

# Production-grade, low-footprint Kubernetes for the Edge



Konstantinos Tsakalozos  
Staff engineer



Yalton Ruiz  
Engineering Manager

ubuntu<sup>®</sup>

CANONICAL



MicroK8s



# MicroK8s

**2018**

First lightweight  
K8s distribution

**115K**

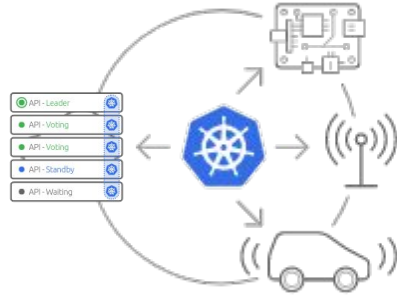
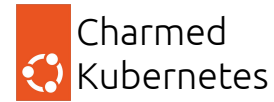
Running clusters  
around the globe

**24h**

Releasing 24  
hours after CNCF



Pioneers in  
Cluster API



- **Lightweight** K8s for resource-constrained environment
- All **Kubernetes services** and most popular addons
- Zero ops for easy clustering
- **Self-healing with HA**
- **Automatic upgrades**
- Secured by default, strict confinement, and long term support
- Compatible with **Linux, Window and macOS**

- Configurable & multi-cloud
- Pluggable CNI, CSI, CRI and monitoring components
- Model-driven Kubernetes for fully customisable deployments
- Carrier-grade and hardware accelerations features
- Support for third-party components and services
- Full life cycle management for host and in-cluster with Juju

# IoT/Edge challenges



Out-of-the-box Security



Reliability

Low-touch



Cost reduction

CapEx + OpEx



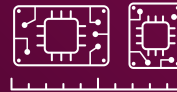
Scale PoC to Product

Time and resources



Growing complexity

and skill (and time) shortage



Hardware resources

and compatibility

# IoT/Edge challenges



## Out-of-the-box Security



## Reliability

Low-touch



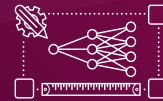
## Cost reduction

CapEx + OpEx



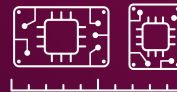
## Scale PoC to Product

Time and resources



## Growing complexity

and skill (and time) shortage



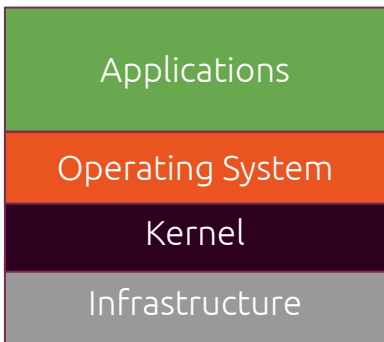
## Hardware resources

and compatibility



# Out of the box security

Each layer needs to be hardened



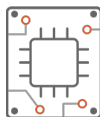
FIPS conformant MicroK8s



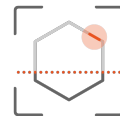
*MicroK8s CIS compliance in the roadmap (1.28)*



Trivy Operator addon for security assessments



Immutability with Ubuntu Core

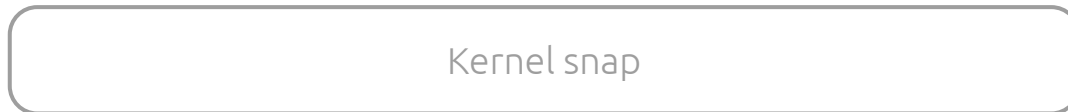
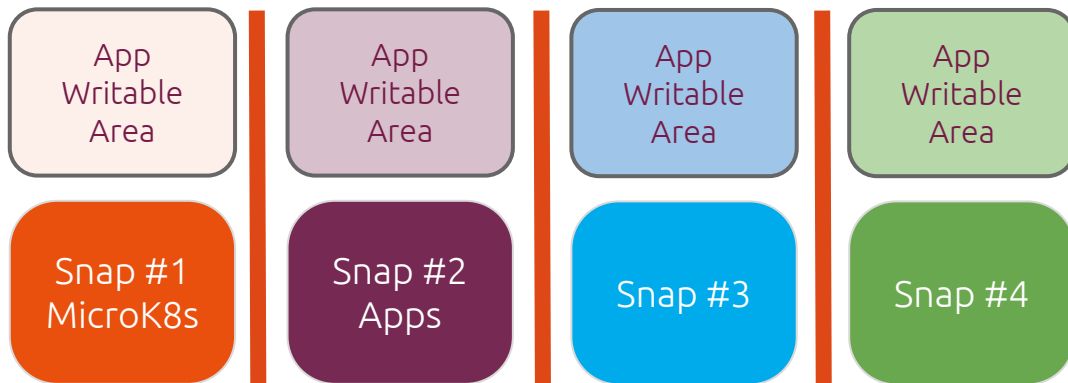


Strict confinement on MicroK8s

# Strict confinement

Fully contained,  
immutable  
packages  
for OS, K8s  
and apps

## Strict Confinement





# MicroK8s channels (1.27)

## Kubernetes 1.27 Classic

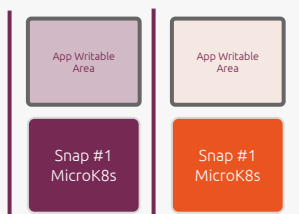
All upstream Kubernetes 1.27 features are available in classic deployment.

All MicroK8s addons

## Kubernetes 1.27 - Strict

All upstream Kubernetes 1.27 features are available in strict deployment.

However due to the security isolation provided by snap, some features might experience restrictions. [More information.](#)



## EKS-D 1.25

The EKS-D channels package includes addons for the specific AWS resources that integrate with Kubernetes. Deploy Amazon Kubernetes on prem with MicroK8s and use AWS addons integrated out of the box:

These addons are;

- IAM Authenticator
- Elastic Block Storage CSI Driver
- Elastic File System CSI Driver

[More information](#)

# IoT/Edge challenges



Out-of-the-box Security



**Reliability**

Low-touch



Cost reduction

CapEx + OpEx



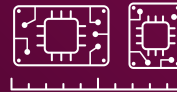
Scale PoC to Product

Time and resources



Growing complexity

and skill (and time) shortage



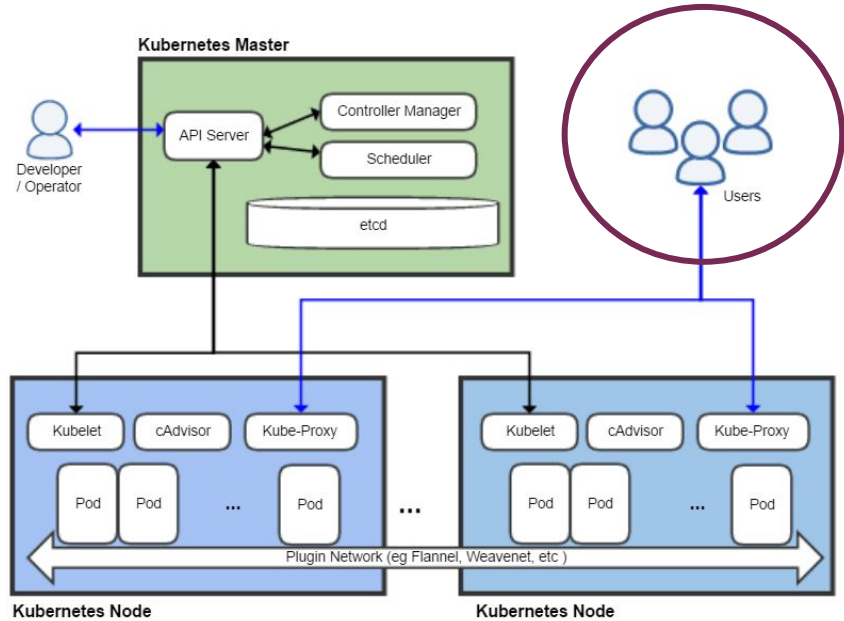
Hardware resources

and compatibility

# High Availability - Users Perspective

For users

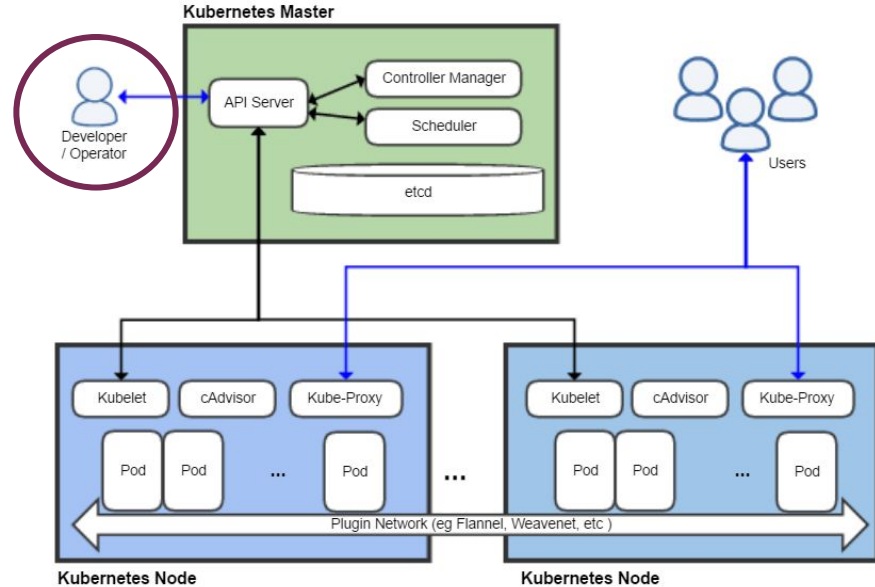
- Services are always available



# High Availability - Admins Perspective

## For admins

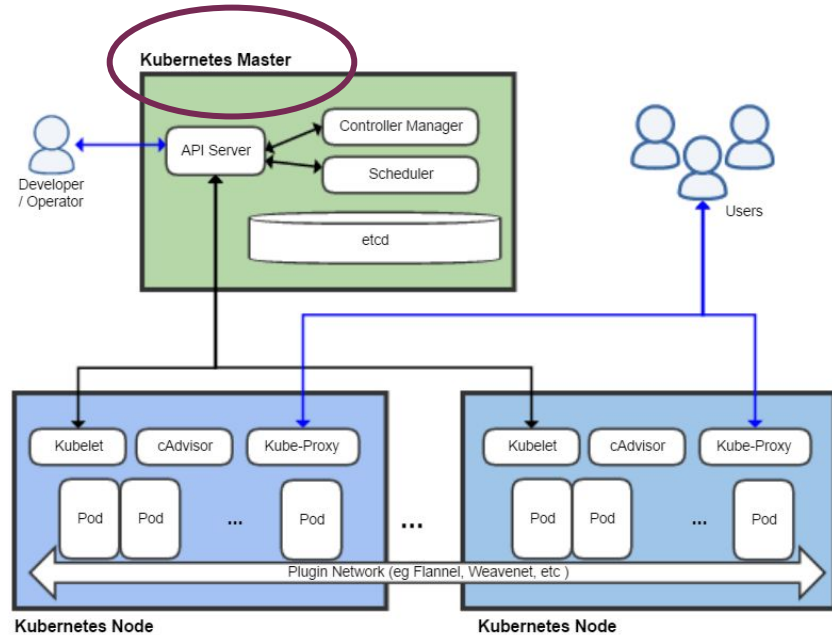
- Control plane is always available
- More than one nodes
- Workloads spread across nodes
- Reliable persistent storage
- Load balancer configured

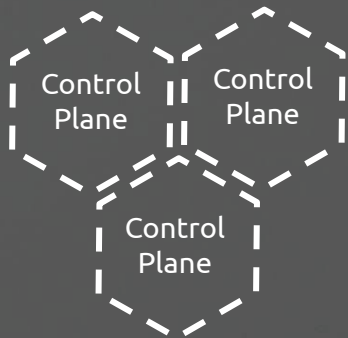


# High Availability - Kubernetes Perspective

## For Kubernetes itself

- Datastore is always available
- Clustering
- Persistent storage configured





# High Availability

With at least 3 control plane nodes

## Stop worrying about **Control Plane**

- Datastore embedded into the API server
- Dqlite: the most popular embedded database made distributed
- At least three nodes needed
- Replication: API server ↔ datastore

## Stop worrying about the **Workers**

- Every node is also a worker
  - API server replication ↔
  - Datastore replication ↔
  - Worker replication

# IoT/Edge challenges



Out-of-the-box Security



Reliability

Low-touch



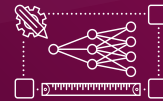
**Cost reduction**

CapEx + OpEx



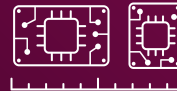
Scale PoC to Product

Time and resources



Growing complexity

and skill (and time) shortage



Hardware resources

and compatibility

# Cost Reduction

## Operations

Deployment time - Addons / Launch config  
Massive deployments - Juju & CAPI

## LCM

Upgrades - Juju or Cluster API  
Updates - Automatically via Snap

## HW&SW cost

Minimal HW footprint  
SW Free to consume

## Enterprise support

Ubuntu PRO (infra and apps)

- Security patching
- Support 24/7
- Compliance

WITH INFRA SUPPORT (24/7)

Support for Kubernetes, LXD

Support for MAAS

Support for Openstack

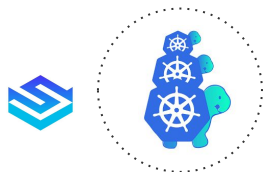
Cluster deployment



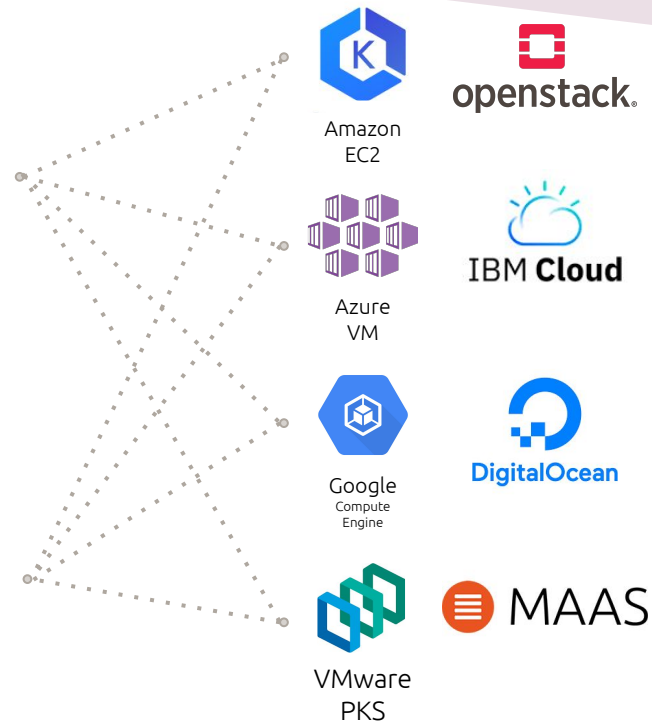
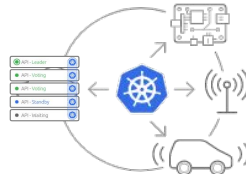
# Canonical Kubernetes solution



Install, maintain, upgrade, and integrate infrastructure and apps across clouds



Automate cluster lifecycle management



2 Provisioning and LCM

2 K8s distributions

+20 Infrastructure providers

# Comprehensive Kubernetes ecosystem



# +40

## Addons

```
> microk8s enable <add-on>
```

### Core

Maintained & supported by Canonical

### Partners

Maintained & supported by Partners of Canonical

### Community

Maintained & supported by OSS Community



# Edge features - MicroK8s init (1.27)

## 2 ways to load configuration and images to MicroK8s cluster

### 1. During cluster deployment

#### (Classic and Strict)

Configuration and side load images

Automate cluster creation with desired configuration

Example dns, ingress, rbac, hostpath-storage and registry addons automatically



Initial configuration file  
/root/snap/microk8s/common/.  
microk8s.yaml

The configuration file will be picked up automatically during the execution of command

```
snap install microk8s
```

### 2. Using content snap

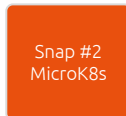
#### (Strict)

Configuration and side load images

Especially useful for large scale deployments and requires snap installation

Update MicroK8s configuration via content interface

Create new snap with MicroK8s configuration and images  
Content snap

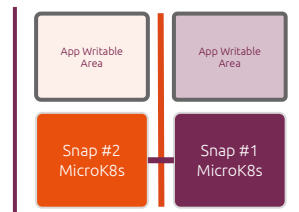


*Transfer snap to host*

*# install content snap from the store...*

```
sudo snap install content-demo-microk8s
```

```
sudo snap connect content-demo-microk8s:configuration microk8s
```



# IoT/Edge challenges



Out-of-the-box Security



Reliability

Low-touch



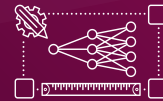
Cost reduction

CapEx + OpEx



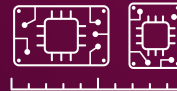
**Scale PoC to Product**

Time and resources



Growing complexity

and skill (and time) shortage



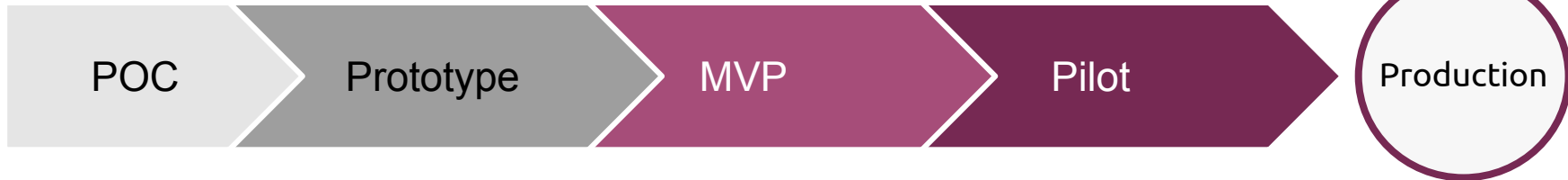
Hardware resources

and compatibility

# From PoC to production

Minimum HW  
Partial solution  
Days  
Freeware

HW optimization  
Full solution  
Years  
Scalability  
Stability  
Enterprise support



Lightweight Kubernetes  
Fast deployment and clustering  
All K8s services and APIs

Production grade  
Enterprise support  
Free

# IoT/Edge challenges



Out-of-the-box Security



Reliability

Low-touch



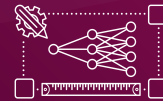
Cost reduction

CapEx + OpEx



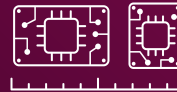
Scale PoC to Product

Time and resources



**Growing complexity**

and skill (and time) shortage



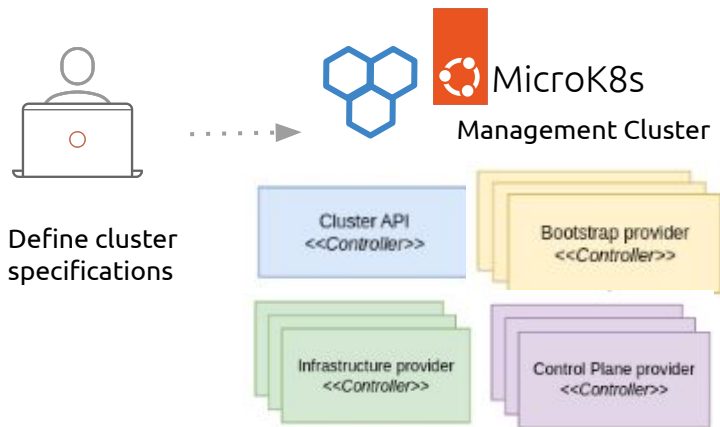
Hardware resources

and compatibility

# Edge features - CAPI (1.27)

## CAPI supported features

- Multi-cluster Lifecycle management
- Scaling
- Self-Healing
- Easy cluster upgrade
- Update and upgrade control



## Bootstrap

- Amazon Elastic Kubernetes Service (EKS)
- Kubeadm
- **MicroK8s**
- Talos

Responsible for generating a cloud-init script to turn a Machine into a Kubernetes Node using MicroK8s

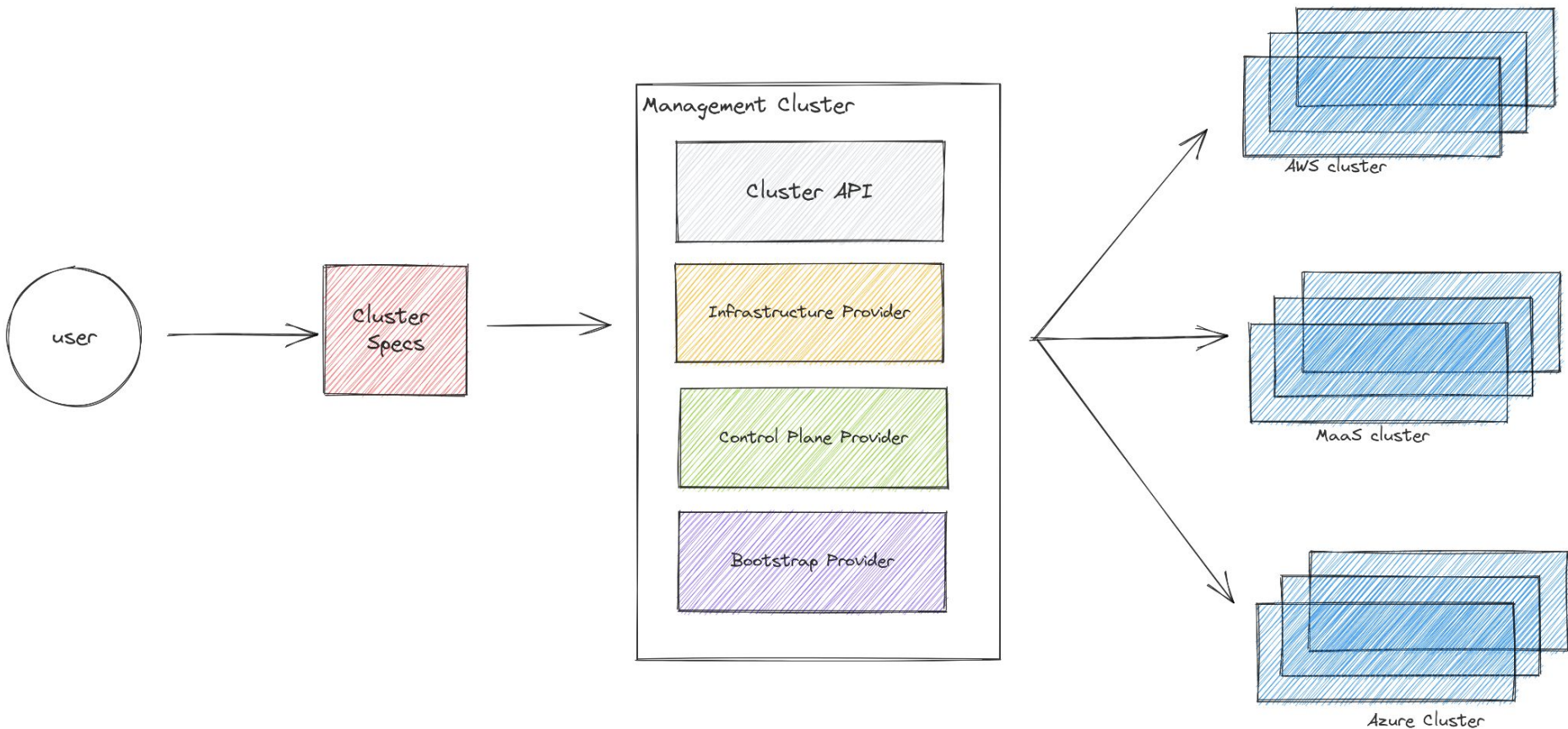
## Control Plane

- Kubeadm
- **MicroK8s**
- Nested
- Talos

Responsible for managing the control plane of the provisioned clusters using MicroK8s.



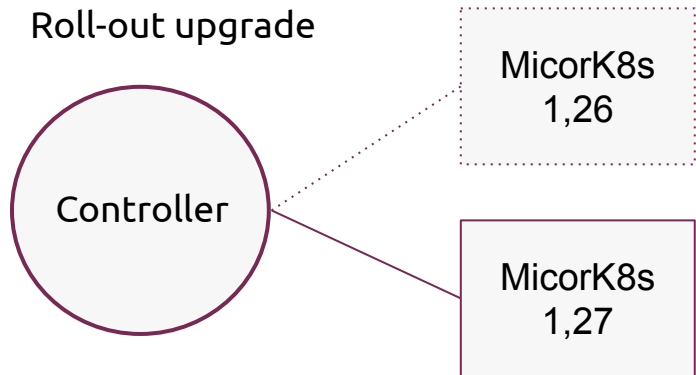
# Cluster API: Multi-cloud and Multi-cluster



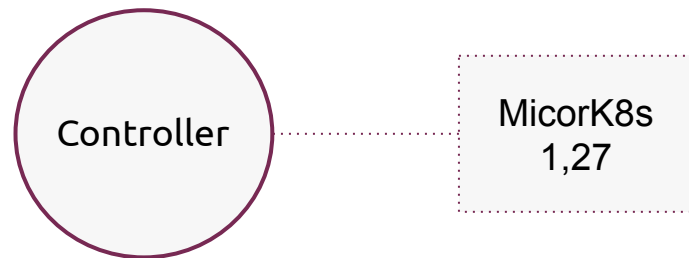


# Cluster upgrades

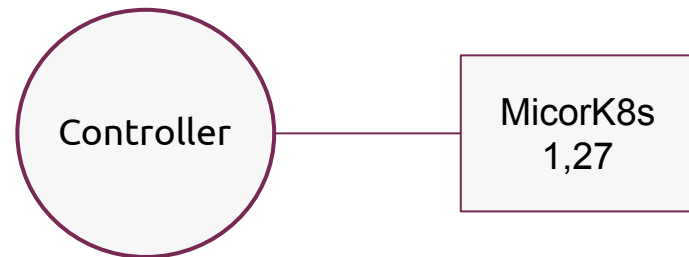
Roll-out upgrade



In-Place upgrade



*Change manifest*



# IoT/Edge challenges



Out-of-the-box Security



Reliability

Low-touch



Cost reduction

CapEx + OpEx



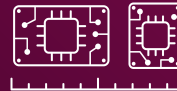
Scale PoC to Product

Time and resources



Growing complexity

and skill (and time) shortage



**Hardware resources**

and compatibility

 HW resources

### Host OS upgrade does not affect MicroK8s

Snap is packaging the application with all the dependencies

### MicroK8s minimal size:

1 CPU and 1G of memory

### New HW architecture

Easier to support new HW architectures as libraries are package in snap

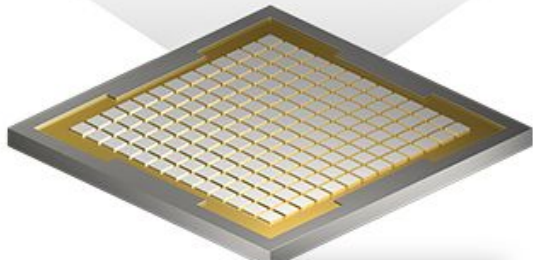
### MicroK8s recommended HW size:

2 CPUs and 4G of memory



# AI/ML with GPU operator & MIG

**NVIDIA**®



```
microk8s enable gpu
```

A GPU can be partitioned into different-sized MIG instances. **NVIDIA H100, A100, and A30** Tensor Core GPUs.

A portion of GPU can be dynamically allocated to different workloads.

## Advantages of MicroK8s support for MIG

- Easier configuration for K8s utilization ([More](#))
- Boosting AI workloads
- Guaranteed quality of service (QoS) for relevant workloads
- Better utilization of HW resources and acceleration.
- Parallel workloads without competing for resources

Demo time...

# High availability demo

## High availability demo

Link: <https://asciinema.org/a/u06P0CwWo00ED1CS7fSQdbdHA>

Time: 9 min

Scope: Get 3 MicroK8s nodes to

- go into High Availability
- show that the cluster can survive the removal of a single node

# Launch configurations & addon creation

## Launch configurations demo

Link: <https://asciinema.org/a/hhEFLxgQU8LjaPpEBqR4n9zAV>

Time: 7 min

scope : Enable ingress and metrics-server

## Addon creation demo

Link: <https://asciinema.org/a/qdazGXjjLzDzbCJWrEGvYyZA>

Time: 18

Scope: Create a simple “hello” bash addon

- Show how to add/remove add-ons
- Create a simple command

Thank you. Questions?



MicroK8s

Where to find us?

[microk8s.io](https://microk8s.io)

[ubuntu.com/kubernetes](https://ubuntu.com/kubernetes)

[github.com/canonical/microk8s](https://github.com/canonical/microk8s)

[meetup.com/microk8s-meetup-group/](https://meetup.com/microk8s-meetup-group/)

Slack MicroK8s

**Community synch**

Every Friday 1:30 GMT

<https://meet.google.com/hjy-uogt-tax>