lunar way®

A tale of 1,5 years with Cloud Native technologies at Lunar Way

GOTO Nights CPH August 2018

Kasper Nissen - @phennex

Who?

Kasper Nissen (@phennex)

- Cloud Architect / SRE @lunarway
- Previous; LEGO Systems, IT Minds, Drivelogger
- Organiser & Co-Founder of Cloud Native Aarhus
- MSc. Computer Engineering
- Founder Cloud Native DK Slack Community
- Occasional speaker at meet ups and conferences

lunar way®

AGENDA

- Cloud Native Infrastructure
- Container Orchestration with Kubernetes
- Monitoring with Prometheus
- Log Management with fluentd and Humio

Cloud Native Infrastructure

https://pixabay.com/en/skyline-new-york-600001/

Cloud Native, the CNCF definition

Cloud native technologies empower organizations to build and run ***scalable applications*** *in modern,* ***dynamic environments*** *such as public, private, and hybrid clouds.* ***Containers****,* ***service meshes****,* ***microservices****,* ***immutable infrastructure****, and* ***declarative APIs*** *exemplify this approach.*

These techniques enable ***loosely coupled*** *systems that are* ***resilient****,* ***manageable****, and* ***observable****. Combined with robust automation, they allow engineers to make* ***high-impact changes frequently*** *and* ***predictably*** *with* ***minimal toil****.*

The Cloud Native Computing Foundation seeks to drive adoption of this ***paradigm*** *by fostering and sustaining an ecosystem of* ***open source, vendor-neutral projects****. We democratize state-of-the-art patterns to make these innovations accessible for everyone.*

https://www.cncf.io/about/faq/

CLOUD NATIVE COMPUTING FOUNDATION

Key characteristics

- Scalable systems
- Dynamic environments
- Containers
- Immutable Infrastructure
- Observability and manageability
- Open source

Why do this?

https://www.pexels.com/photo/flight-sky-earth-space-2159/

Speed!

Cloud Native Landscape

v20180525

See the interactive landscape at l.cncf.io

Greyed logos are not open source

App Definition and Development

Database and Data Warehouse

- Vitess (CNCF Incubating)
- ArangoDB, BlocksDB, CarbonData, Cassandra, CockroachDB, Couchbase, CrateDB, Druid, FoundationDB, Hadoop, IBM DB2, Igniteo, InfinisDB, MarkLogic, MemSQL, SQL Server, MongoDB, MySQL, neo4j, Oracle, OrientDB, Postgres, pilosa, PostgreSQL, presto, Qubole, redis, RethinkDB, Scylla, Snowflake, Software AG, SQL Data Warehouse, Stardog, TiDB, Vertica, YugaByte

Streaming

- NATS (CNCF Incubating)
- cloudevents (CNCF Sandbox)
- Amazon Kinesis, APEX, Heron, nifi, Apache Pulsar, Spark, Storm, Flink, Google Cloud Dataflow, kafka, OpenMessaging, Pulsar, RabbitMQ, Stream Serve

Source Code Management

- Bitbucket, GitHub, GitLab, Microsoft Team Foundation Service

Application Definition and Image Build

- Telepresence (CNCF Sandbox)
- apache brooklyn, binaml, DRAFT, hobitat, HELM, Kaniko, kompose, Vulcan, minikube, OpenFaaS Initiative, Packer, Google Cloud Skaffold, Fission

Continuous Integration / Continuous Delivery (CI/CD)

- AppVeyor, argo, Bamboo, BRIGADE, buddybuild, Buildkite, circleci, Skycap, cloudbees, codefresh, Concourse, CodeShip, Drone, GitLab Runner, go, Habitus, Jenkins, JenkinsX, openstack Zuul, Screwdriver, shippable, Spinnaker, TeamCity, Travis CI, weave flux, wercker, OPLOY

Orchestration & Management

Scheduling & Orchestration

- kubernetes (CNCF Graduated)
- Amazon ECS, Docker Swarm, MESOS, Kubernetes Service Fabric, Nomad

Coordination & Service Discovery

- CoreDNS (CNCF Incubating)
- Apache Zookeeper, Consul, ContainerPilot, etcd, NETFLIX EUREKA, airbnb SmartStack

Service Management

- GRPC (CNCF Incubating)
- envoy (CNCF Incubating)
- LINKERD (CNCF Incubating)
- Open Policy Agent (CNCF Sandbox)
- SCALE, Apache Thrift, AVI Networks, SOFA MESH, backplane, CONDUIT, f5, HAPROXY, heptio Contour, Hyperlink, Istio, Kong, NETFLIX RIBBON, NETFLIX ZUUL, NGINX, traefik, vamp

Runtime

Cloud-Native Storage

- ROOK (CNCF Sandbox)
- ceph, CSI, DATERA, DELL EMC, Diamanti, GLUSTER, HITACHI, HEDVIG, Hewlett Packard Enterprise, kasten, LeoFS, MINIO, NetApp, OpenEBS, OpenIO, OpenSDS, portworx, PURE STORAGE, Quobyte, Rook, ROBIN, Sheepdog, storagelibrary, SWIFT, Triton Object Storage

Container Runtime

- containerd (CNCF Incubating)
- rkt (CNCF Incubating)
- cri-o, Google Cloud gVisor, Intel Clear Containers, kata, lxd, Pouch, runV, Singularity, SmartOS

Cloud-Native Network

- CNI (CNCF Incubating)
- alcide, Apoteo, AVISTX, Skydive, cilium, Contiv, CUMULUS, flannel, GuardiCore, LIGATO, midonet, VMware NSX, navigatorworks, OvS, OPENCONTRAIL, PLUMgrid, CALICO, ROMANA, weave net

Provisioning

Host Management / Tooling

- ANSIBLE, CFEngine, CHEF, kube-bench, LINUXKIT, puppet, RUN DECK, SALTSTACK, BlackStorm

Infrastructure Automation

- Intelligent Infrastructure, BOSH, CLOUDIFY, Intelekt, juju, Pulumi, ManageIQ, Terraform

Container Registries

- Amazon ECR, Azure Registry, codefresh REGISTRY, REGISTRY, HARBOR, IBM Cloud, JFrog Artifactory, Portus, ATOMIC, QUAY

Secure Images

- Notary (CNCF Incubating)
- TUF (CNCF Incubating)
- anchore, aqua, BLACK DUCK, clair, Grafeas, NeuVector, OpenSCAP, Specialty Linux Images, Twistlock

Key Management

- spiffe (CNCF Sandbox)
- SPIRE (CNCF Sandbox)
- CYBERARK CONJUR, KeyWhiz, KNOX, LYA, ORACLE POLICY AUTOMATION, Teleport, Vault

Cloud

Public

- aws, Azure, Baidu Cloud, DigitalOcean, FUJITSU, Google Cloud, HUAWEI, IBM Cloud, Joyent, ORACLE, packet, Tencent Cloud

Private

- DIGITAL REBAR, FORMAN, MAAS, openstack, vmware

l.cncf.io

This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path.

CLOUD NATIVE Landscape

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint Amplify

Platforms

Certified Kubernetes - Distribution

- AppCode, caicloud, CANONICAL, Container Linux, containerize, DaoCloud, Diamanti, docker, Fowlcloud, Giant Swarm, IBM Cloud, Huawei Cloud, JD.COM, Joyent, FUJITSU, kubeoperator, Kubermatic, kubir, MESOSphere, MIRANTIS, navops, NetLinks Cloud, OPENSHIFT, ORACLE, Pivotal, PLATFORM9, QHoudon, RANCHER, STACKRIOT, SUSE, TECTONIC, Telekube, Typhoon, vmware, 嘉云数科

Certified Kubernetes - Hosted

- 天翼云, Alibaba Cloud, Amazon EKS, Azure Container Service, Azure, Baidu Cloud, 博云, CISCO, EasyStack, Google Cloud Platform, Huawei Cloud, 新云科技, LYSIN, HUAWEI, IBM Cloud, nirmata, ORACLE, SAP, Tencent Cloud, 阿里云, ZTE

Certified Kubernetes - Installer

- APPFORM, CoreOS, Gardener, Google Cloud, Heptio, ORACLE, SANDBOX SDS, WEAVEWORKS

Non-Certified Kubernetes

- Maestro, gyrdonic, HUAWEI, STRATOSCALE, SUPERGIANT

PaaS/Container Service

- APPFORM, AMAZON ALTOSTRATUS, Flynn, CLOUDFOUNDRY, Hasura, HYPER.SH, Jiaozi, KONTENA, Lightbend, MANTL, Ku Cloud, Pivotal, platform.sh, portainer.io, Scalingo, Tsuru

Observability & Analysis

Monitoring

- Prometheus (CNCF Incubating)
- Amazon Web Services, APPNAMICS, AppPoint, CloudHealth, COSCALE, DATADOG, dynatrace, Google Stackdriver, Grafana, graphite, Heroic, Icinga, influxdb, INSTANA, IRONdb, librato, Logzio, Los Alamos, Nagios, NETSIL, New Relic, NOSOURCE, Outlyer, sensu, SENTRY, SignalFx, StackRox, Sysdig, Thanos, weave cloud, weave scope

Logging

- fluentd (CNCF Incubating)
- elastic, graylog, Humio, LOGGLY, logz.io, loom systems, remotest, splunk, sumologic

Tracing

- JAEGER (CNCF Incubating)
- OPENTRACING (CNCF Incubating)
- Stackdriver, Stackdriver, ZIPKIN

Serverless

Special

Kubernetes Certified Service Provider

- accenture, 博云, Baidu Cloud, binaml, 嘉云数科, caicloud, CANONICAL, claranet, CloudOps, Container Solutions, CONTINO, CoreOS, Cloud Native Computing Foundation, DaoCloud, EasyStack, GlobalSoft, Heptio, HUAWEI, IBM Cloud, Pulumi, JFROG, KENZAN, kin.cloud, kumina, LIVEWYRE, loodse, MIRANTIS, noitaways, NOW CONTINUE, nirmata, NTT DATA, OCTO, OPENSOURCE, RX-M, SAMSUNG SDS, SAP, Tencent Cloud, 天翼云, THOUGHTWORKS, weaveworks, 新云科技, 阿里云

Kubernetes Training Partner

- CloudOps, Container Solutions, DaoCloud, Pulumi, loodse, RX-M, THE LINUX FOUNDATION TRAINING

Serverless Cloud Native Landscape

v20180525

See the serverless interactive landscape at **s.cncf.io**

Greyed logos are not open source

Tools

dashbird, Epsagon, event gateway, Glue, IO|pipe, Iron.io, Node Lambda, OVERCLOCK, python-λ, SIGMA, STACKERY, THUNDRA

Security

intrinsic, Protego, PURESEC, snyk

Framework

APEX, .arc Architect, aws Chalice, AWS SAM, Claudia.js, FLOGO, gun.io, serverless, Shep, SPARTA, Spring Cloud Function

Platform

Hosted

ALGORITHMIA, AWS Lambda, Azure Functions, BINARIS, Clay, cloudboost, CLOUDFLARE, Google Cloud Functions, HUAWEI, HYPER.SH, IBM Cloud, NANO-LAMBDA, PubNub FUNCTIONS, spotinst, stdlib, syncano, twillo, WEBLAB, ZEIT

Installable

APACHE OpenWhisk, AppScale, DISPATCH, fission, fn, GALACTIC FOG, Kubeless, LunchBadger, nuclio, OPENFAAS, OpenLambda, riff

s.cncf.io

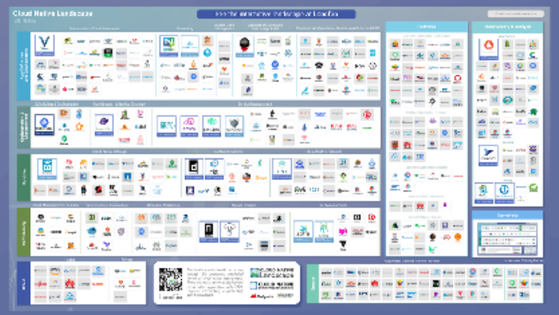
Serverless computing refers to a new model of cloud native computing, enabled by architectures that do not require server management to build and run applications. This landscape illustrates a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled, and billed in response to the exact demand needed at the moment.

CLOUD NATIVE Landscape

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint

Cloud Native Landscape



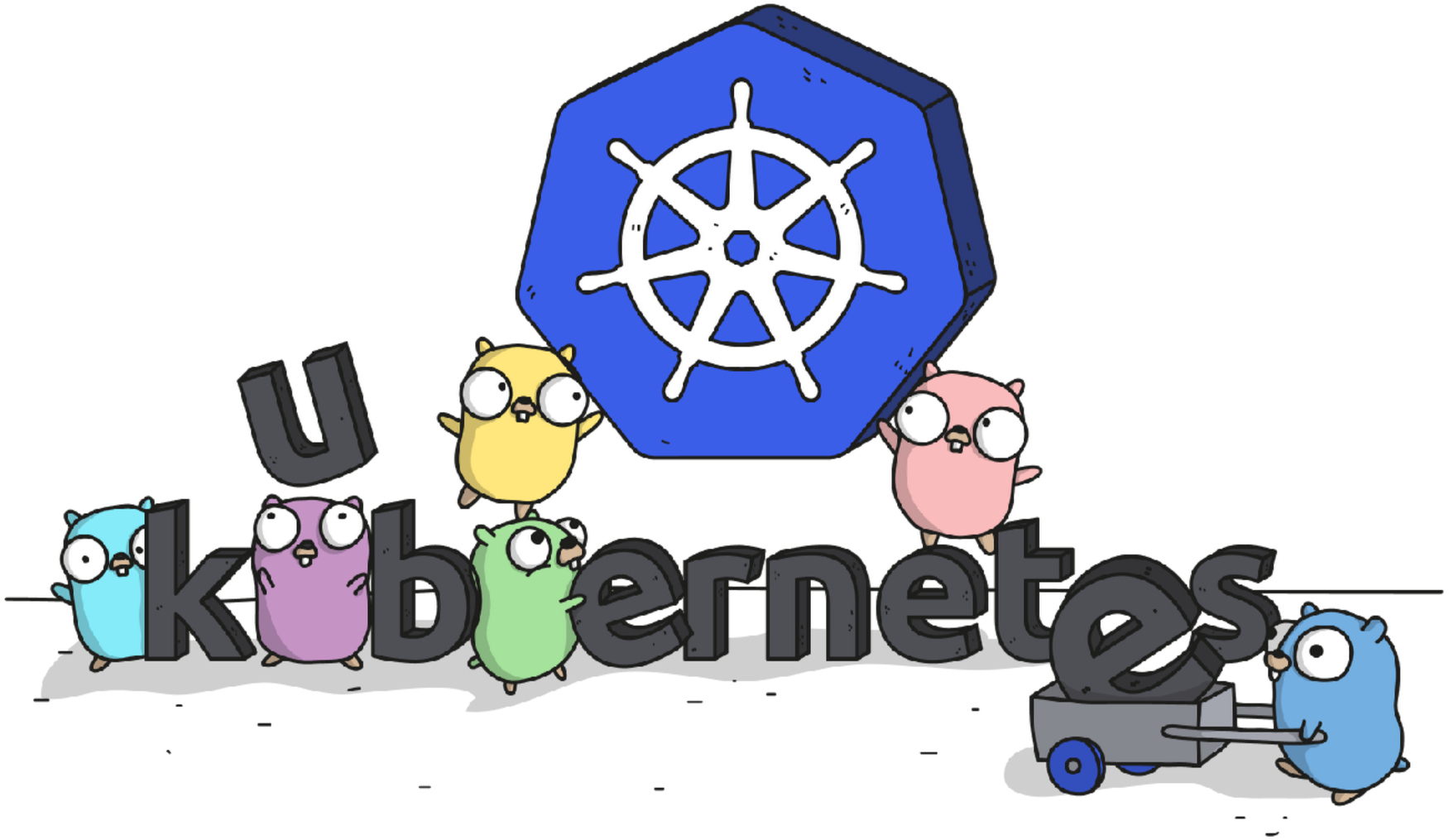
Today, we will setup...

- A highly available Kubernetes cluster across 3 availability zones
- Route traffic from the internet to the cluster
- Monitoring with Prometheus
- Log collection with fluentd
- Log management with Humio

.... and hopefully Thomas will build upon this setup!

Container Orchestration with Kubernetes

https://www.pexels.com/photo/shallow-focus-photography-of-black-ship-1095814/



https://github.com/ashleymcnamara/gophers/blob/master/KUBERNETES_GOPHER.png

Kubernetes, what is it?

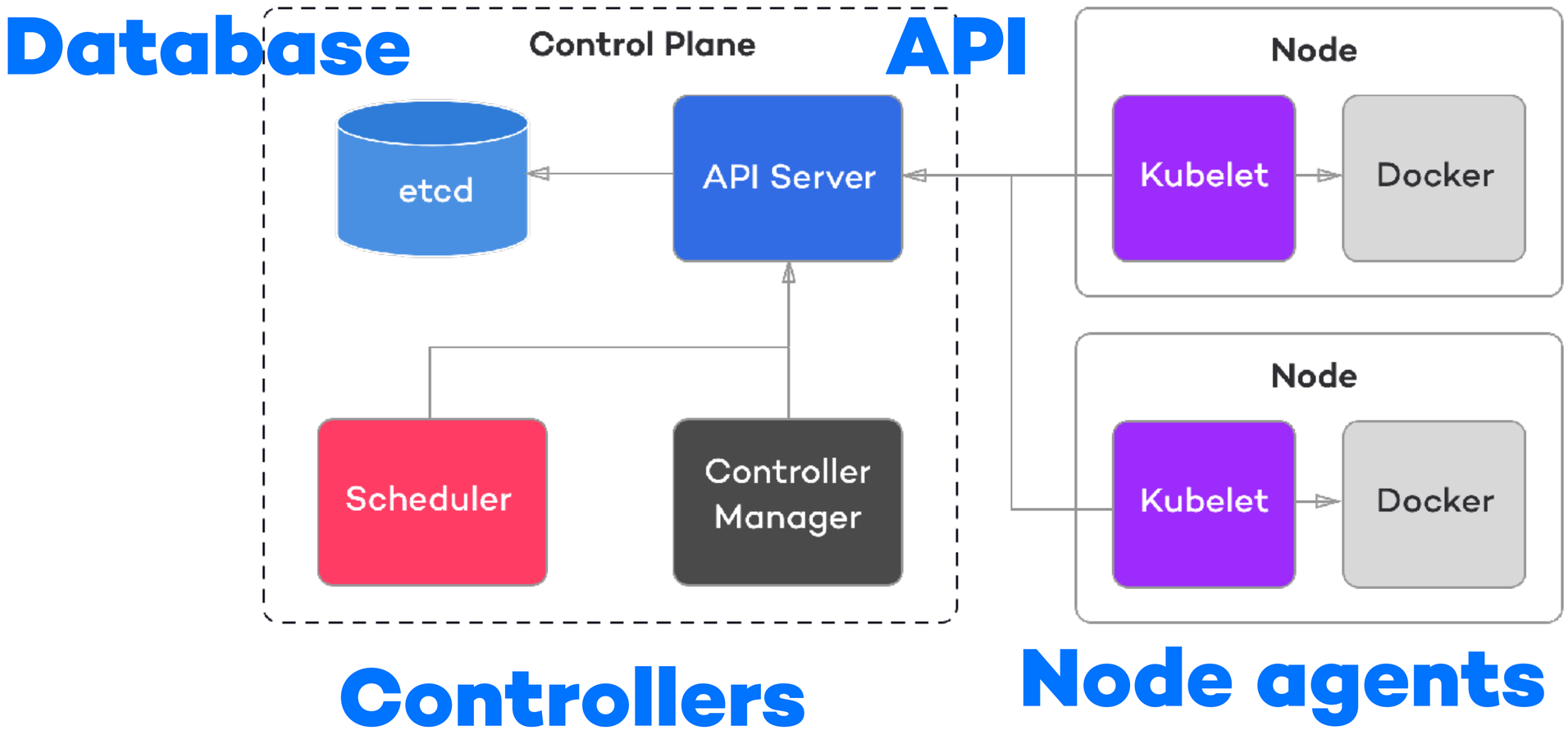
A Cluster Orchestrator

- Makes decisions on which nodes containers should be started
- Provides Service Discovery
- Integrate with networking

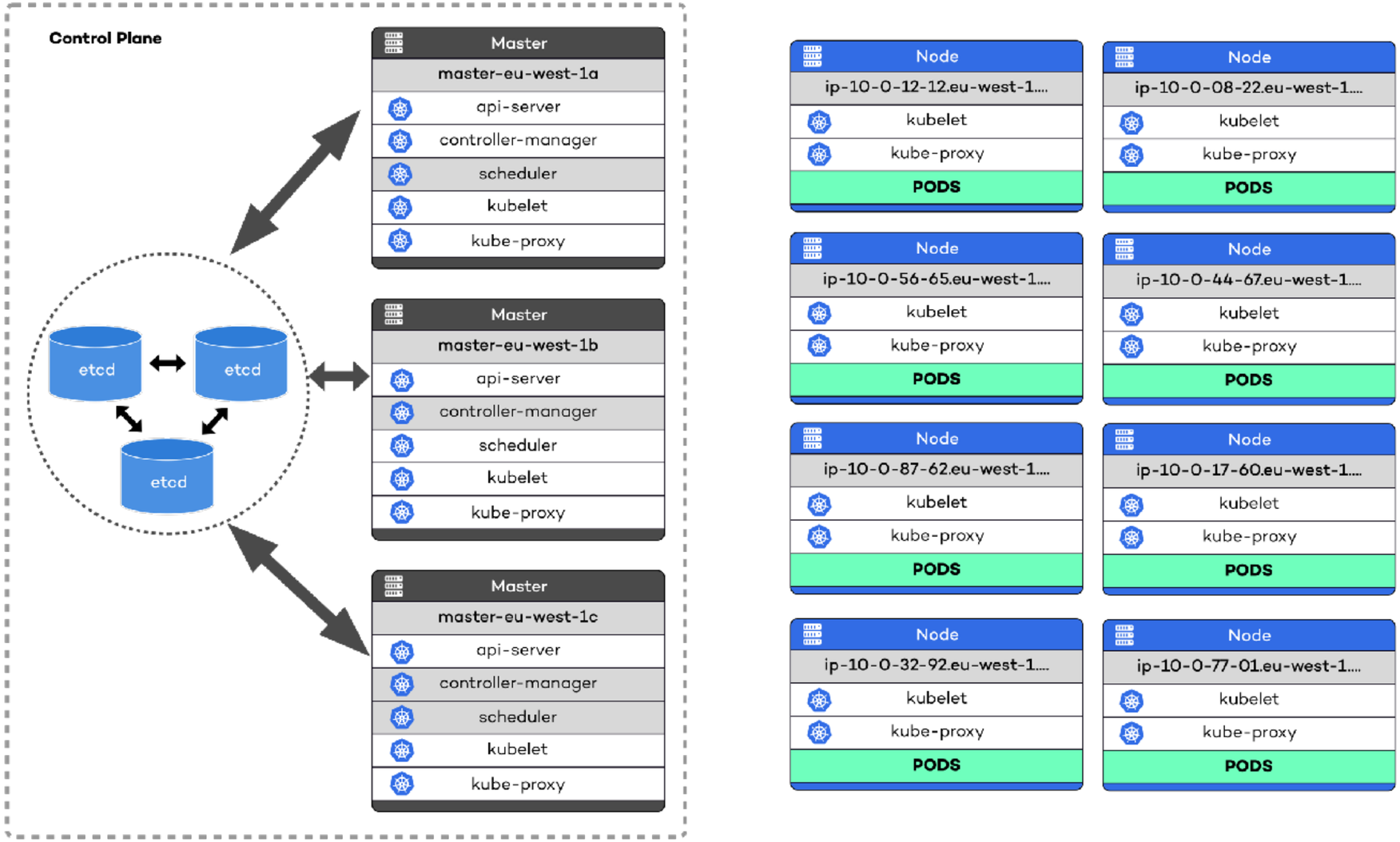
Value Proposition

- Increased developer and operator productivity
- Better resource utilisation (bin packing)
- Resilience

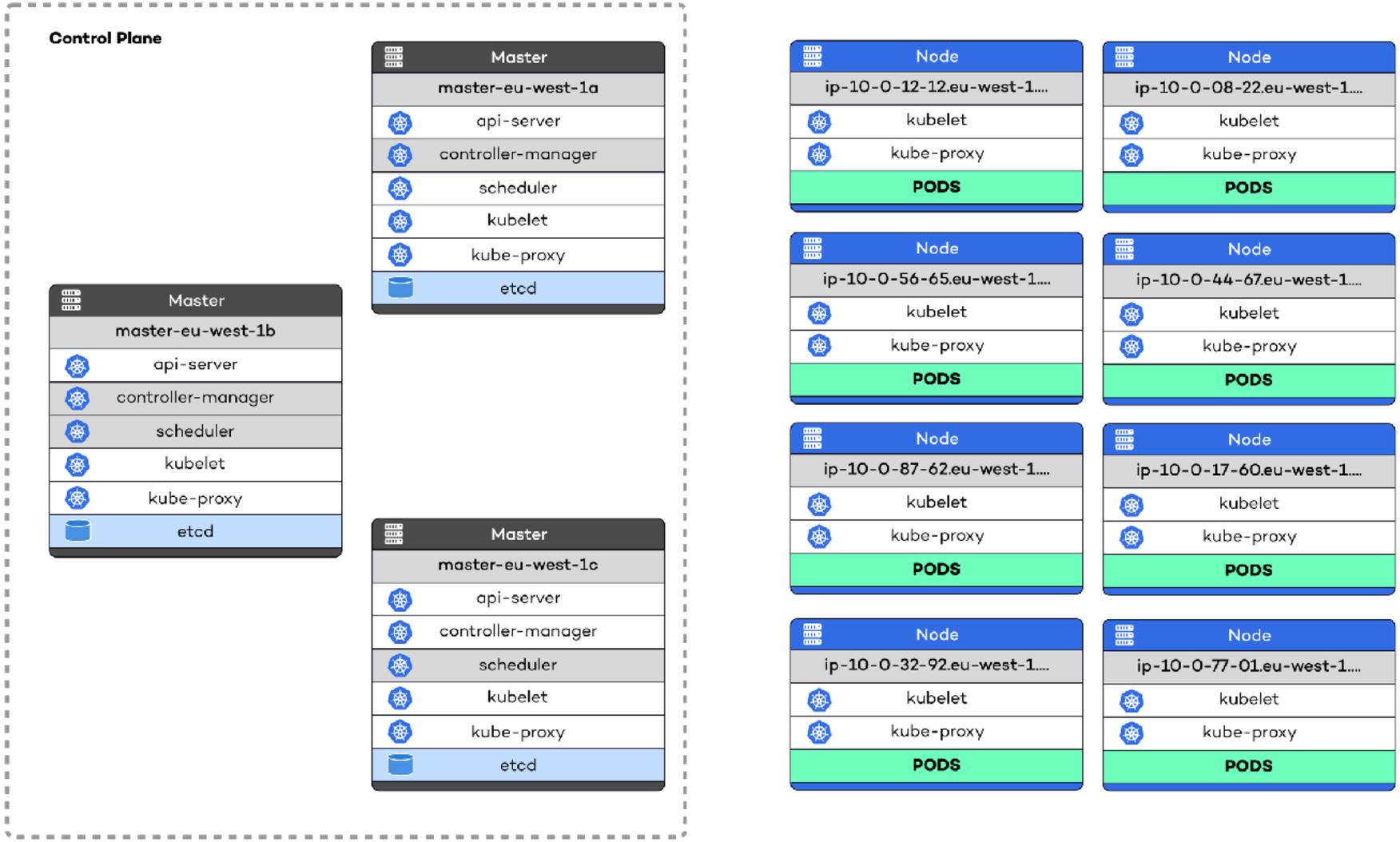
Kubernetes at a high-level



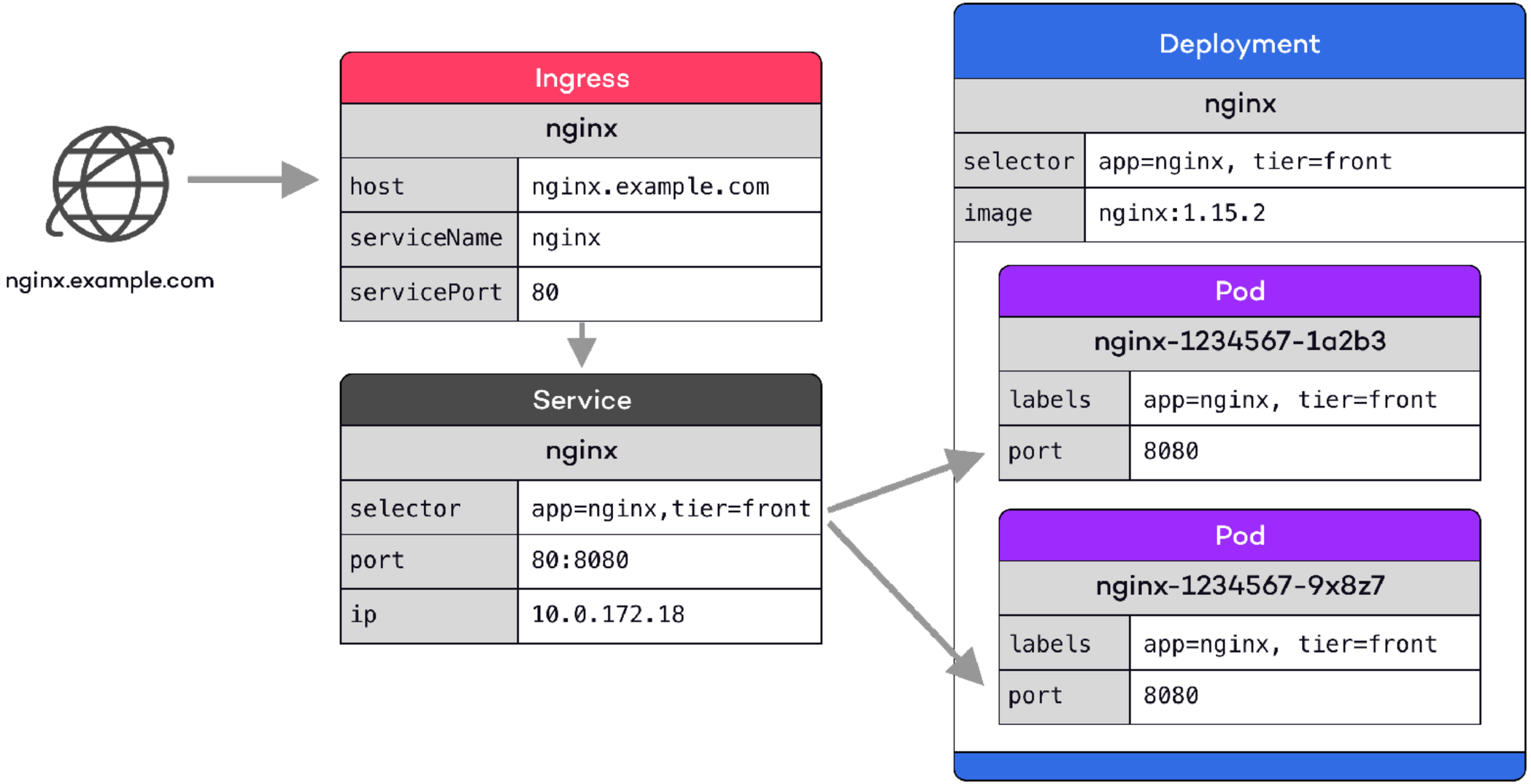
Highly Available Kubernetes



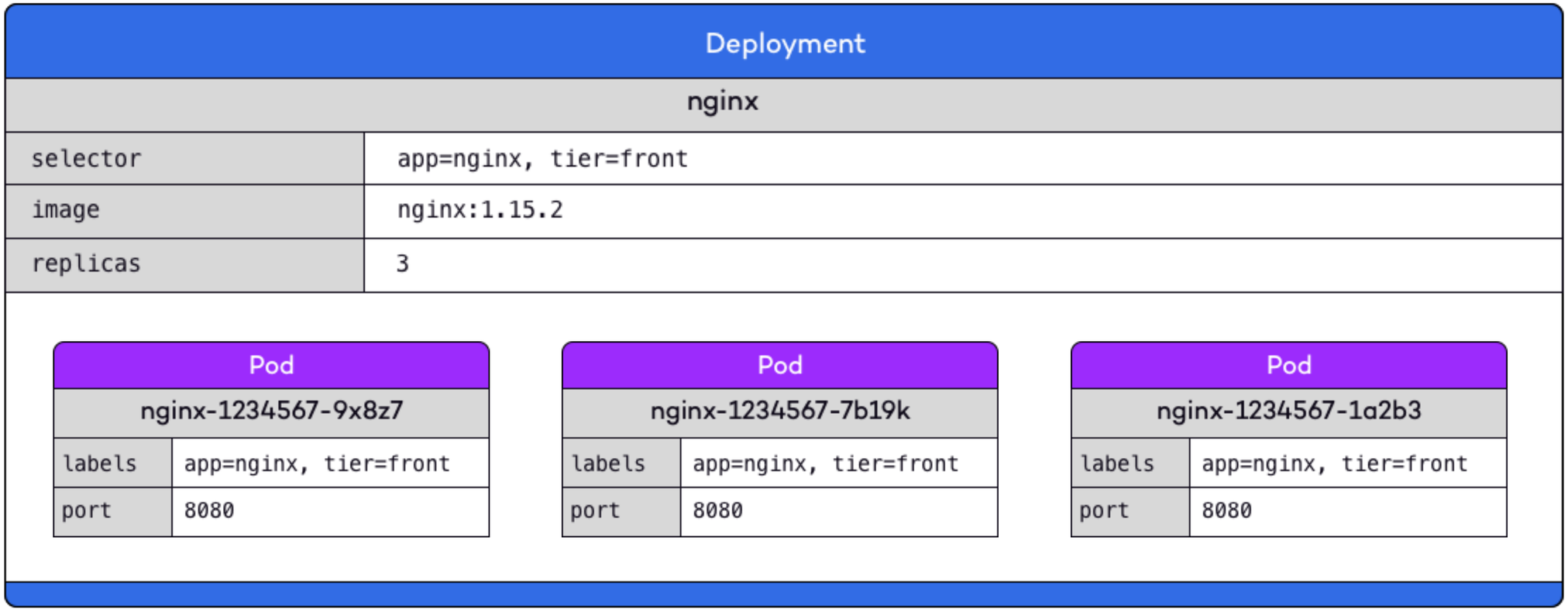
Highly Available Kubernetes



Core Concepts



Declarative Reconciliation Loop

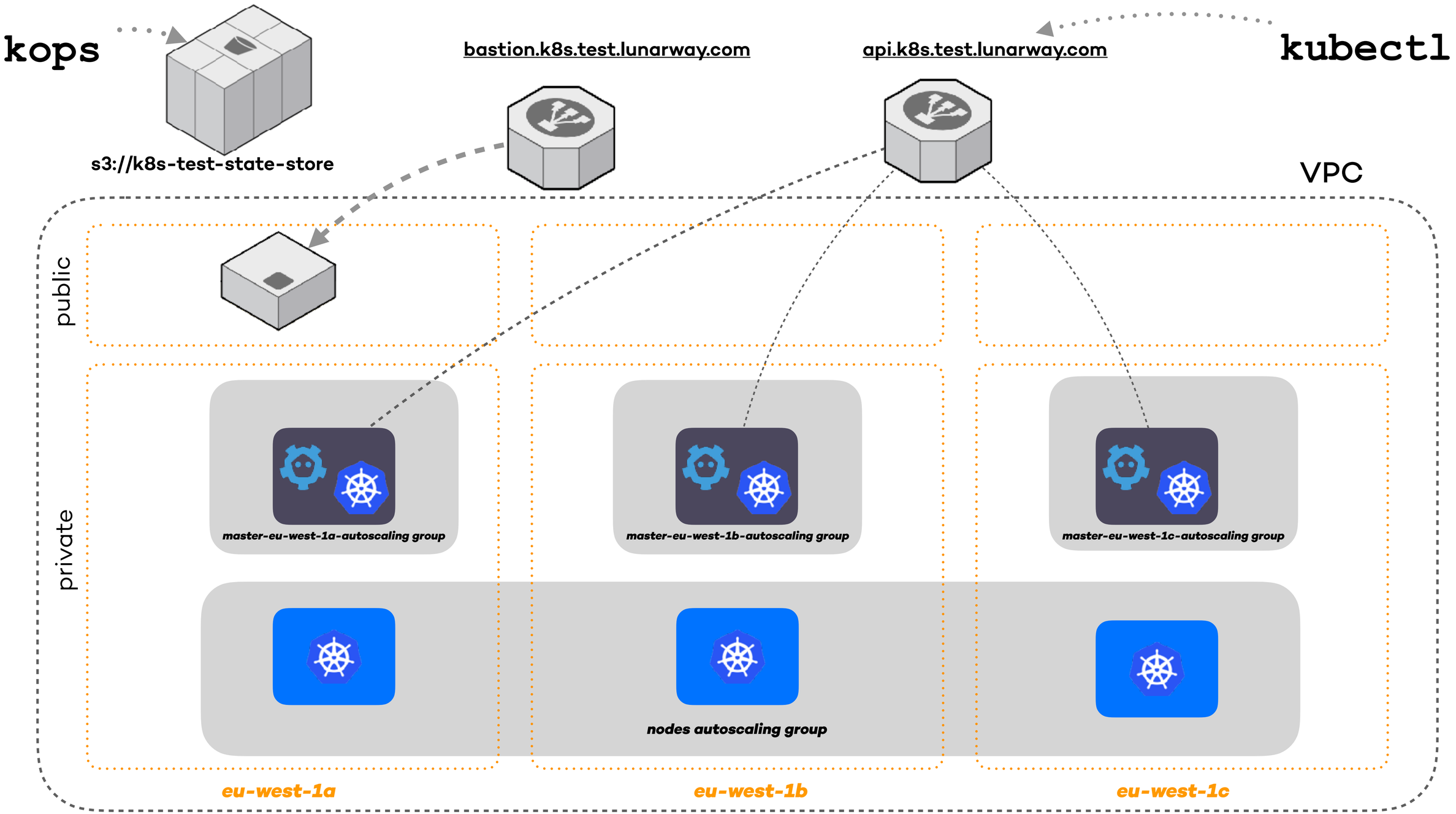


Kubernetes at Lunar Way

- Kubernetes in production since March 2017
- Three clusters at the moment (dev, staging, prod)
- Kubernetes Operations (Kops) with quite a lot of configuration
- Production environment is a multi-master highly available cluster
- Started at Kubernetes 1.5 and are now at 1.9.6

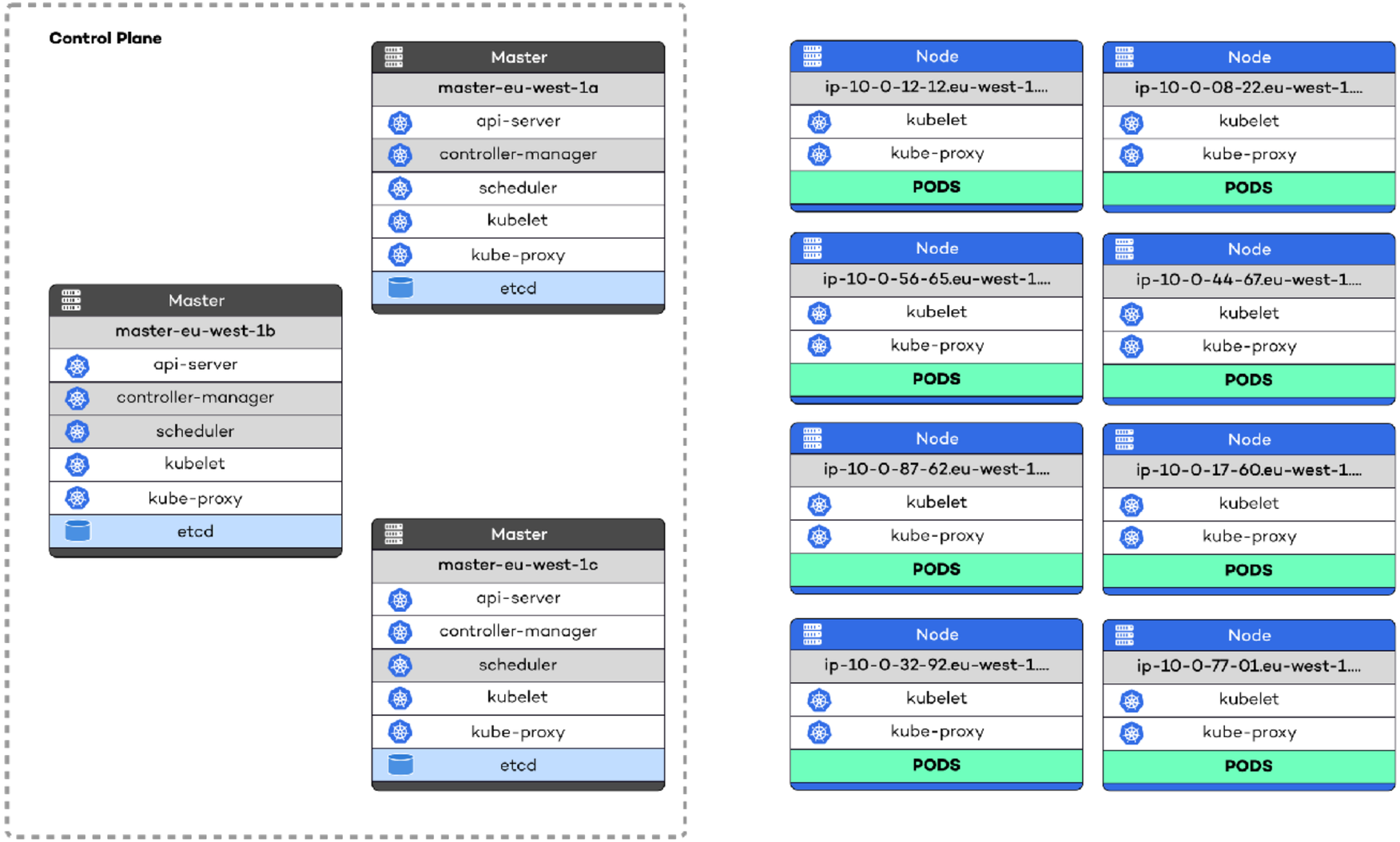
Kops

```
kops create cluster \
  --name dev.example.com \
  --state s3://some-s3-bucket \
  --node-count 3 \
  --zones eu-west-1a,eu-west-1b,eu-west-1c \
  --master-zones eu-west-1a,eu-west-1b,eu-west-1c \
  --dns private \
  --node-size m4.large \
  --master-size m4.large \
  --topology private \
  --networking weave \
  --yes
```


kops
s3://k8s-test-state-store
bastion.k8s.test.lunarway.com
api.k8s.test.lunarway.com
kubectl
VPC
public
private
master-eu-west-1a-autoscaling group
master-eu-west-1b-autoscaling group
master-eu-west-1c-autoscaling group
nodes autoscaling group
eu-west-1a
eu-west-1b
eu-west-1c

Let’s spin up
a cluster!

Highly Available Kubernetes



Declarative Cluster Spec

```
apiVersion: kops/v1alpha2
kind: Cluster
metadata:
  name: k8s.test.lunarway.com
spec:
  api:
    loadBalancer:
      type: Public
  authorization:
    rbac: {}
  channel: stable
  cloudProvider: aws
  configBase: s3://somebucket/k8s.test.lunarway.com
  dnsZone: DNSZONE
  etcdClusters:
  - etcdMembers:
    - instanceGroup: master-eu-west-1a
      name: a
    name: main
  - etcdMembers:
    - instanceGroup: master-eu-west-1a
      name: a
    name: events
  kubeAPIServer:
    auditLogMaxAge: 30
    auditLogMaxBackups: 10
```

Kops - Pros/Cons

Pros

- Very easy to spin up clusters
- Highly configurable
- Declarative cluster specifications
- Possible to output to terraform if needed

Cons

- Previously pretty bad defaults
- Release cadence is lacking a couple of months behind upstream Kubernetes

Kubernetes - Pros/Cons

Pros

- Allow us to easily deploy services independently
- Rolling back is fast
- Provides us with resilience
- Makes management of services easy and immutable

Cons

- Very complex system
- Sometimes to high velocity

Other interesting tales

- First upgrades of the productions clusters caused a 30 min outage, because of network congestion fetching pods
- Some services where just moved to this dynamic environment, not handling termination very well
- Problems with kubelet increasing resource consumption

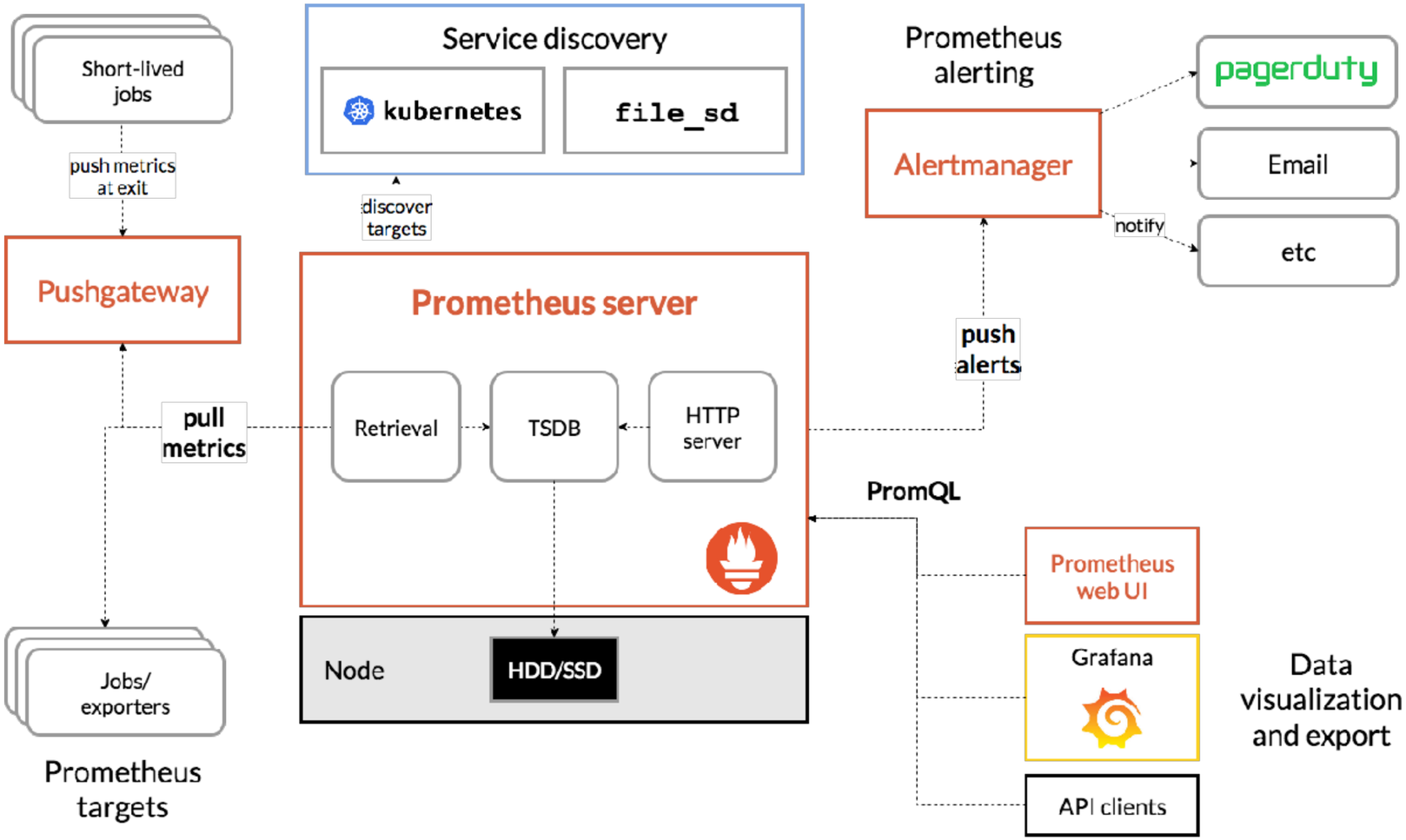
Monitoring with Prometheus

https://www.pexels.com/photo/men-working-at-night-256219/

What is Prometheus?

- Monitoring system and Timeseries Database
- Instrumentation
- Metrics collection and storage
- Querying
- Alerting
- Dashboard / Graphing / Trending

Prometheus, an overview



Why Monitor?

- Analysing long-term trends
- Comparing over time or experiment groups
- Alerting
- Building dashboards to gain insights
- Conducting ad hoc retrospective analysis

Basically, being able to find out what is broken and why...

and ... even better... know it before it impacts customers..

Let’s set up some monitoring of our cluster!

Why do we need it?

- We need insights into all the moving parts of our infrastructure
- We need to monitor error rates, saturation, latency, etc.
- We need a dynamic and powerful query language
- We need a way to get alerted when things go wrong

Prometheus at Lunar Way

- Prometheus provides insights in to all our different systems running, whether it's services running in Kubernetes or outside
- We currently run Prometheus in our Kubernetes cluster
- It runs as a simple deployment, and managed with our own set of yaml configurations

Prometheus - Pros/Cons

Pros

- Provides great insights to all of our services
- Makes it easy for developers to instrument their services
- Integrates well with many different services

Cons

- Prometheus do not support clustered setup

Other interesting tales

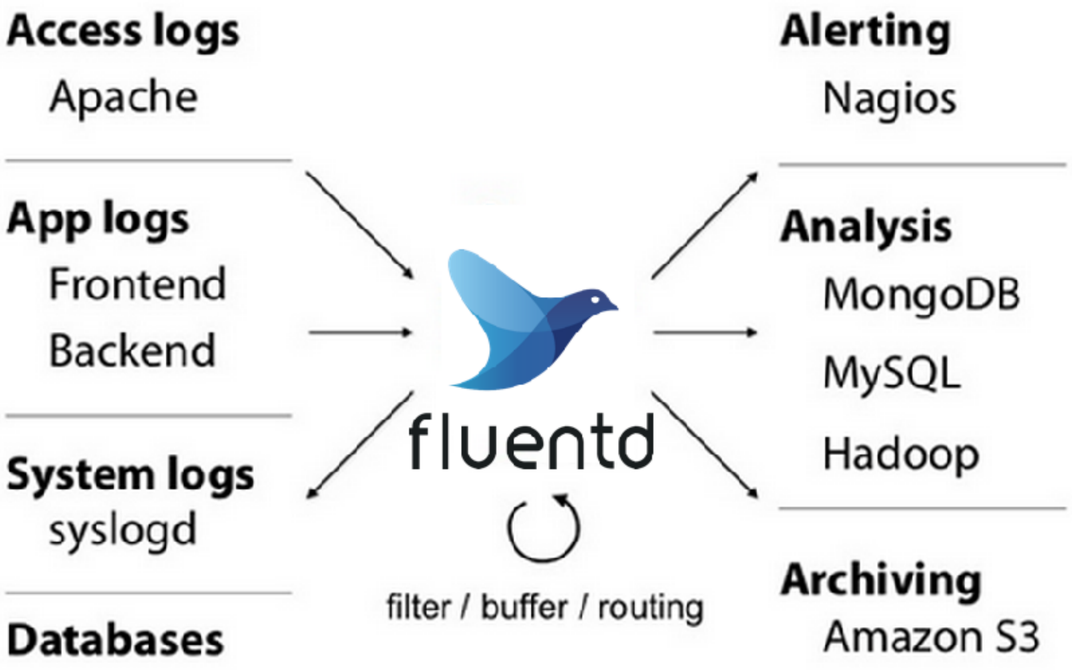
- We've had many internal discussions on when to use logs and when to use metrics
- Before Prometheus 2.0 we had a lot of difficulties with high memory consumption
- We write exporters internally for monitoring external partners

https://www.pexels.com/photo/art-classic-contemporary-design-345046/

Log Management with fluentd and Humio

fluentd, what is it?

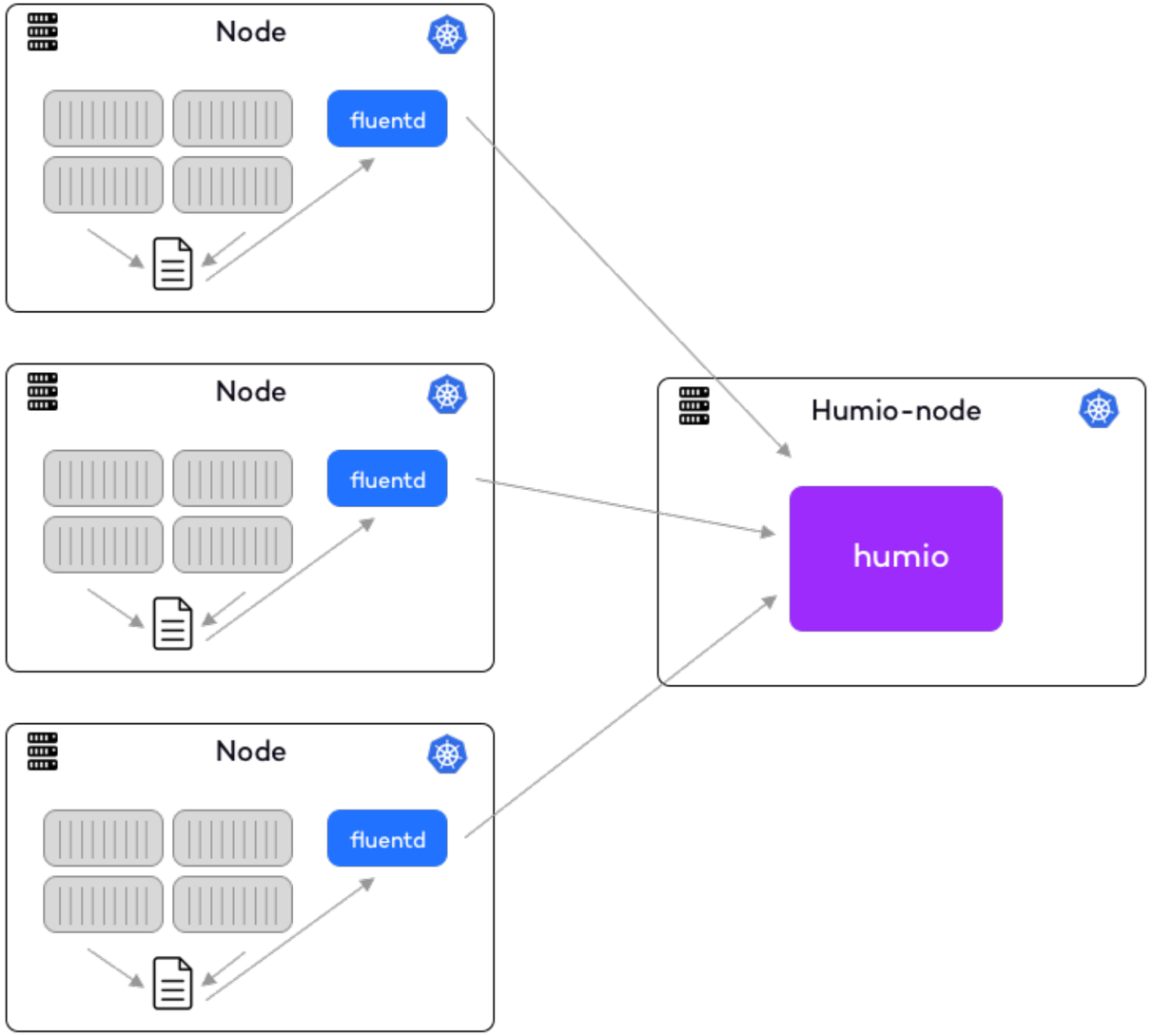
- Fluentd is a log collector
- Hosted by the CNCF



humio, what is it?

- Humio is a log management solution (unfortunately not open-source)
- Humio provides a simple and developer friendly query language for getting insights into your logs

Architecture overview



fluentd/humio - why this combo?

- fluentd is easy to get up and running
- fluentd makes it fairly easy to add additional data and parse up structured fields
- fluentd comes with a build in buffer mechanism in case of connection failures
- Humio provides an elasticsearch ingest api which works great with fluentd
- The most important reason for us choosing Humio, is it's developer friendly query language and speed

Moving towards fluent bit

- Our fluentd setup has become quite advanced, and a lot of different parsing is happening at this layer
- fluentd uses quite a lot of resources on each machine
- We have seen fluentd spike to 1.5gb memory consumption
- We wan't to move parsing further up the stack (in Humio)

For many of these reasons we are looking at fluent bit as an alternative!

Let’s start
reading our logs!

What's next?

- Utilising Custom Resource Definitions and a controller to do Release Management
- Services Mesh
- Adopting more Operators to ease operations of e.g. Prometheus.
- Provide a FaaS on top of Kubernetes
- Extend Kubernetes with virtual-kubelet for additional serverless resources

Wrapping up

Key takeaways if entering kubernetes land

Kubernetes is complex and has a steep learning curve, but it enables so many possibilities

Monitoring and alerts are very important in such a dynamic environment, but be aware of alert fatigue

Read your logs and make them easily accessible for all your developers

lunar way®

Questions?

Thank you!

@phennex