

Logical KaaS(Kubernetes Cluster as a Service) over Distributed Clouds

승실대학교 DCN 연구실
박사과정 이장원
(jangwon.lee@dcn.ssu.ac.kr)

Contents

- ✚ Background
- ✚ Problem
- ✚ L-KaaS Architecture
- ✚ L-KaaS Scenario
- ✚ L-KaaS Resources
- ✚ Demo

DCN (Distributed Cloud and Network) Lab at SSU

Research Topics

Cloud

- Infrastructure Design (LCM, and Closed Loop Framework)
- Infrastructure and Workload Acceleration
- Serverless Interworking on Cross Domains
- Workload Placement and Management
- Stateful Workload Migration
- Cloud with AI (MLOps)

5G+/6G Network

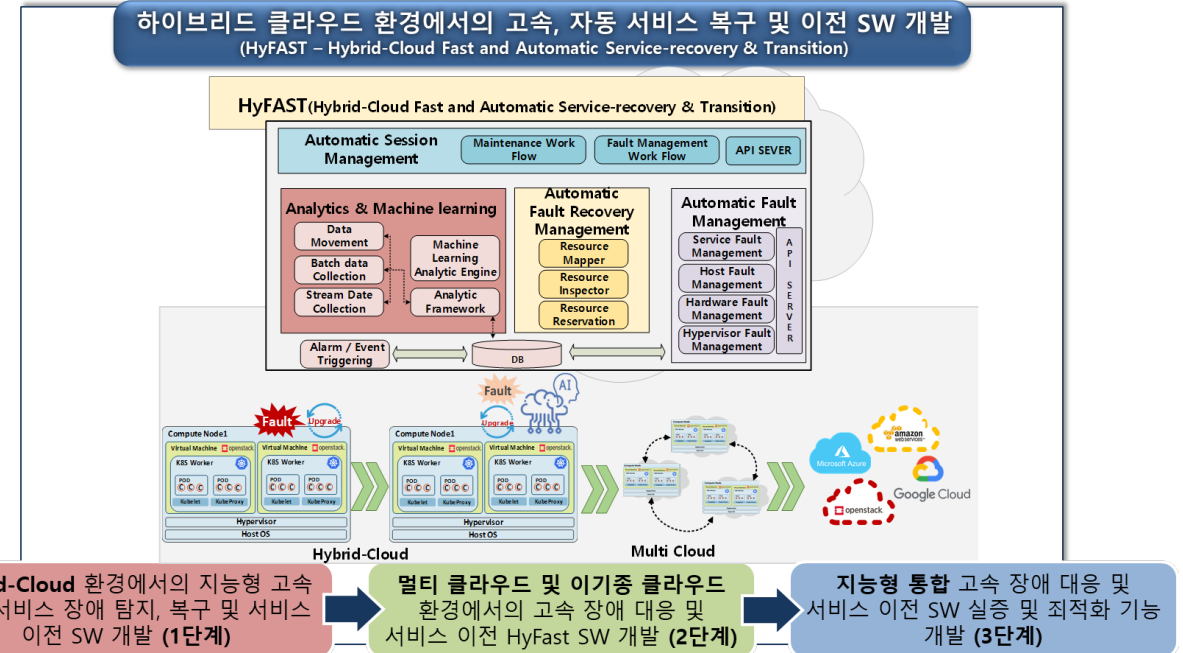
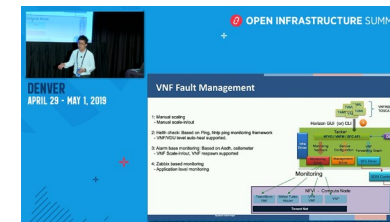
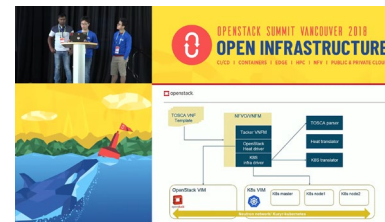
- SRv6+LISP for 5G underlay network
- MEC Service Mobility

OpenSource Contributions (OpenStack Tacker)

- VIM Rolling Upgrades
- Adding Multi-Interface for C-VNF
- Adding Kubernetes as VIM and C-VNF type
- VNFFG's Chain and Classifier
- Monitoring in VNFM using Zabbix

Presentations

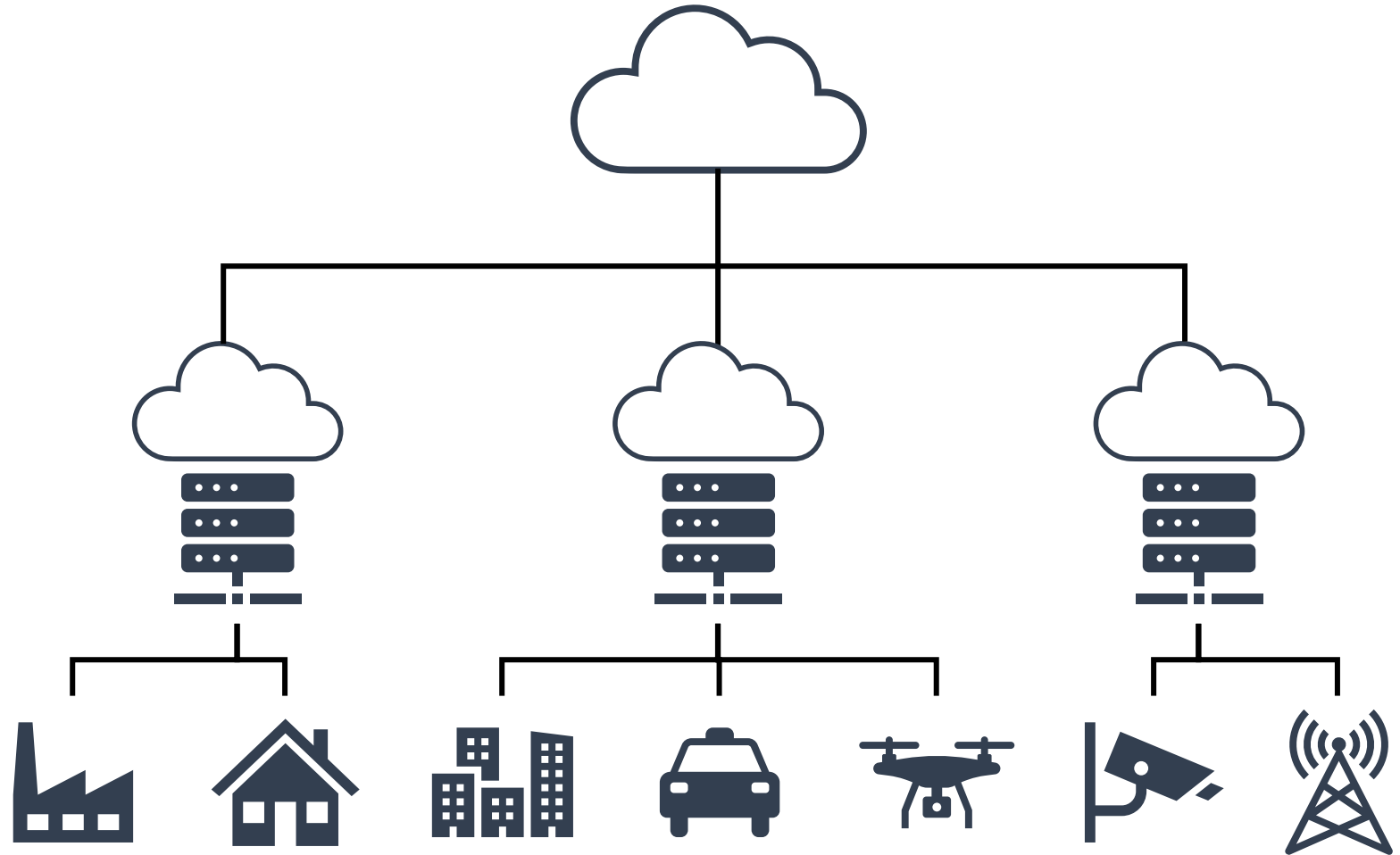
- OpenStack Summit 2018: Hybrid VNF across OpenStack and Kubernetes VIM
- OpenStack Summit 2019: A New Era for NFV with Tacker



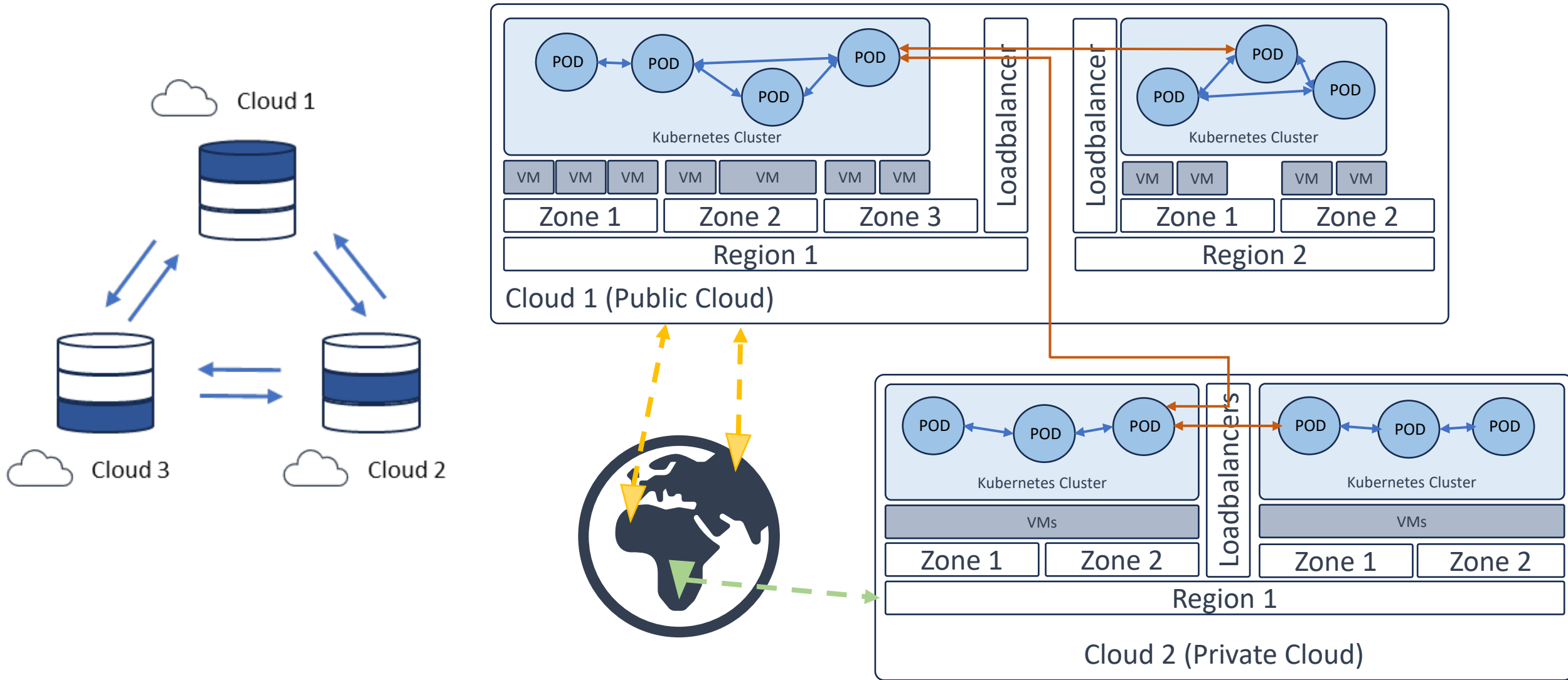
Background – Cloud and Edge

Centralized Cloud,
Public and Private Cloud

Edge Computing



Logical Cloud over Multi-Cloud, Multi-Region



Background – Required Steps for Cluster Set-UP

✦ Cluster Setup

- ✦ Installation (Kubeadm, Minikube, Rancher, Cluster API, ...)
- ✦ ETCD
- ✦ DNS (CoreDNS, ...)
- ✦ CCM (Cloud Controller Manager related)

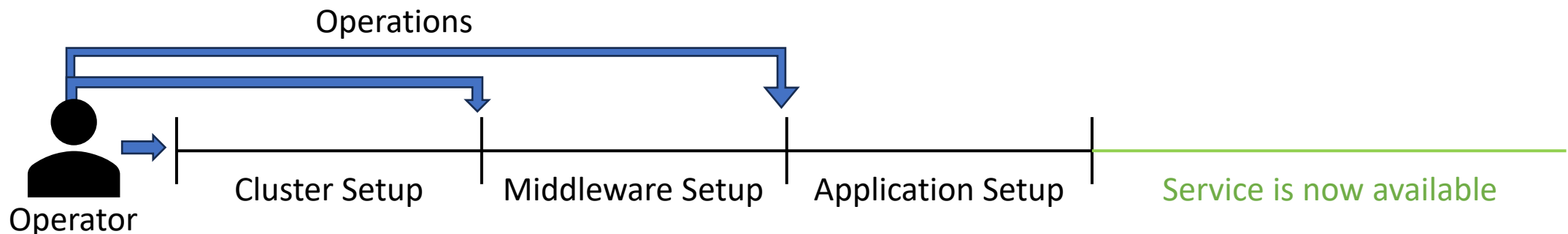
✦ Networking

- ✦ CNI Plugins (Calico, Flannel, Cilium, Weave, VPC related, ..)
- ✦ Service Mesh (Istio, NSM, ...)

✦ Storage

- ✦ CSI Plugins (NFS, Ceph, Cloud related Storage Service)

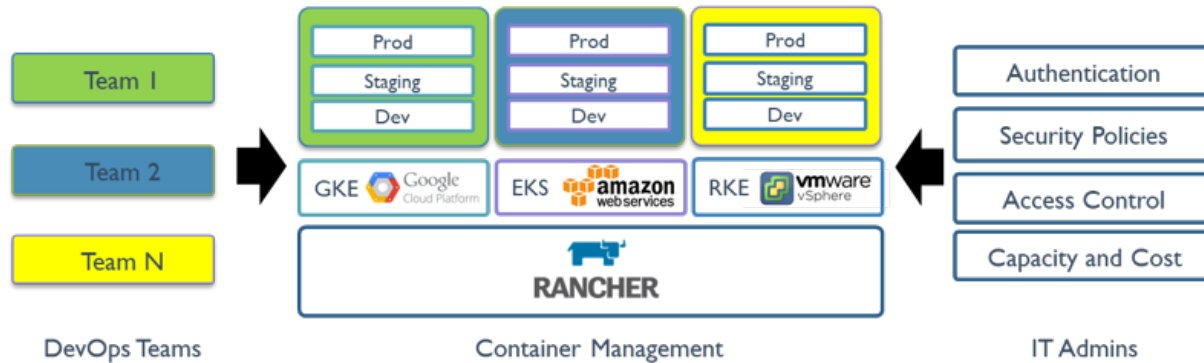
✦ Workloads or 3rd party things



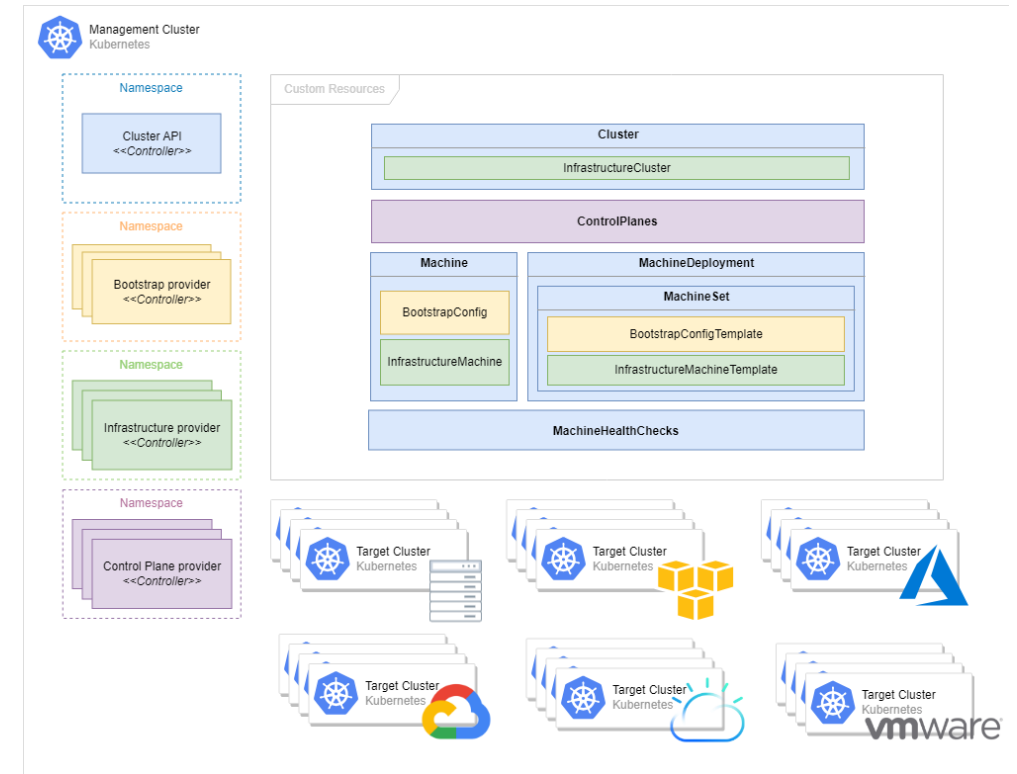
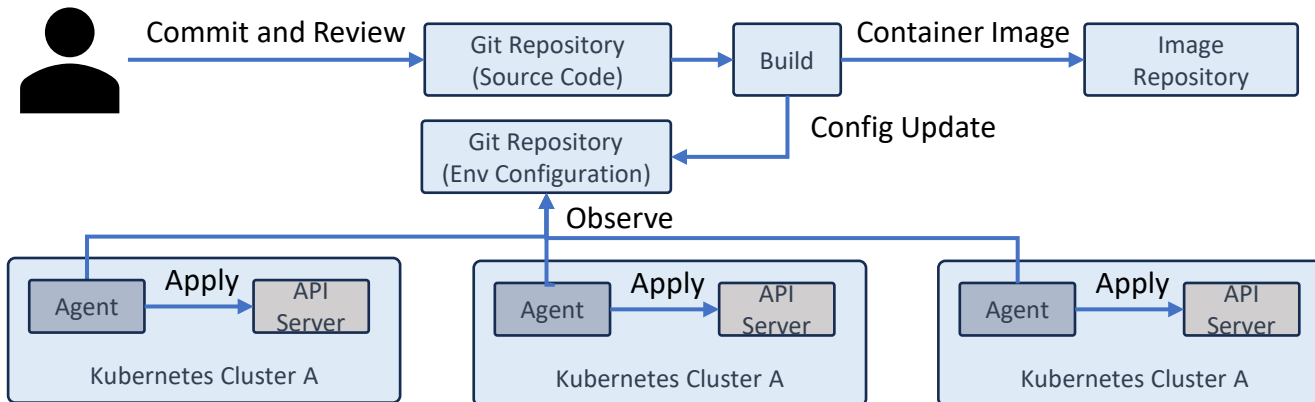
Background – Trends for creating multiple clusters

✦ Configure a Cluster as a Resource (Template based)

✦ CI/CD, Propagation, GitOps



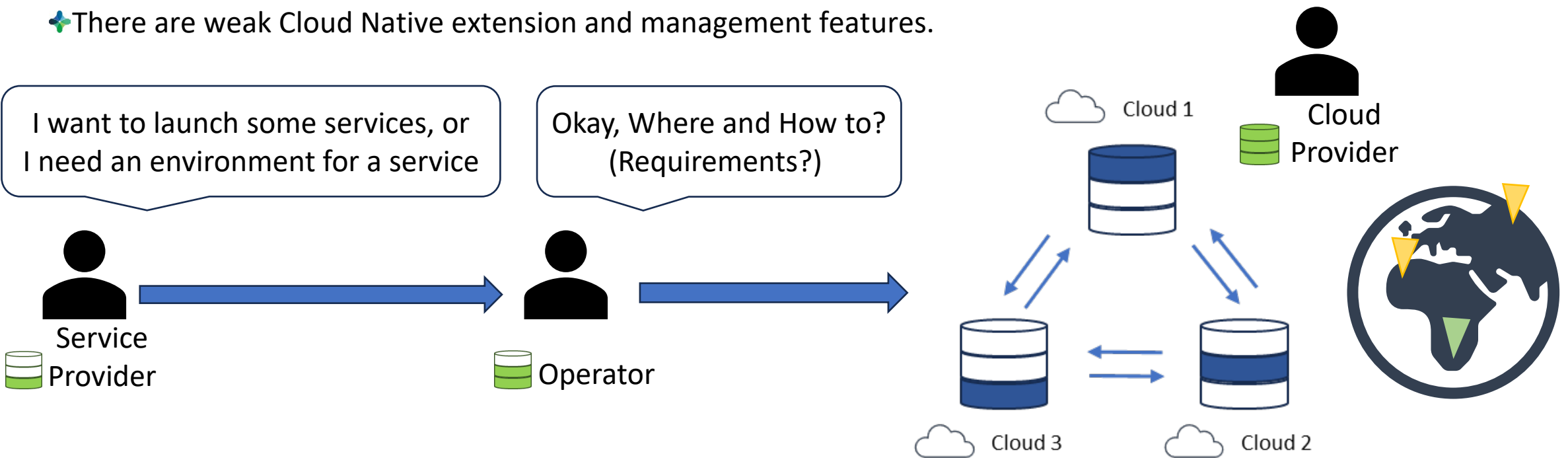
출처: Rancher



출처: Cluster API

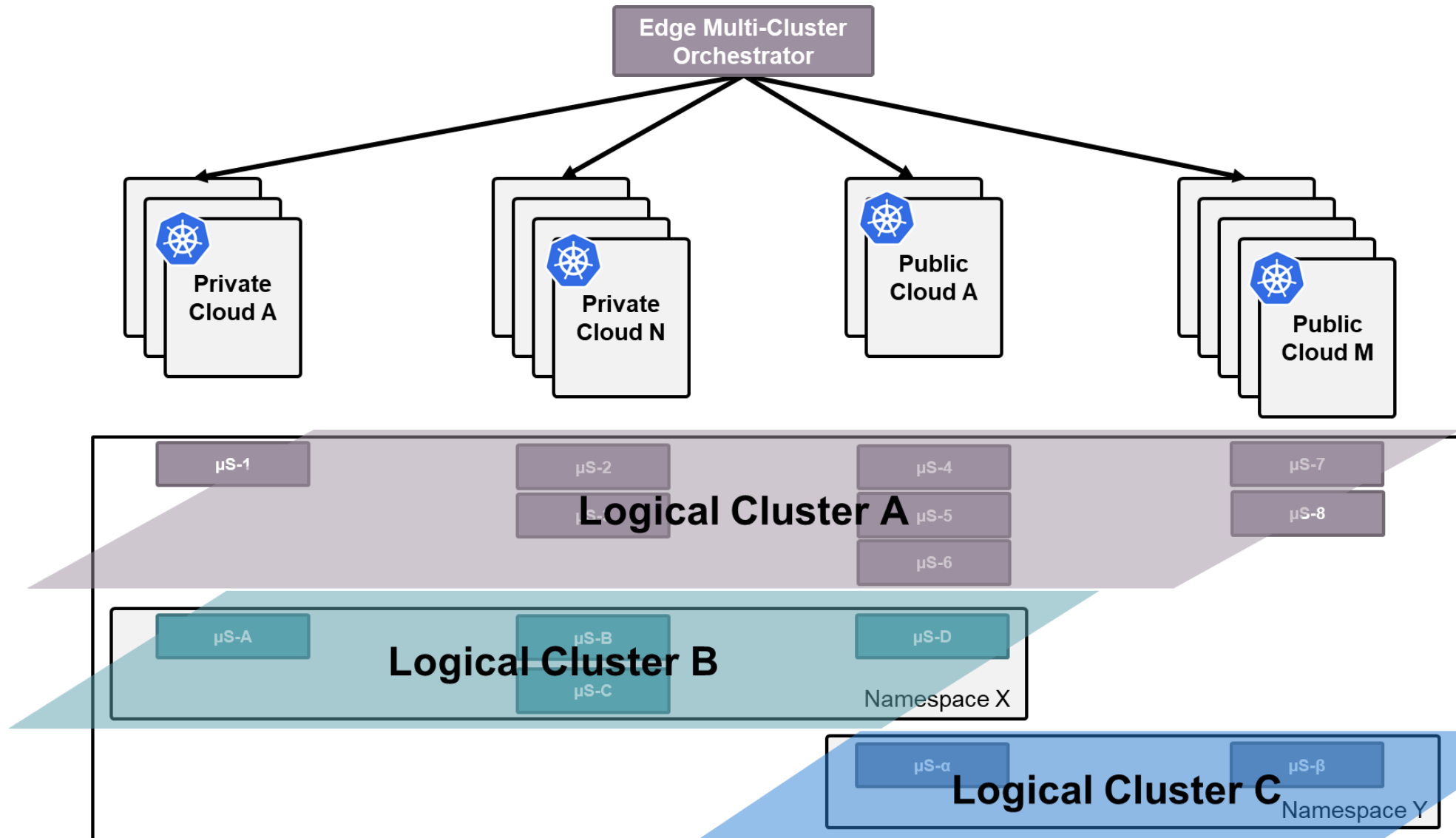
Problem: from Day0 to Day 2

- ✦ Require to have a deep knowledge of Kubernetes and a ton of Kubernetes-related configuration.
- ✦ If system consists of multiple clusters, or tens of clusters, administrators must put in a lot of effort to maintain, upgrade, and monitor.
- ✦ There is no open-source project that provides provisioning, managing, or operating a system that has multiple clusters operating and running in a distributed environment.
- ✦ There are weak Cloud Native extension and management features.



 Cloud Knowledge (Weak to Strong)

Problem: Multitenancy for Multi Clusters on Multi Cloud



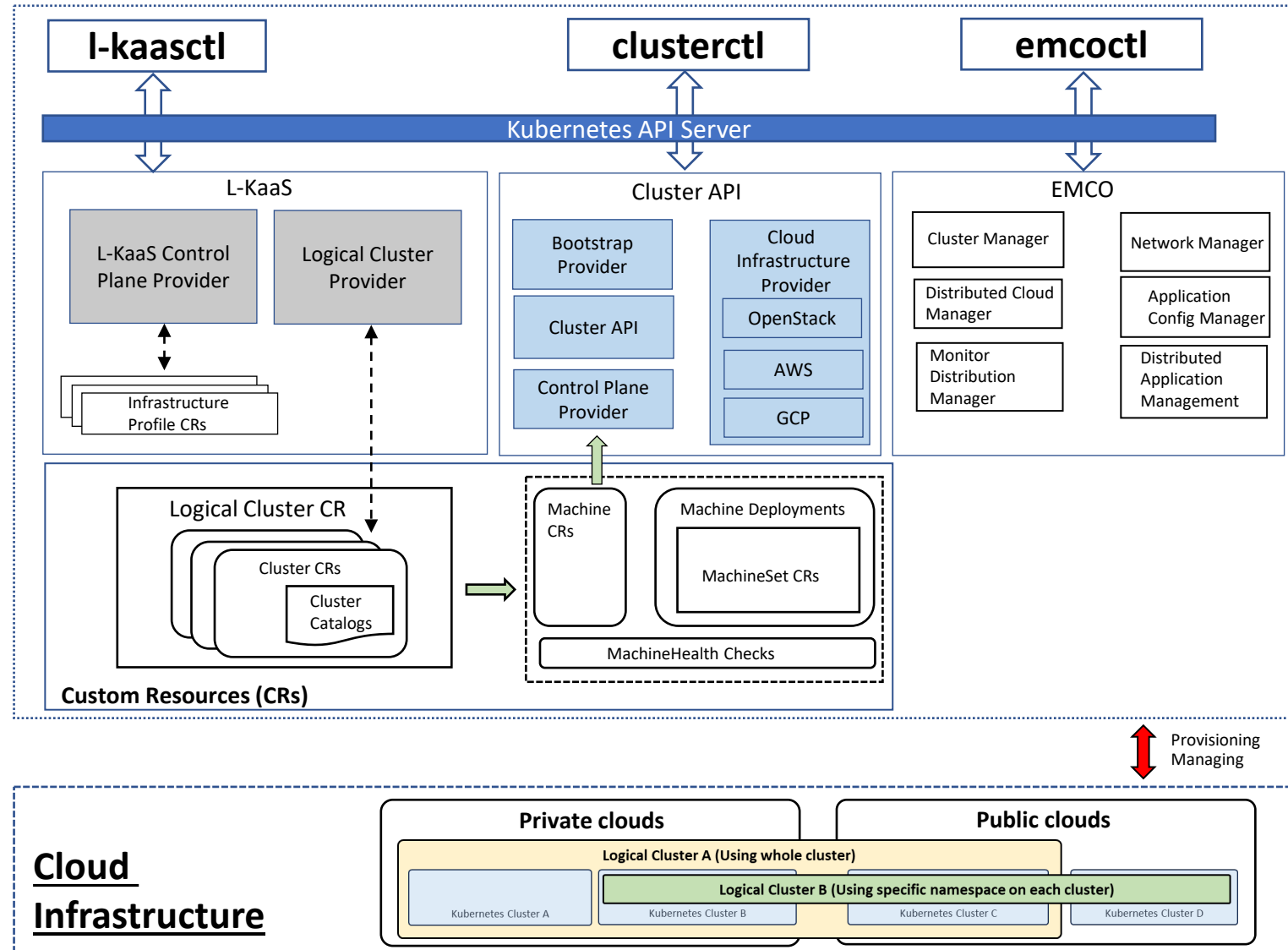
L-KaaS Architecture

✦ Role

- ✦ L-KaaS: LCM for Logical Cluster
- ✦ Cluster API: Actuator for a Cluster
- ✦ EMCO: Actuator for Workload Deployment and Management

✦ EMCO(Edge Multi Cluster Orchestrator)

- ✦ LFN(Linux Foundation Networking) Project
- ✦ Benefits
 - ✦ Scheduling-based Workload Management
 - ✦ Multiple Network Management
 - ✦ Module-based Scalability



L-KaaS

✦ Logical Kubernetes Cluster as a Service (L-KaaS)

- ✦ L-KaaS is a project focused on **providing declarative APIs and tooling to simplify, abstract, be easy to use for users who don't have deep technical knowledge of infrastructure and shield them from low-level concepts and technologies.**
- ✦ L-KaaS uses Kubernetes Resources Model and Kubernetes environment to **provide an abstraction high-level & automation framework that automates clusters, logical clusters.**
- ✦ The L-KaaS is a project standing on the Cluster API and EMCO Project.

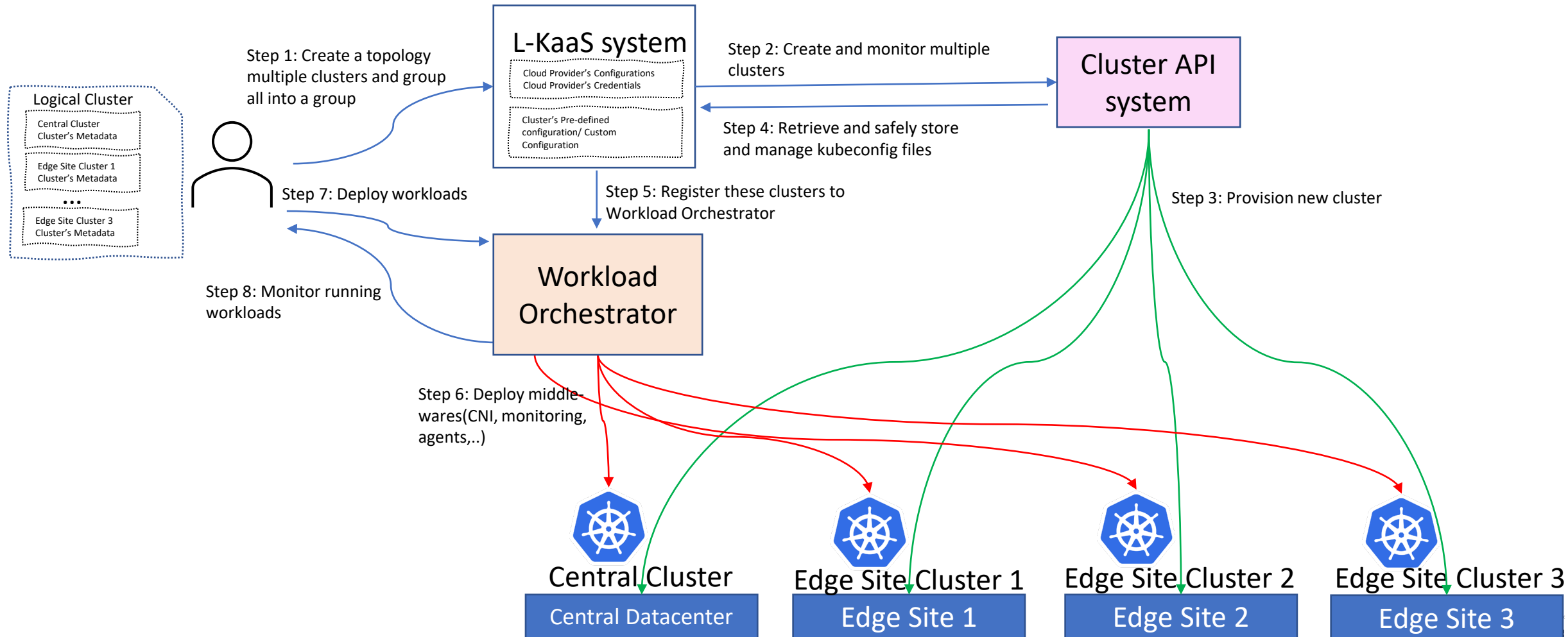
✦ Why L-KaaS

- ✦ Managing Multiple Clusters System Lifecycle from Day 0 (creation) through Day 2
- ✦ Consistent, declarative control to Kubernetes clusters on different types of infrastructure
- ✦ Managing Multi-Clusters environment with GitOps
- ✦ L-KaaS abstracts the low-integrated infrastructure configuration of cloud provider and standardize it across numerous cloud vendor
- ✦ More control over the configuration and installed software, a standardized approach to multi-cluster management

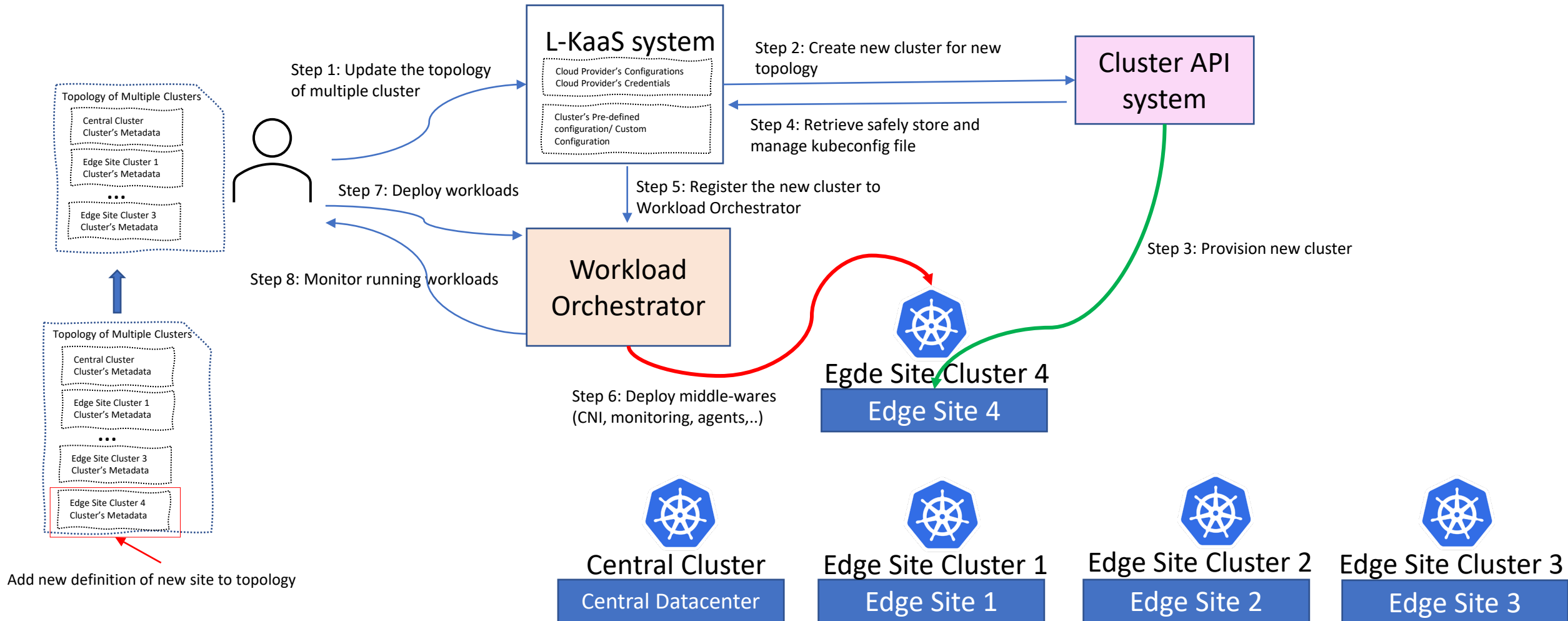
Comparison with other Projects

Category	Action	Rancher	Cluster API	L-KaaS
Cluster Setup	Goal	Cluster Ready	Cluster Creation (to Init)	Service Ready
	Recommend Setup	Dashboard / Configuration with yaml	Kubernetes resources	Kubernetes resources
	Template Repository	Helm chart	No	Yes (Git or URL), GitOps TBD
Cluster Management	Adding Existing Cluster	Yes	TBD	TBD
	Managing Cluster Members	Limited (Created by RKE engine)	Health Monitor CA (Cluster Autoscaler)	Cluster API + CA + Self System(TBD)
	Editing and Upgrading Clusters	Yes	Yes (based on Topology Features)	Yes
	Managing Persistent Volumes and Storage Classes	Yes	No	Yes
	Managing Projects, Namespaces	Yes	No	Yes
Workload Management	Using App Catalogs	Yes (Helm)	No	Yes (Profile resources)
	Continuous Delivery	Yes (GitOps with Fleet)	No	Yes (GitOps with Flux)
	Deploy a Workload to a Cluster	Yes (Dashboard)	Limited (based on ClusterResourceSet)	Yes (Helm)
	Deploy a Workload to Multi-clusters	Yes (Propagation, Helm)	Limited (Propagation, Label based)	Yes (Scheduling and Propagation)
	Init Container	Yes	No	No (Considerable)
	Upgrade, Rollback, ...	Yes	No	Yes
	Stateful Workload Management	No	No	TBD
Networking	Multiple Interfaces	Yes	No	Yes
	Gateway and Service Setup	Yes (Default: Nginx Ingress and Optional using Istio)	No	Yes (Optional automation step using Istio, TBD)
	Traffic Management	Yes (Specific CNI and Istio)	No	Yes (Specific CNI and Istio)

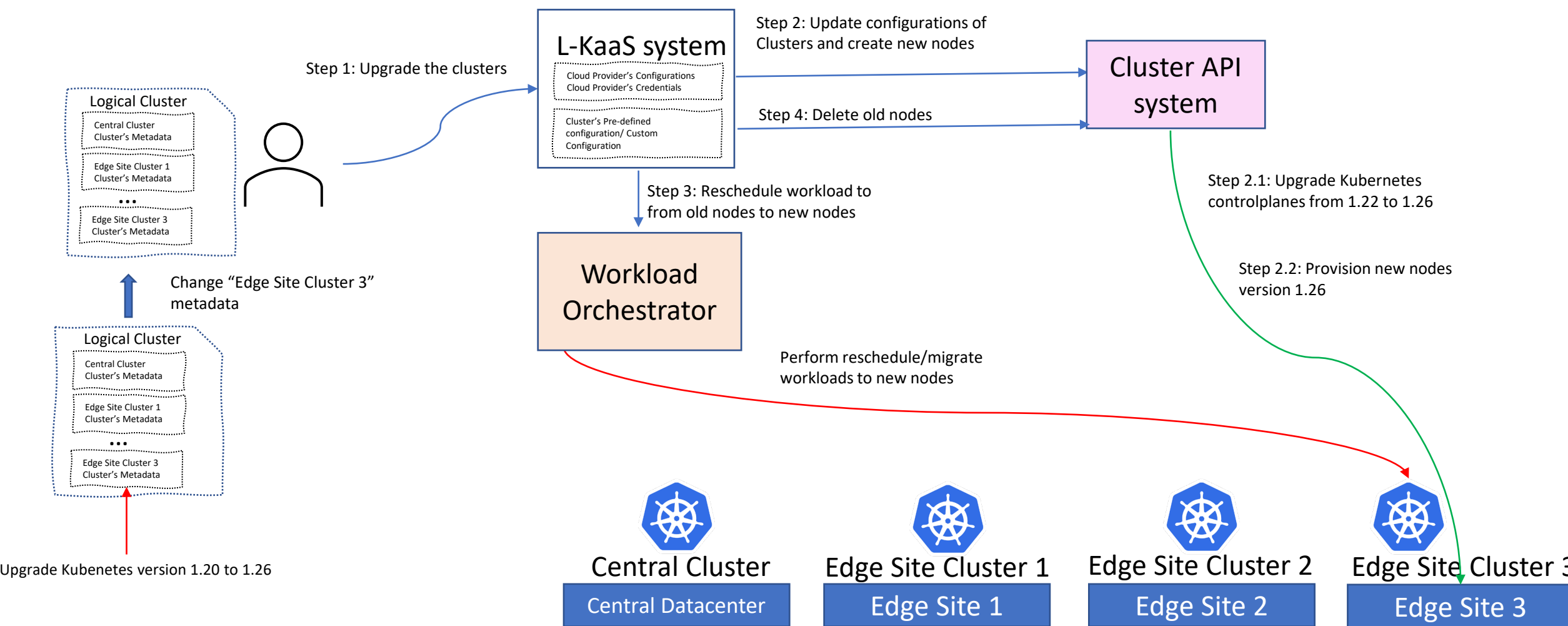
Create Multiple Cluster in L-KaaS



Spin up a new edge site with L-KaaS



Example: Automation in upgrading the version Kubernetes of “Edge Site Clusters” with zero-downtime



L-KaaS Controller Functions

✦ Logical Cluster Provider:

✦ Primarily Functions:

- ✦ Managing Infrastructure Profile resources, Cluster resources, Logical Cluster Resources
- ✦ Transforming Logical Cluster Resources, Cluster Resources with Infrastructure Profile to CAPI Resources.
- ✦ Make sure CAPI Resources match states with Cluster Resources, and Logical Cluster Resources (re-transform Cluster Resources and Physical if it is changed).
- ✦ Versioning and Syncing Logical Clusters, Cluster Catalog, Cluster Resources resources to GitHub repositories.
- ✦ Reconciling Cluster Resource, Logical Cluster Resource to match states with physical cluster

✦ Logical Cluster Controlplane:

✦ Primary Functions:

- ✦ Managing the start of L-KaaS system, and downstream consumers.
- ✦ Managing secret, kubeconfig, system configuration, and provider configurations
- ✦ Providing an interface for user to interact, CRUD with L-KaaS
- ✦ Registering a new logical cluster to EMCO in order to manage workload, and applications.

L-KaaS Resources

✦ There are 4 kinds of Custom Resources

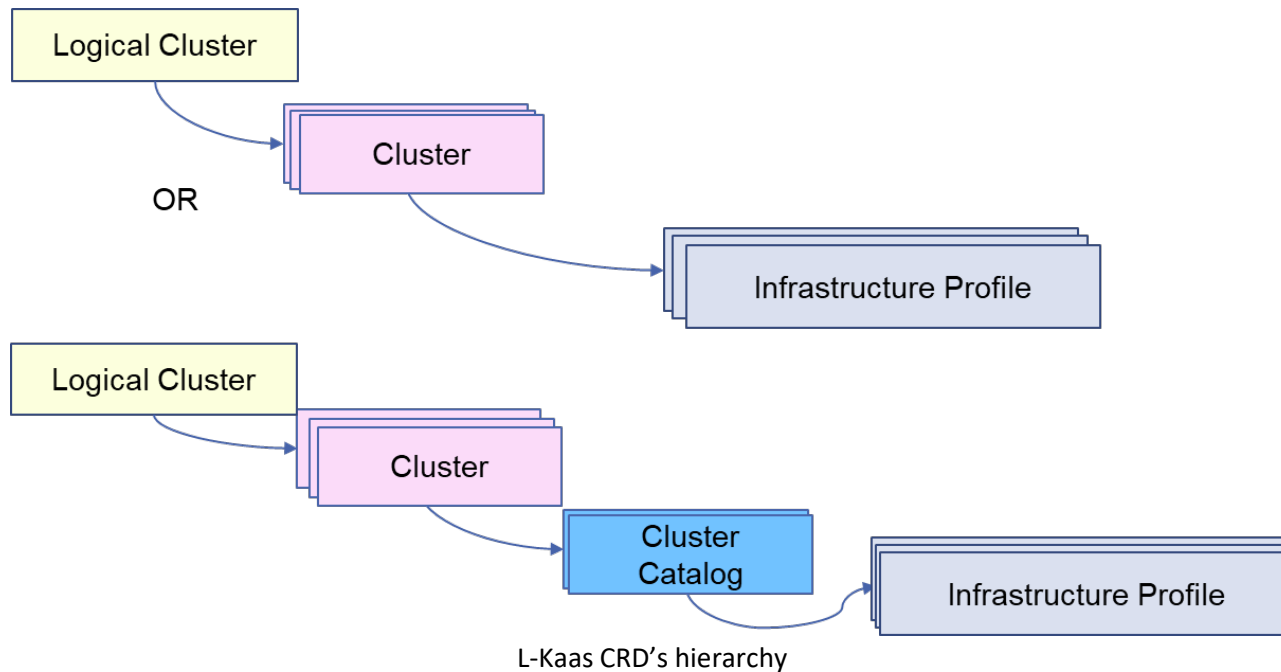
✦ Logical Cluster Resources:

✦ Cluster Catalog Resources:

✦ Cluster Resources:

✦ Infrastructure Profile Resources

✦ Logical Cluster, Cluster, Cluster Catalog, and Infrastructure Profile are consumed by **Logical Cluster Provider**.



L-KaaS Cluster Resource

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: Cluster
metadata:
```

```
  name: cluster-sample
  namespace: starlab
  tenant: starlab
  labels:
    automation.dcn.ssu.ac.kr/region: kr-seoul
    automation.dcn.ssu.ac.kr/zone: public
```

Cluster Metadata

```
spec:
```

```
  infrastructure:
    spec:
      provider: openstack
      profile:
        name: openstack-small-cluster
        type: Infrastructure
        revision: 1.0.0
      override:
        kubernetesVersion: v1.25.0
```

OpenStack Infrastructure Profile and cluster characteristics

Cluster API Profile associated with OpenStack provider

```
  network:
    spec:
      name: calico-v1
      profile:
        name: calico-v1
        type: Network
        revision: 1.0.0
```

Infrastructure related to network

```
  software:
    - name: prometheus
      profile:
        name: prometheus-default
        type: Software
        revision: 1.0.0
```

Infrastructure related to software

L-KaaS Logical Cluster Resource

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: ClusterCatalog
metadata:
  name: Core5GCluster
  namespace: starlab
spec:
  infrastructure:
    spec:
      provider:
        name: openstack-v1
        profile:
          name: openstack-medium-cluster-v1
          type: Infrastructure
          revision: 1.0.0
        override:
          kubernetesVersion: v1.25.0
      network:
        spec:
          name: cilium-v1
          profile:
            name: cilium-v1-default
            type: Network
            revision: 1.0.0
          software:
            - name: prometheus-v2.2
              profile: prometheus-default
              type: ApplicationTemplate
              revision: 1.0.0
```

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: ClusterCatalog
metadata:
  name: MECClusterIn5G
  namespace: starlab
spec:
  infrastructure:
    spec:
      provider:
        name: openstack-v1
        profile:
          name: openstack-small-cluster-v1
          type: Infrastructure
          revision: 1.0.0
        override:
          kubernetesVersion: v1.25.0
      network:
        spec:
          name: calico-v1
          profile:
            name: calico-v1-default
            type: Network
            revision: 1.0.0
          software:
            - name: prometheus-v2.2
              profile: prometheus-default
              type: Software
              revision: 1.0.0
```

```
apiVersion: intent.automation.dcn.ssu.ac.kr/v1
kind: LogicalCluster
metadata:
  name: mec-cluster
  labels:
    automation.dcn.ssu.ac.kr/project: dcn
    tenant: starlab
spec:
  clusters:
    - cluster:
        name: central-cluster
        metadata:
          labels:
            automation.dcn.ssu.ac.kr/location: kr-seoul
            automation.dcn.ssu.ac.kr/managed-by: emco-mec-central-mgmt
          spec:
            clustercatalog: openstack-medium-catalog
      - cluster:
        name: seoul-edge-cluster
        metadata:
          labels:
            automation.dcn.ssu.ac.kr/location: kr-seoul
            automation.dcn.ssu.ac.kr/managed-by: emco-mec-central-mgmt
          spec:
            clustercatalog: openstack-medium-catalog
      - cluster:
        name: busan-edge-cluster
        metadata:
          labels:
            automation.dcn.ssu.ac.kr/location: kr-busan
            automation.dcn.ssu.ac.kr/managed-by: emco-mec-central-mgmt
          spec:
            clustercatalog: openstack-medium-catalog
      - cluster:
        name: gwangju-edge-cluster
        metadata:
          labels:
            automation.dcn.ssu.ac.kr/location: kr-gwangju
            automation.dcn.ssu.ac.kr/managed-by: emco-mec-central-mgmt
          spec:
            clustercatalog: openstack-medium-catalog
```


L-KaaS Infrastructure Profile Resources

- ✦ **Infrastructure Profile Resources:** referred to **pre-defined configurations** that contains configurations about Cloud provider, metadata, cluster settings,...
- ✦ Infrastructure Profile are created by administrators/operators at Day 0 (At beginning of L-KaaS setup or after setting up a new Infrastructure)
- ✦ Infrastructure Profile are stored in both the Kubernetes cluster and Github repository and can be used by tenants.
- ✦ Example of Infrastructure Profile:

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: InfrastructureProfile
metadata:
  name: openstack-small-cluster
  type: Infrastructure
  namespace: default
  revision: 1.0.0
spec:
```

Identifiers of Infrastructure Profile

```
  profile:
    name: clusterapi-resource-template-default
    type: Infrastructure
    revision: 1.0.0
    kubernetesVersion: v1.24.8
    controlPlaneMachineCount: 1
    workerMachineCount: 3
    controlplaneFlavor: m1.medium
    workerFlavor: m1.large
```

Profile refers to another profile in order to reuse configurations.

Infrastructure Profile values

L-KaaS Infrastructure Profile Resources

Cluster and Infrastructure Profile

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: InfrastructureProfile
metadata:
  name: openstack-small-cluster
  type: Infrastructure
  namespace: default
  revision: 1.0.0
spec:
  profile:
    name: clusterapi-resource-template-default
    type: Infrastructure
    revision: 1.0.0
    kubernetesVersion: v1.24.8
    controlPlaneMachineCount: 1
    workerMachineCount: 3
    controlplaneFlavor: m1.medium
    workerFlavor: m1.large
```

Calico CNI Profile

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: InfrastructureProfile
metadata:
  name: calico-v1
  type: Network
  namespace: default
  revision: 1.0.0
spec:
  cni: calico
  version: 0.20.0
  podCIDR: ["10.96.0.0/12"]
  serviceCIDR: ["192.168.0.0/16"]
```

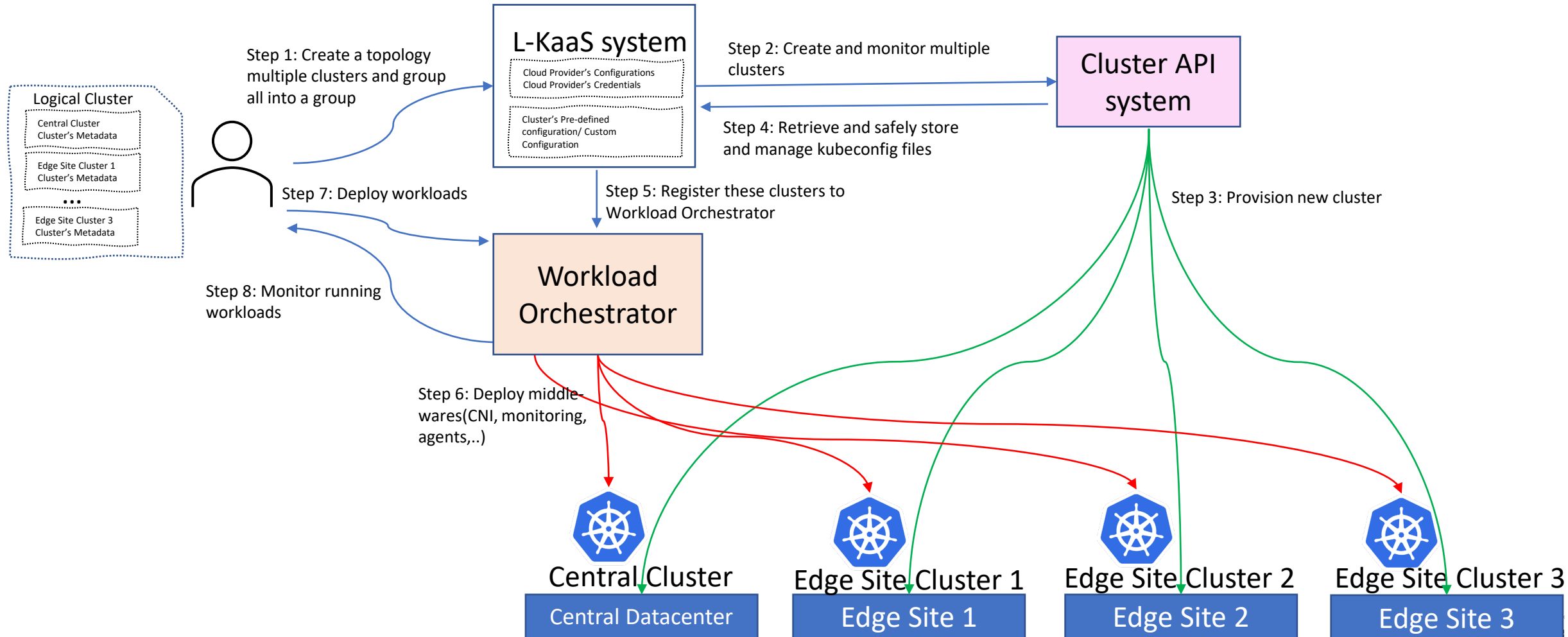
Cluster API Profile

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: InfrastructureProfile
metadata:
  name: clusterapi-resource-template-default
  type: Infrastructure
  namespace: default
  revision: 1.0.0
spec:
  filename: cluster-template.yaml
  url: https://github.com/kubernetes-sigs/cluster-api-provider-openstack/releases
  version: v0.7.0
```

Prometheus Profile

```
apiVersion: automation.dcn.ssu.ac.kr/v1alpha1
kind: InfrastructureProfile
metadata:
  name: prometheus-default
  type: Software
  namespace: default
  revision: 1.0.0
spec:
  version: 2.41.0
  volumeMounts:
    - name: config
  volumes:
    - name: config
      configMap:
        name: prometheus-config
```


Demo: Create a Logical Cluster and Deploy a middleware



QnA