kublr

Centralizing Kubernetes and Container Operations

History

- Custom software development company
- Dozens of projects per year
- Varying target environments:
  clouds, on prem, hybrid
- Unified application delivery platform wanted:
  monitoring, logs, security, multiple env, ...

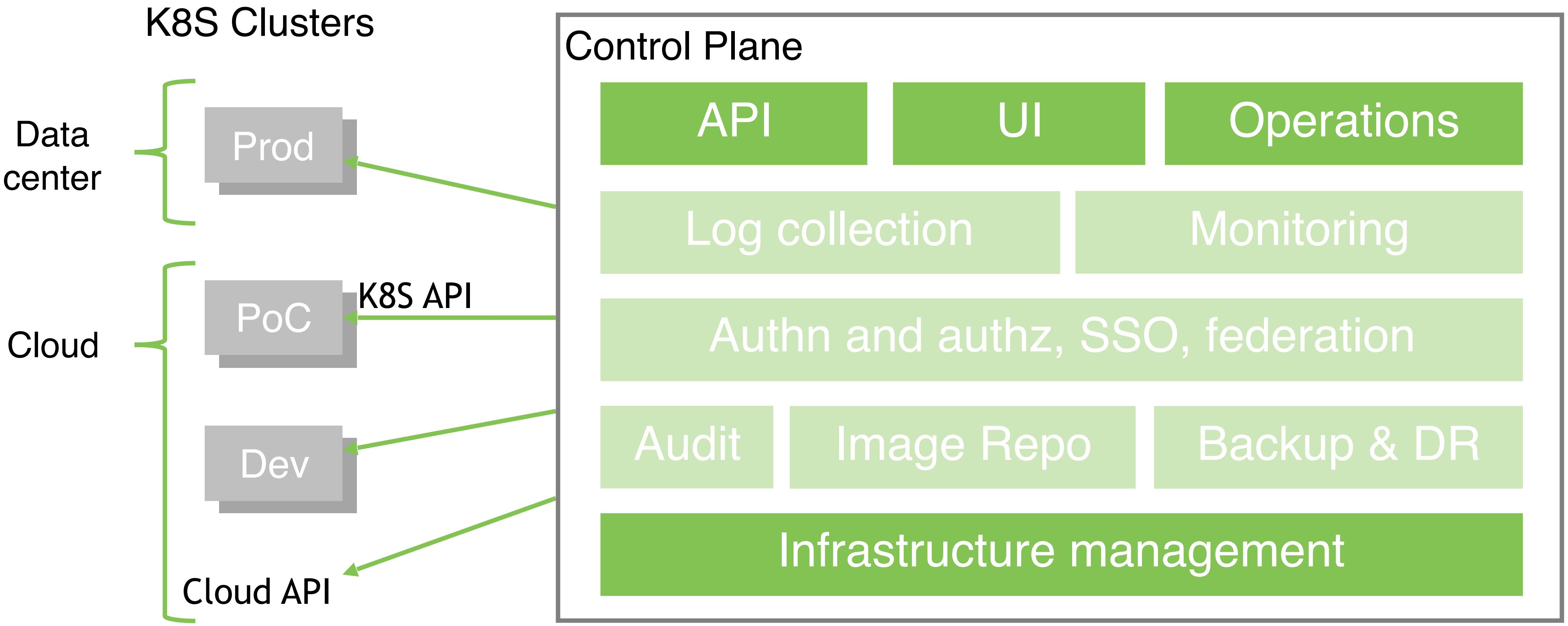
Docker and Kubernetes to the Rescue

- Docker is great, but local
- Kubernetes is great
- ... when it is set up and running
- Who sets up and operates K8S?
- Who takes care of operational aspects at scale?

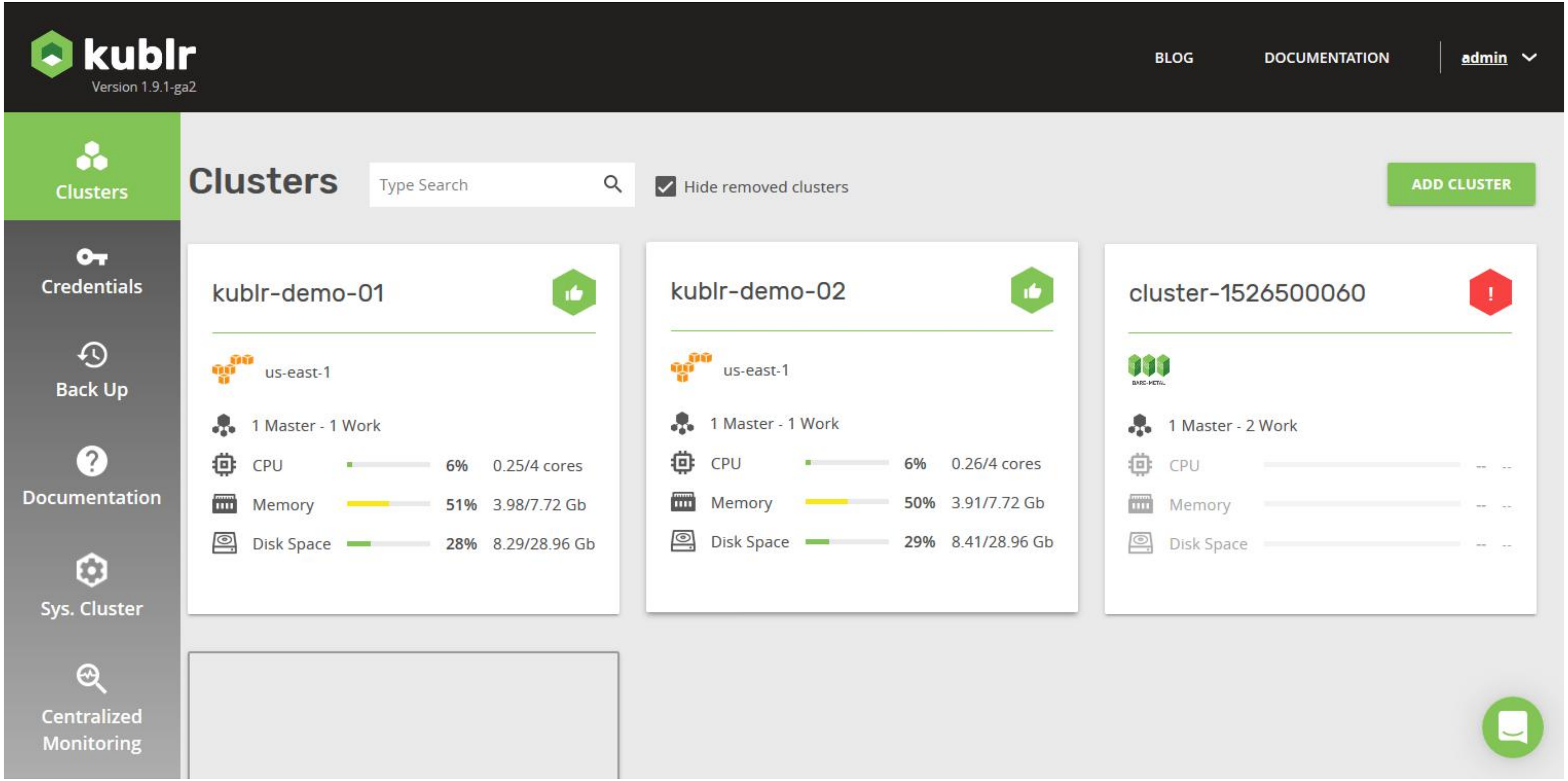
Kubernetes Management Platform Wanted

- Portability – cloud, on prem, isolated deployments
- Centralized multi-cluster operations
  Multiple environments (dev, prod, ...); resource overhead
- Reliability – self-healing, cluster self-reliance, modularity
- Limited management profile – cloud and K8S API
- Architecture – flexible, open, pluggable
- Security, scalability, HA, DR etc.

Central Control Plane: Operations



Central Control Plane: Operations



Clusters

☑ Hide removed clusters

kublr-demo-01

us-east-1

1 Master - 1 Work

CPU	6%	0.25/4 cores
Memory	51%	3.98/7.72 Gb
Disk Space	28%	8.29/28.96 Gb

kublr-demo-02

us-east-1

1 Master - 1 Work

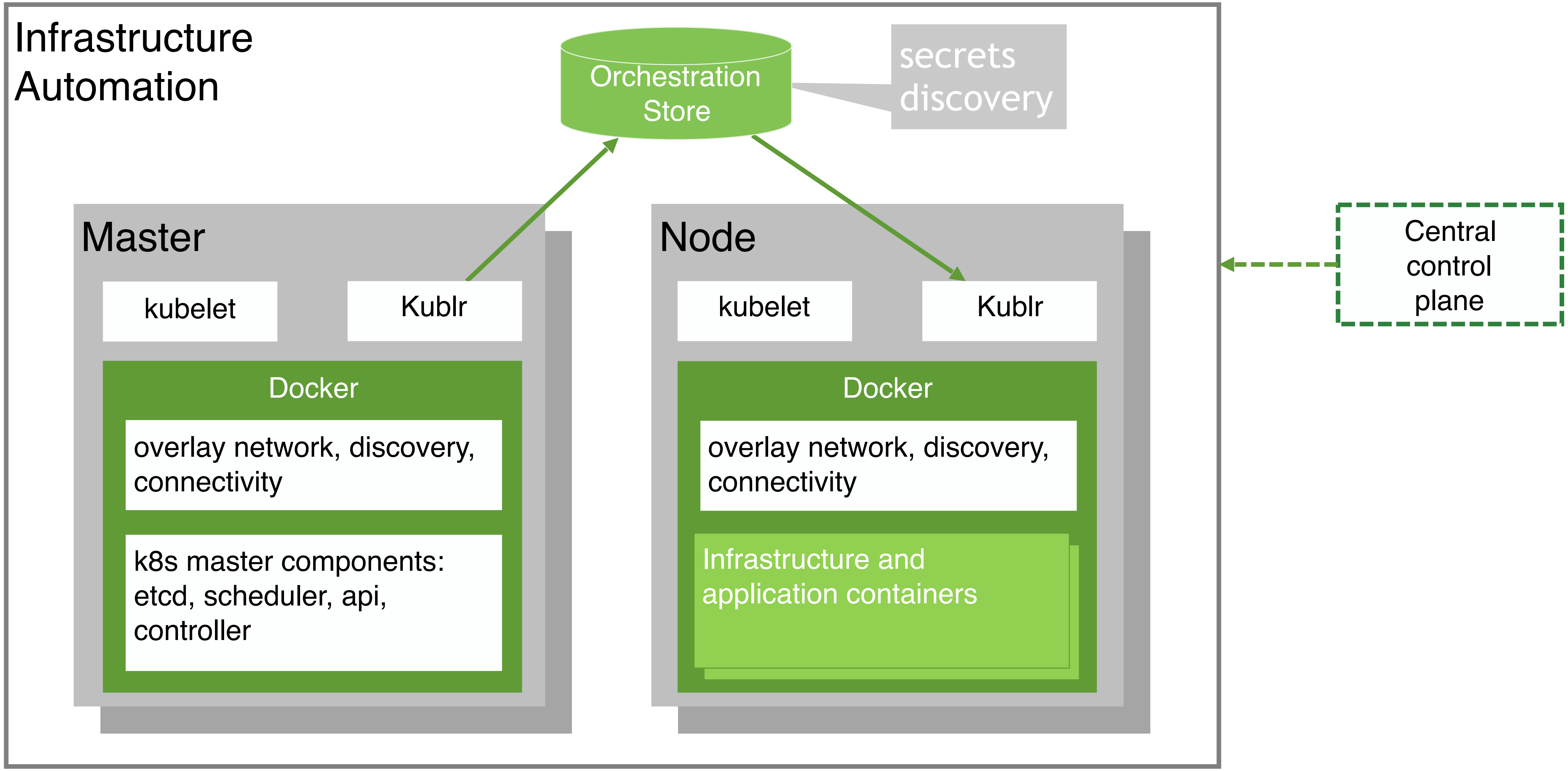
CPU	6%	0.26/4 cores
Memory	50%	3.91/7.72 Gb
Disk Space	29%	8.41/28.96 Gb

cluster-1526500060

1 Master - 2 Work

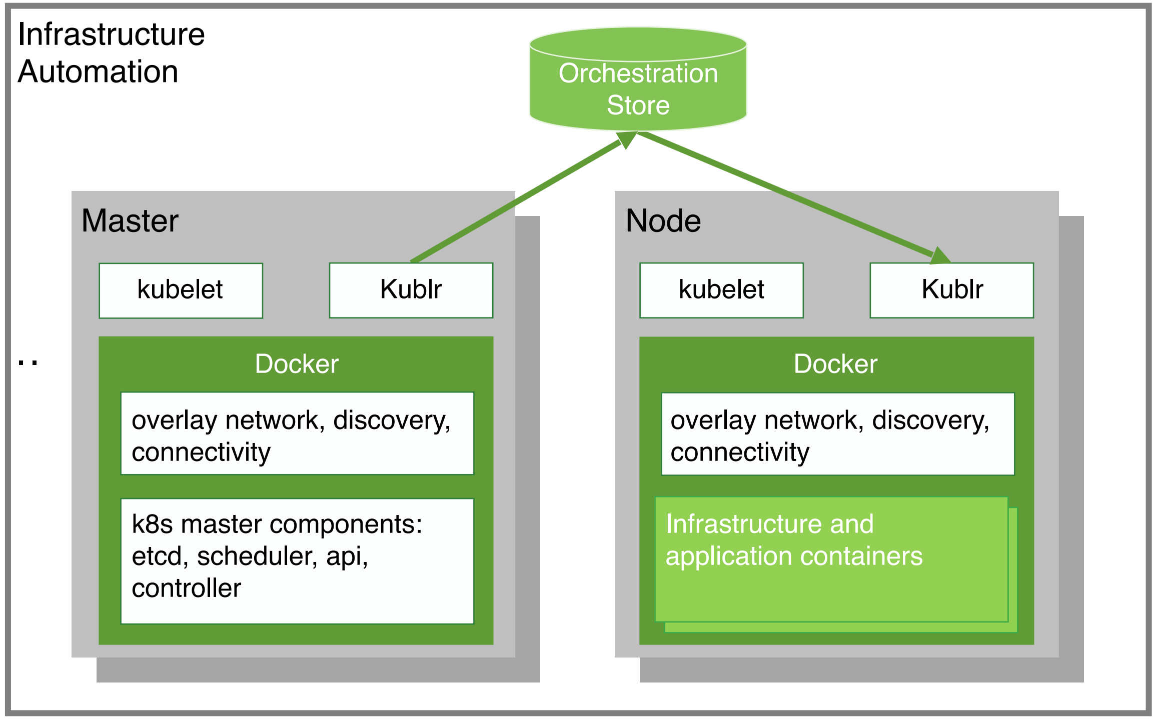
CPU	--	--
Memory	--	--
Disk Space	--	--

Cluster: Self-Sufficiency


Infrastructure Automation
Orchestration Store
secrets
discovery
Master
kubelet
Kublr
Docker
overlay network, discovery, connectivity
k8s master components: etcd, scheduler, api, controller
Node
kubelet
Kublr
Docker
overlay network, discovery, connectivity
Infrastructure and application containers
Central control plane

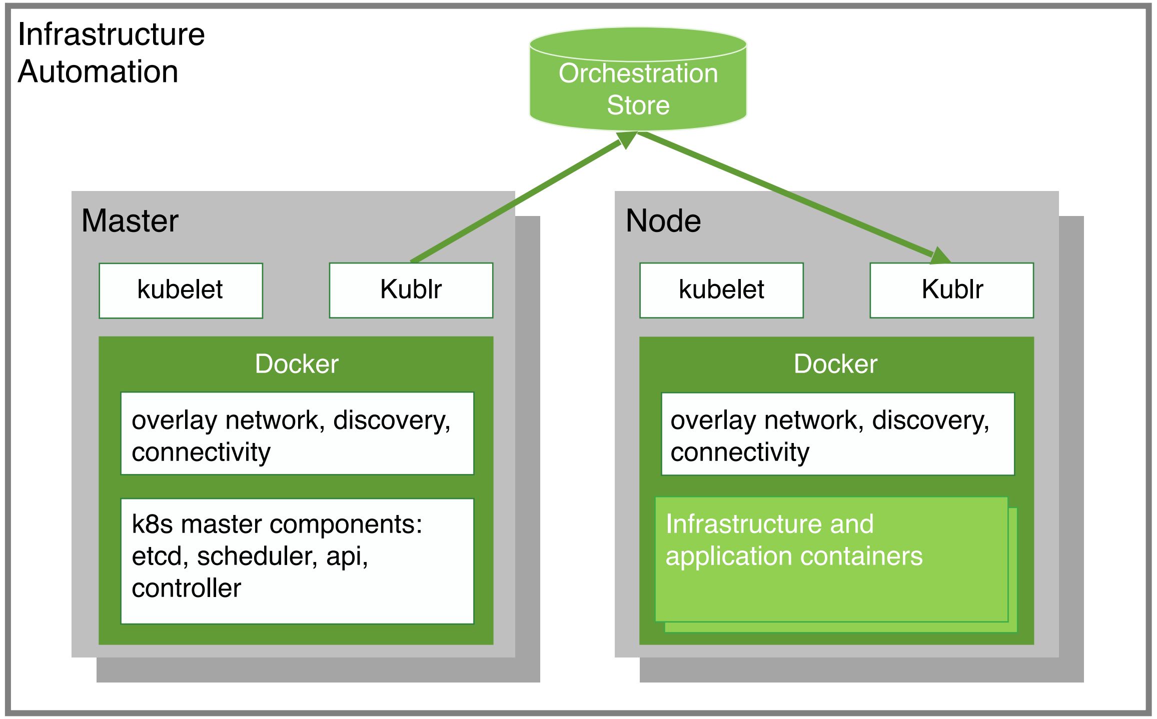
Cluster: Portability

- Everything runs in containers
- Single-binary agent
- Minimal store requirements
  - Shared, eventually consistent
  - RW files for masters, RO for nodes
  - Thus the store can be anything: S3, SA, NFS, rsynced dir, provided files, ..
- Minimal infra automation requirements
  - Configure and run Kublr agent
  - Enable access to the store
  - Can be AWS CF, Azure ARM, BOSH, Ansible, ...

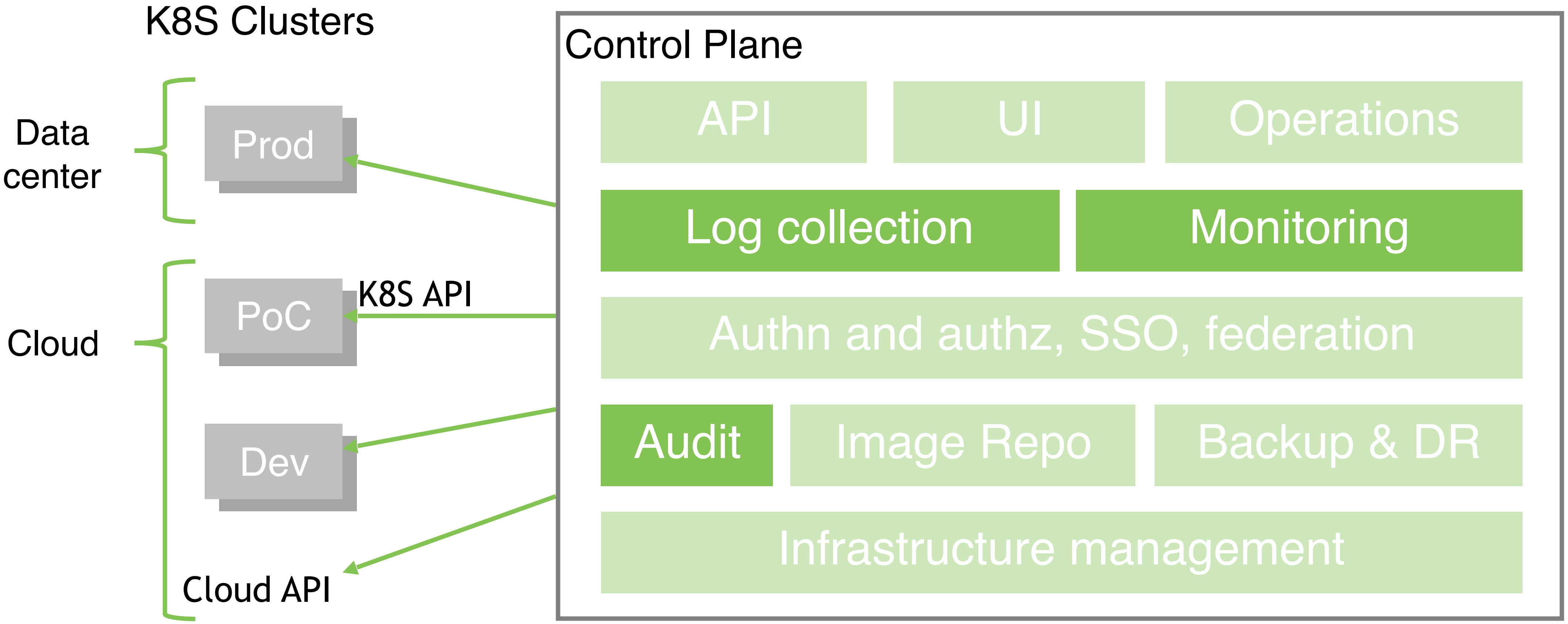

Infrastructure Automation
Orchestration Store
Master
kubelet
Kublr
Docker
overlay network, discovery, connectivity
k8s master components: etcd, scheduler, api, controller
Node
kubelet
Kublr
Docker
overlay network, discovery, connectivity
Infrastructure and application containers

Cluster: Reliability

- Rely on underlying platform only, if possible
  - ASG on AWS
  - IAM on AWS for store access
  - SA on Azure, S3 on AWS
  - ARM on Azure, CF on AWS
- Minimal infrastructure SLA tolerate temporary failures
- Resources - requests and limits


Infrastructure Automation
Orchestration Store
Master
kubelet
Kublr
Docker
overlay network, discovery, connectivity
k8s master components: etcd, scheduler, api, controller
Node
kubelet
Kublr
Docker
overlay network, discovery, connectivity
Infrastructure and application containers

Central Control Plane: Logs and Metrics

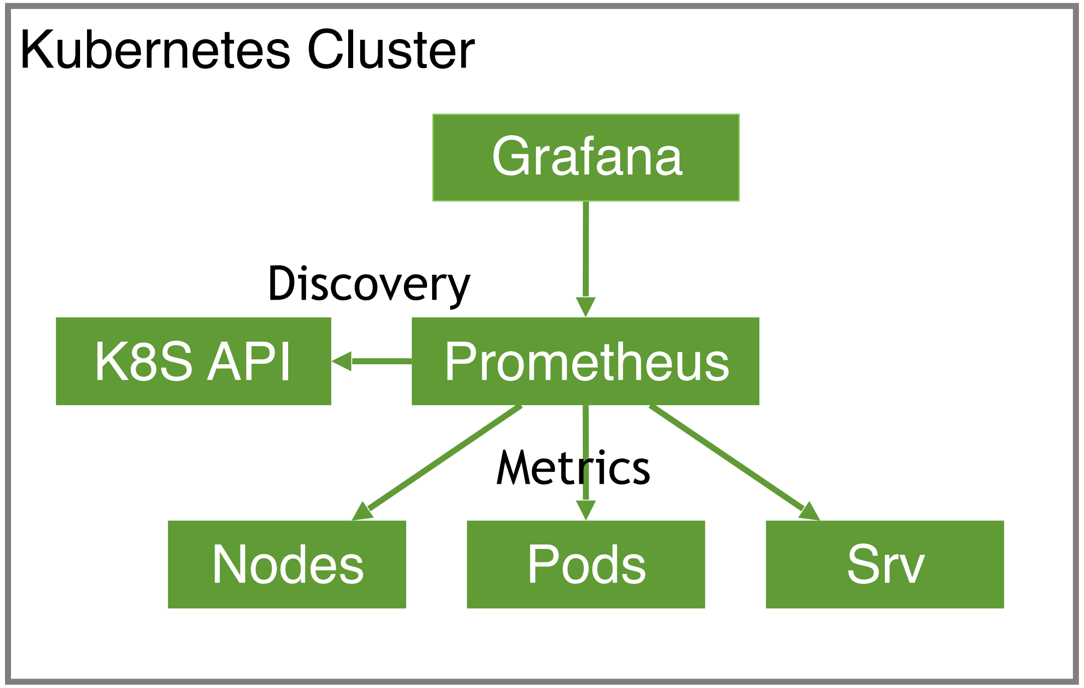


Centralized Monitoring and Logs. Why Bother?

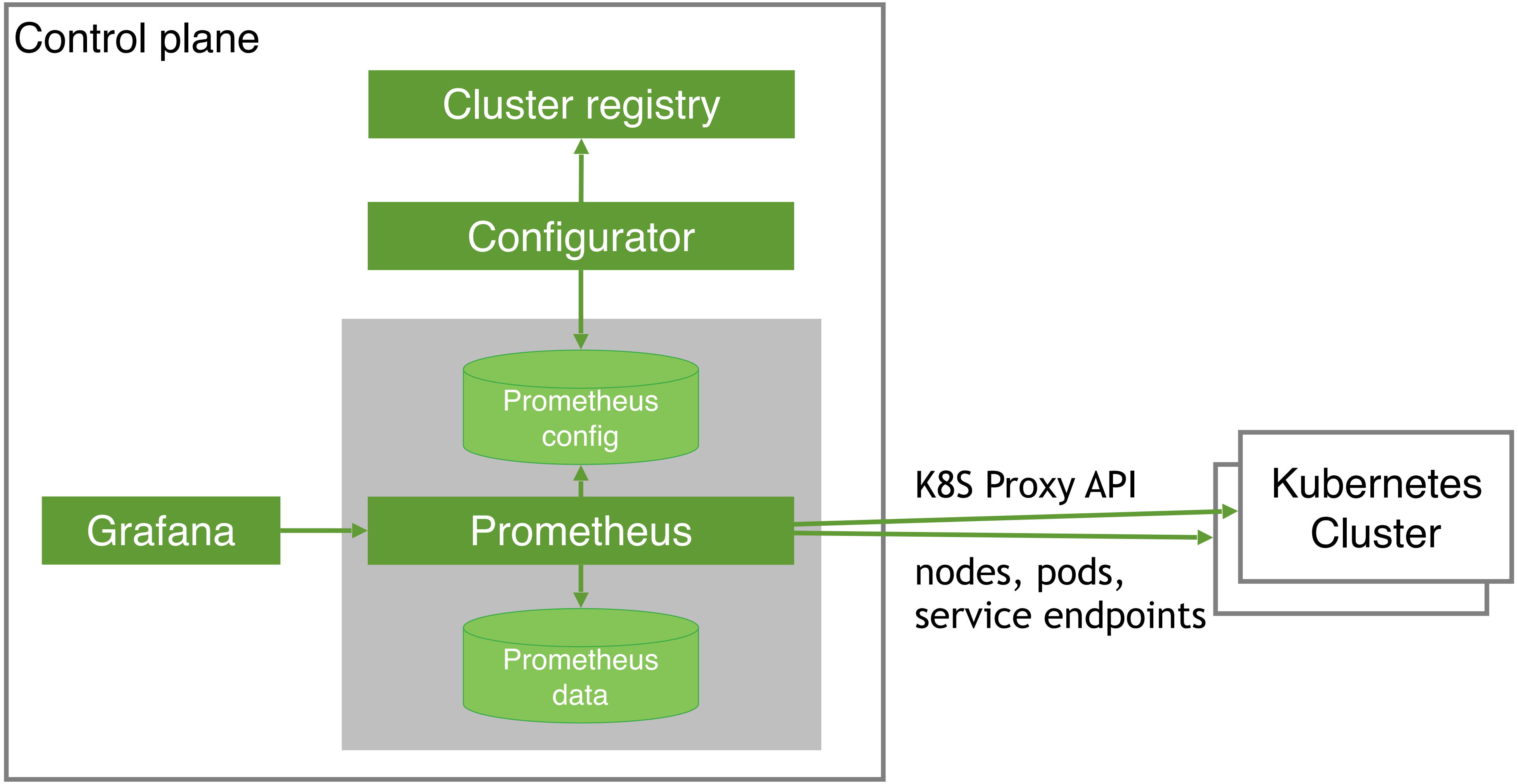
- Prometheus and ELK are heavy
  Need at least 4-8 Gb RAM... each... per cluster
- Cloud monitoring is not always allowed
- Existing monitoring is often not container-aware
- No aggregated view and analysis
- No alerting governance

K8S Monitoring with Prometheus

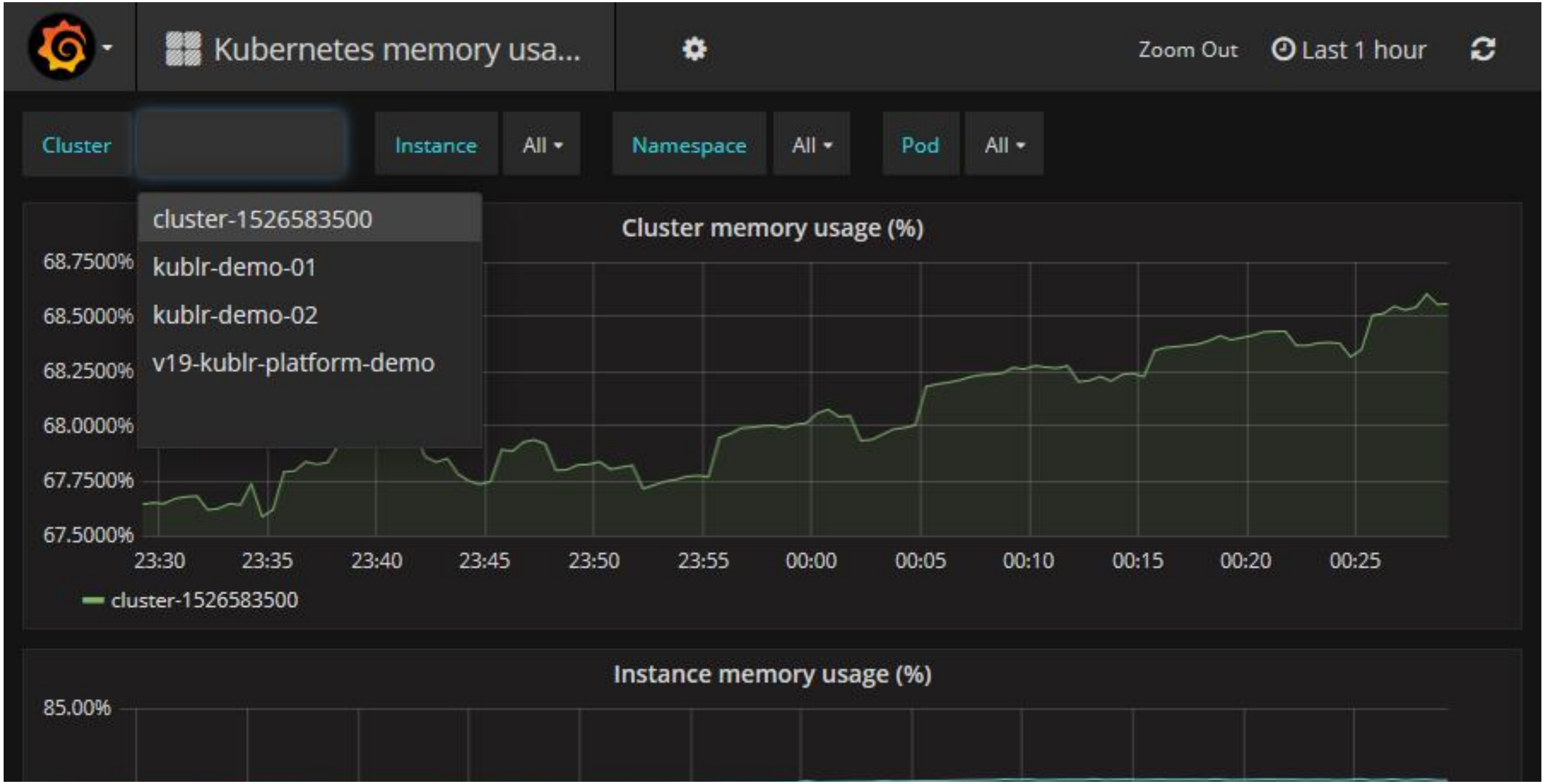
- Discover nodes, services, pods via K8S API
- Query metrics from discovered endpoints
- Endpoint are accessed directly via internal cluster addresses



Centralized Monitoring



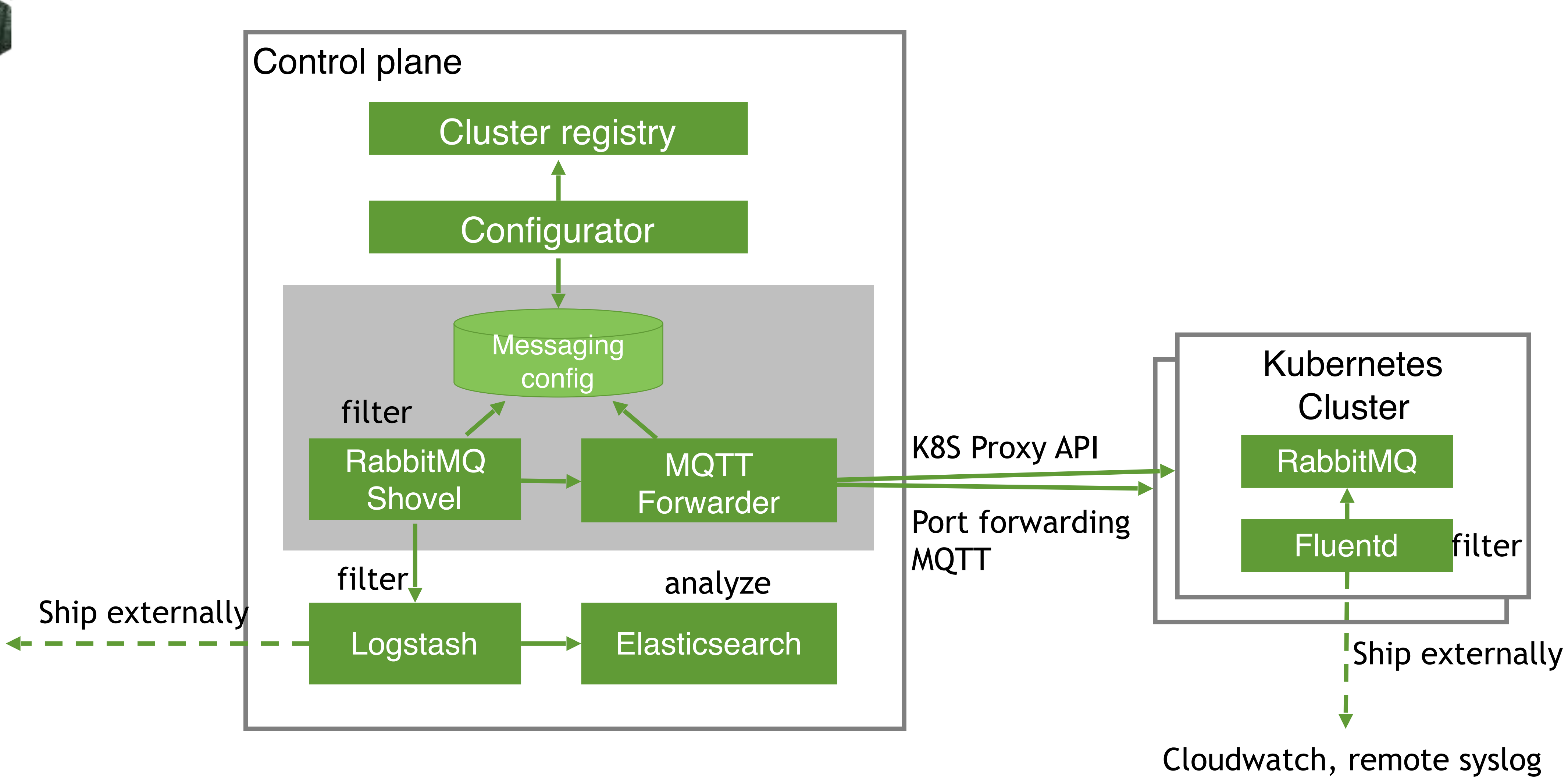
Centralized Monitoring



Centralized Monitoring

- Resource tuning – limits, scale, central part limits
- Long term storage - custom retention, m3db
- Sharding
- Metrics labelling – pod name, container name
- Additional load on k8s API

Centralized Log Collection



The Rest

- Identity management – KeyCloak
  Users, Authn, Autzn, SSO, RBAC
- Backup and disaster recovery
  Scheduled snapshots; k8s and apps data
  Full cluster recovery or copy
- Docker image management – Nexus or Artifactory
  Docker registry, image scanning, isolated env, etc.

Q&A