

# Operator Lifecycle Manager

# Late 2017...

[← Back to All Blogs](#)

## Tectonic 1.8 and the Open Cloud Services Catalog are now available

December 19, 2017 • By Neil McAllister

Tags: [announcements](#) [Tectonic](#) [Open Cloud Services](#) [Kubernetes](#)



Today we're proud to announce Tectonic 1.8, the latest release of our enterprise-ready Kubernetes platform, is now available. This version, which we announced ahead of [KubeCon + CloudNativeCon North America](#) earlier this month, is the first to ship with a catalog of industry-first Open Cloud Services that enable users to deploy key infrastructure components with the ease of managed cloud vendor offerings but without cloud vendor lock-in.

The first three offerings in the CoreOS Tectonic Open Cloud Service Catalog are based on etcd, Prometheus, and Vault. You can learn more about what each of these services delivers in [our previous blog post](#), and we'll dive into each of them in separate posts soon. What all Open Cloud Services share in common, though, is that they automate key operational tasks such as security updates, scaling and handling failover, making backups, and so on, allowing users to concentrate their focus on managing their applications and not the infrastructure they run on.

These powerful services are the latest building blocks in the complete Tectonic platform, which combines pure, upstream open source Kubernetes with features like authentication and authorization, monitoring and alerting, and ease of installation and management. For more information on what's new with Tectonic 1.8, [check out the release notes](#).

# Tectonic 1.8 Dashboard with OCS

The screenshot displays the Tectonic 1.8 Dashboard interface. On the left is a dark blue sidebar with the Tectonic logo and a list of navigation items: Applications, Available Applications, Open Cloud Catalog, Workloads, Daemon Sets, Deployments, Replica Sets, Replication Controllers, Persistent Volume Claims, Stateful Sets, Jobs, Pods, Config Maps, Secrets, Resource Quotas, Open Cloud Services, Routing, and Troubleshooting. The main content area is divided into two sections. The top section, 'Open Cloud Services', shows a table with one entry: 'Open Cloud Services' provided by 'CoreOS, Inc'. The bottom section, 'Applications', features a search bar and a table of installed applications. The table lists 'etcd', 'Prometheus', and 'Vault', all of which are 'Enabled (1 namespace)' and include links to 'Show namespace'. Each application entry also has an 'Enable' button.

**Tectonic**  
by CoreOS

Applications

Available Applications

Open Cloud Catalog

Workloads

Daemon Sets

Deployments

Replica Sets

Replication Controllers

Persistent Volume Claims

Stateful Sets

Jobs

Pods

Config Maps

Secrets

Resource Quotas

Open Cloud Services

Routing




Troubleshooting

### Open Cloud Services

NAME	PROVIDER
Open Cloud Services	CoreOS, Inc

### Applications

Filter Applications by name...

NAME ↑	STATUS
 <b>etcd</b> 0.6.1 by CoreOS, Inc	Enabled (1 namespace) <a href="#">Show namespace</a>
 <b>Prometheus</b> 0.14.0 by CoreOS, Inc	Enabled (1 namespace) <a href="#">Show namespace</a>
 <b>Vault</b> 0.1.3 by CoreOS, Inc	Enabled (1 namespace) <a href="#">Show namespace</a>

# What is the Operator Lifecycle Manager (OLM)?

OLM is the..

**Installation** (`InstallPlan`)

**Definition** (`ClusterServiceVersion`)

**Resolution** (`CatalogSource`)

**Upgrading** (`Subscription/Package`)

of....

# Operators!

Okay..but what **IS** it?

# OLM...

- Provides an App-Store like experience for discovering and installing Operators.
- Automated upgrades for Operators.
- Provides a framework building rich, reusable UIs.
- Package management and dependency resolution.



That's all good..but seriously...

What IS IT?

An Operator.

# But not just one Operator.


**OLM-Operator**

## Three Operators.

**OLM-Operator**

**Catalog-Operator**

**Package-Server**



Before we dive into the details, let's take a couple of steps back and visualize our environment in the context of OLM.



Admin

# OLM OpenShift Environment

accounting

marketing



User



openshift-operator-lifecycle-manager

sourceType: External

CatalogSource1  
Publisher: Google

- etcd.package.yaml
- etcdbackup.crd.yaml
- etcdcluster.crd.yaml
- etcdoperator.clusterserviceversi...
- etcdoperator.v0.9.0.clusterservi...
- etcdoperator.v0.9.2.clusterservi...
- etcdrestore.crd.yaml

OLM-Operator

olm-operator-serviceaccount

Catalog-Operator

olm-operator-serviceaccount

sourceType: Internal

CatalogSource2  
Publisher: Red Hat

- etcd.package.yaml
- etcdbackup.crd.yaml
- etcdcluster.crd.yaml
- etcdoperator.clusterserviceversi...
- etcdoperator.v0.9.0.clusterservi...
- etcdoperator.v0.9.2.clusterservi...
- etcdrestore.crd.yaml

Package-Server

olm-operator-serviceaccount

ConfigMap

CRDs

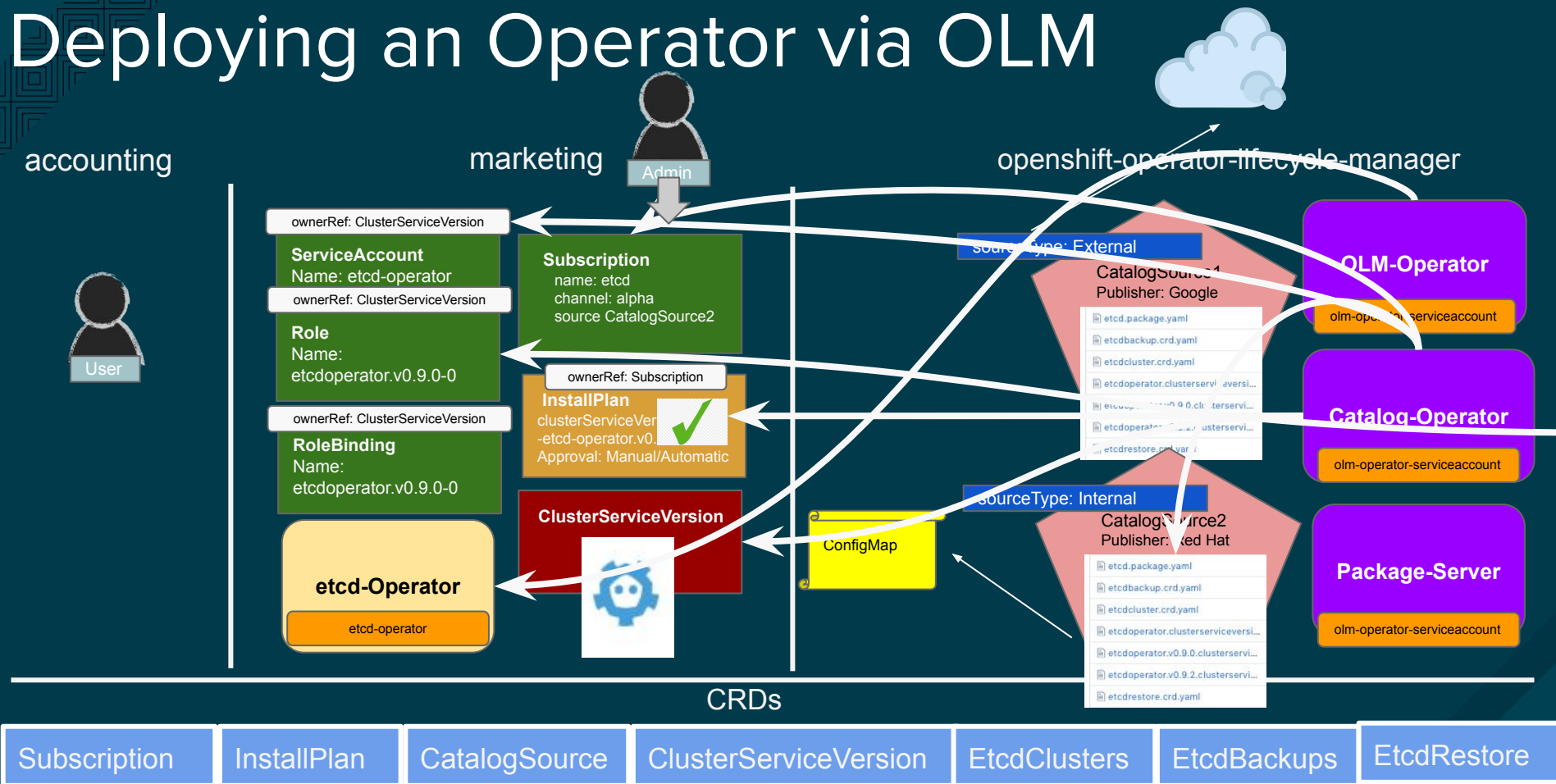
Subscription

InstallPlan

CatalogSource

ClusterServiceVersion

# Deploying an Operator via OLM

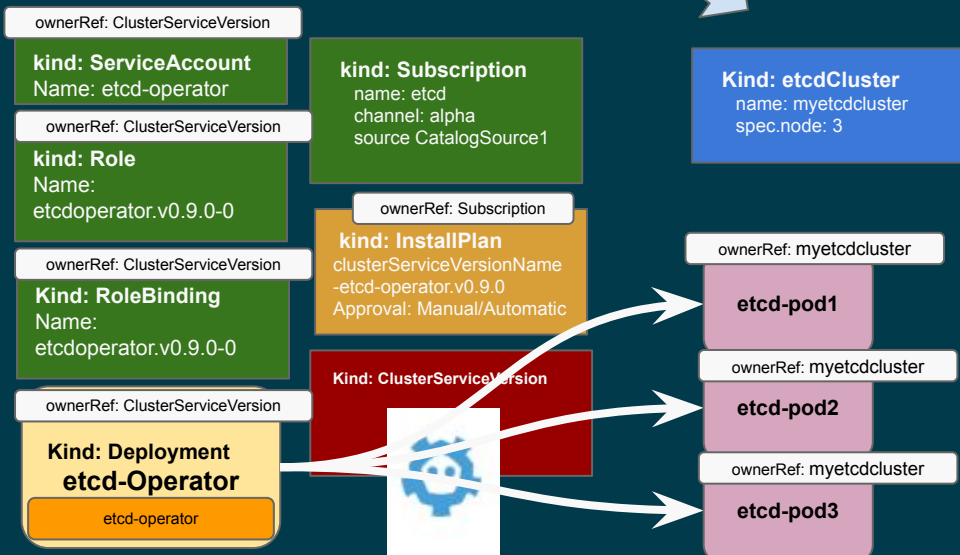


# User Creates CR

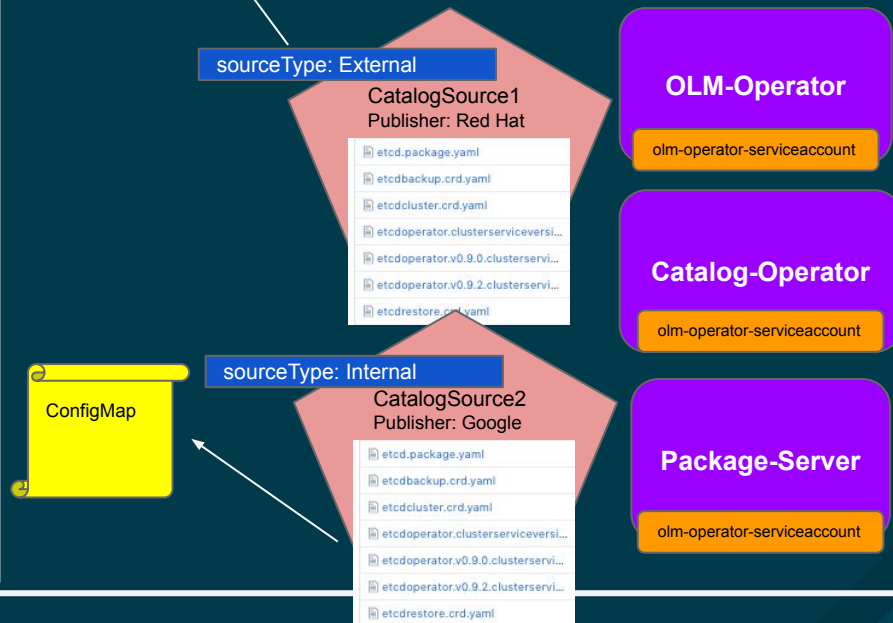
marketing



User



openshift-operator-lifecycle-manager



CRDs

Subscription

InstallPlan

CatalogSource

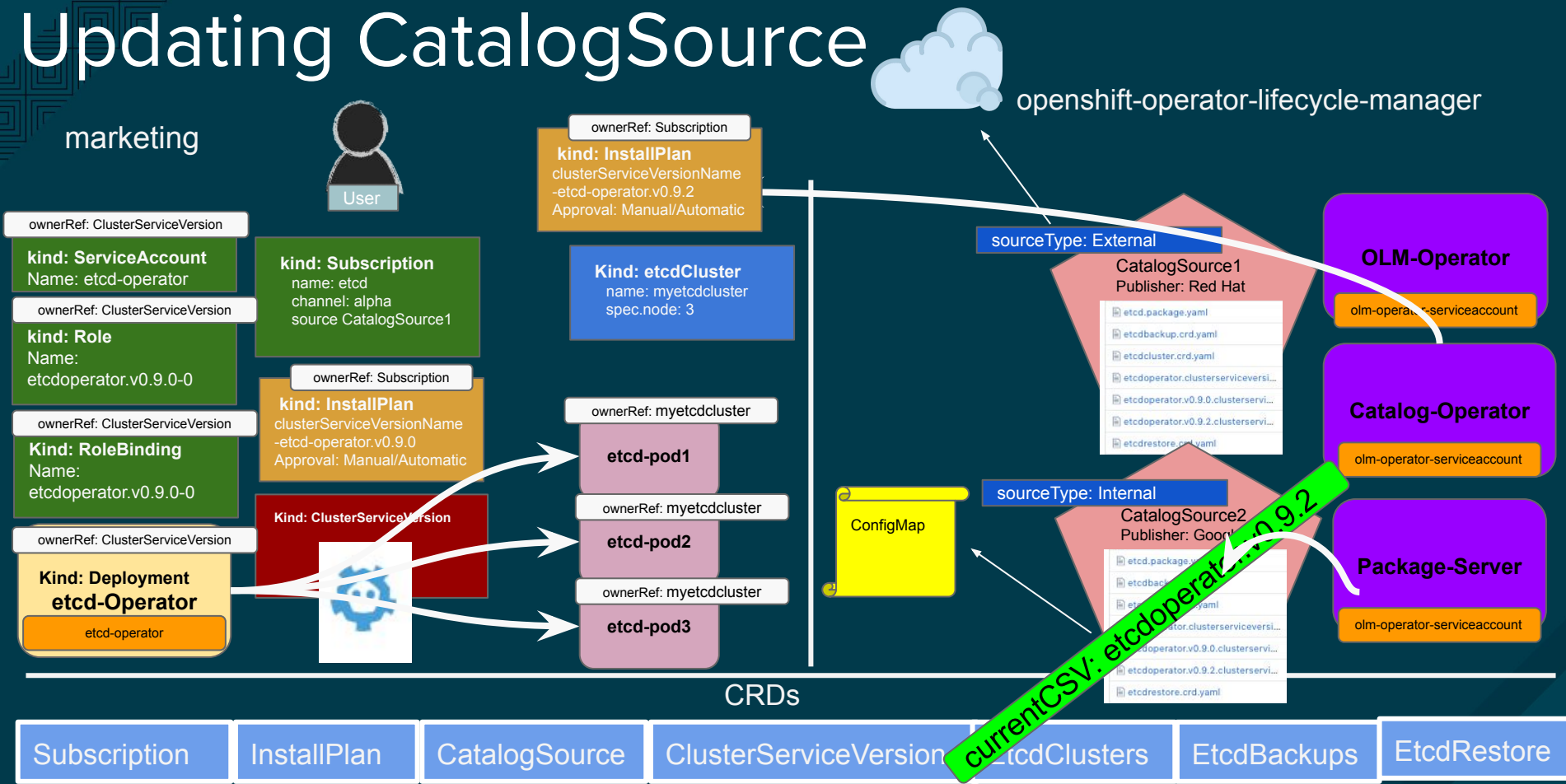
ClusterServiceVersion

EtcdClusters

EtcdBackups

EtcdRestore

# Updating CatalogSource





# ClusterServiceVersion is Analogous to a Package

ClusterServiceVersion

=

=

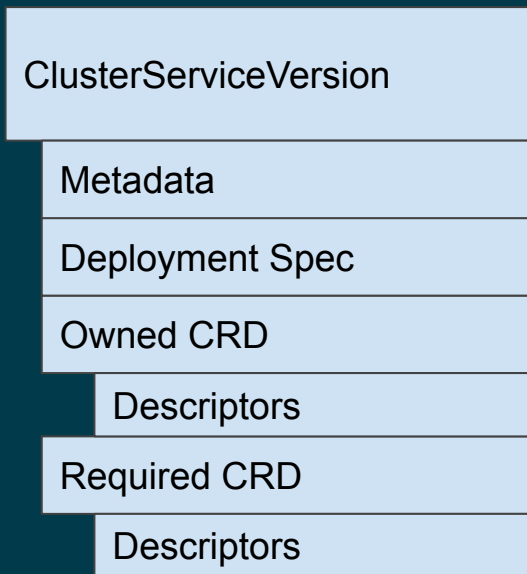
deb

rpm

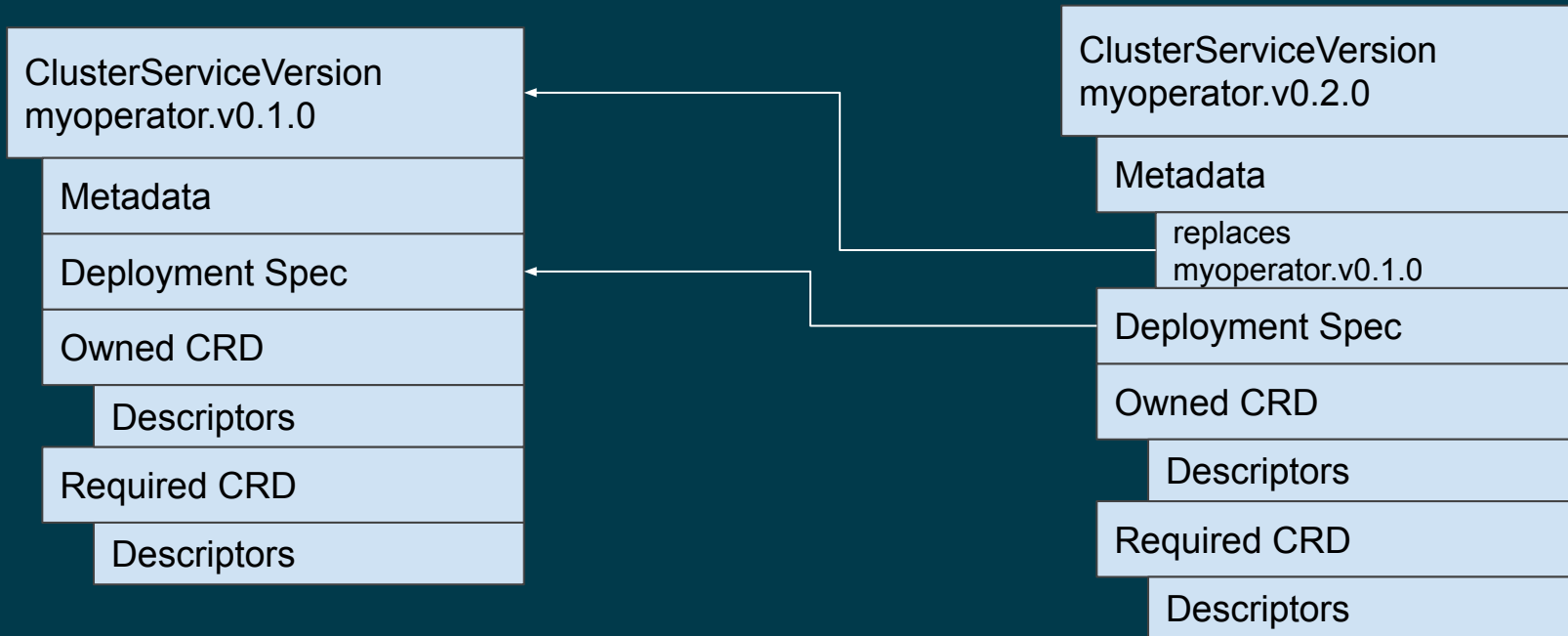
setup.py

formula

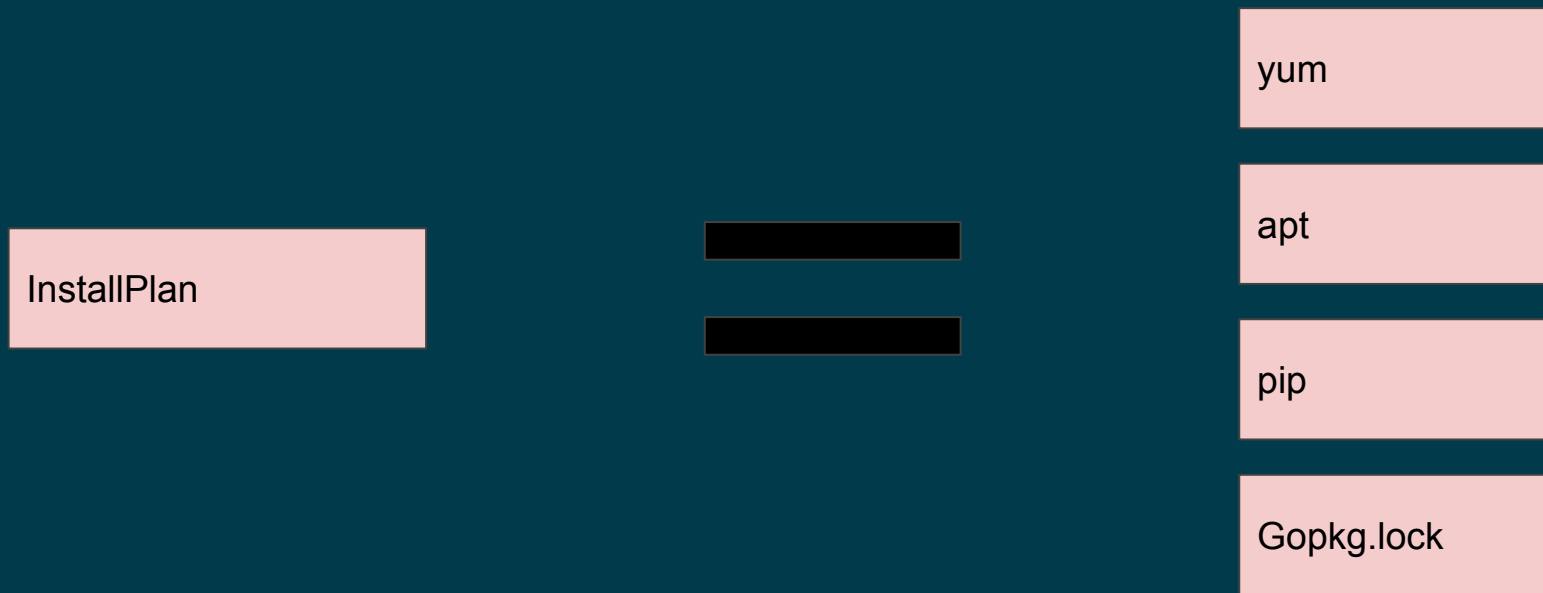
# ClusterServiceVersion Manifest



# Updating to a New CSV



# InstallPlan is similar to a Package Manager



# InstallPlan Manifest

InstallPlan

Desired CSVs

Status

Resolved CSVs

Resolved CRDs

# InstallPlan Manifest

InstallPlan

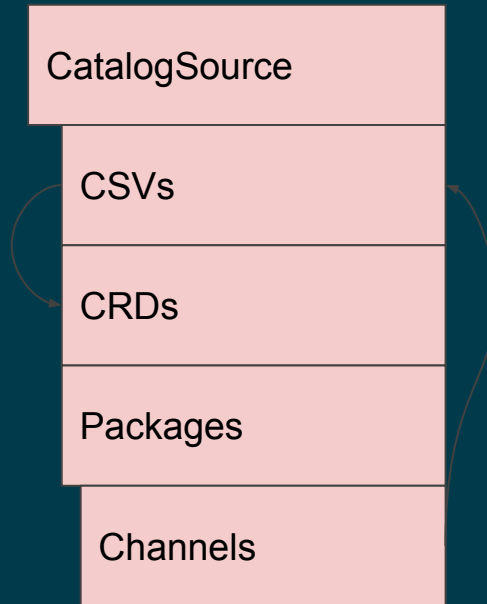
Desired CSVs

Status

Resolved CSVs

Resolved CRDs

# Catalog Source Manifest



# Subscription Manifest

Subscriptions

Package

CatalogSource



# Dependency Management!

