Cluster API Deep Dive With a Tencent Case Study
闵峰  Feng Min
Staff Software Engineer at Google Kubernetes team.

洪志国  Zhiguo Hong
Software Architect at Tencent Cloud.
Agenda

• What’s Cluster API
• Upgrade as example
• Bootstrapping
• How to deploy
• Demo
Where we are

- Cluster Management is a hard job.
- Fragmented tooling and ecosystem.
- Too many tools to choose from.
- Reinventing wheels.
What’s Cluster API?

<table>
<thead>
<tr>
<th>Declarative API</th>
<th>Common Logic</th>
<th>Pluggable Architecture</th>
<th>Tooling, Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Cluster</td>
<td>● Machine Lifecycle (incl. Provisioning)</td>
<td>● Infrastructure platform (vSphere, GCP, AWS, etc.)</td>
<td>● Cluster Bootstrapping, Upgrade</td>
</tr>
<tr>
<td>● Machine</td>
<td>● Machine Upgrade</td>
<td>● Support for various Operating Systems</td>
<td>● Auto-scaling, Repair, Node Auto-provisioning</td>
</tr>
<tr>
<td>● Machine Set + Machine Deployment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What’s Cluster API

Cluster Config with Declarative API
What is configured?

- **Cluster-level Configuration**
  - **Network**
  - Machine setup
    - # of machines?
  - Platform-specific configuration
    - GCP project, machine type
  - Software on those machines
    - Kubelet Version
    - Control Plane

```yaml
apiVersion: "cluster-api.k8s.io/v1alpha1"
kind: Cluster
metadata:
  name: kubecon-demo
spec:
  clusterNetwork:
    services:
      cidrBlocks: ["10.96.0.0/12"]
    pods:
      cidrBlocks: ["192.168.0.0/16"]
  serviceDomain: "cluster.local"
```
What is configured?

- **Cluster-level Configuration**
  - Network
- **Machine setup**
  - # of machines?
- **Platform-specific configuration**
  - GCP project, machine type
- **Software on those machines**
  - Kubelet Version
  - Control Plane

```yaml
apiVersion: "cluster.k8s.io/v1alpha1"
kind: MachineSet
metadata:
  name: my-first-machineset
spec:
  replicas: 3
  template:
    metadata:
      name: my-first-machineset
    spec:
      roles:
      - Node
```
What is configured?

- **Cluster-level Configuration**
  - Network
- **Machine setup**
  - # of machines?
- **Platform-specific configuration**
  - GCP project, machine type
- **Software on those machines**
  - Kubelet Version
  - Control Plane

```yaml
apiVersion: "cluster.k8s.io/v1alpha1"
kind: Machine
metadata:
  name: gce-master-kubecon
spec:
  providerConfig: >
  {
    "project": "gke-kubecon",
    "zone": "us-central1-f",
    "machineType": "n1-standard-2",
    "image": "/path/to/ubuntu-1604-lts"
  }
```
What is configured?

- **Cluster-level Configuration**
  - Network
- **Machine setup**
  - # of machines?
- **Platform-specific configuration**
  - GCP project, machine type
- **Software on those machines**
  - Kubelet Version
  - Control Plane

```yaml
apiVersion: "cluster.k8s.io/v1alpha1"
kind: Machine
metadata:
  name: gce-master-kubecon
spec:
  providerConfig: {
    ...
  }

  versions:
  - kubelet: 1.7.4
  - controlPlane: 1.7.4
```
What’s Cluster API

Tooling, YAML, CRDs, Controllers
Logical Flow - YAML to Objects

Toolings:
- upgrade
- autoscaling
- `kubectl` + `Cluster.yaml` + `machines.yaml`
- repair
- `clusterctl`
- `kops`

API:
- API Server + ClusterAPI CRDs

Objects:
- MachineDeployment
- Cluster
- MachineSet
- Machine
Logical Flow - Objects to Machines
Case Study: Upgrade

How do we upgrade K8s Cluster using Cluster API?
MachineDeployment
- replicas: 3
- nodes: MyNodes
- maxSurge: 1
- maxUnavailable: 0
- version: v1.8.3
**MachineDeployment**
- **replicas**: 3
- **nodes**: MyNodes
- **maxSurge**: 1
- **maxUnavailable**: 0
- **version**: v1.9.3
MachineSet
- replicas: 3
- version: v1.8.3
- selector:
  - nodes: MyNodes
Upgrade

MachineSet
- replicas: 3
- version: v1.8.3
- selector:
  - nodes: MyNodes

MachineSet
- replicas: 0
- version: v1.9.3
- selector:
  - nodes: MyNodes
MachineSet
- replicas: 3
- version: v1.8.3
- selector:
  - nodes: MyNodes

MachineSet
- replicas: 1
- version: v1.9.3
- selector:
  - nodes: MyNodes

Upgrade
MachineSet
  - replicas: 2
  - version: v1.8.3
  - selector:
    - nodes: MyNodes

MachineSet
  - replicas: 2
  - version: v1.9.3
  - selector:
    - nodes: MyNodes
Upgrade

MachineSet
- replicas: 1
- version: v1.8.3
- selector:
  - nodes: MyNodes

MachineSet
- replicas: 3
- version: v1.9.3
- selector:
  - nodes: MyNodes
MachineSet
- replicas: 0
- version: v1.8.3
- selector:
  - nodes: MyNodes

MachineSet
- replicas: 3
- version: v1.9.3
- selector:
  - nodes: MyNodes
How to get cluster api components up and running?
Bootstrapping - 10 km view
Bootstrapping - 10 km view
Bootstrapping - 10 km view
Bootstrapping - 10 km view

Cluster
- Machine Controller
- Cluster Controller

Control Plane
- Machine A
- Cluster A

CLI

User

Local Temporary External Cluster
- Machine Controller
- Cluster Controller
- Machine A
- Cluster A
Bootstrapping - 10 km view

Cluster

Machine Controller

Cluster Controller

Machine A

Cluster A

Control Plane

CLI

User
In Cluster or not?
Deploy - In Cluster
Deploy - Out of Cluster

Manager Cluster:
- kubelet
- Machine Controller
- Control Plane

Child Cluster:
- Machine
- Machine
- Machine
Provider Support

- AWS
- Azure
- DigitalOcean
- GCP
- OpenStack
- Tencent Cloud
- VSphere
- ...

DEMO on TKE
More Information

- [https://github.com/kubernetes-sigs/cluster-api](https://github.com/kubernetes-sigs/cluster-api)
- Joining Kubernetes-sig-cluster-lifecycle
- Weekly meeting on Wed @ 10:00 PT
- Office Hours: Weekly on Wed @ 06:00 AM and Tues @ 12:00 PT
- Slack: #cluster-api
Question?