



KCD - Melbourne - 2026

What Managed Kubernetes Hid and Why It Matters Now

Olga Mirensky

Senior Staff SRE - Ping Identity

Today I will take you on a journey

KUBERNETES HISTORY AND DEPLOYMENTS



2014

Kubernetes Open Sourced



Constrained IoT Devices
Planet Scale AI/ML clusters

OPERATIONAL COMPLEXITY SPECTRUM



Kubernetes
the Hard Way

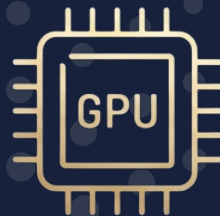


Kubernetes the Hard Way
the Easy Way



Managed
Kubernetes

DEEP DIVE



Local Inference on Kubernetes, Hybrid Clusters

Kubernetes at planet scale

Hyperscaler		Nodes in Single Cluster	GPUs / TPUs	Date	AI Workload
	GKE	65,000	250,000 TPUs	Nov 2024	
	GKE	130,000	Not verified	Nov 2025	
	EKS	100,000	800,000 NVIDIA GPUs	Jul 2025	Claude training
	AKS	5,000	Not verified	same as upstream k8s limit	
80 Clusters Fleet					
	AKS	70,000	140,000 CPU Cores GPU - unknown	Apr 2025	OpenAI

What made this possible

01 Persistent storage

Replace etcd with Spanner (GKE)
Smart watch cache for kubelets (GKE)
Key-Space partitioning & external journaling, replace Raft (EKS)
Dedicated Event State Stores - separate high frequency `Event`, `Lease`

02 Workload Aware Scheduling

Gang / All or Nothing scheduling
Topology, interconnect aware co-location
New SIGs - LeaderWorkerSet
High-throughput preemption engine, bypassing kube-scheduler

03 Container Runtime & Image Acceleration

Eliminate boot delays
Lazy image loading, image streaming
CDI - Container Device Interface
DRA - Dynamic Resource Allocation

04 High-Throughput Networking

eBPF data path $O(n) \rightarrow O(1)$ k8s svc
RDMA - Remote Direct Memory Access
Pre-assigned IP prefix delegation

05 Multi-Cluster Fleet Management

Fleet-Level workload placement
Blast radius isolation

Years of optimisation from hardware and networking to application layer.

Engineering blog posts on massive cluster milestones is a great source to learn systems scaling, not just Kubernetes



Infinite Application Versatility

The freedom to build, deploy and scale any workload across an endless variety of shapes, constraints and operational environments.

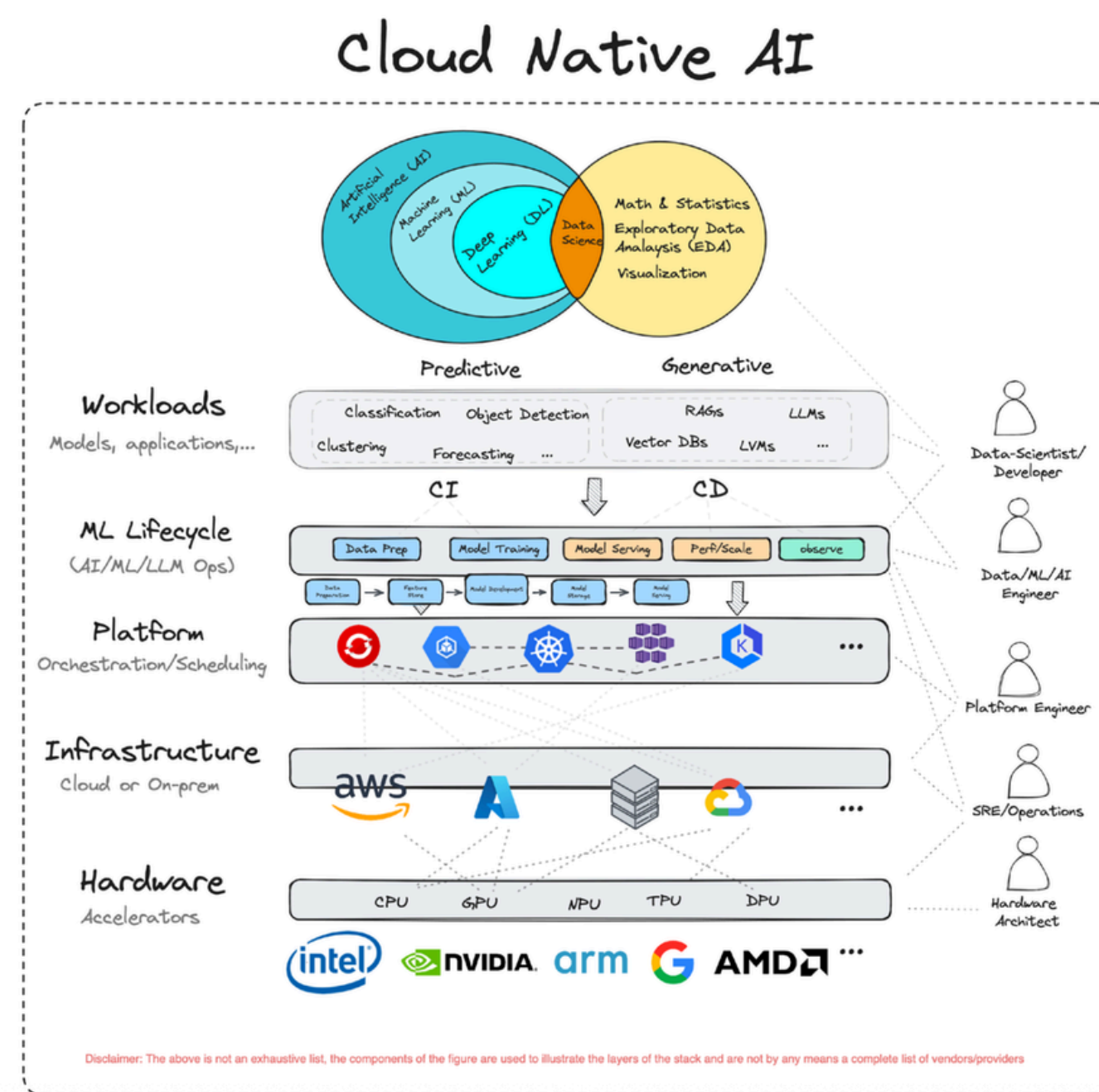
API Contract



Infrastructure

Hyperscalers, vendors and Kubernetes maintainers can re-engineer Kubernetes and the underlying infrastructure as long as the contract remains backward-compatible.

Cloud Native Artificial Intelligence



Any size, any shape, any challenge

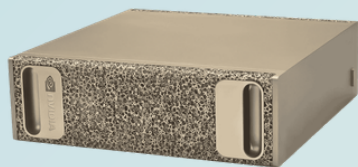
130,000 nodes (GKE) — 500-node enterprise — 3-node homelab — 1 GPU under a desk — Raspberry Pi, Robotics

Large scale AI
workloads
Frontier labs

Your typical enterprise
microservices architecture



And other form factors
for local inference

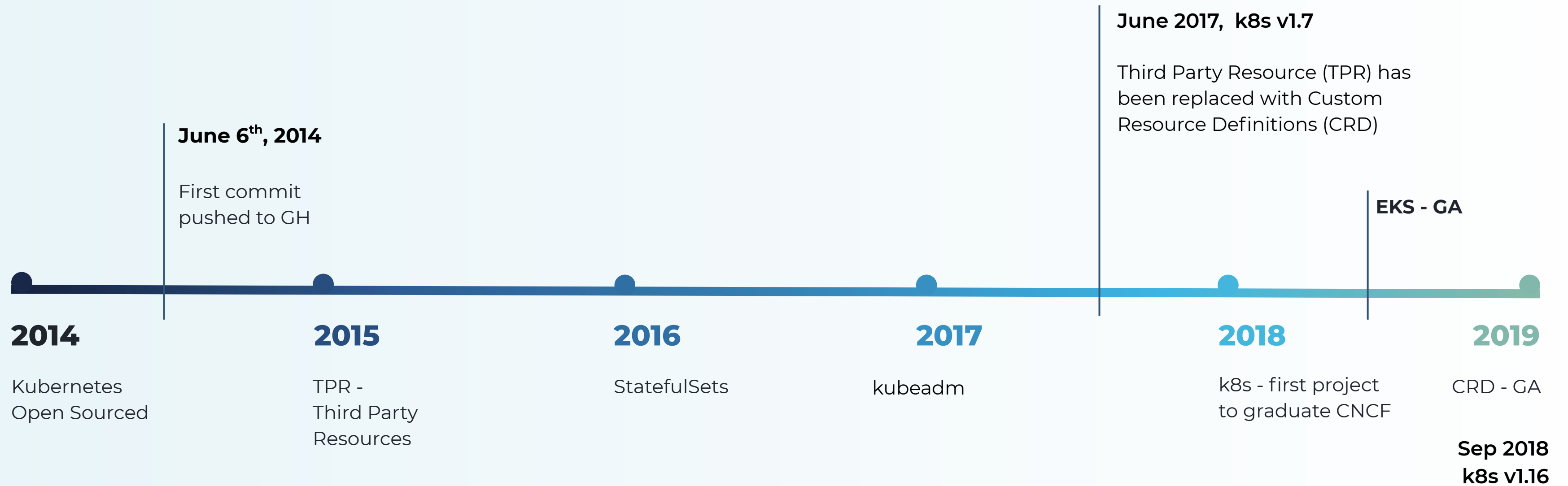


Offline autonomy,
unreliable network
airgapped envs

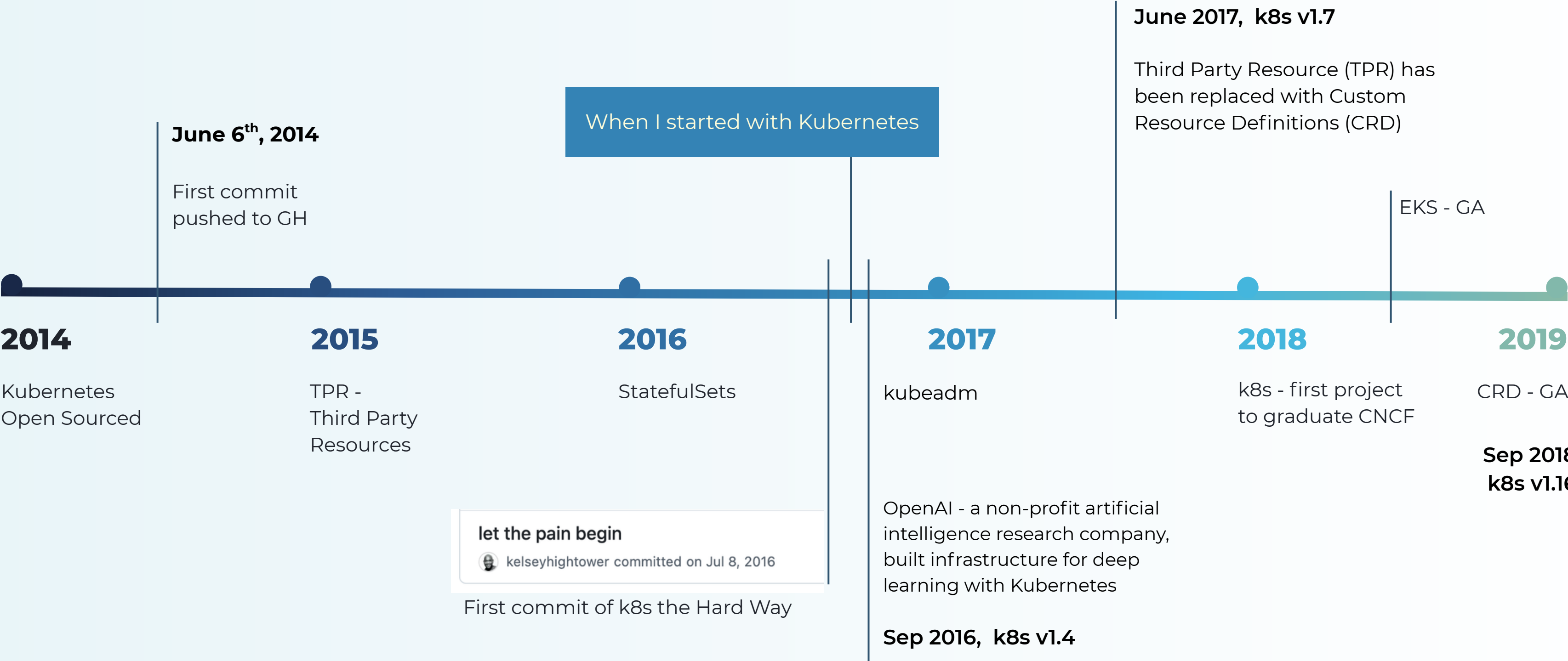


same kubelet, same API, same primitives.

Kubernetes first years



Kubernetes first years

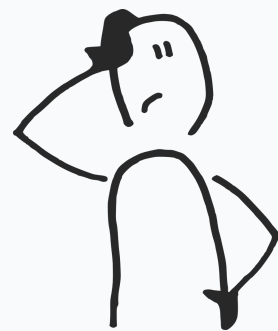


Real-life stories from the early days

PagerDuty

Half of the pods disappeared
in production cluster

HALF THE TEAM



“What is a Pod anyway?”

Real-life stories from the early days

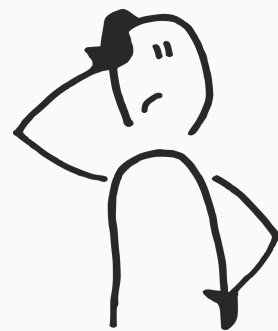
PagerDuty

Half of the pods disappeared
in production cluster

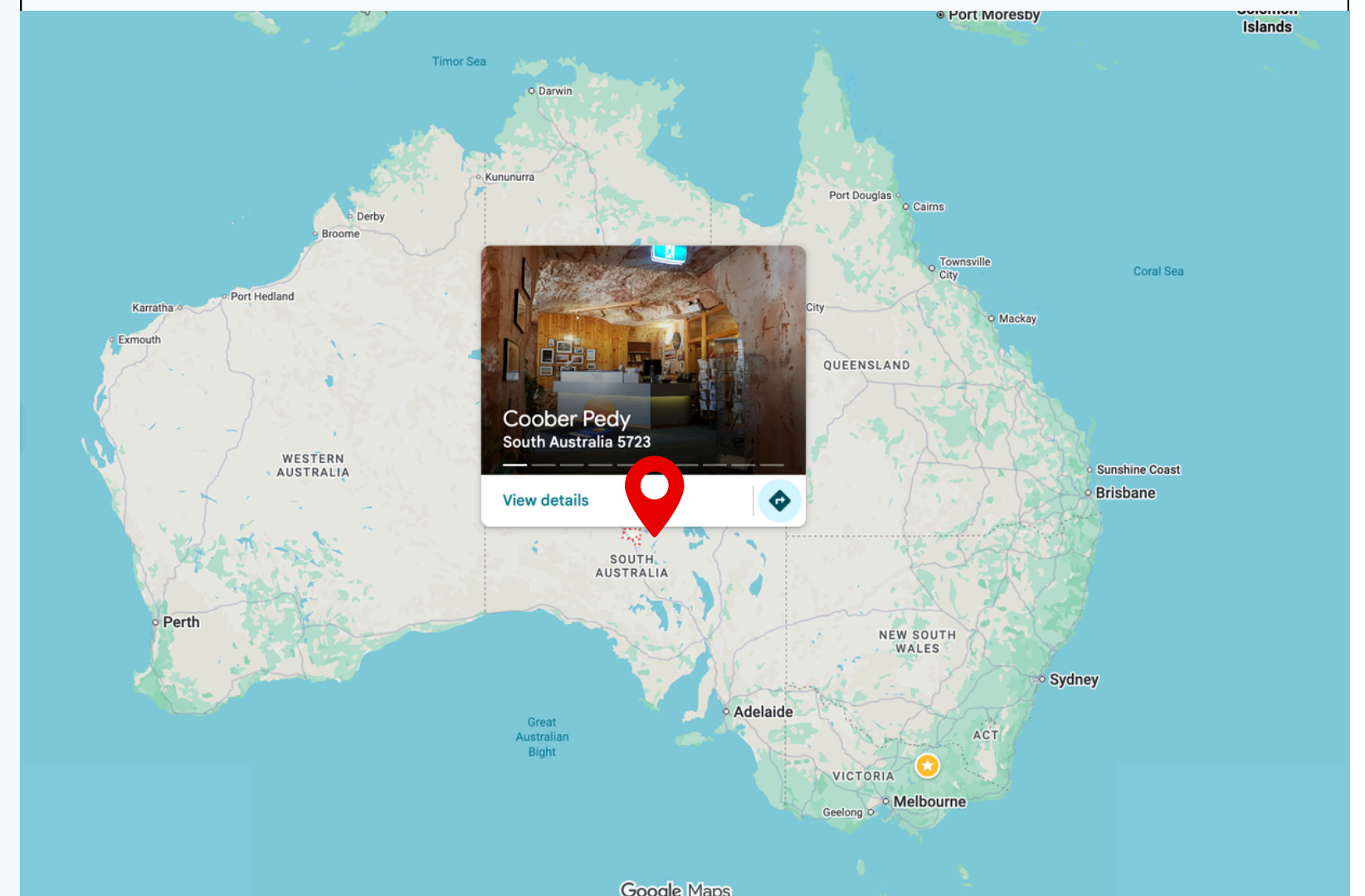
LEADERSHIP

“Why are we going to Coober Pedy?”

HALF THE TEAM



“What is a Pod anyway?”



COHERENT PRODUCTION-GRADE
CONTAINER ORCHESTRATION PLATFORM

RAW MATERIAL:
A PILE OF VMs

WAREHOUSE
FLOOR

KUBERNETES
HARD WAY

K3s
(LIGHTWEIGHT K8s)

K0s
(ZERO-OS K8s)

API
SERVER

CONTROLLER
MANAGER

ETCD
(STATE)

SCHEDULER

SCHEDULER

CNI
(NETWORK
OVERLAY)

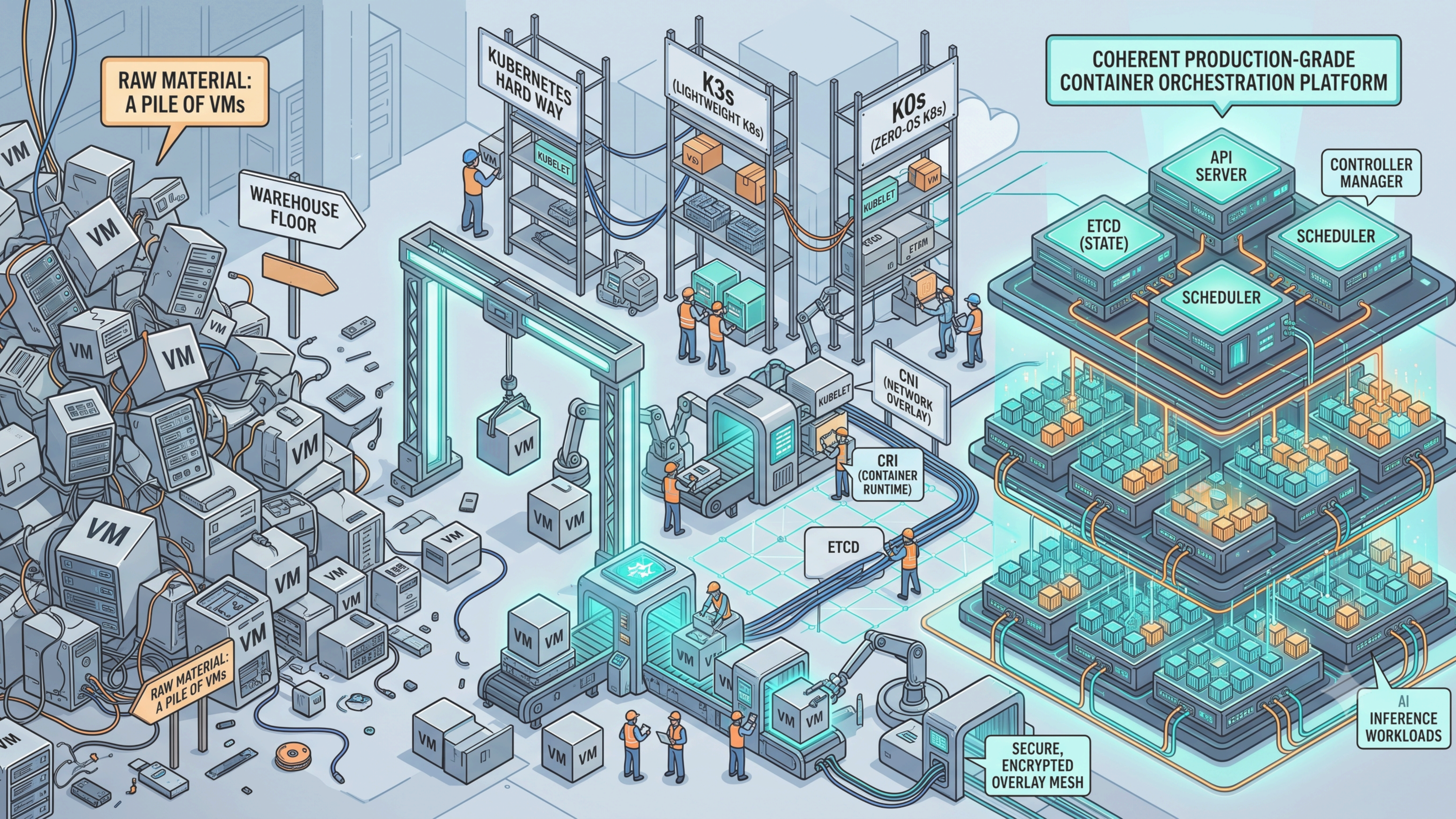
CRI
(CONTAINER
RUNTIME)

ETCD

AI
INFERENCE
WORKLOADS

SECURE,
ENCRYPTED
OVERLAY MESH

RAW MATERIAL:
A PILE OF VMs



Kubernetes the hard way

kubernetes-the-hard-way / docs /

kelseyhightower support arm64 and amd64

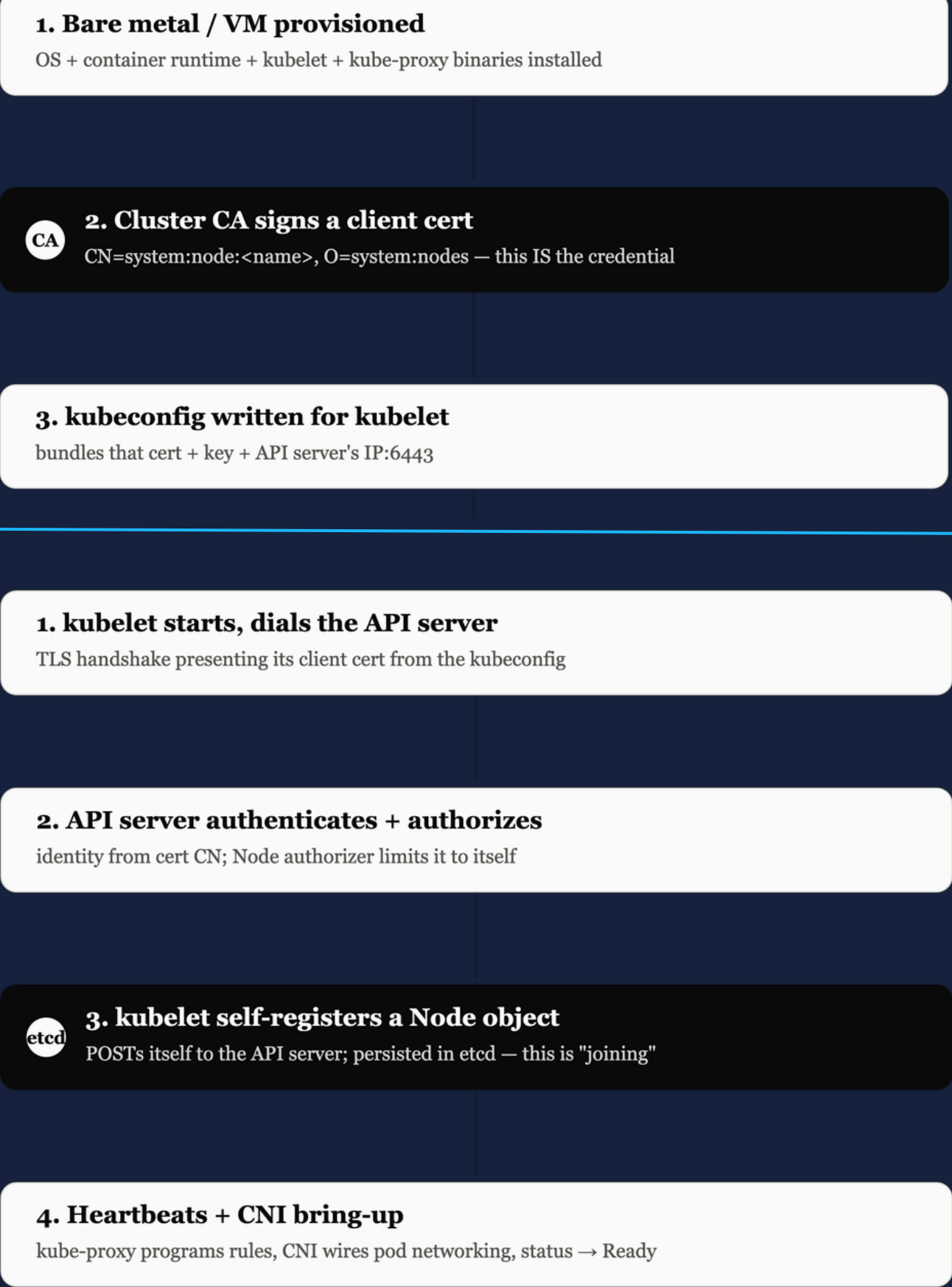
Name
..
01-prerequisites.md
02-jumpbox.md
03-compute-resources.md
04-certificate-authority.md
05-kubernetes-configuration-files.md
06-data-encryption-keys.md
07-bootstrapping-etcd.md
08-bootstrapping-kubernetes-controllers.md
09-bootstrapping-kubernetes-workers.md
10-configuring-kubectl.md
11-pod-network-routes.md
12-smoke-test.md
13-cleanup.md

compute, PKI

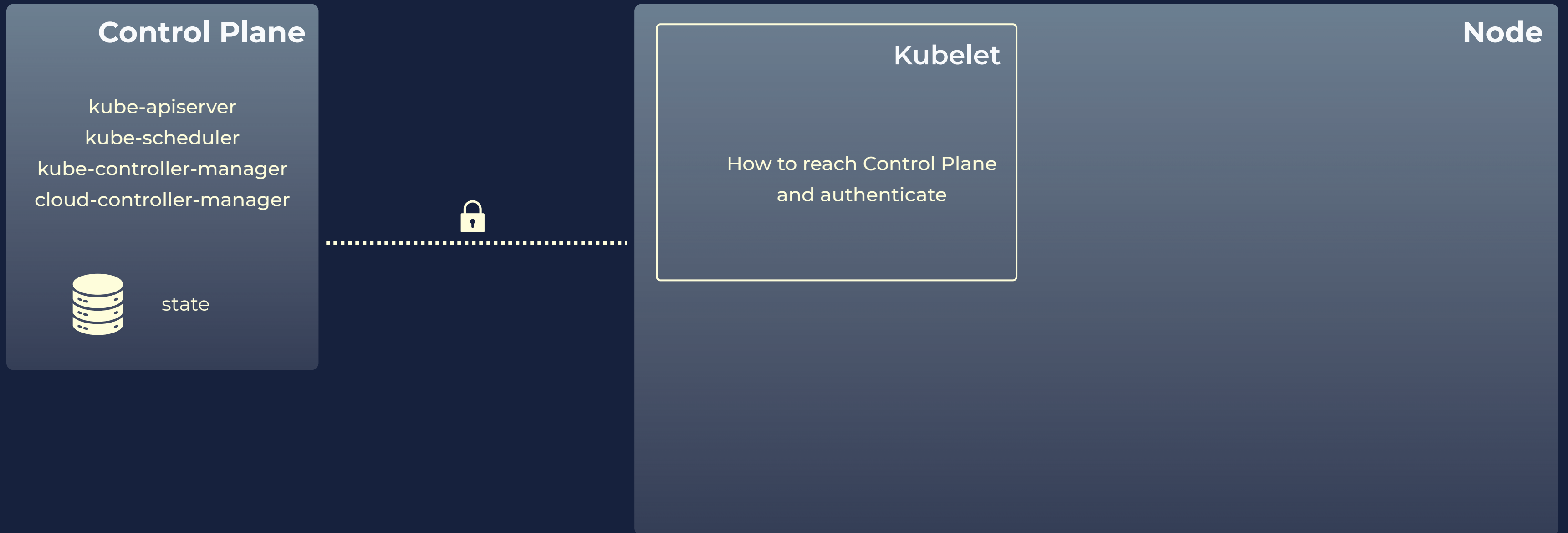
kubelet kubeconfig

setting up control plane

node joins



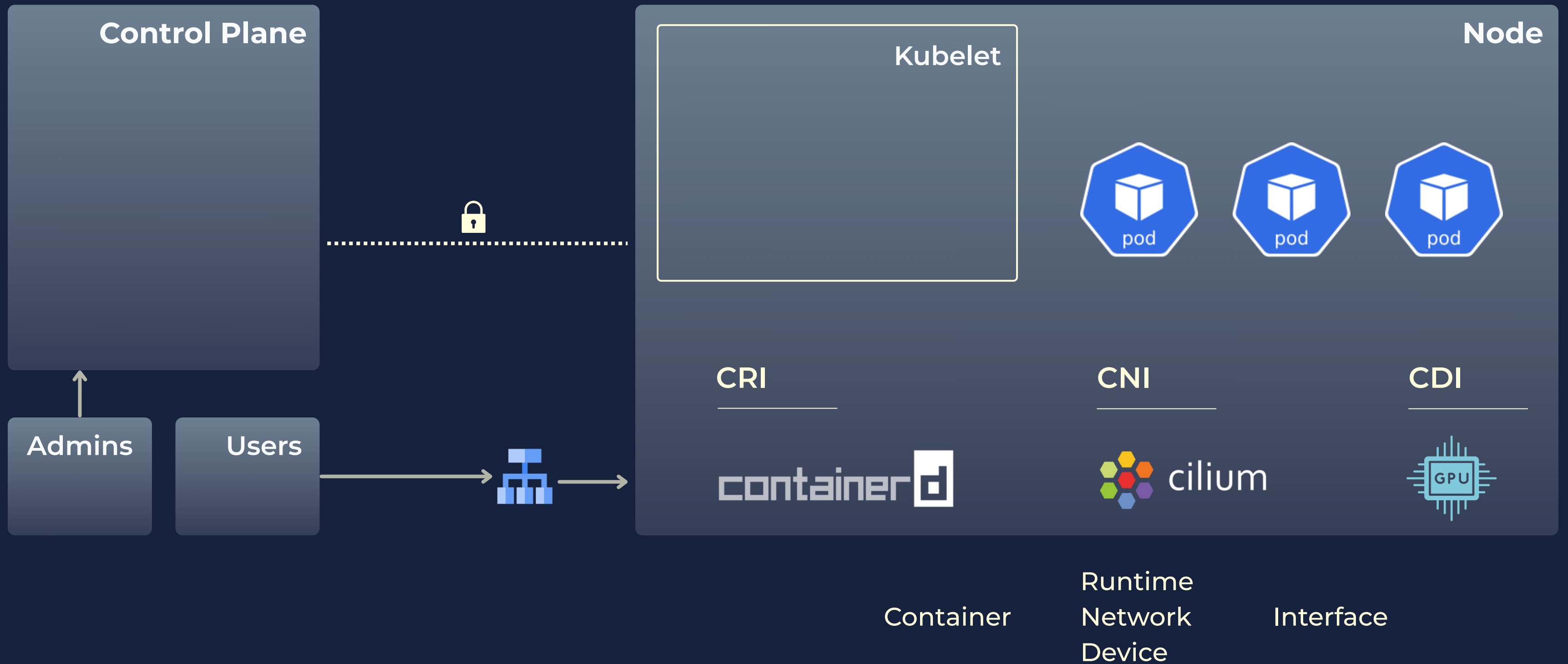
What is Kubernetes?



What is Kubernetes?



What is Kubernetes?



How does this apply to local inference and hybrid clusters



Local compute: drivers & workload patterns

Privacy & data sovereignty

Local data cleanup, PII stripping, patient/customer data

Sub-millisecond latency

Edge computer vision, real-time safety alerts, hazard detection. Process data right where it originated.

Predictable economics

Fixed hardware CapEx vs. volatile token/API OpEx at sustained utilisation.

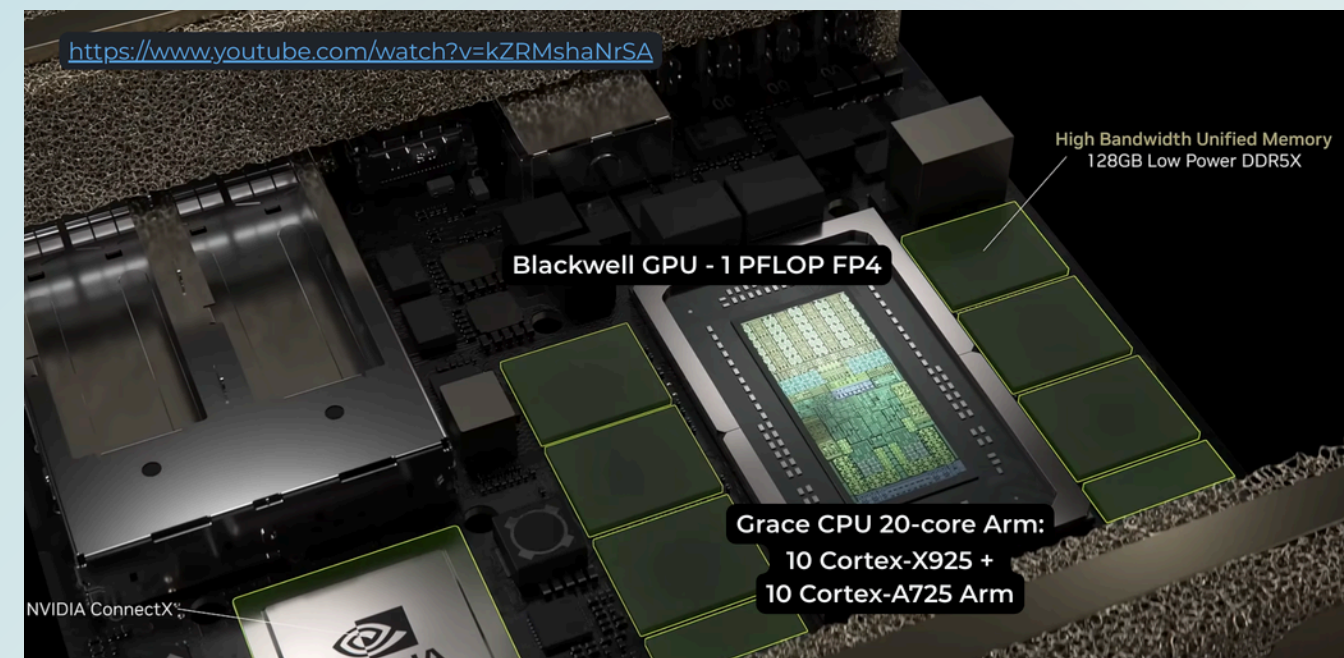
Even with 1 GPU Node

`Qwen3.6-35B-A3B` - @178 Tok/s

`NVIDIA-Nemotron-3-Nano-30B-A3B` - @104 Tok/s

~100B param model, or larger models with streaming MoE

Computer vision, SML, safety filtering, data cleanup, fine tuning & LoRA



Choosing the right tool for the job

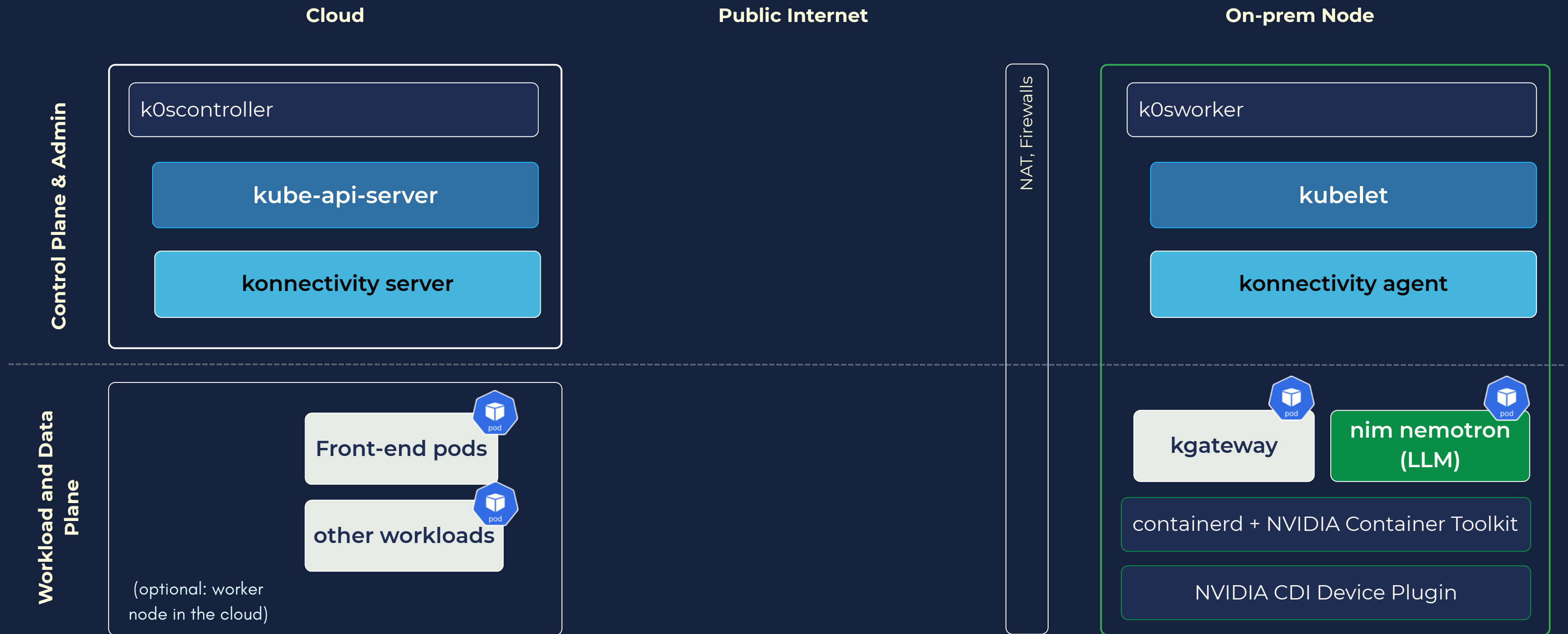
Certified Kubernetes - Distribution	Tagline	Ownership	Use case	CNCF AI Conformance
 K3S	Certified Kubernetes distribution built for IoT & Edge computing		Low-memory Kubernetes distribution packaged as a single binary - 10% of upstream Kubernetes size. Best for resource-constrained edge devices and IoT	not certified officially
 Talos Linux	Bare metal to Kubernetes in under a minute. Linux OS.	 (Sidero Labs)	Declarative, immutable purpose-built Kubernetes OS. Ideal for bare-metal	✓
 k0S	The Zero Friction Kubernetes		zero-dependency Kubernetes distribution, installs upstream vanilla k8s	✓

A distribution is when a vendor takes core Kubernetes and packages it for redistribution. Usually this entails finding and validating the Kubernetes software and providing a mechanism to handle cluster installation and upgrades. [\(source: CNCF landscape documentation\)](#)

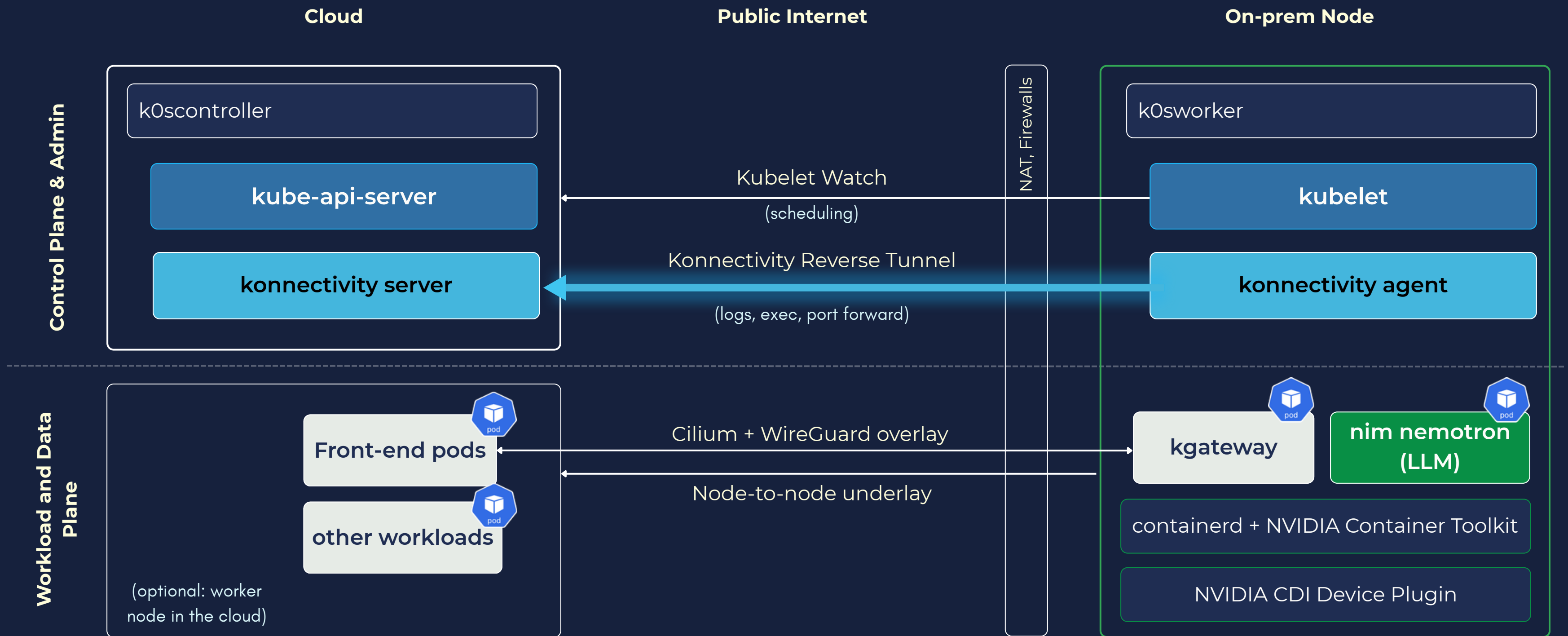
CNCF AI Conformance program introduces a community-led effort to define and validate standards for running artificial intelligence (AI) workloads reliably and consistently on Kubernetes.

[\(source: CNCF blog\)](#)

Hybrid Cluster Architecture (k0s)



Hybrid Cluster Architecture (k0s)



Demo - Infrastructure

```
k0s install controller \  
  --config /etc/k0s/k0s.yaml \  
  --enable-worker \  
  --enable-k0s-cloud-provider
```

```
k0s start
```

```
home@k0s-control-plane:~$ ps aux | grep kube-apiserver  
... /var/lib/k0s/bin/kube-apiserver  
--client-ca-file=/var/lib/k0s/pki/ca.crt  
--authorization-mode=Node,RBAC  
--secure-port=6443  
--kubelet-client-key=/var/lib/k0s/pki/apiserver-kubelet-client.key  
--kubelet-certificate-authority=/var/lib/k0s/pki/ca.crt  
...
```

```
home@k0s-control-plane:~$ k0s status  
Version: v1.36.2+k0s.0  
Process ID: 950  
Role: controller  
Workloads: true
```

```
k0s install worker \  
  --token-file /etc/k0s/join-token \  
  --enable-cloud-provider \  
  --cri-socket remote:unix:///run/containerd/containerd.sock
```

```
k0s start
```

```
home@spark-ce50:~$ ps aux | grep kubelet  
... /var/lib/k0s/bin/kubelet  
--cert-dir=/var/lib/k0s/kubelet/pki  
--cloud-provider=external  
--config=/run/k0s/kubelet/config.yaml  
--kubeconfig=/var/lib/k0s/kubelet.conf  
--root-dir=/var/lib/k0s/kubelet  
...
```

```
home@spark-ce50:~$ k0s status  
Version: v1.36.2+k0s.0  
Process ID: 2125376  
Role: worker  
Workloads: true
```

Demo - Nodes

```
% k get node -o wide
NAME                                STATUS  ROLES          VERSION      INTERNAL-IP  EXTERNAL-IP  KERNEL-VERSION
k0s-control-pl...k8s.internal    Ready   control-plane  v1.36.2+k0s  10.241.x.x   x.x.x.x      6.17.0-1021-gcp (amd64)
spark-ce50                        Ready   <none>         v1.36.2+k0s  192.168.x.x  210.x.x.x    6.17.0-1014-nvidia (arm64)
```

```
% k describe nodes spark-ce50 | grep -A 8 Allocatable
Allocatable:
  cpu:                20
  ephemeral-storage:  3628684392175
  ...
  memory:              127498828Ki
  nvidia.com/gpu:      1
```


Demo - inference workload and routing

NAMESPACE	NAME	READY	STATUS	NODE	
kgateway-system	kgateway-5794df7945-bggw9	1/1	Running	spark-ce50	# GW Control Plane watches Gateway, HTTPRoute
nim-inference	nemotron-nim-68b5cfdffb-zct9j	1/1	Running	spark-ce50	# NIM (NVIDIA Inference Microservice)
nim-inference	nim-gateway-868c7f54fb-nbbgv	1/1	Running	spark-ce50	# gateway dataplane
kube-system	nvidia-device-plugin-daemonset-lspb7	1/1	Running	spark-ce50	

```
- image: nvcr.io/nim/nvidia/nemotron-3-nano-omni-30b-a3b-reasoning:latest
  name: nemotron-nim
  resources:
    limits:
      nvidia.com/gpu: "1"
    requests:
      nvidia.com/gpu: "1"
  volumeMounts:
    - mountPath: /opt/nim/.cache
      name: nim-cache
    - mountPath: /nim-fp8
      name: nim-fp8-weights
      readOnly: true
```

Model

FP8 Model Weights:	~31 GB
vLLM KV Cache:	~30 GB
NIM/vLLM overhead:	~3 GB

Total container:	~64 GB

System

OS + system:	~3-4 GB
k0s worker + kubelet:	~1 GB
Cilium + containerd:	~1 GB
NIM process overhead:	~3-5 GB

Demo - inference

nvidia/nemotron-3-nano-omni-30b-a3b-reasoning

- MoE (Mixture of Experts) architecture
- Native multimodality (text, video, audio, image),
- Long-context processing up to 256K tokens,
- Explicit chain-of-thought reasoning, multi-turn workflows

```
curl http://<LOAD_BALANCER>/v1/chat/completions \
  -H "Content-Type: application/json" \
  -H "Authorization: Bearer sk-nim-<API_KEY_AUTH_IN_KGATEWAY>" \
  -d '{
    "model": "nvidia/nemotron-3-nano-omni-30b-a3b-reasoning",
    "messages": [{"role": "user", "content": "Explain CNCF in 2 sentences"}],
    "max_tokens": 100
  }'

{"id": "chatcmpl-cc519ed2-aeb6-498e-8fda-42047be0cf9d", "object": "chat.completion", "created": 1785190754, "model": "nvidia/nemotron-3-nano-omni-30b-a3b-reasoning", "choices": [{"index": 0, "message": {"role": "assistant", "content": "\n\nThe Cloud Native Computing Foundation (CNCf) is an open-source foundation under the Linux Foundation that incubates and promotes cloud-native technologies such as containers, Kubernetes, and service meshes. It provides a collaborative ecosystem where developers, operators, and vendors can share tools, best practices, and standards for building, deploying, and scaling resilient, portable applications in modern data centers", "refusal": null, "annotations": null, "audio": null, "function_call": null, "tool_calls": [], "reasoning": "We need to explain CNCF in 2 sentences. CNCF = Cloud Native Computing Foundation....\n"}, ...}
```


Open source blueprint

ORCHESTRATION & CONTROL

CNCF Sandbox



Lightweight, single-binary distribution with decoupled control plane & worker roles.

GATEWAY & ROUTING

CNCF Sandbox



Envoy-powered, Kubernetes-native Gateway API with native LLM/AI model routing.

DATAPATH & SECURITY

CNCF Graduated

OSS (wireguard)



eBPF networking with kernel-level encrypted node-to-node overlay tunnels.

HARDWARE ACCELERATION

Open Standard

NVIDIA Container Toolkit & CDI

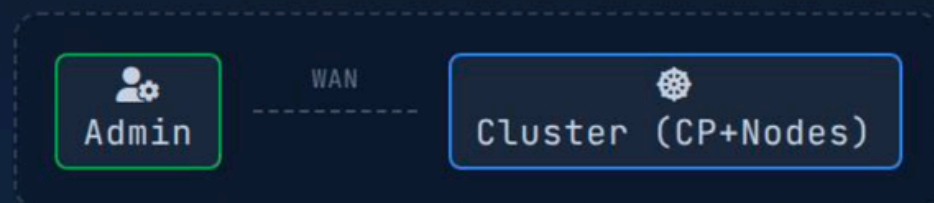
Standardised Container Device Interface (CDI) mapping host GPUs directly into Pod specs.

Hyperscaler Hybrid Philosophies Compared

Google Cloud

Control plane and nodes always deployed together.

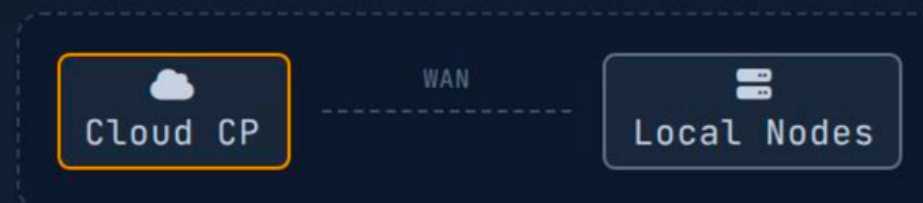
- Attach **clusters**, never individual nodes over WAN.
- High local autonomy & resilience.
- Platform: GKE Enterprise (Anthos).



AWS

EKS Hybrid Nodes decouple the topology.

- Stretches cloud-hosted control planes to on-prem worker nodes.
- Requires constant, high-quality network link.
- Alternatives: AWS Outposts, EKS Anywhere.



Microsoft Azure

Centralized management, local cluster execution.

- Attach existing CNCF clusters via **Azure Arc**.
- Prefers native local control planes (AKS Arc) for survivability.
- Ideal for custom lightweight setups (e.g., kOs).



References

k0s and AI Conformance

<https://k0sproject.io/>

<https://docs.k0sproject.io/stable/governance/cncf/ai-conformance/>

Cilium and WireGuard encryption

<https://docs.cilium.io/en/stable/security/network/encryption-wireguard/>

GKE 130,000 nodes cluster engineering blog

<https://cloud.google.com/blog/products/containers-kubernetes/how-we-built-a-130000-node-gke-cluster/>

Gateway API: kgateway (demo-ed), agentgateway

<https://kgateway.dev/docs/>

<https://agentgateway.dev/docs/kubernetes/latest/>

CNCF Use-case: k3s on-prem infra

<https://www.cncf.io/blog/2026/04/17/k3s-on-on-prem-infrastructures-the-gitops-way-writing-a-custom-k0rden-template-from-scratch>

Local inference ROI

<https://lenovopress.lenovo.com/lp2368-on-premise-vs-cloud-generative-ai-total-cost-of-ownership-2026-edition>



Olga Mirensky



olga-mir.github.io



github.com/olga-mir