

Leveraging Cluster API for Production-Ready Multi-Regional Infrastructures

Kotaro Inoue

*Software Engineer,
LY Corporation*

Shotaro Gotanda

*Software Engineer,
LY Corporation*

Agenda

01

Overview of our Platform

02

Adoption of Cluster API

03

Multi-Region and Multi-AZ with Cluster API

04

In-place Migration to Cluster API

05

Obstacles and Insights

Agenda

01 Overview of our Platform

02 Adoption of Cluster API

03 Multi-Region and Multi-AZ with Cluster API

04 In-place Migration to Cluster API

05 Obstacles and Insights

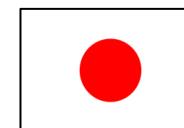


LINE

A communication app that connects people, services, and information

Launched on June 23, 2011

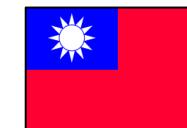
MAU



95M



54M



22M



6M

As of Mar. 2023

Private Cloud Platform: Verda

for LINE Service Developers



VM/Baremetal



Kubernetes



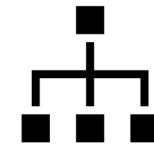
Block Storage



OpenSearch



NAT



Load Balancer



And more...

40+ Services in Verda

Verda Prod Project vks-manual-test Product Overview Region Tokyo User

Project & Support	Compute & Container	Network	Database	Contents Delivery & Storage	Application Service
Overview Overview for selected project	Servers Virtual/Physical servers and Persistent Block Storage in the LINE data center	DNS Operate DNS records of LINE	MySQL Service Easily configure and manage MySQL high availability and support shard feature	VOS for Internal Object Storage service comes with an S3 compatible Object Storage API	Kafka Provides creation and permission management of CRUD topics to Kafka cluster managed by IMF
Approvals Approvals requested by the project	Kubernetes Service Kubernetes as a service managed by Verda	Load Balancer Load Balancer is the component which offers load distribution and high availability of your application	MongoDB Service Fully managed and automated MongoDB service makes running document workloads easy	VOS for CDN Object Storage service comes with an S3 compatible Object Storage API	GeoIP API
Manage Member Manage members and roles of the project	App Runner Easiest way to run your apps. Deploy from source code or container image.	Internet Gateway Provides reliable internet connectivity without attaching Public-IP for your computing instance	Redis Service Can launch Redis servers easily and simple	VSFS (Shared File System) POSIX compliant shared file system for Verda	
Service Accounts Manage service accounts of the project	Functions Serverless computing platform service that allows you to run code without having to provision or manage servers	API Gateway Can create REST API define related resources and methods	OpenSearch(Elasticsearch) Service Helps developers build OpenSearch(Elasticsearch) cluster easily and promptly	CDN Content delivery network is a service that provides CDN service for your project	
Notice Learn about new releases, latest updates, and maintenances		VPC Provides isolated networking and fine-grained access control for high security.	DBS for MySQL (Deprecated)	CDN Purge	
Documents Technical documentations for all Verda Products	Container Registry Image registry service to easily store and manage container images.		MySQL (Deprecated)		
API Doc API reference for all Verda Products					
Help Verda Communication channel to receive improvement feedbacks					
Infra Tools	CI/CD & Repository	Code Quality	Observability & Auth & Audit	Deployment	Media
Voyager Certificate management service	PIPE CI/CD Pipeline as a Service	SonarQube Static code analysis tool	Audit Log Project audit log	Inventory Inventory information management service	OBS Media platform for LINE and LINE-family services
IDMS2 Server ACL management service	Jenkins Continuous integration tool for build, test, and deploy your application		Alexander OAuth2 authentication and authorization service	PMC Application management service that provides role management, deployment and application configuration management feature	Video Hub Encoding, player, subtitle, analysis of video for superior streaming experience on multiple devices
Asset	Circle CI Another continuous integration tool		IMON Observability service for your application and infrastructure	PMC Deployment UI User-friendly UI to deploy and restart application on PMC	
ACL Tracking	Nexus Binary repository manager service		DBMON Database monitoring service	Forestry Deployment pipeline service based on PMC	
Staff Finder	Harbor Docker registry with policies and role-based access control		Promgen Prometheus Monitoring and Alerting Service	Abyss Static file deployment tool for Front-end	
	LandPress Host static web apps and websites by providing a simple CI/CD				

Large Scale Platform

@ October, 2023





Verda Kubernetes Service (VKS)



- Managed Kubernetes Platform
- Provide Native Integration of Verda and Corporate Platform
- Self-Serviced via REST API
- Launched in 2019

@ October 16th, 2023

Agenda

01 Overview of our Platform

02 Adoption of Cluster API

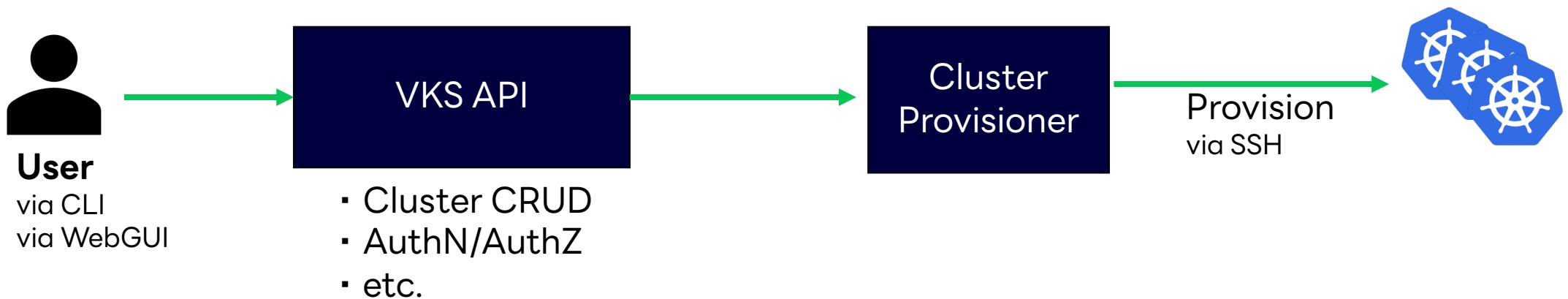
03 Multi-Region and Multi-AZ with Cluster API

04 In-place Migration to Cluster API

05 Obstacles and Insights

Overview of Legacy

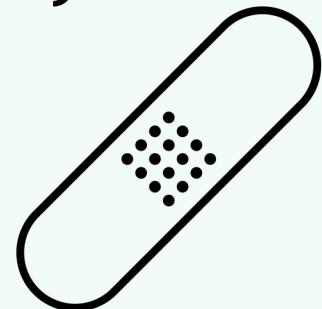
As of 2022



Management of Cluster Provisioner

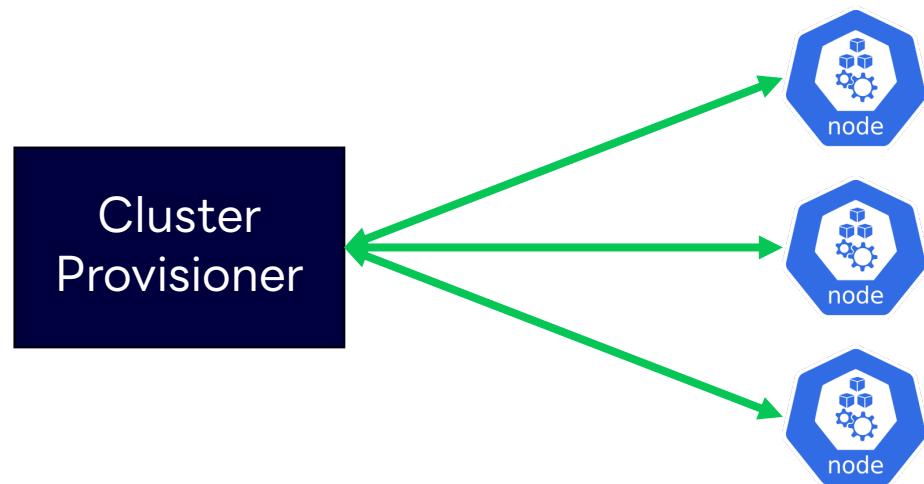
Issue #1

- Multiple patches into our Fork (**6** repositories in total)
 - Custom Features
 - Bug Fixes
- Outdated Version
- Hard to backport upstream changes
 - Complex codebase because of our patches



Node Management via Stateful Connections

Issue #2



- Establish connections to all nodes
- Connections often become unstable
 - Need to restart controllers manually for mitigation

Local cluster: ClusterOwnerDisappeared

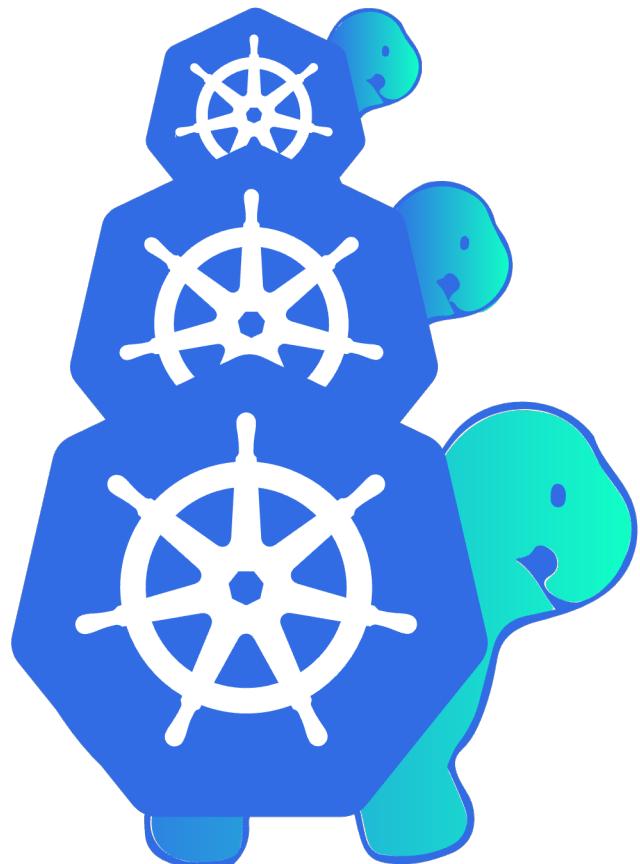
Runbook - Graph

- Owners of 26.18181818181818 clusters disappeared. Web socket sessions are unstable

Requirement for new Cluster Provisioner

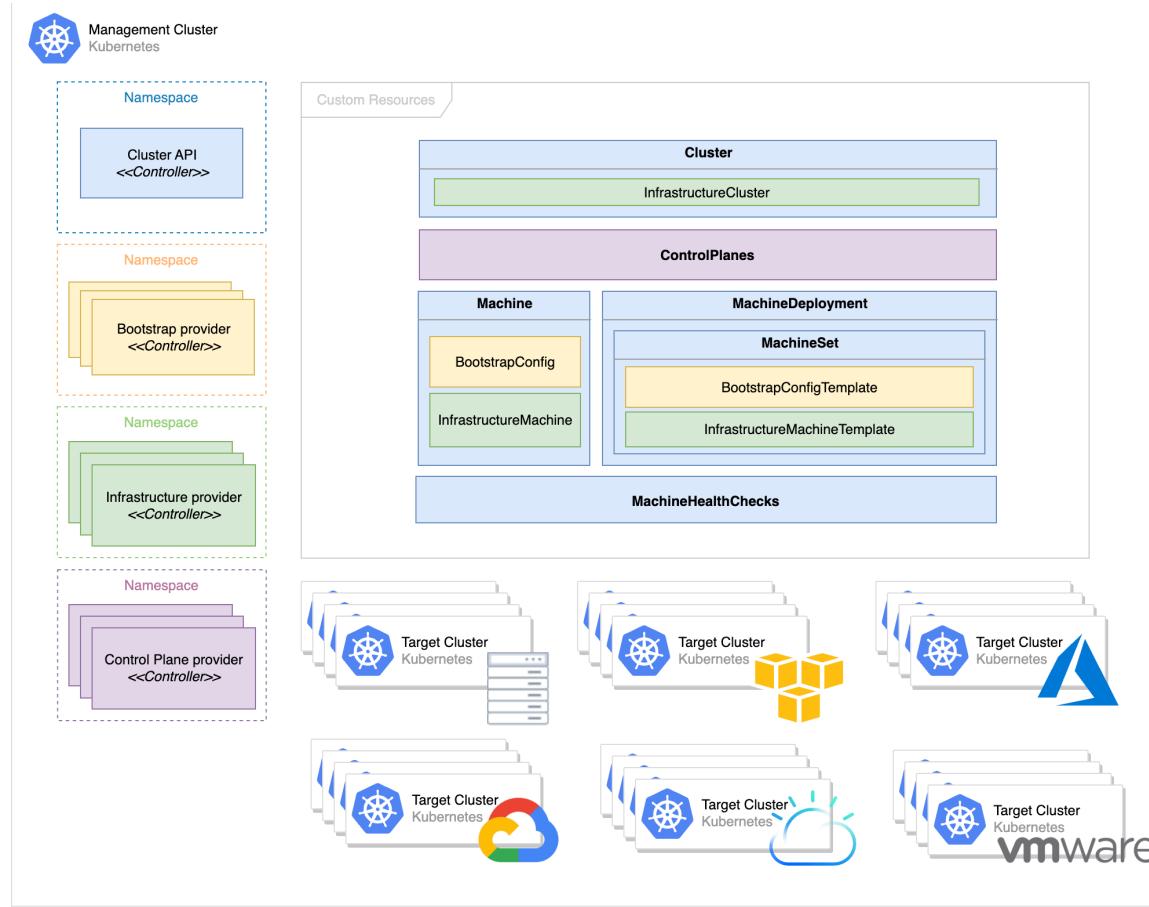
- **Open Source**
- **Lower maintenance cost**
 - Pluggable Interfaces
- **Scalable**
 - No stateful connections for node management

Cluster API (CAPI)



- Open Source Cluster Provisioner
- SIG Cluster Lifecycle
- Declarative API
 - Help to provision/operate multiple k8s clusters
- Set of Tools
 - Useful for k8s cluster/node management

Cluster API ❤ Any Platforms



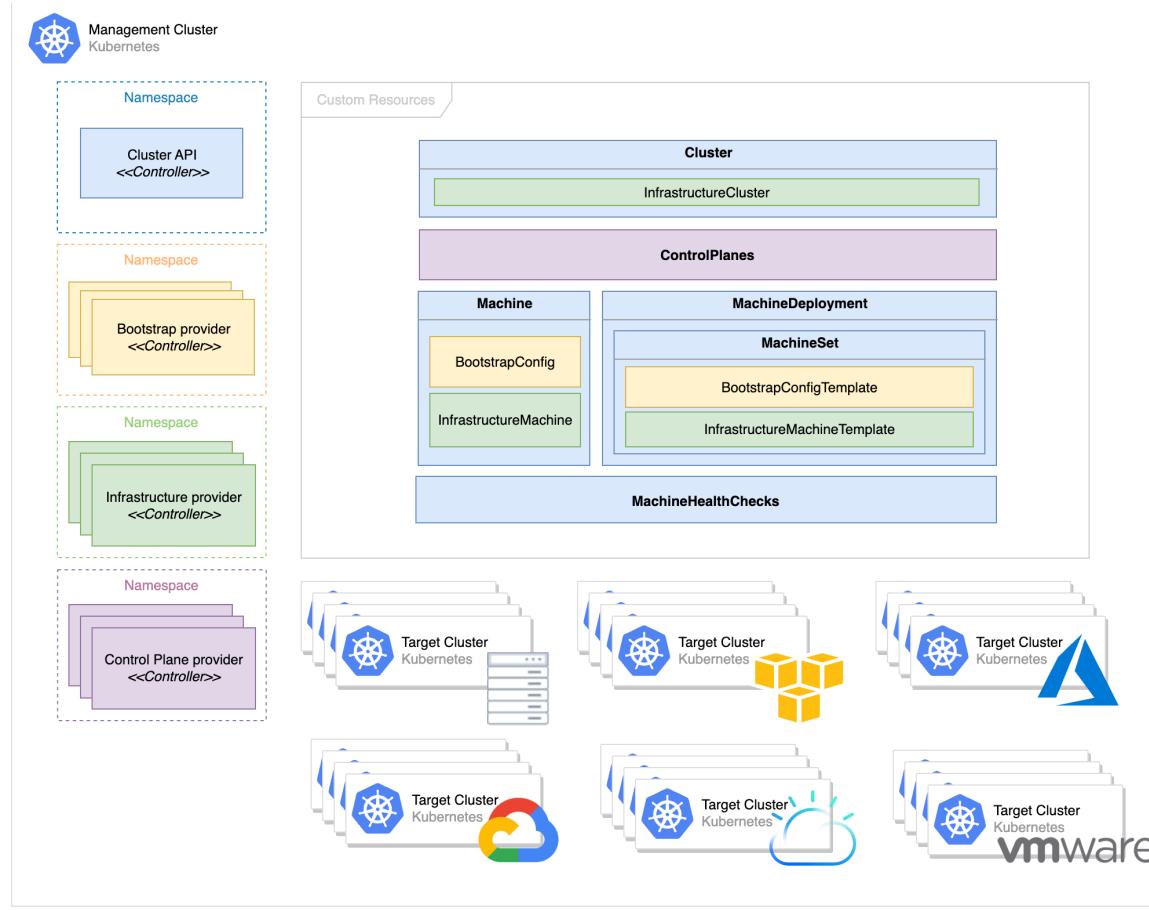
Pluggable Interfaces

- **Infrastructure Provider**
 - Prepare underlying resources
 - LoadBalancer, VM, Baremetal
 - Example: OpenStack Provider
- **Bootstrap Provider**
 - Turn Machines into k8s Nodes
 - Cloud-Init, Ignition
 - Example: kubeadm
- **Control Plane Provider**
 - Instantiate Kubernetes Control Plane
 - Example: kubeadm



<https://cluster-api.sigs.k8s.io/user/concepts>

Choose Providers for Verda



Pluggable Interfaces

- **Infrastructure Provider**
→ Cluster API Provider OpenStack (CAPO) ...?
- **Bootstrap Provider**
→ Cluster API Bootstrap Provider Kubeadm (CABPK)
- **Control Plane Provider**
→ Cluster API Control Plane Provider Kubeadm (CACPK)

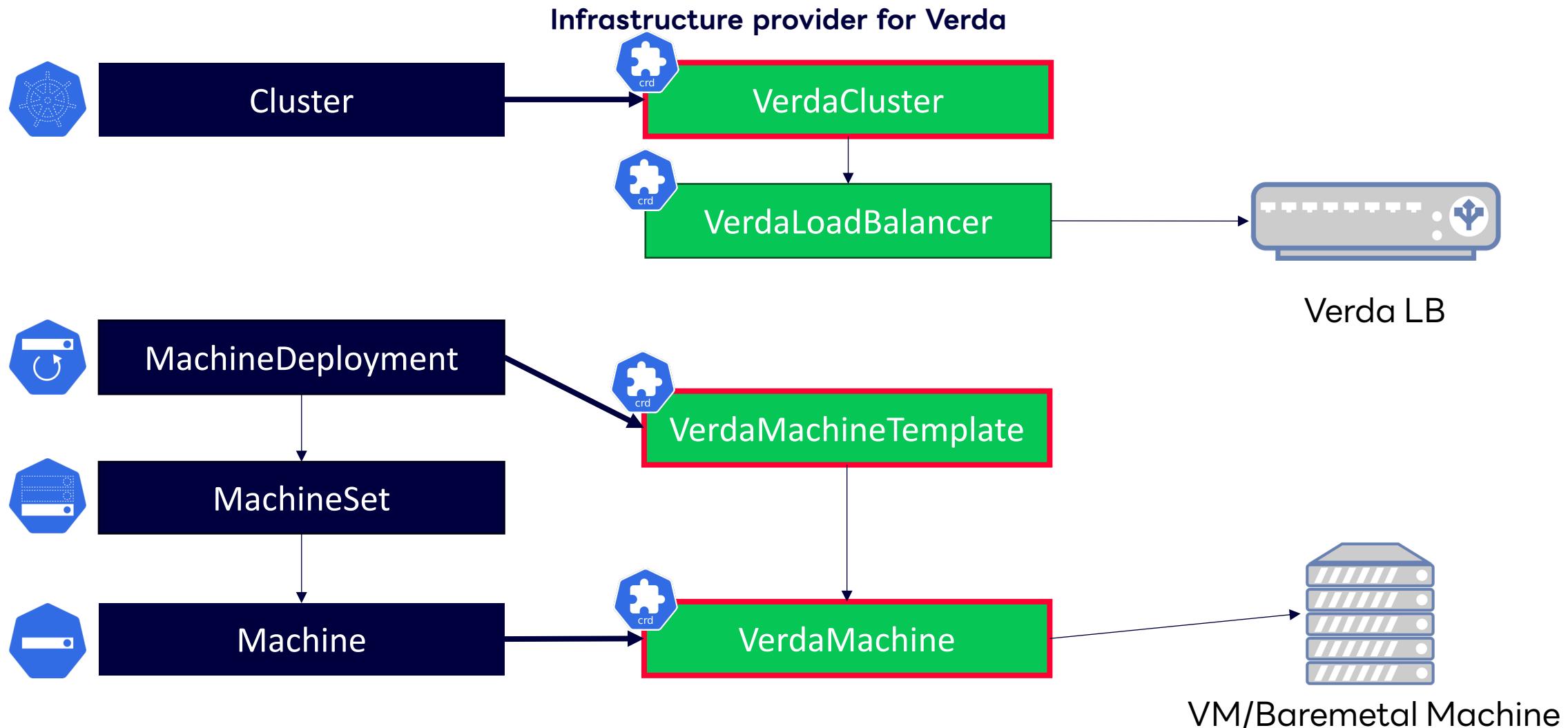


<https://cluster-api.sigs.k8s.io/user/concepts>

Potential Blocker for CAPO

- Verda doesn't meet the requirements of CAPO
 - Customized IaaS API
 - Own LBaaS API (!= OpenStack Octavia API)
 - Cannot use kubernetes-sigs/image-builder directly

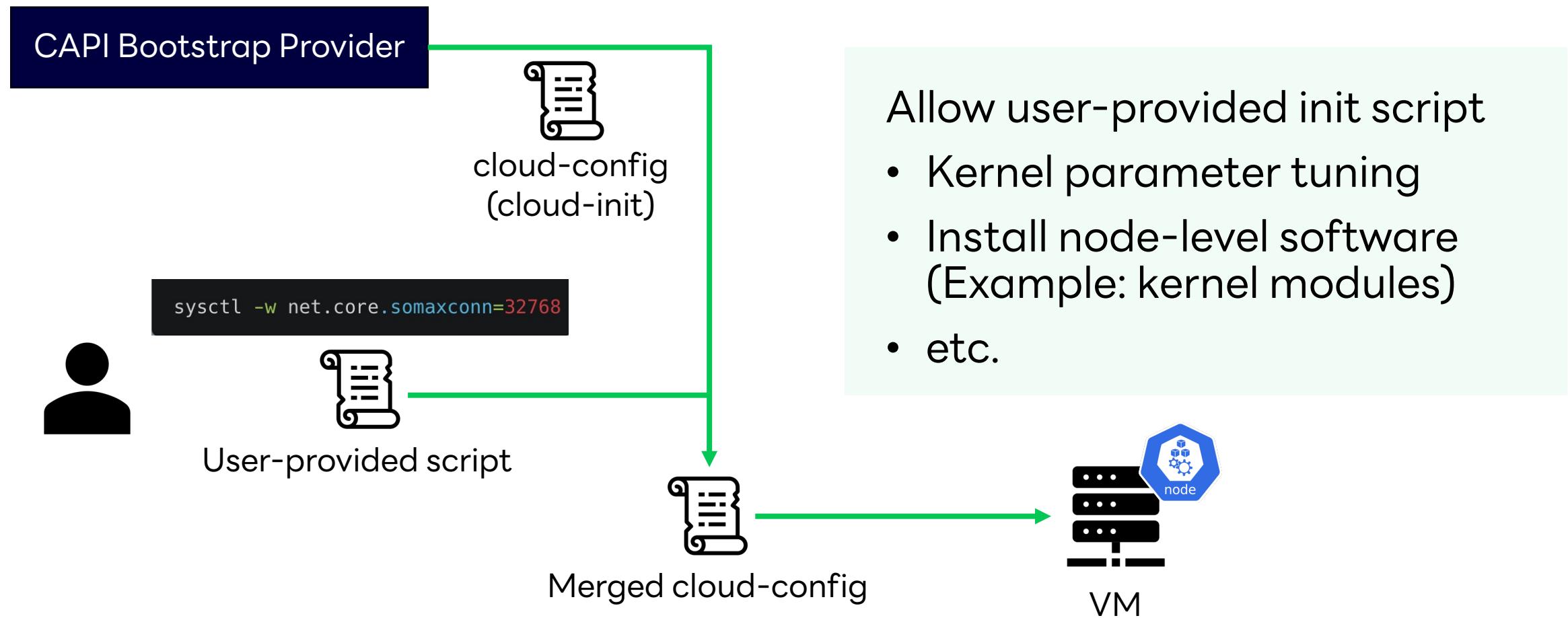
Cluster API Provider Verda (CAPV)



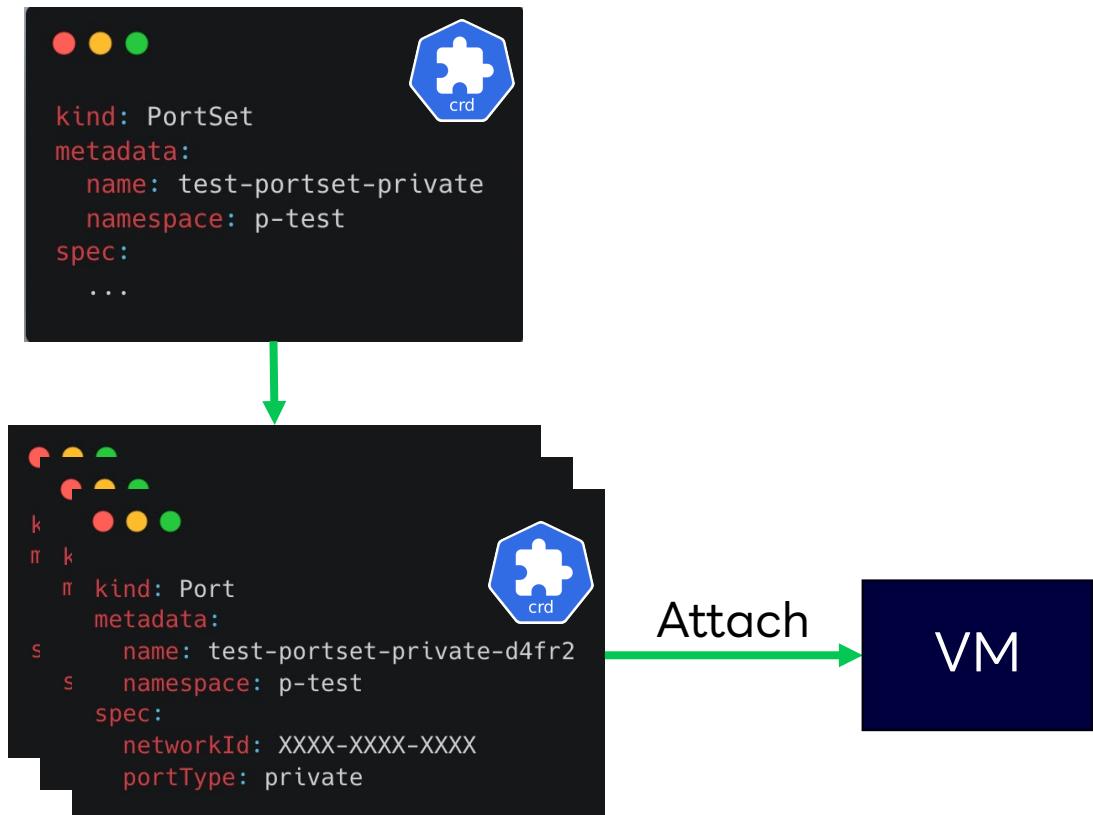
Custom Features of CAPV

- User Script
- Static IP Node Pool
- etc.

Custom feature: User Script



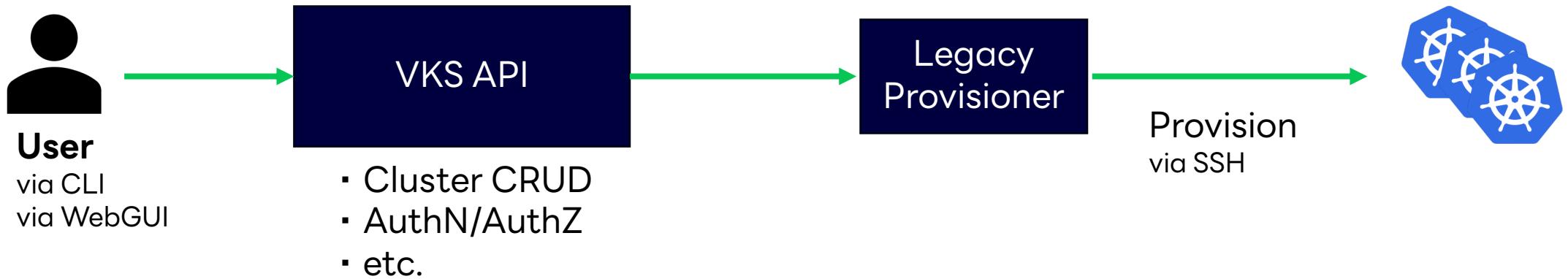
Custom feature: Static IP Node Pool



- Reuse OpenStack Neutron Ports to assign reserved fixed IPs
 - New Custom Resource to manage Neutron Ports
- Useful for IP ACLs
 - Having the same IP even recreating nodes

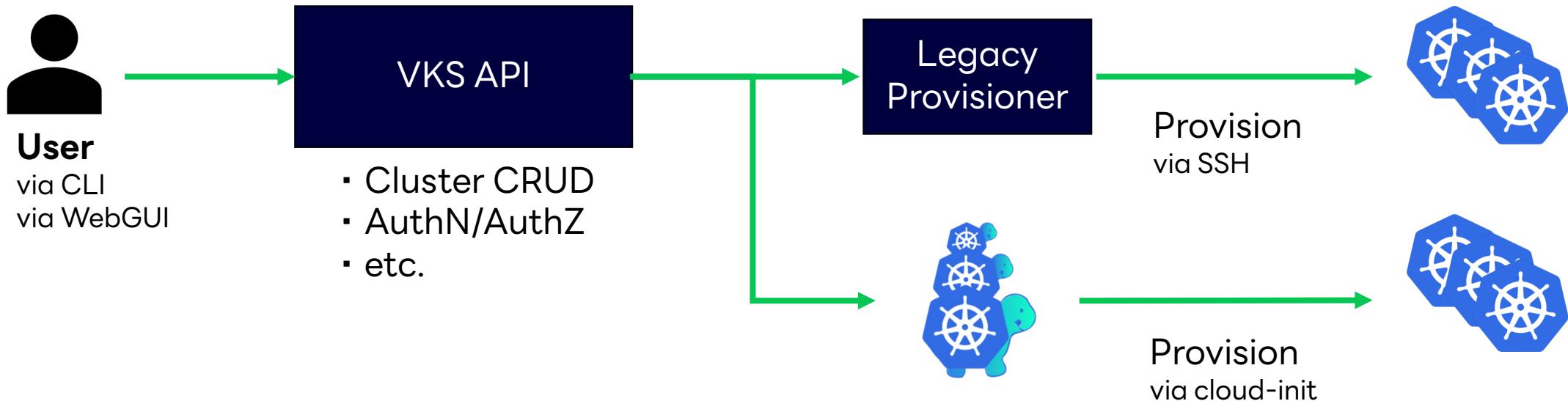
Overview of Legacy (Recap)

As of 2022



Same Interface, Same Experience

Transparent thanks to the VKS API in front of cluster provisioners



Agenda

01 Overview of our Platform

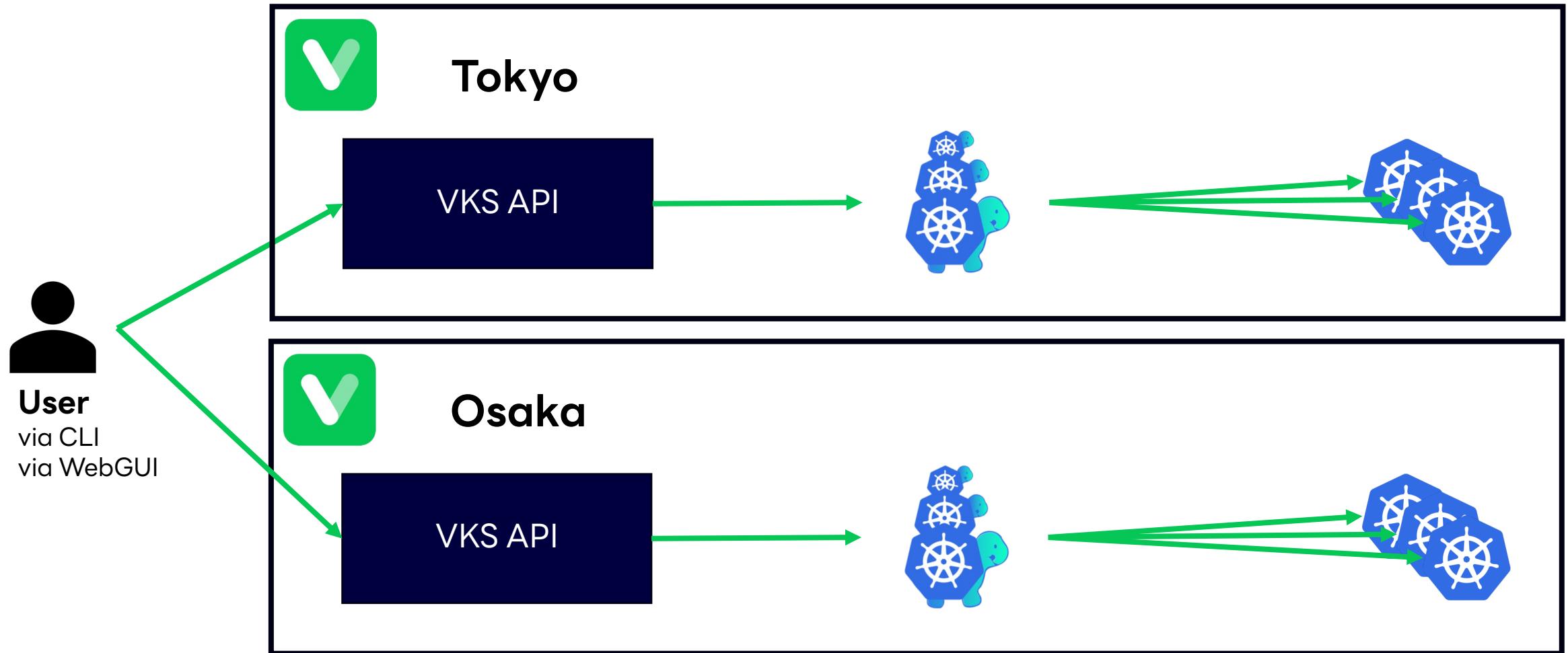
02 Adoption of Cluster API

03 Multi-Region and Multi-AZ with Cluster API

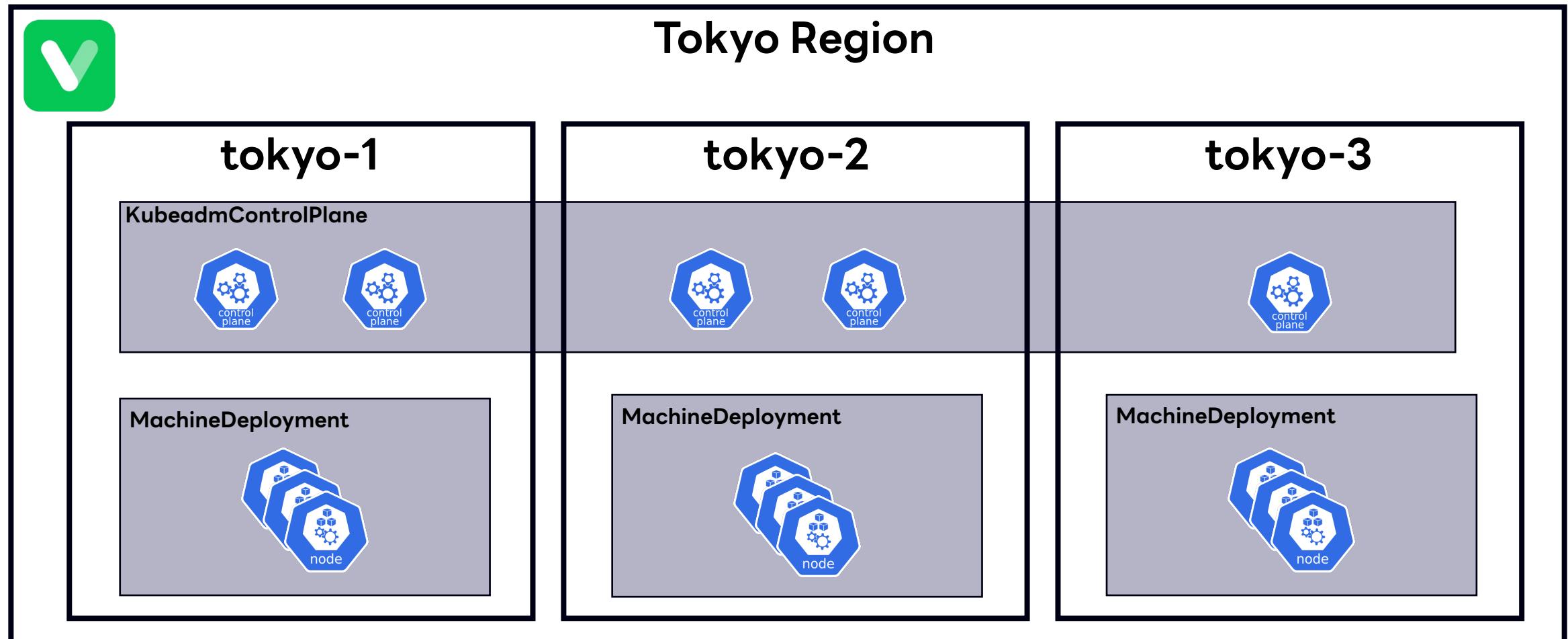
04 In-place Migration to Cluster API

05 Obstacles and Insights

Multi Region



Multi AZ



CAPI is Multi-AZ friendly

```
● ● ●  
kind: VerdaCluster  
metadata:  
  name: test  
  namespace: p-test  
spec:  
...  
failureDomains:  
  tokyo-1:  
    controlPlane: true  
  tokyo-2:  
    controlPlane: true  
  tokyo-3:  
    controlPlane: true  
  ...
```

Propagate

```
● ● ●  
kind: Cluster  
metadata:  
  name: test  
  namespace: p-test  
spec:  
...  
status:  
...  
failureDomains:  
  tokyo-1:  
    controlPlane: true  
  tokyo-2:  
    controlPlane: true  
  tokyo-3:  
    controlPlane: true  
  ...
```

One picked

```
● ● ●  
kind: Machine  
metadata:  
  name: test-cp-2v65h  
  namespace: p-test  
spec:  
...  
failureDomain: tokyo-1  
...  
● ● ●  
kind: VerdaMachine  
metadata:  
  name: test-cp-2v65h  
  namespace: p-test  
spec:  
...  
availabilityZone: tokyo-1  
...
```

Agenda

01 Overview of our Platform

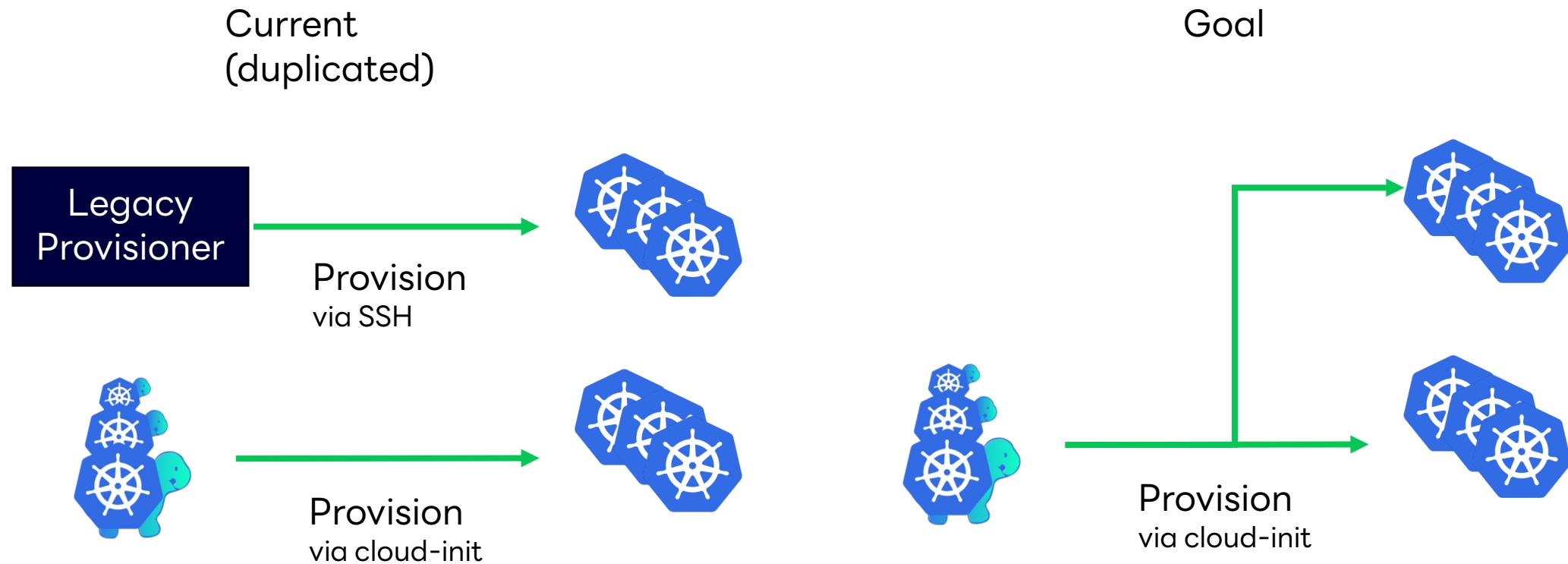
02 Adoption of Cluster API

03 Multi-Region and Multi-AZ with Cluster API

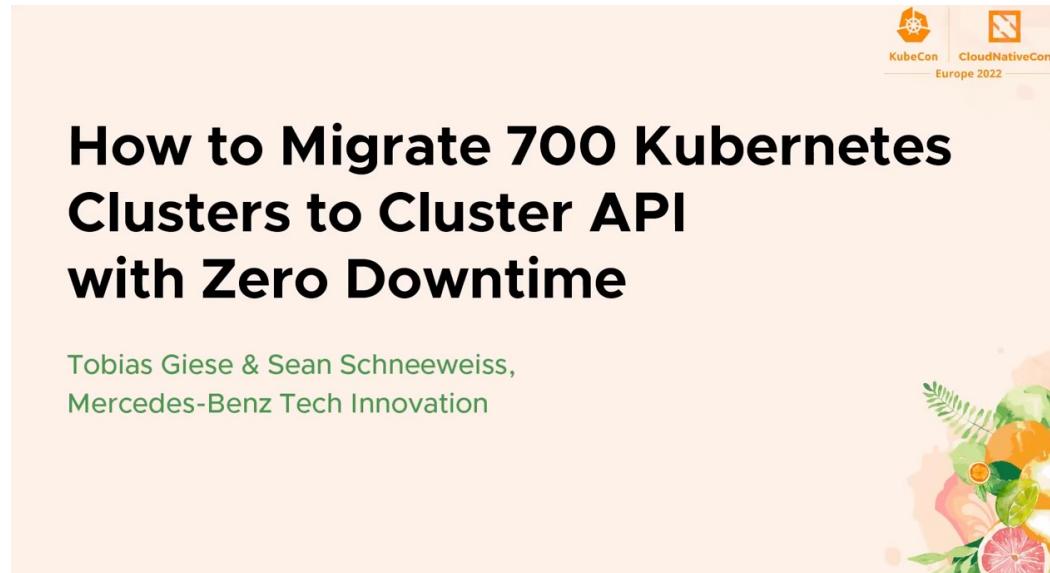
04 In-place Migration to Cluster API

05 Obstacles and Insights

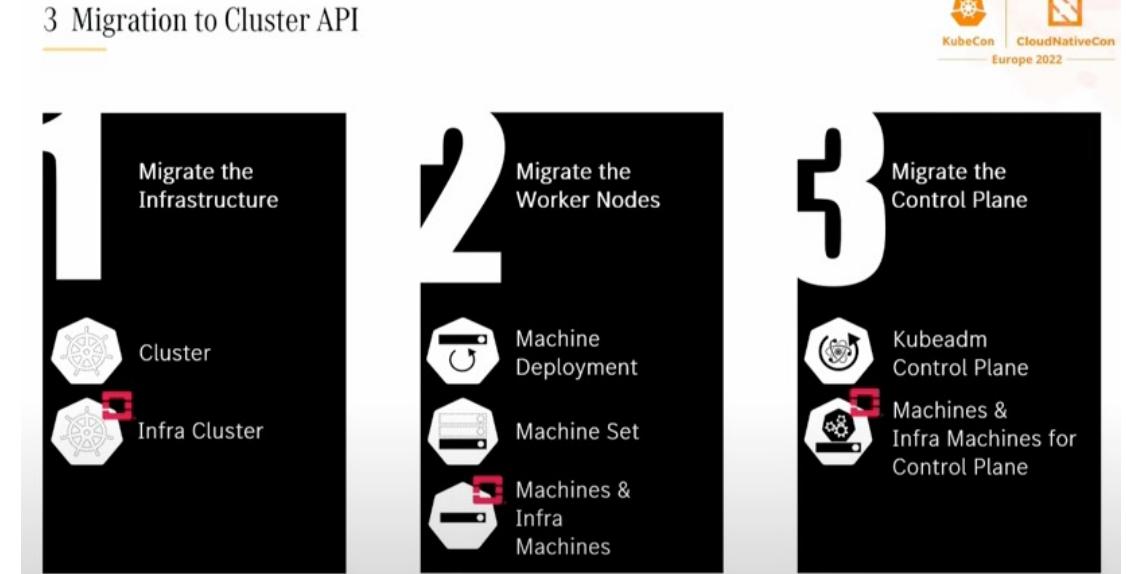
Goal: Deprecating Legacy Provisioner



Cluster Migration to Cluster API



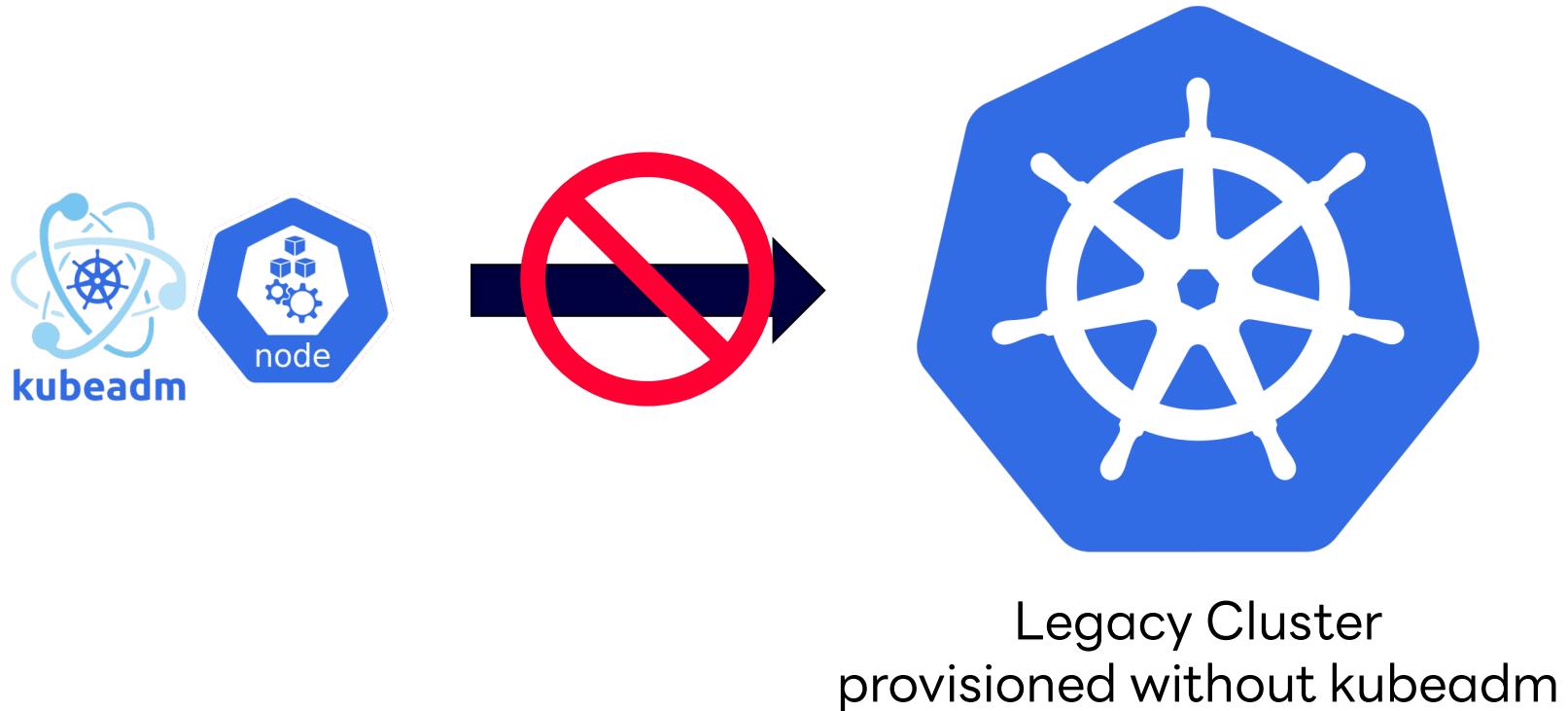
3 Migration to Cluster API



Kubecon EU 2022: How to Migrate 700 Kubernetes Clusters to Cluster API with Zero Downtime
https://www.youtube.com/watch?v=KzYV-fJ_wH0

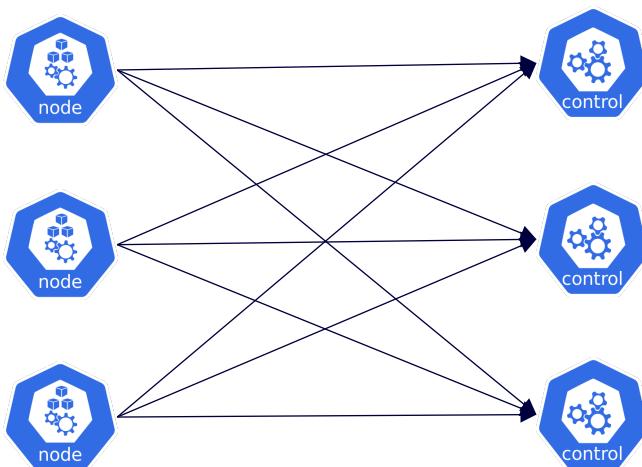
Problem: Node installer inconsistency

Legacy Cluster doesn't accept Kubeadm Nodes.

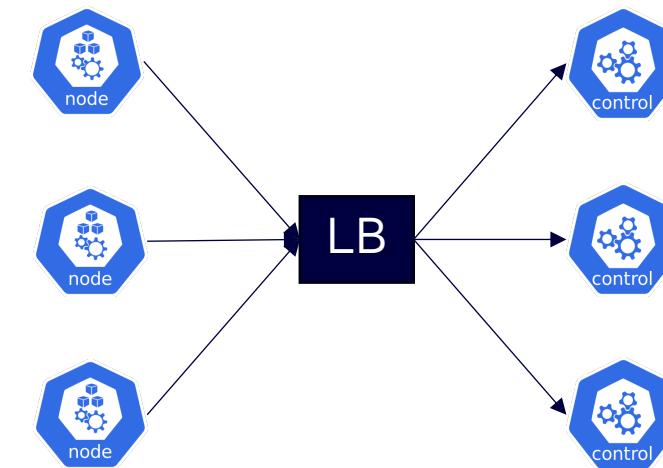


GAP 1: API Request Load Balancing

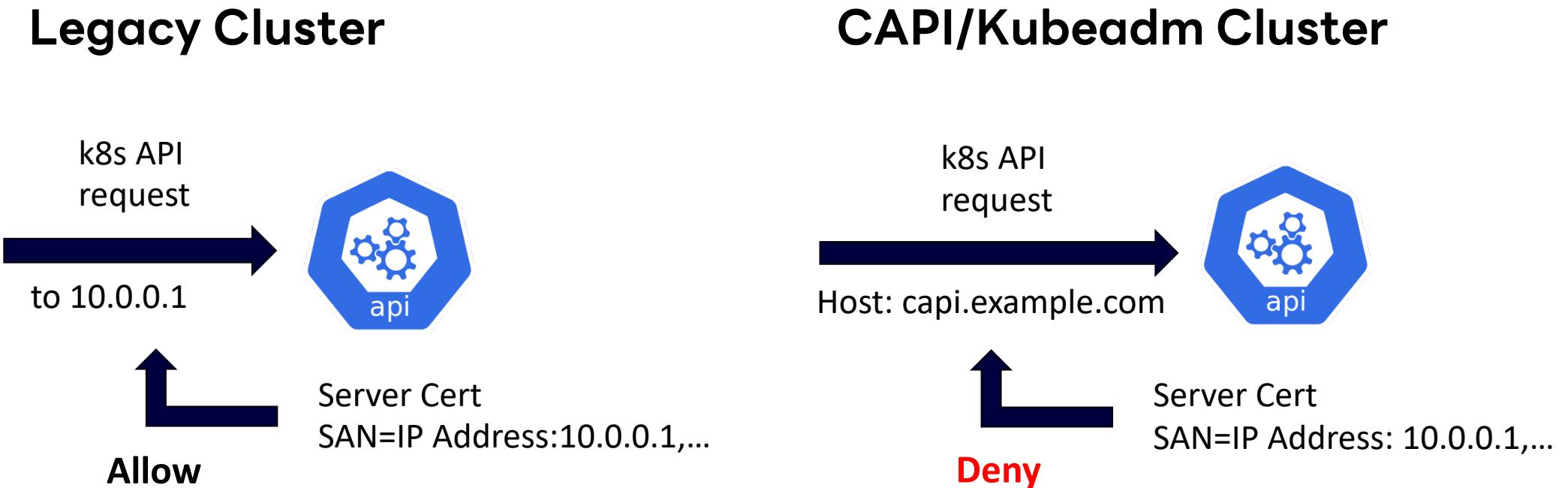
Legacy Cluster



CAPI/Kubeadm Cluster



Gap 2: SAN in Server Cert



Gap 3: Config Management



```
kind: ConfigMap
metadata:
  namespace: kube-system
  name: kubeadm-config
data:
  ClusterConfiguration: |
    apiVersion: kubeadm.k8s.io/v1beta3
    kind: ClusterConfiguration
    clusterName: mycluster
    controlPlaneEndpoint: mycluster.example.com:443
  etcd:
    local:
      dataDir: /var/lib/etcd
  kubernetesVersion: v1.26.1
```

- Kubeadm manage configs with ConfigMaps.
 - kubeadm-config
 - kubelet-config
 - kube-proxy
 - ...

Gap 4: Etcd member discovery

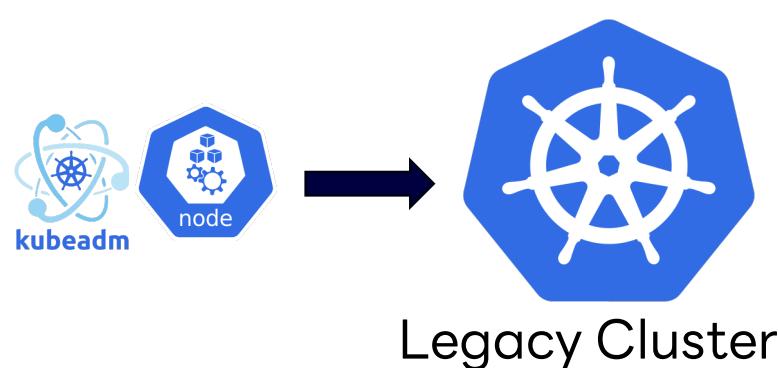


```
kind: Pod
metadata:
  namespace: kube-system
  name: etcd-mycluster-cp-2x58s
  labels:
    component: etcd
    tier: control-plane
```

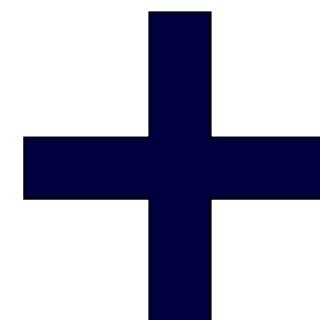
- Kubeadm get etcd member IPs by listing pod
 - component=etcd, tier=control-plane
- This issue was handled with **Dummy Pod**.

How to support migration to CAPI

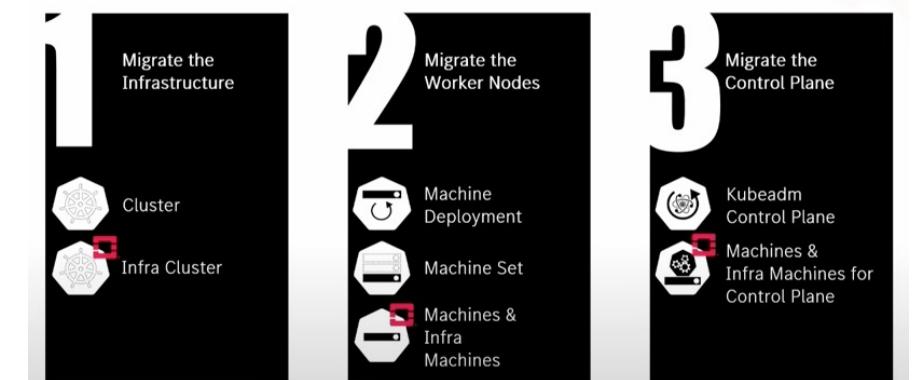
Modify Legacy Cluster for kubeadm



Handle Cluster API Custom Resources



3 Migration to Cluster API



How to Migrate 700 Kubernetes Clusters to Cluster API with Zero Downtime
https://www.youtube.com/watch?v=KzYV-fJ_wH0

Agenda

01 Overview of our Platform

02 Adoption of Cluster API

03 Multi-Region and Multi-AZ with Cluster API

04 In-place Migration to Cluster API

05 Obstacles and Insights

Obstacles and Insights

- Controller Scalability
 - Etcd Snapshot/Restore
- ...and more

Obstacles and Insights

- Controller Scalability
- Etcd Snapshot/Restore

...and more

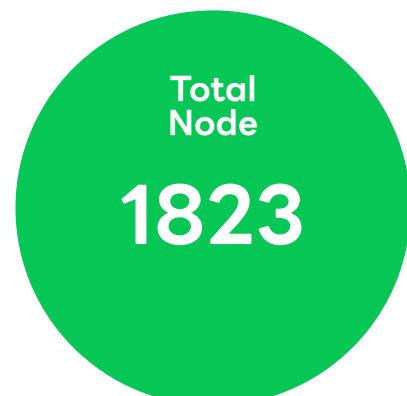
CAPI Controller Scalability



Verda Prod



Verda Dev



CAPI Controller Scalability

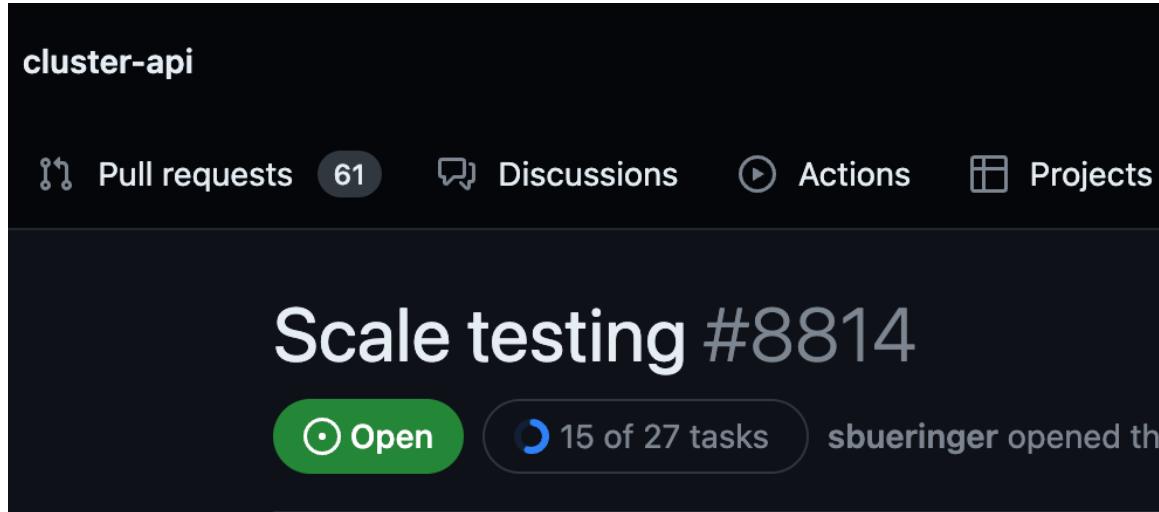


- Constantly Long Workqueue
- Any operation takes time
 - Cluster creation
 - Node provision



- Scale-Up controllers
+ Increase Concurrency
- Scale-Out(Sharding) controllers
→ leader-follower architecture makes it hard

CAPI Controller Scalability



e2e test and test framework:

- Implement scale test automation:
 - Cluster topologies:
 - Small workload cluster: x * (1 control-plane + 1 worker node)
 - Small medium workload cluster: x * (3 control-plane + 10 worker node)
 - Medium Workload Cluster: x * (3 control-plane + 50 worker nodes)
 - Large Workload Cluster: x * (3 control-plane + 500 worker nodes)
 - Dimensions: # of MachineDeployments

- Scalability of CAPI is one of our interests.
- E2E for ensuring scalability seems to be now ongoing on the upstream.

Our Target:
Max 1000 nodes / cluster

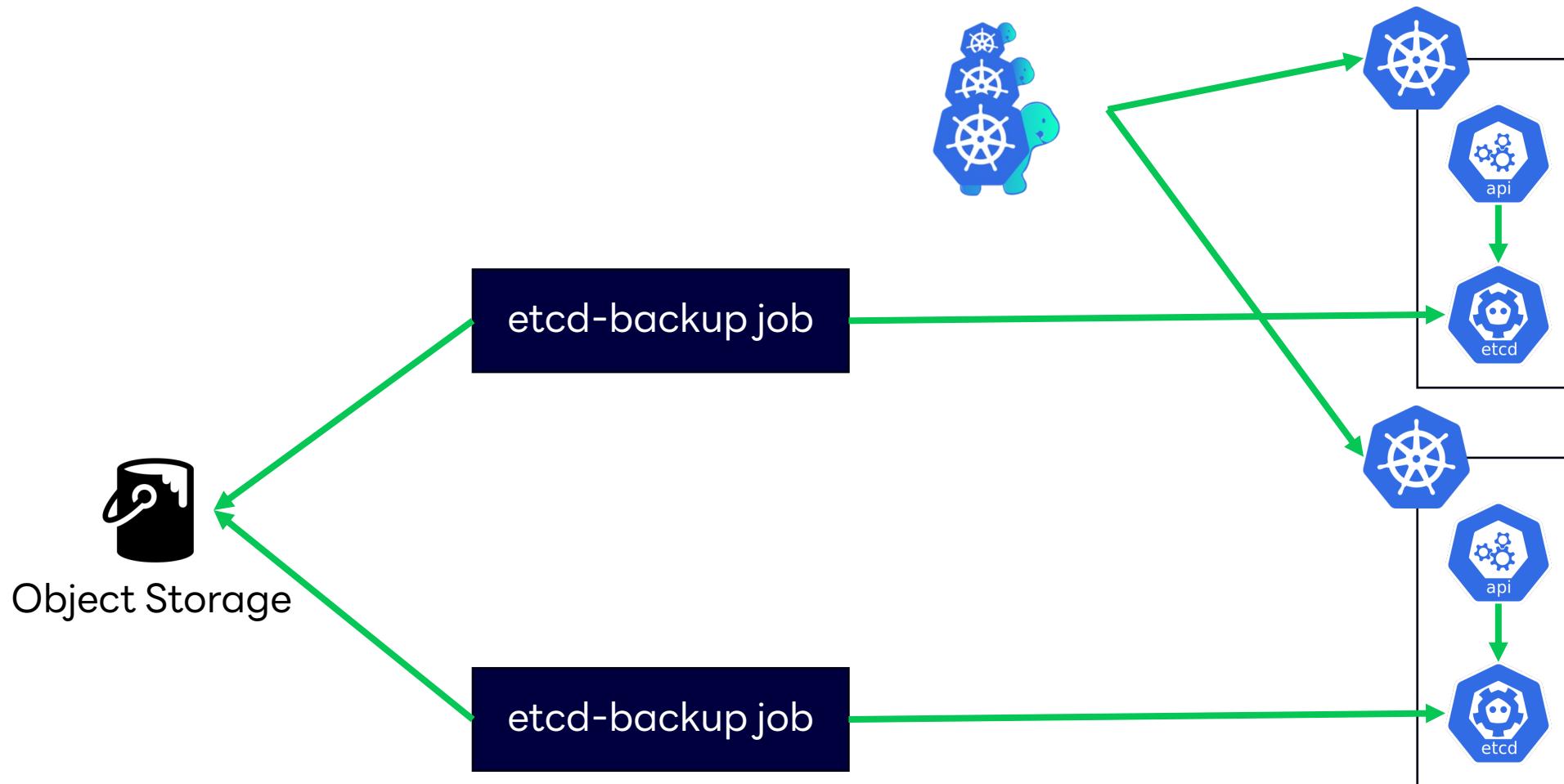
Obstacles and Insights

- Controller Scalability
- Etcd Snapshot/Restore

...and more

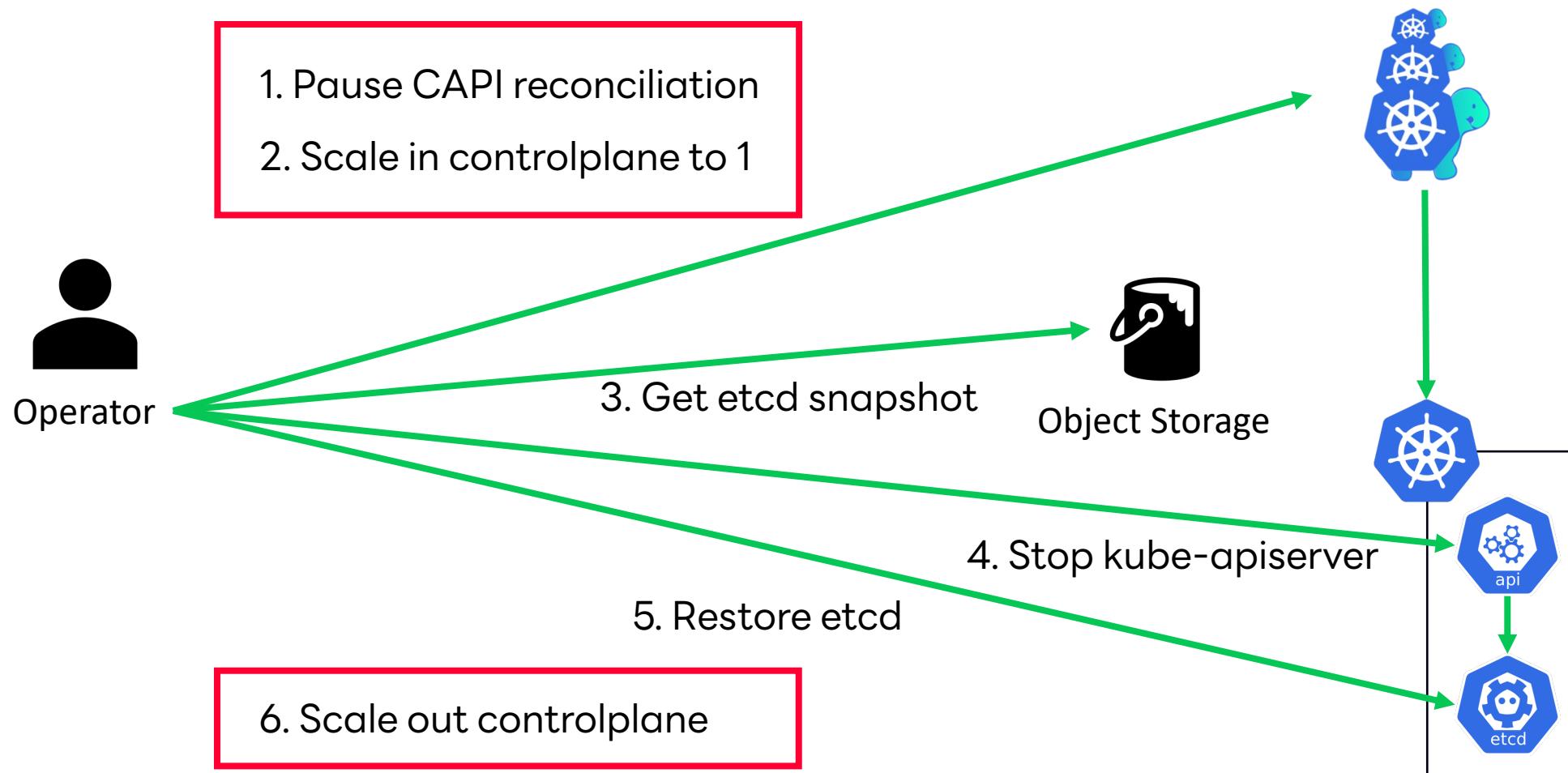
Etcd Backup/Restore

Automation of Backup is easy, but…



Etcd Backup/Restore

Done by Manual Operation for now



Etcd Backup/Restore

ETCD snapshot/restore support #7796

Open musaprg opened this issue on Dec 22, 2022 · 10 comments

 musaprg commented on Dec 22, 2022 · edited

User Story

As an operator, I'd like to maintain the etcd snapshot/restore functionalities with Cluster API (KubeadmControlPlane).

Detailed Description

The etcd snapshot and restore are usually crucial for administrators. We can achieve those functionalities by using community-provided operators (e.g., `etcd-operator`, which is already archived though...) or `etcdctl` directly. However, sometimes restore tasks should be considered as one of the cluster lifecycles since it requires to stop/start kube-apiserver before/after restoring. It would be nice to provide the etcd snapshot/restore functionalities by the CAPI side so that we can easily maintain them.

(I couldn't find any discussions related to this except for [#7399](#), so I filed this topic as a new issue. Please let me know if there are any places where we already have this kind of discussion.)

Anything else you would like to add:

(TBD)

Related Issues/PRs

- [External etcd lifecycle support #7399](#)
- [CAEP to add support for managed external etcd clusters in CAPI #4659](#)

/kind feature

+ Add tasklist

Unsubscribe

Customize

No one assigned

Labels

kind/feature kind/proposal triage/accepted

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

Notifications

You're receiving notifications because you were assigned.

7 participants

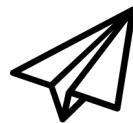
Summary

- **Cluster API is great for Private Cloud Platform**
 - Minimum code management cost
 - Extensibility for custom features
 - Multi-regional support with FailureDomain
- **Minimize various costs for cluster operators**
 - Consistent UI/UX leveraging API abstraction
 - In-place migration working with Cluster API & kubeadm

Thank you!



Kubernetes: @musaprg OR @sgotnd



Email:

inoue.kotaro@lycorp.co.jp
shotaro.gotanda@lycorp.co.jp



Feel free to ask us after this session



PromCon
North America 2021

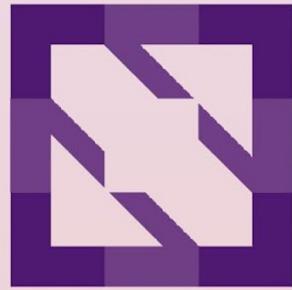


Please scan the QR Code above
to leave feedback on this session

LY



KubeCon



CloudNativeCon

North America 2023