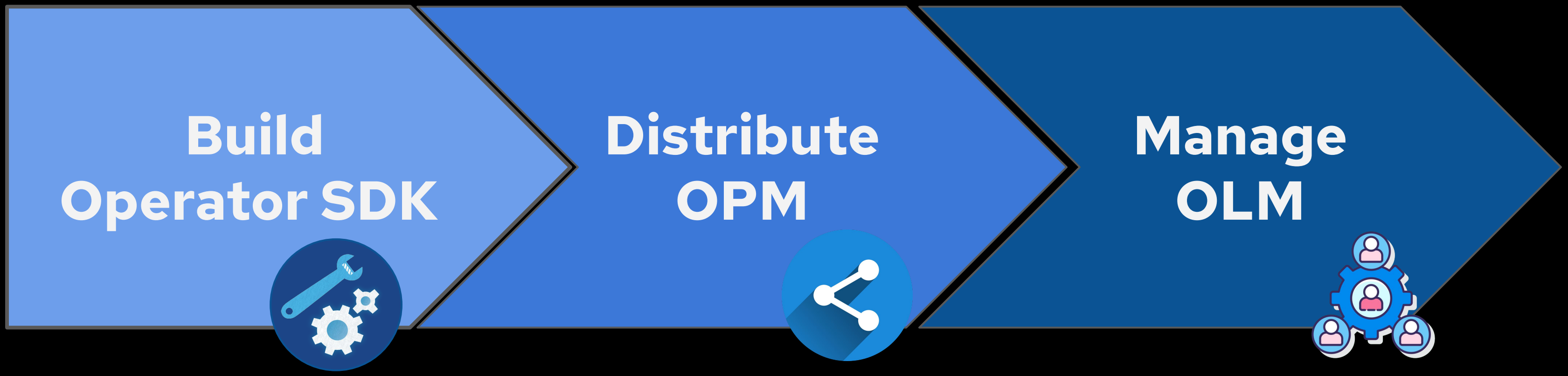




OPERATOR FRAMEWORK

The Operator Framework is an open source toolkit to manage Kubernetes native applications, called Operators, in an effective, automated, and scalable way.

What’s In The Framework



Build
Operator SDK

Distribute
OPM

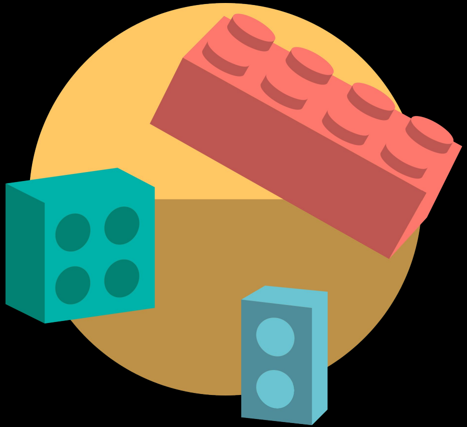
Manage
OLM

Java Operator SDK is joining Operator Framework!

Java Operator SDK is joining Operator Framework!

- Joining Operator Framework as an official subproject
- Project consists of:
  - Java framework for implementing controllers (similar to controller-runtime for Golang operators)
  - Operator SDK plugin for generating Java operators
  - Testing framework and support tooling
- Foster closer collaboration for better integration with OSDK, OLM

Build: Supporting External Plugins



- Plugins - responsible for implementing the code executed when KB sub-commands are called.
- Hoping to dogfood this by re-implementing core OSDK libraries (Go, Helm, Ansible) as external plugins
- Leverage in Kubebuilder as library to create your own plugins.
- Use the stack and language that you are familiar with.

Build: Want To Learn More?

Operator SDK: https://sdk.operatorframework.io/
Plugins:

- https://book.kubebuilder.io/plugins/creating-plugins.html
- https://sdk.operatorframework.io/docs/contribution-guidelines/plugins

Java Operator SDK: https://javaoperatorsdk.io/

Grafana Plugin: https://book.kubebuilder.io/plugins/grafana-v1-alpha.html

Deploy Image Plugin: https://book.kubebuilder.io/plugins/deploy-image-plugin-v1-alpha.html
Operator SDK Run Bundle: https://sdk.operatorframework.io/docs/cli/operator-sdk_run_bundle/
Operator SDK Validators: https://sdk.operatorframework.io/docs/cli/operator-sdk_bundle_validate/

Manage: Operator Lifecycle Manager (OLM)

OLM extends Kubernetes to provide a declarative way to install, manage, and upgrade Operators and their dependencies in a cluster.

What's New?

- OLM v1
- RukPak
- Catalogd
- Deppy

Manage: OLM V1

News:

- OLM is moving to a new set of APIs! (Coming soon!)

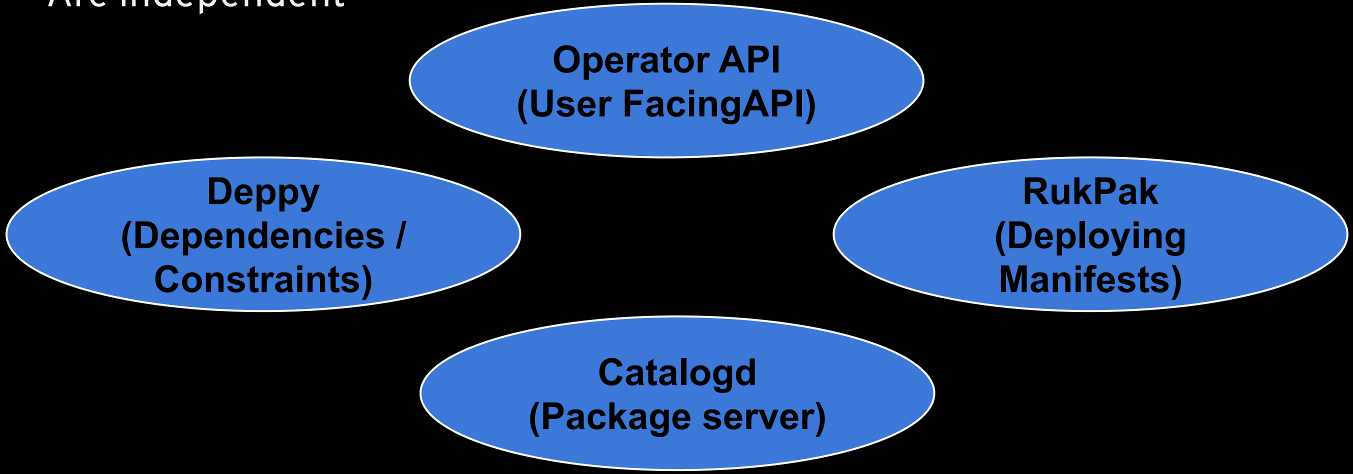
Why:

- Existing APIs fulfill too many purposes
    - It takes time to learn how to use OLM
- OLM was designed when CRDs were in their beta format
- Some design decisions that made sense at the time do not make sense today

Manage: OLM V1

What to Expect:

- OLM will consist of a set of focused/scoped components
- New controllers:
  - Will introduce tightly scoped APIs that can be customized or built upon
  - Are less opinionated
  - Are independent



Operator API
(User FacingAPI)

Deppy
(Dependencies / Constraints)

RukPak
(Deploying Manifests)

Catalogd
(Package server)

DEMO

Manage: RukPak - Concepts

Bundle CRD

- Defines manifests that must be installed

BundleDeployment CRD

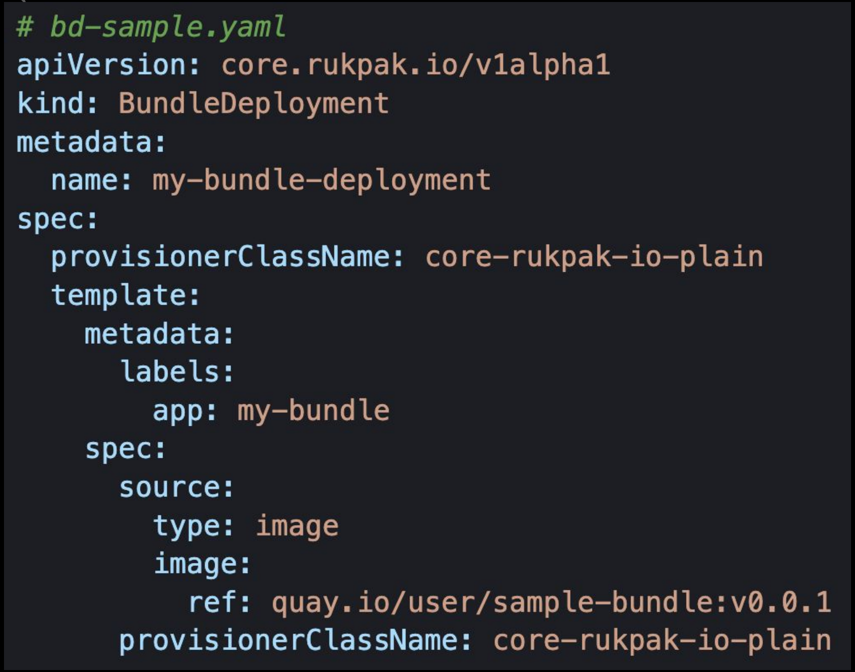
- Instantiation of a *Bundle* on a cluster

Provisioners

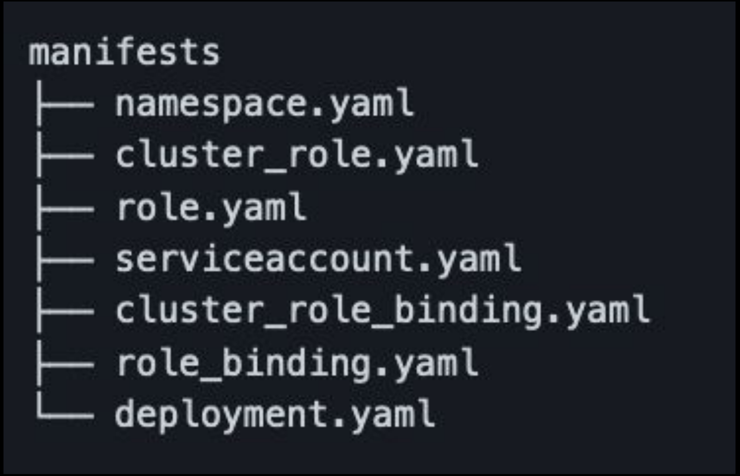
- Controllers aware of how to reconcile *Bundle* formats
- Provisioners will be responsible for:
    - Retrieving bundles from some storage medium
    - Installing the manifests associated with a bundle onto the cluster
- BYO Provisioner

DEMO

Manage: RukPak - Example



```yaml
# bd-sample.yaml
apiVersion: core.rukpak.io/v1alpha1
kind: BundleDeployment
metadata:
  name: my-bundle-deployment
spec:
  provisionerClassName: core-rukpak-io-plain
  template:
    metadata:
      labels:
        app: my-bundle
    spec:
      source:
        type: image
        image:
          ref: quay.io/user/sample-bundle:v0.0.1
      provisionerClassName: core-rukpak-io-plain
```



```
manifests
├── namespace.yaml
├── cluster_role.yaml
├── role.yaml
├── serviceaccount.yaml
├── cluster_role_binding.yaml
├── role_binding.yaml
└── deployment.yaml
```

Manage: Catalogd

A repository of Operator content and packages that define an application.

V1 design introduces:

- An aggregated server and dedicated etcd storage to serve FBC content.
- Cluster scoped APIs for exposing package and bundle level contents.

It is in early stages of development. We welcome contributions!

Manage: Deppy

Variable

- A literal in a boolean clause
- Has constraints (e.g. Mandatory, Prohibited, AtMost, Dependency)
- Used to represent bundles and complex constraints (e.g. at most 1 bundle / package)
- Composed to form a boolean statement that is fed to the SAT solver

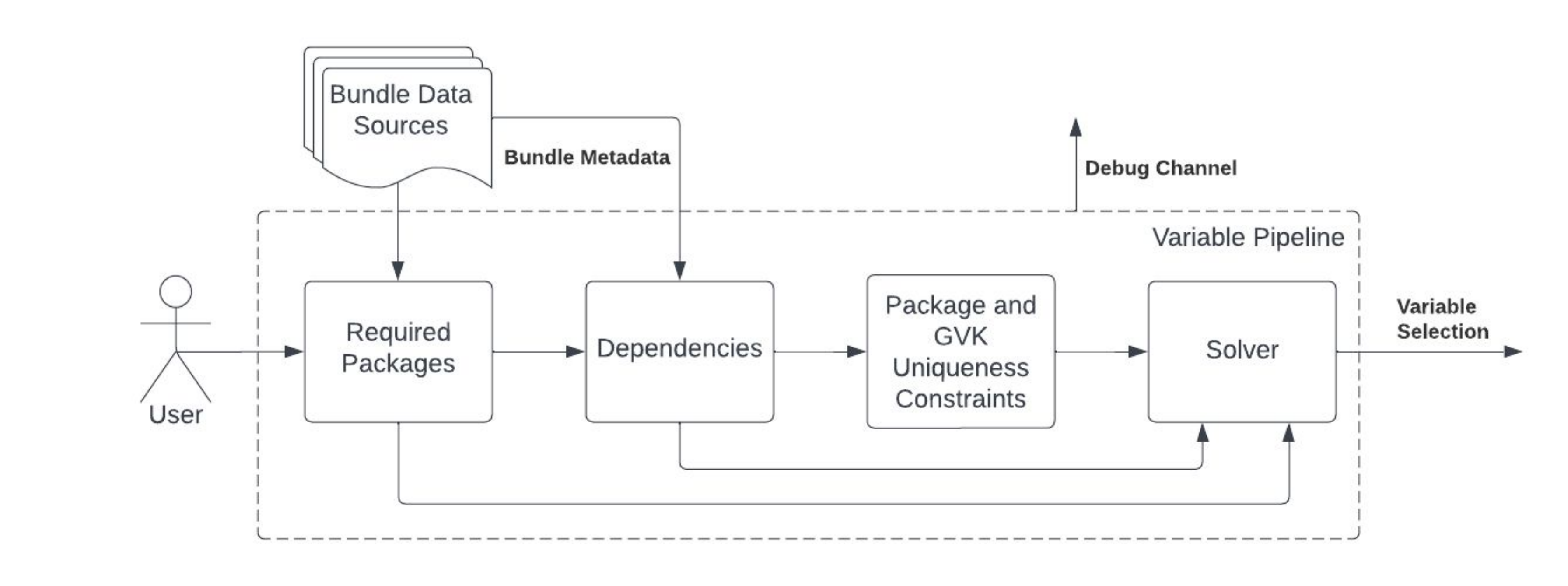
SAT Solver

- Is there an assignment (T or F) to each literal that satisfies the boolean statement (makes it evaluate to true)

Variable Sources

- A pipeline framework is used to produce the variables from domain specific data sources (e.g. Catalog Sources, the cluster)

Manage: Deppy - Extensible Variable Pipelines



DEMO

Manage: Operator Controller

Operator CR (work in progress)

- Defines the application you want to install (package, version, channel, etc.)

Reconciliation

- Employs Deppy to ensure operator requirements are met (fostering deployment stability)
- Relies on RukPak for installation and updates

Current State

- Initial stages, perfect for new contributors!
- We’re checking all previous assumptions
- We’re experimenting with ways to have an eventually consistent resolver
- Want to make big usability improvements over OLM v0

Manage: Join us! And together we can rule the galaxy!

RukPak:

- Github Repository: https://www.github.com/operator-framework/rukpak/
- K8s Slack Channel: #rukpak-dev | https://tinyurl.com/3dd2va3w

Deppy:

- Github Repository: https://github.com/operator-framework/deppy/
- K8s Slack Channel: #deppy-dev | https://tinyurl.com/4uwpy4wv

Operator Controller:

- Github Repository: https://github.com/operator-framework/operator-controller
- K8s Slack Channel: #olm-dev | https://tinyurl.com/yw75nkfy

Catalogd:

- Github Repository: https://github.com/operator-framework/catalogd

Please provide us your feedback :)

Any Questions?