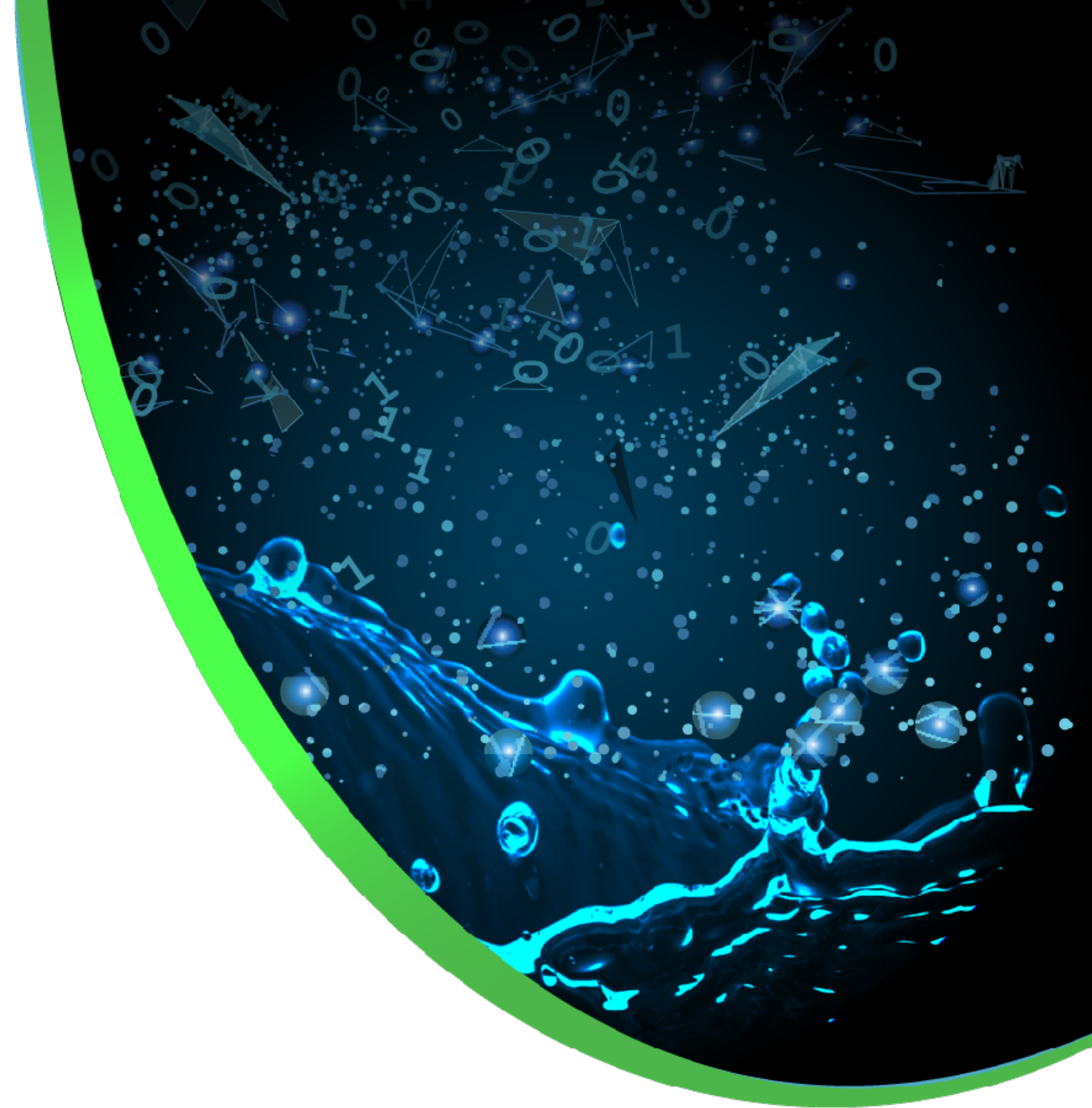


# Managing artifacts cross multiple Kubernetes clusters with JFrog Artifactory and Rancher

Alena Prokharchyk, Principal Software  
Engineer @ Rancher Labs

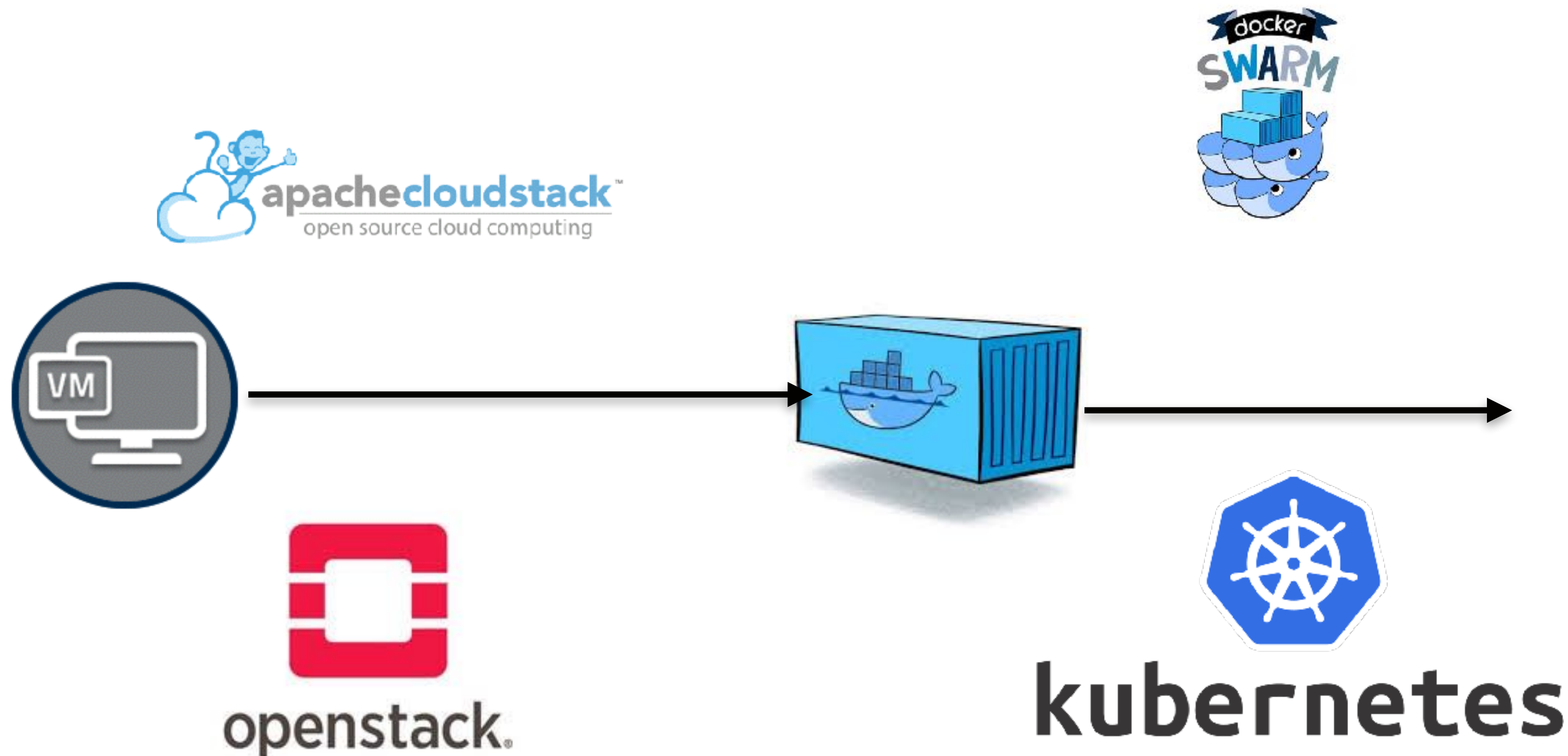




What was your favorite subject at school? Mine was history



# Data Center infrastructure evolution timeline



The data center of the future



# Containers started shaping infrastructure space back in 2013

Containers are

- Lightweight and portable
- An easy way to pack, ship, and run software
- Provide secure isolation for user applications

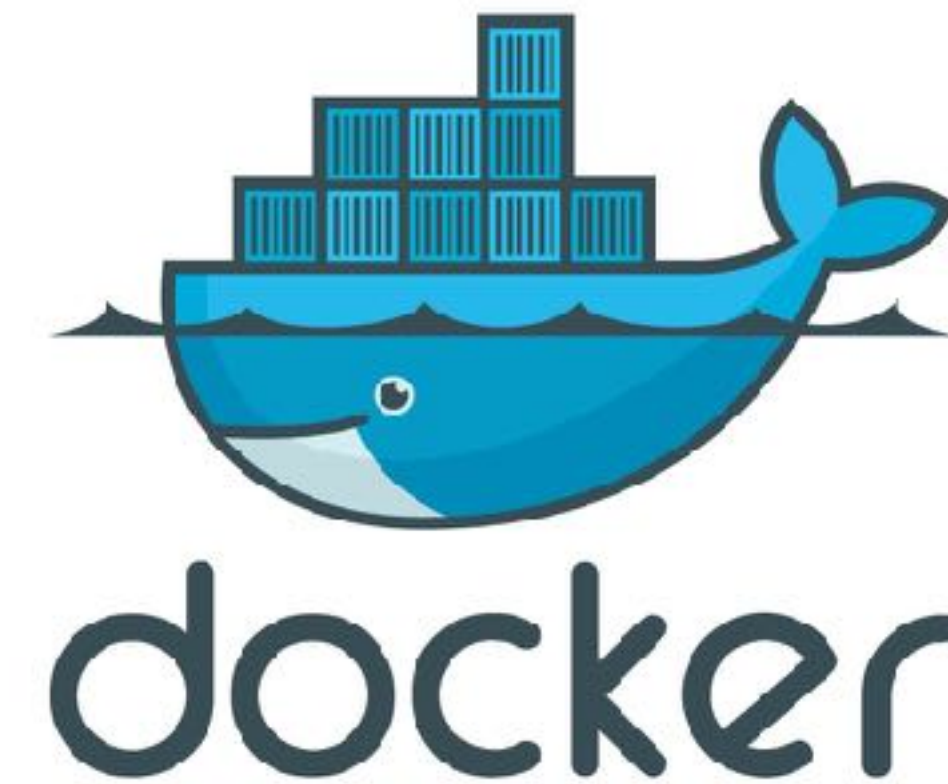


Image = Artifact = Essential piece of a docker container





# Managing artifacts - easy with JFrog

- Secure private registry with fine-grained access control
- Multiple docker registries support
- Images caching helps to reduce network overhead
- Advanced search
- Promoting Images to production



# Deploying artifacts...not so easy

# You start with deploying an image on a single host

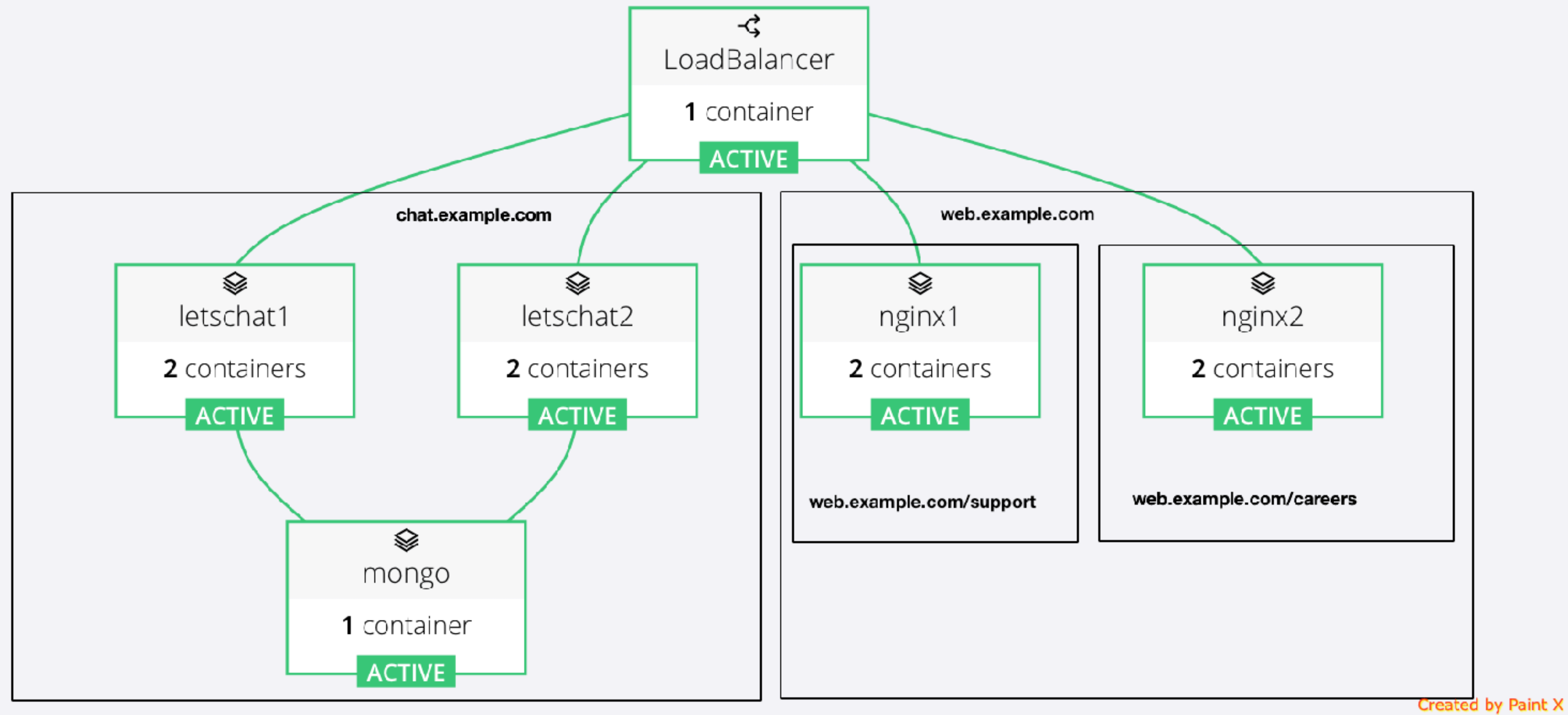
can be a developer laptop

or VM

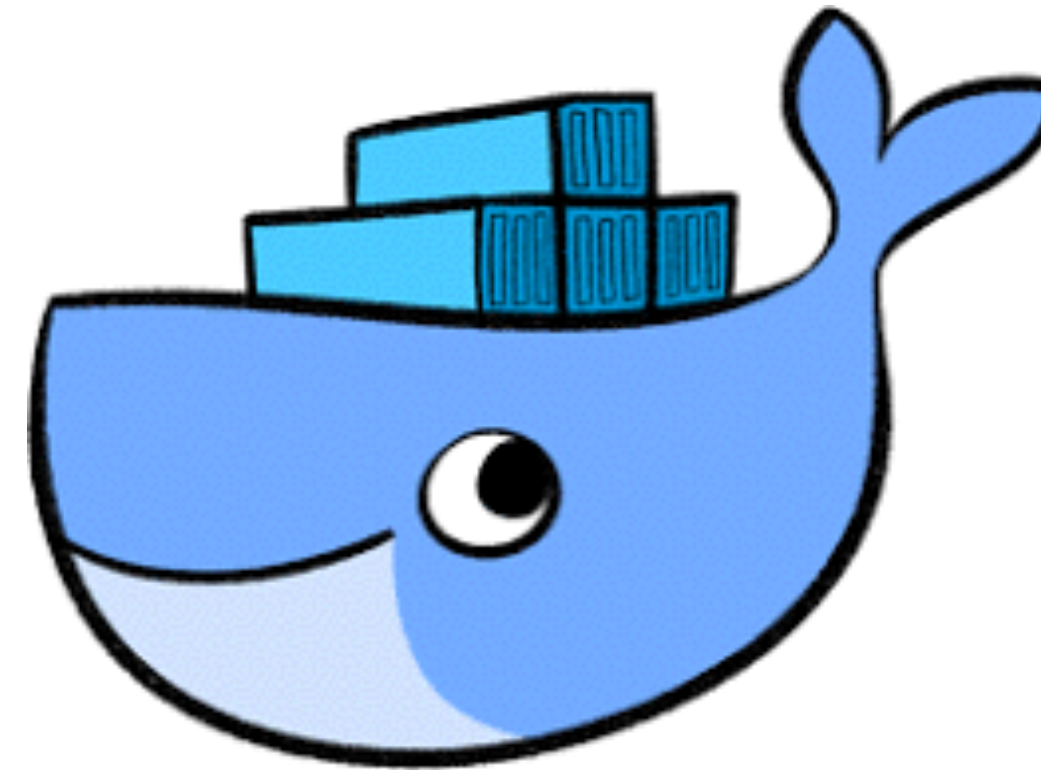




# But then realize your application is more complex



# Also the app has to be scheduled on multiple hosts



# You need an orchestrator



- To manage application scheduling
- To configure and monitor application health
- To make application upgrades simple





# kubernetes

proved to be the most popular orchestrator

# But nobody said Kubernetes is easy

## Kubernetes The Hard Way

- <https://github.com/kelseyhightower/kubernetes-the-hard-way>
- A Github repository created by Google engineer Kelsey Hightower ([@kelseyhightower](#))
- Manually configure the cluster step-by-step
- The tutorial is based on Google Cloud Engine
- The best way to learn Kubernetes
  - Showing the distributed nature of a Kubernetes cluster
  - HA for master nodes
  - Authentication method by bootstrap tokens and certificates/keys

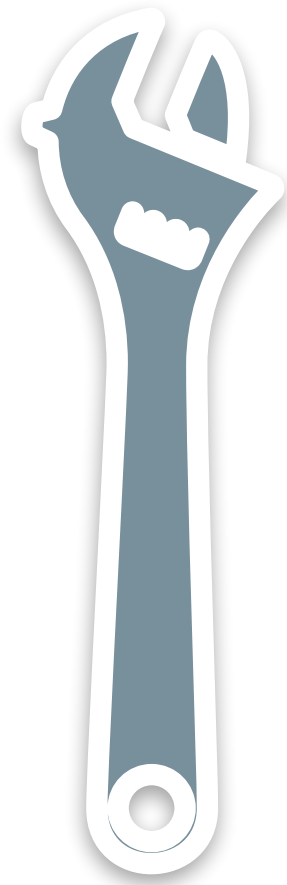
# Orchestrating orchestrator was challenging

- Installation was hard
- Upgrade was even harder
- Setup requirements varied depending on a cloud

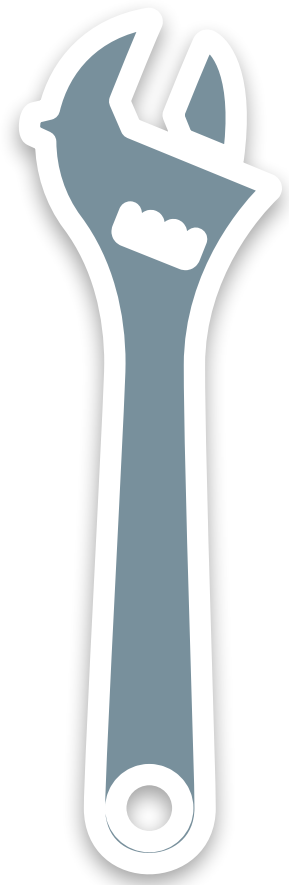


# But it got better

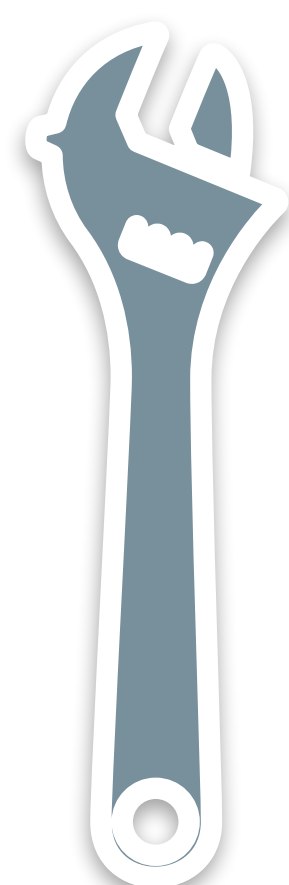
A lot of Kubernetes installation tools emerged to address the challenge



**kops**

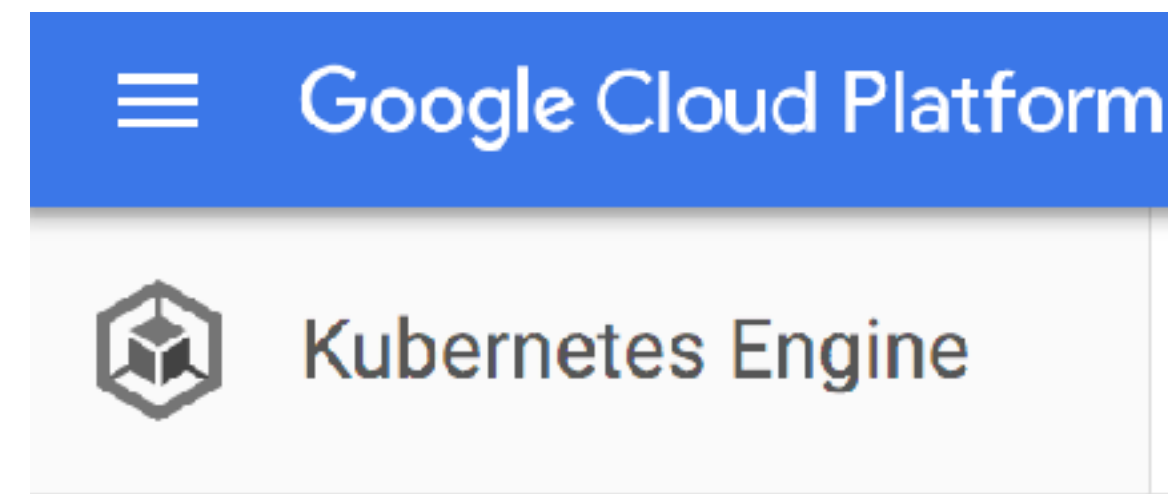


**kubespray**

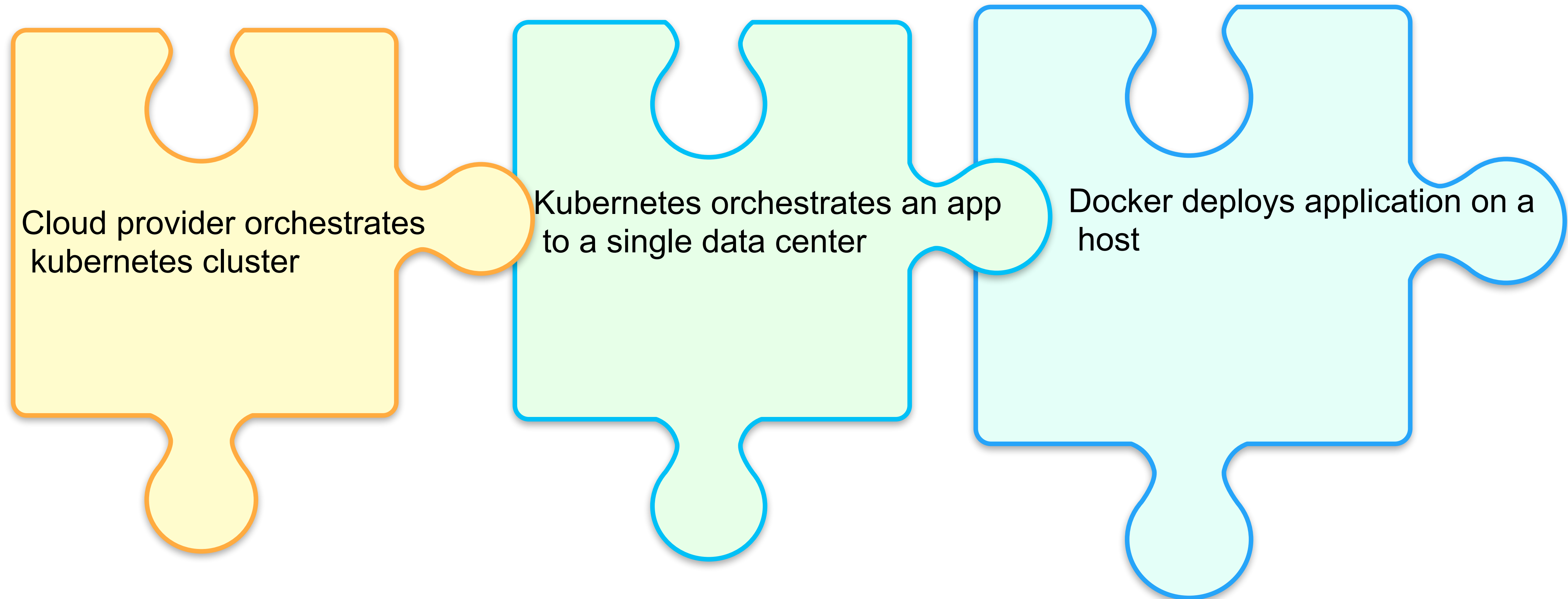


**kubeadm**

Kubernetes as a public offering was made available on several clouds

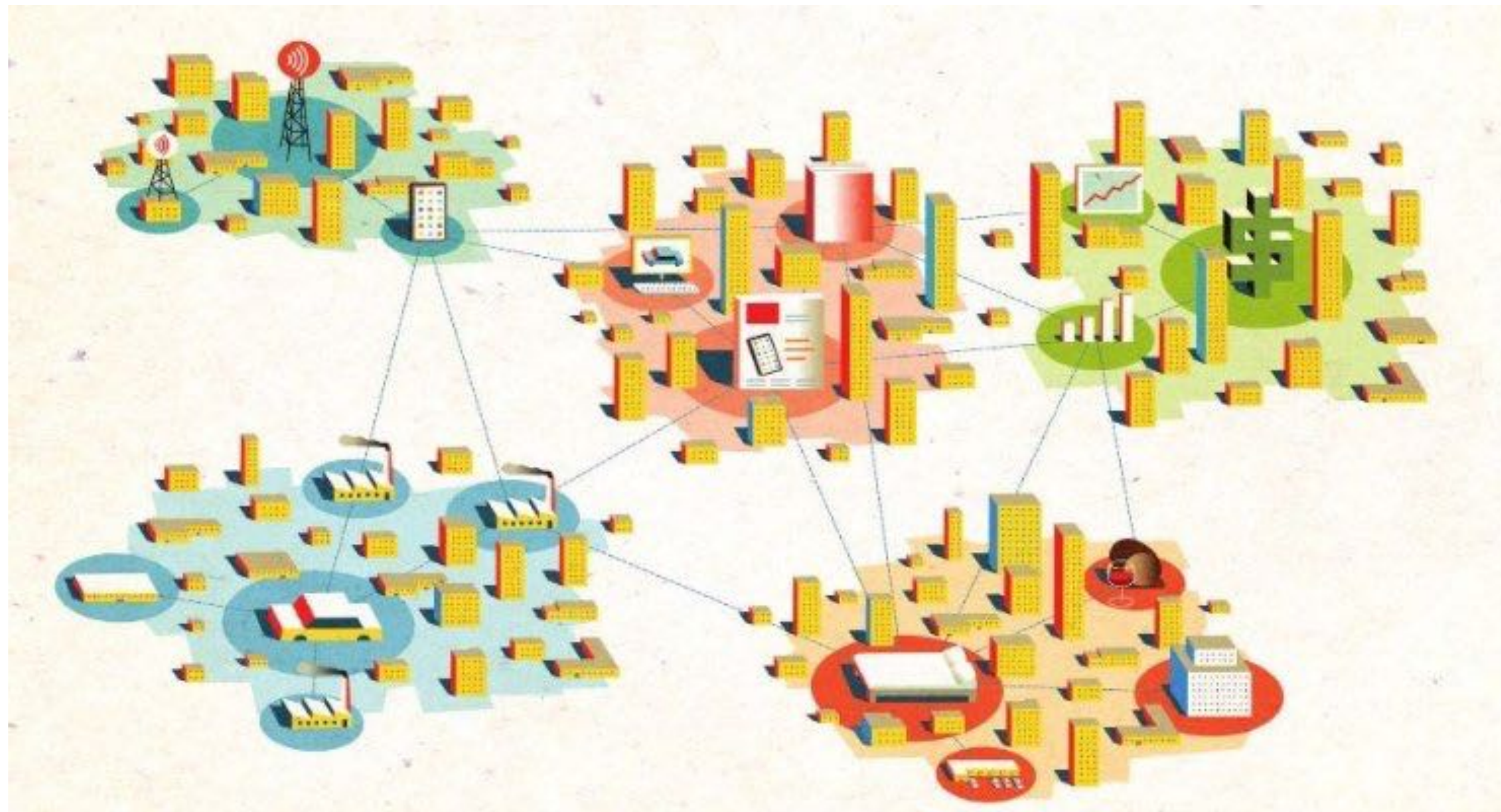


# Is the puzzle complete?





# You might need more than 1 Kubernetes cluster to deploy your app





# Use cases

- You have multiple data centers, multiple clouds or multiple regions
- Your team might spread across Kubernetes clusters
- You want to maintain different environments - developer, QA - each in its own dedicated cluster

- You need multi K8s installer/operator/manager
- Centralized auth and security policies is a requirement
- Secure built-in solution for artifactory cross clusters is highly desired
- And it should all look easy









**COMPLETE  
CONTAINER  
MANAGEMENT  
PLATFORM**

## User Experience

User Interface • App Catalog • Docker run • kubectl • API • CLI

## Multi-cluster Management

Provisioning • Upgrades • RBAC • Monitoring • Health Checks • Backup

### *Kubernetes Cluster*



### *Kubernetes Cluster*



### *GKE Cluster*



## Rancher Kubernetes Distribution

Storage Drivers • Container Networking • Infrastructure Management • Kubernetes Master • etcd

## Imported Clusters

GKE • ACS •  
Build Your Own

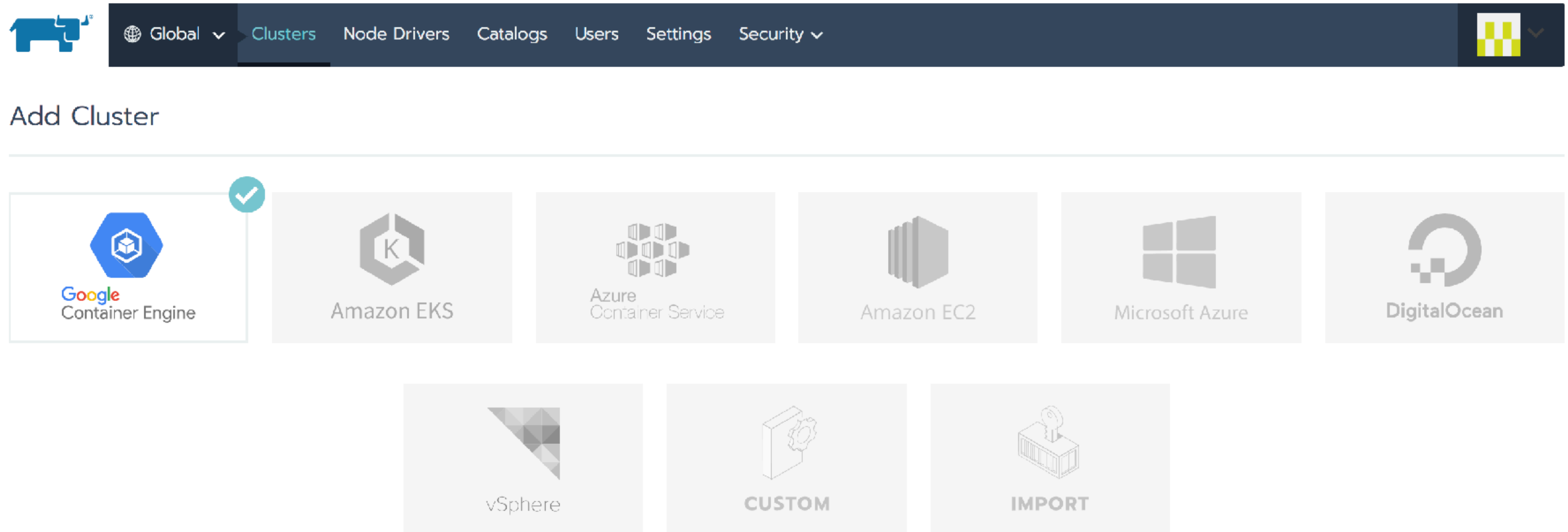
## Multi-cloud Management



# Install Rancher with just one click

```
$ sudo docker run -d --restart=unless-stopped -p 80:80 -p  
443:443 rancher/rancher
```

# Start with Kubernetes cluster provisioning



The screenshot shows the JFrog Kubernetes cluster provisioning interface. At the top, there is a navigation bar with a blue elephant icon, a 'Global' dropdown, and links for 'Clusters', 'Node Drivers', 'Catalogs', 'Users', 'Settings', and 'Security'. The 'Clusters' link is highlighted. Below the navigation bar, the text 'Add Cluster' is displayed. A horizontal line separates the header from the cluster options. There are eight cluster options, each in a light gray box with a logo and name. The first option, 'Google Container Engine', is highlighted with a blue border and a green checkmark in the top right corner. The other options are 'Amazon EKS', 'Azure Container Service', 'Amazon EC2', 'Microsoft Azure', 'DigitalOcean', 'vSphere', 'CUSTOM', and 'IMPORT'.

Global ▾ Clusters Node Drivers Catalogs Users Settings Security ▾

Add Cluster

Google Container Engine

Amazon EKS

Azure Container Service

Amazon EC2

Microsoft Azure

DigitalOcean

vSphere

CUSTOM

IMPORT



# Provide cross clusters proxy auth and RBAC

Active Directory

Azure AD

GITHUB

LOCAL

OpenLDAP

Global

Clusters

Node Drivers

Catalogs

Users

Settings

Security

Roles

Pod Security Policies

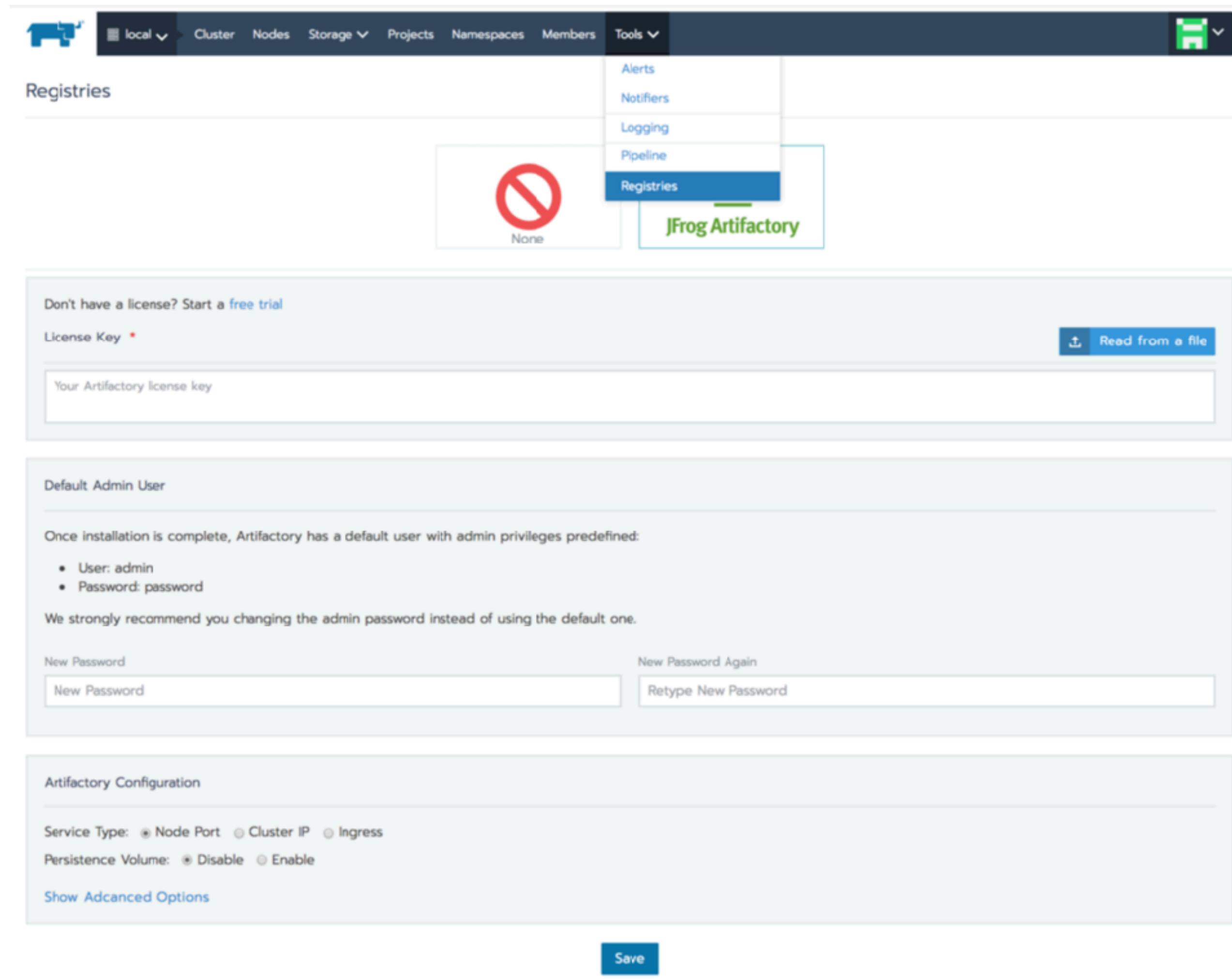
Authentication

Add Role

Search

Name	Built-In	Created
Cluster Member	✓	4:57 PM
Cluster Owner	✓	4:57 PM
Create Namespaces	✓	4:57 PM
Create Projects	✓	4:57 PM
Manage Cluster Members	✓	4:57 PM
Manage Config Maps	✓	4:57 PM

# Enable JFrog artifactory cross clusters



The screenshot shows the JFrog Artifactory configuration interface. At the top, there is a navigation bar with tabs: local, Cluster, Nodes, Storage, Projects, Namespaces, Members, and Tools. The Tools tab is selected, and a dropdown menu is open showing options: Alerts, Notifiers, Logging, Pipeline, Registries, and JFrog Artifactory. The Registries option is highlighted. Below the navigation bar, the main content area is titled 'Registries'. It features a red prohibition sign over a box labeled 'None', indicating that no registry is currently configured. Below this, there is a section for 'License Key' with a text input field and a 'Read from a file' button. The next section is 'Default Admin User', which provides information about the default user (admin) and password (password), and includes fields for 'New Password' and 'New Password Again'. The final section is 'Artifactory Configuration', which includes radio buttons for 'Service Type' (Node Port, Cluster IP, Ingress) and 'Persistence Volume' (Disable, Enable). A 'Show Advanced Options' link is also present. A 'Save' button is located at the bottom of the configuration section.

local Cluster Nodes Storage Projects Namespaces Members Tools

Registries

None

Alerts  
Notifiers  
Logging  
Pipeline  
Registries  
JFrog Artifactory

Don't have a license? Start a [free trial](#)

License Key \*

[Read from a file](#)

Your Artifactory license key

Default Admin User

Once installation is complete, Artifactory has a default user with admin privileges predefined:

- User: admin
- Password: password

We strongly recommend you changing the admin password instead of using the default one.

New Password

New Password Again

New Password Retype New Password

Artifactory Configuration

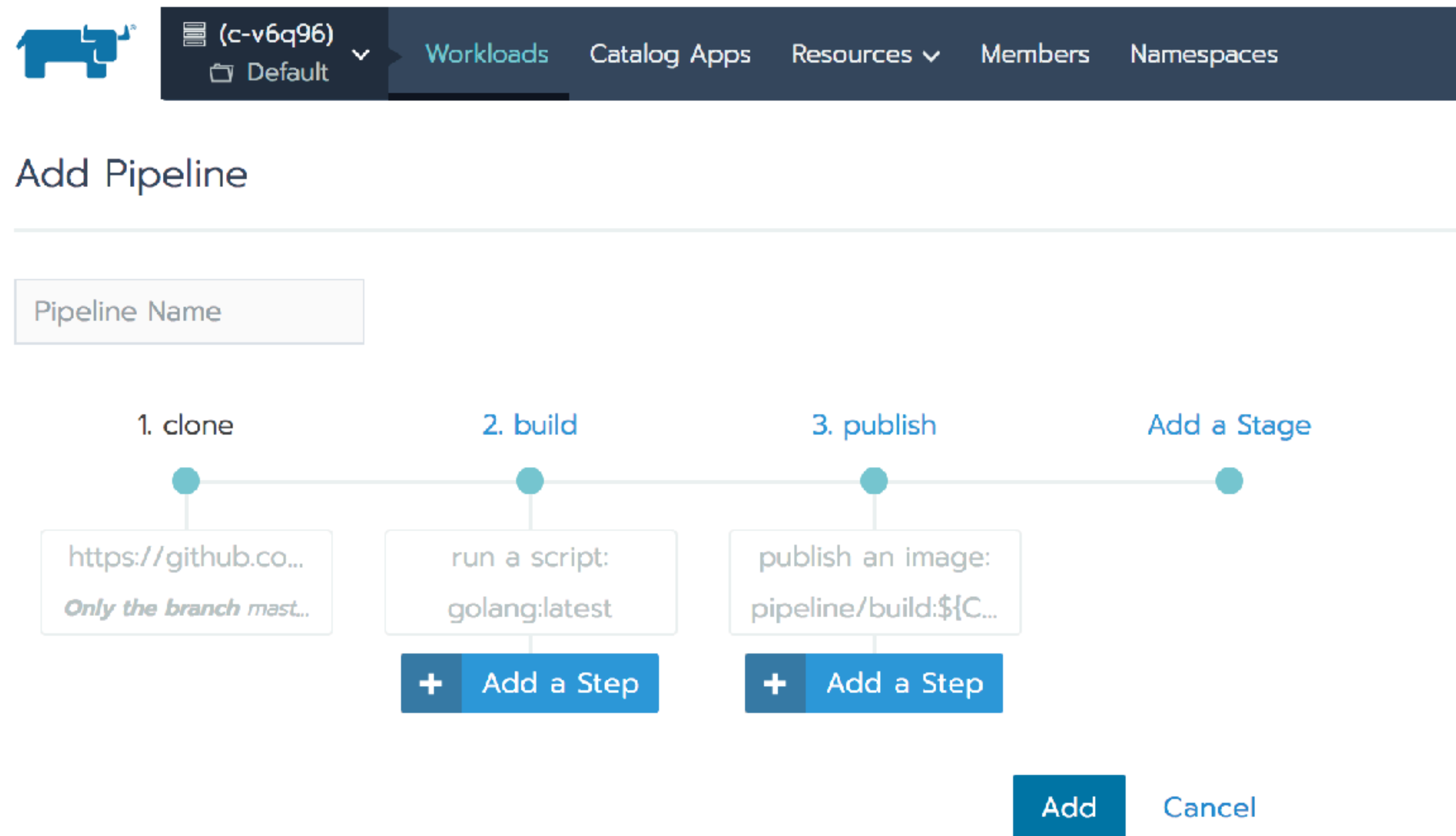
Service Type: ☒ Node Port ☐ Cluster IP ☐ Ingress

Persistence Volume: ☒ Disable ☐ Enable

[Show Advanced Options](#)

Save

# Configure CI/CD to push artifact and deploy







+



=



- JFrog Artifactory installation on demand as a service with just a couple of clicks
- Artifactory service health monitoring and recovery
- CI/CD pipeline integration
- User intuitive workload deployment interface using jFrog registry



# Feel free to reach out if you have questions



@lemonjet



alena1108



# THANK YOU!

