Highly scalable SaaS Apps on Oracle Kubernetes Engine (OKE) : Real Life Case Studies

Practical Experiences with OKE in Production at Scale

Ram Kailasanathan
Senior Director
Product, Oracle
Cloud Native

Richard Bair
Senior Director
Engineering, Oracle IoT
Apps and BlockChain

Oracle Cloud Infrastructure: Complete set of services

Governance
IAM, Tagging, Cost Analysis

Security
IAM, Audit, KMS, CASB

Management
Monitoring, Notifications, Alarms

Automation
Resource Manager, Ansible

Analytics / Integration / SOA Suite / Identity / Management / Content / API Platform / Developer / Visual Builder / Digital Assistant/ *DataFlow / Data Science / Data Safe*

Containers
Containers and Kubernetes

Fully managed, certified Kubernetes service with Docker containers

Data Movement
Storage appliance, Data Transfer

Software NAS gateway, data ingest service with full chain of custody (HDD or appliance)

Autonomous Database
Transactions, Data Warehouse

Fast provisioning. Automatic tuning, patching, securing. 99.995% availability.

Cloud Native
Events, Streaming, Functions

Fully-managed FaaS, event-triggered functions, high-volume data ingest, notifications

Compute
Bare metal/VM, CPUs/GPUs

Up to 64 CPU cores, 8 GPUs, 768 GB RAM, 51 TB local NVMe SSD, 5M IOPS, AMD and Intel processors

Storage
NVMe, Block, File, Object, Archive

Predictable IOPS Block Storage for up to 98% less, storage for whole lifecycle

Database
Bare metal, VMs, Exadata

Millions of TPS; Full RAC and Active Data Guard support

Networking
VCN, LBaaS, FastConnect, VPN

Isolated networks with reserved IPs, security lists, firewalls, lowest cost private connectivity

Public regions

Government regions

Copyright © 2020, Oracle and/or its affiliates. All rights reserved.

Oracle Strategy for Cloud Native

Complete Cloud Native Stack

Deliver tools and services that are complete, integrated and open

- Continuous Integration & Deployment, Container Registry, Orchestration /Scheduling, Management /Operations, Analytics /Introspection
- With an application development platform for serverless and microservices

Open Source

Actively participate in community driven open source container technologies

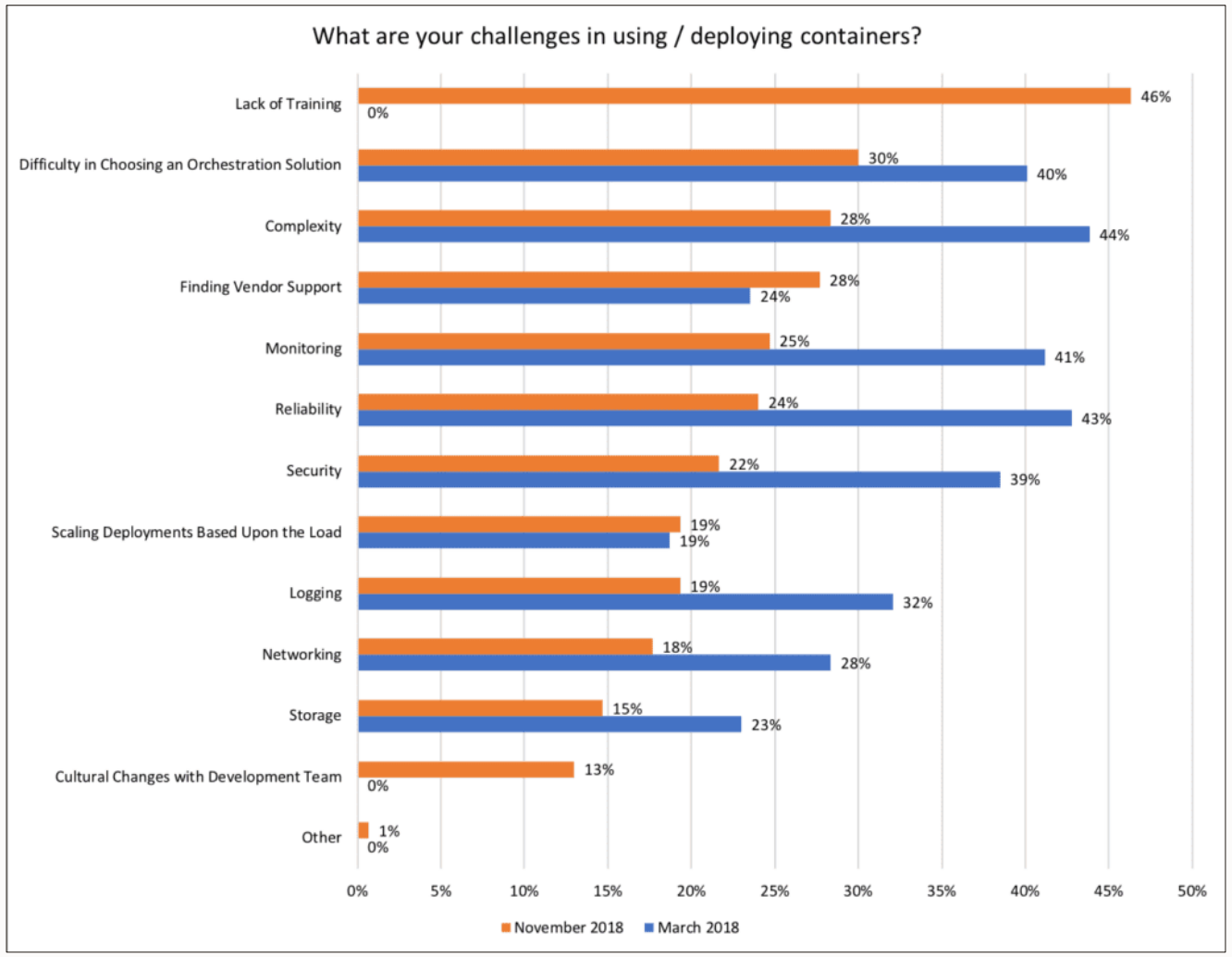
- Investing in Kubernetes, Docker, Fn, & CNCF, DevTools, and DevOps, with engineering resources, code contributions & sponsorships
- Active support from Oracle's portfolio of open source assets (Java, etc.)

Managed Services

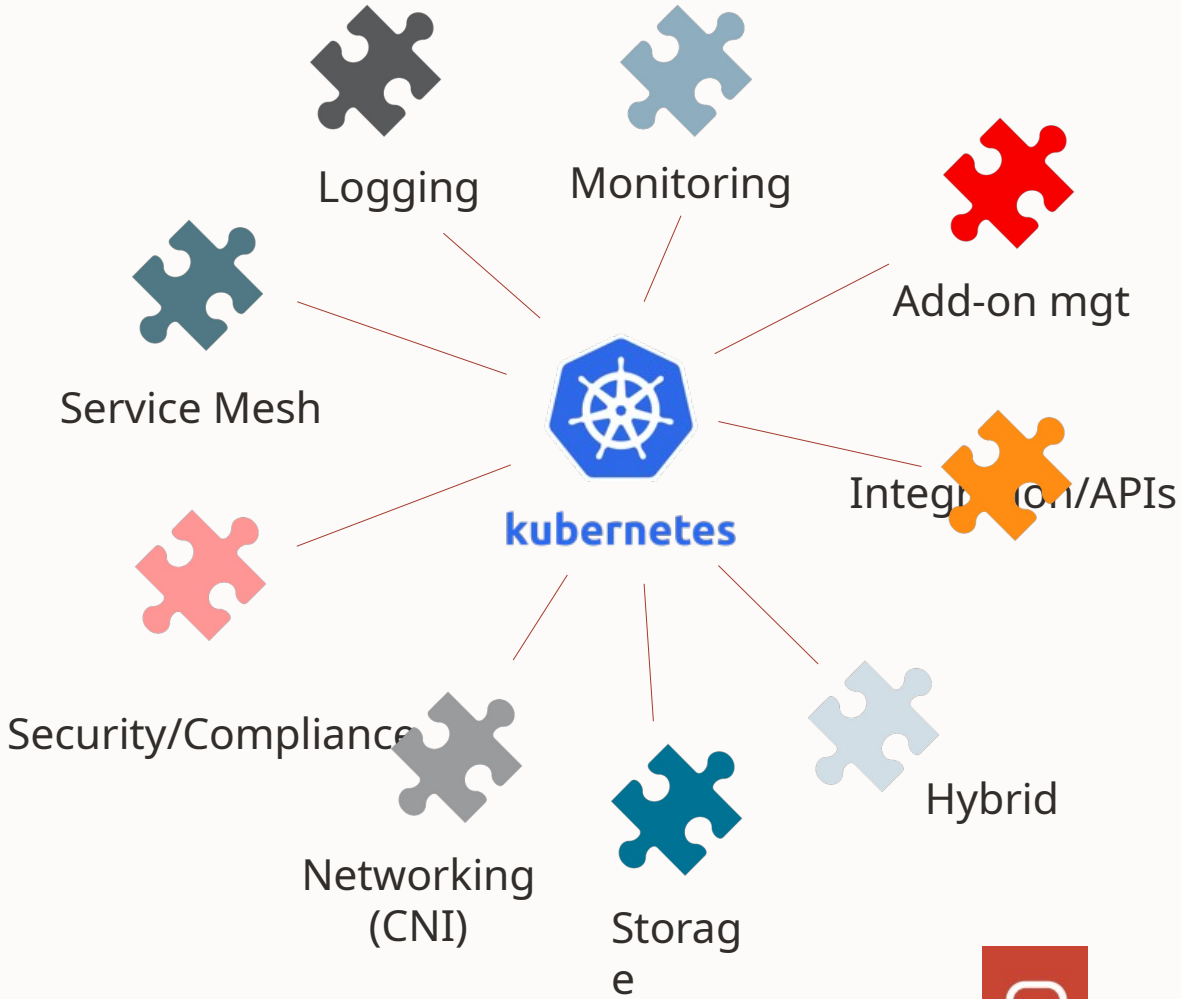
Differentiate on quality of implementation, service and operational excellence

- Full, transparent management
- Open Source Compatible
- Standards **compliant**
- Deployed to Oracle Cloud Infrastructure
- Enterprise grade performance, **security**, HA, and governance

Kubernetes – Necessary but not sufficient



Source: CNCF Survey



Comprehensive Developer Services Portfolio

Easy to Start
(Adopt)

App Development
(Develop)

Deploy

Operate

Console

QuickStart

SDK/CLI

Marketplace

Always Free Tier

API Design* apiary

Cloud Shell

Resource Manager

Source Control*

Container Registry

Cloud Native Java

helidon GraalVM ORACLE Coherence

CI/CD*

API Gateway

Container Engine (OKE)

Functions

Streaming

fluentd

CLOUD NATIVE COMPUTING FOUNDATION

cloudevents

Monitoring

Logging

Events

Notification

Email

OCI Core Infrastructure Services

*Under Development

OKE @ Scale

Big Clusters, and a lot of ‘em

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

17,000+

Cores

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

50+

Clusters

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

OKE all the way

1. Started using OKE ~ late 2018
2. We use OKE on OCI for all dev, test, staging, and production workloads
3. The OKE team has been a very good partner in allowing us to run at scale
4. OKE makes Kubernetes a **lot** easier than running it ourselves

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Lesson #1

Things die. Deal with it.

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Kubernetes provides all the tools for high availability

1. Deployments and Stateful Sets try to maintain a specified number of *replicas*
2. Kubernetes monitors each pod's *readiness* and *health*. Unhealthy pods are automatically replaced
3. *Anti-affinity* rules can be defined to cause Kubernetes to place pods into different availability / fault domains
4. *Pod disruption budgets* will prevent Kubernetes from bringing down more pods than your application can handle

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Things die. Deal with it.

A pod may die if:

- The pod itself becomes unhealthy (stops responding, or runs out of memory, etc)
- Somebody kills the pod manually (*kubectl delete pod*)
- The cluster was scaled out and you want to rebalance the workload
- The node that the pod is on dies

Why would a node die?

- Upgrading the VM image requires recreating the node
- Upgrading the kubelet requires recreating the node
- It just stops working.*

* This happens to us a lot with X6 shapes.

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Lesson #2

Persistent pains aren’t just in old age.

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Persistent volumes caused us persistent problems

1. Kubernetes pods run on Virtual Machine nodes. Each OCI VM has a limit on the number of attached volumes they support. The more persistent volumes, the fewer pods per node.
2. Sometimes we found persistent volumes would get mounted read-only
3. A persistent volume is per-AD. If a pod moves from one AD to another, then it will fail to start because it was unable to attach to its persistent volume.
   - CSI (replaces flex-volume) is designed to help resolve this
4. Persistent volumes have a minimum size of 50GB in OCI
   - This is far too expensive for development, so we setup our own Ceph cluster on some Dense IO bare mental machines in dev so that we could hand out much smaller volumes
5. Too many persistent volumes cause too much API traffic to the block volume service and you will have attach errors
6. I/O intensive tasks with block volumes are slow unless you have very large block volumes and VMs.

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Try using FSS and ephemeral storage

File System Service is NFS for OCI

- If you need a distributed filesystem, you can use an FSS-backed PV and attach it to many pods
- Store heap dumps on an FSS-backed PV

Ephemeral storage for logs and tmp

- Logs are always shipped to ElasticSearch / Kibana ASAP
  - We only need enough ephemeral storage as a buffer
- Every pod in Kubernetes can have ephemeral storage
  - Very useful for very small disk usage requirements
  - Not to be used for storing customer data
  - Ephemeral storage **dies with the pod** and is stored locally on the OKE Node

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Lesson #3

Know Thy Limits

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Every pod should have requests and limits specified

1. A *request* tells Kubernetes what a pod's *minimum* resource needs are. When Kubernetes schedules your pod, it will make sure the node it schedules the pod on has sufficient resources.
   - Requests ensure a pod has the resources it needs
2. A *limit* tells Kubernetes what a pod's *maximum* resource needs are. The pod's CPU will be limited at the Linux Kernel level. If the pod exceeds its memory *limit*, then the pod will die.
   - Limits prevent a pod from stealing resources from other pods
3. If the node begins to encounter pressure, *request* and *limit* are taken into account to decide what to evict first.
4. Problems due to missing limits and requests may not be visible when cluster load is low, but when the load increases, problems start to appear

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Lesson #4

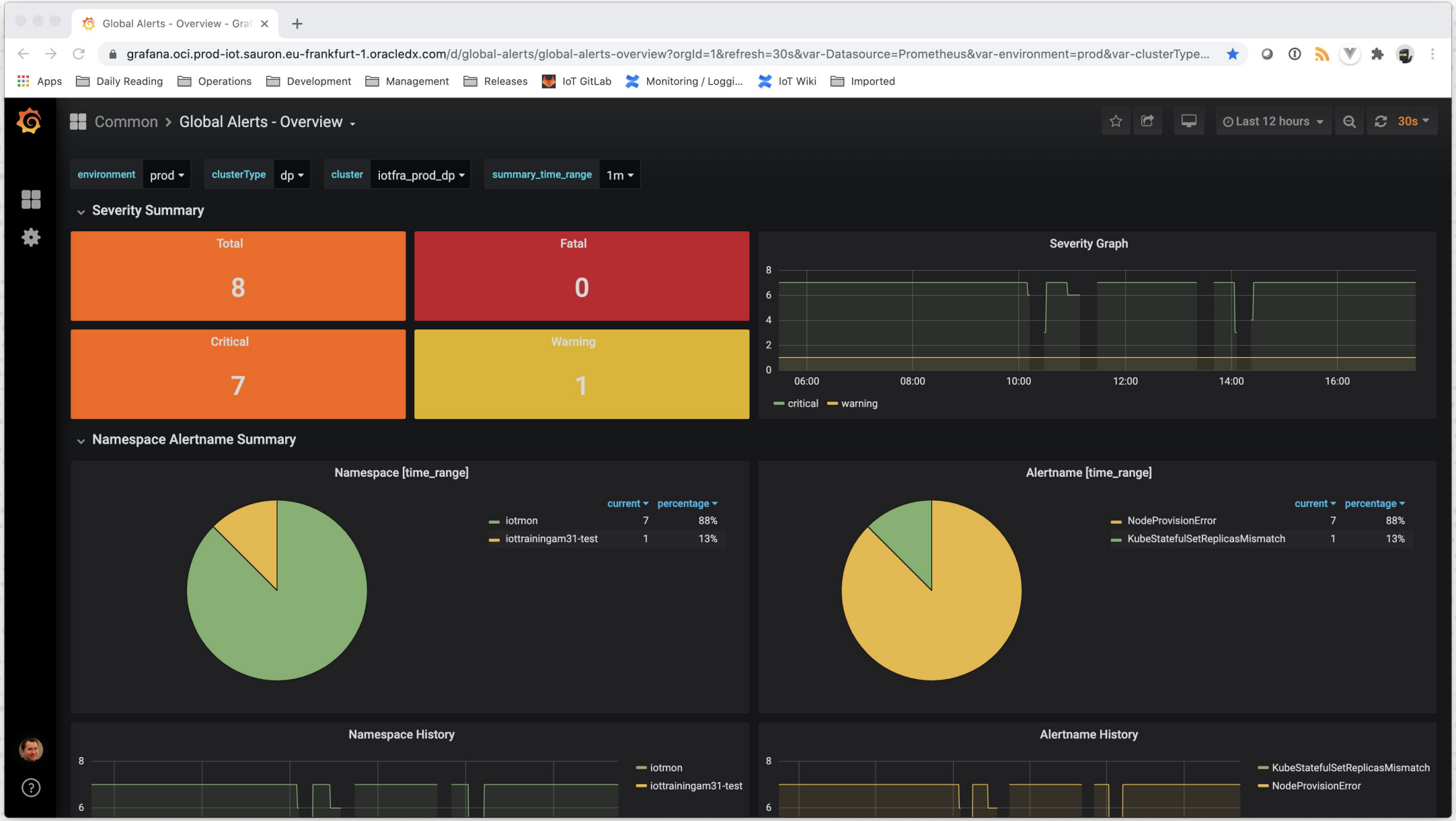
Monitoring is essential

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

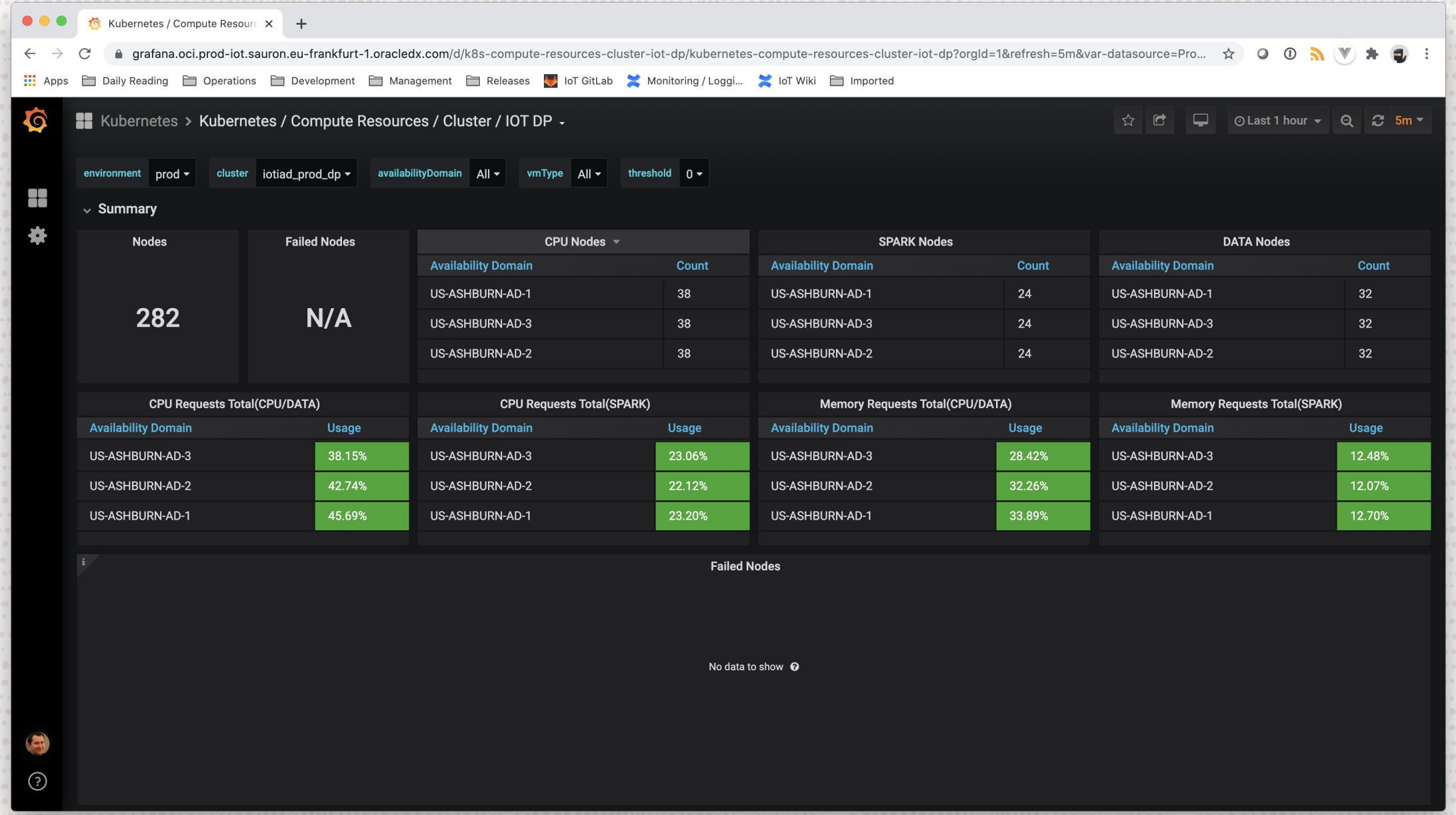
Without good monitoring tools we'd have no chance

1. Every OKE cluster we create runs Prometheus to gather metrics
2. Thanos provides distributed Prometheus queries giving us a single pane of glass for monitoring everything!
3. Single Kibana / ES cluster for collecting all our logs
4. Alert Manager raises PagerDuty alerts based on Kibana / Prometheus data
5. Considering providing some "health check" REST endpoint to give infrastructure **and** application level health data and make it available to monitoring.
   - A pod may be alive, but the underlying application still unhealthy. You need to know this
     - We add health check status to our Prometheus metrics to expose them to Graphana

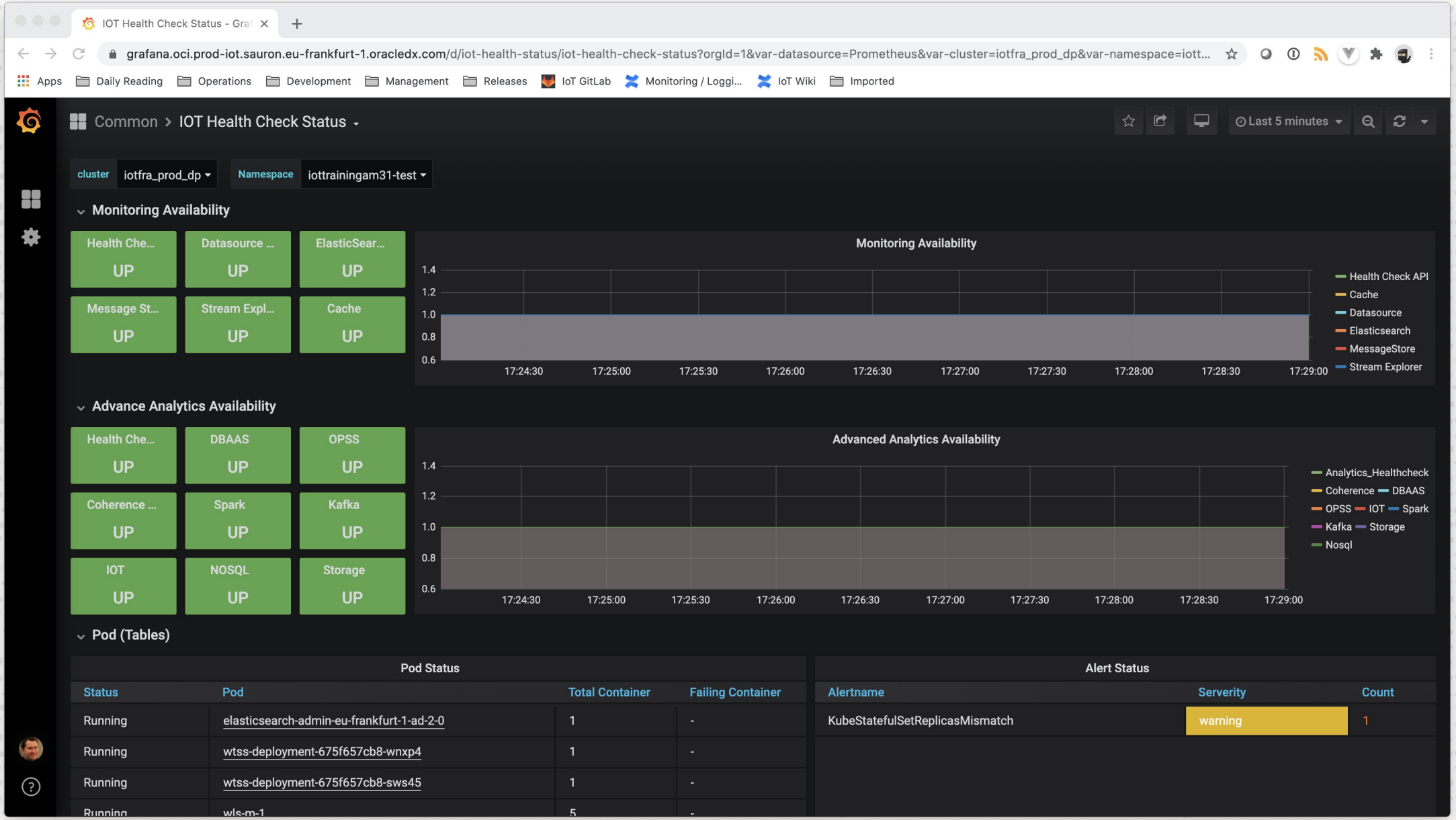
Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted



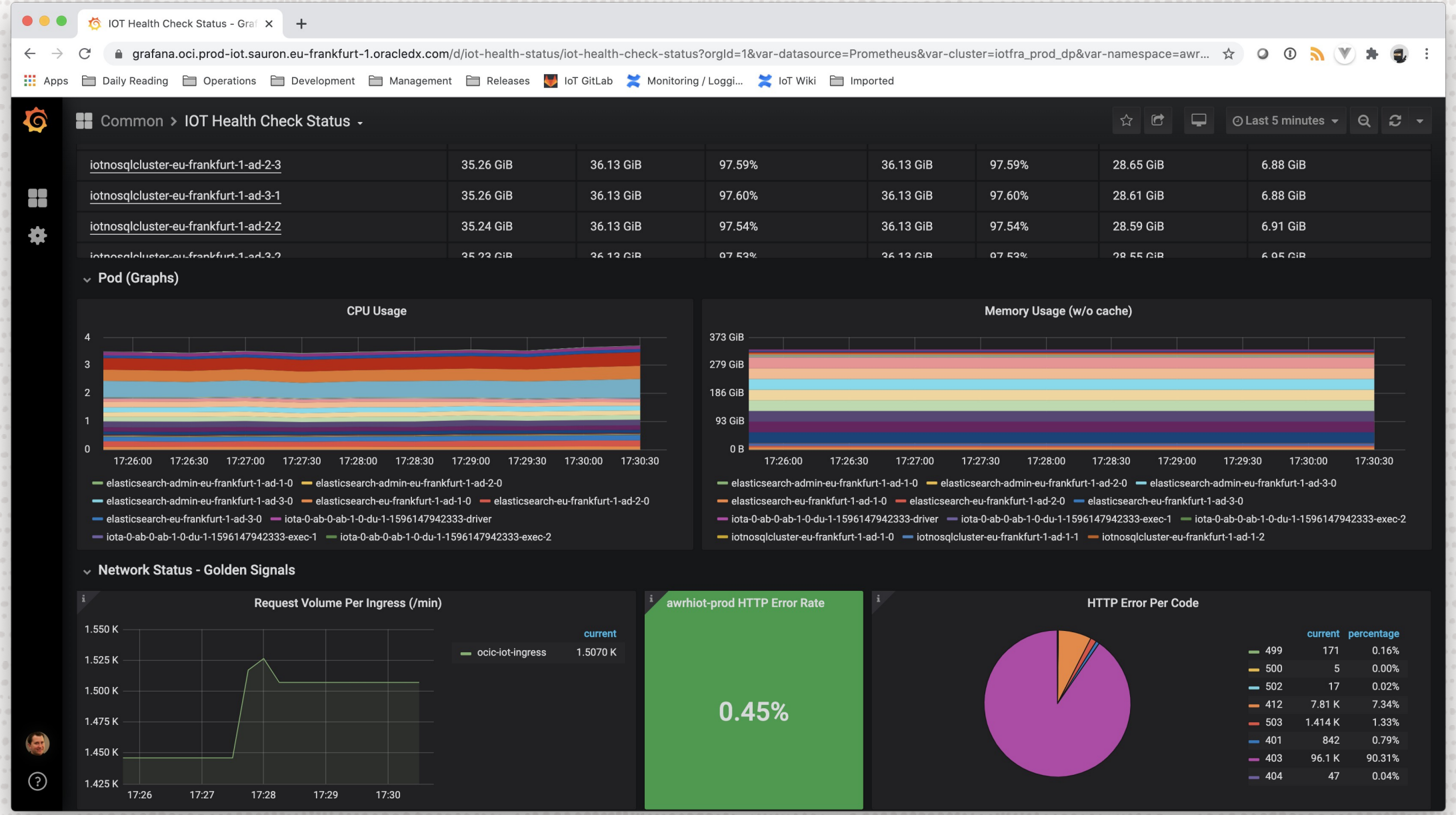
Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted



Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted



Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted



Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Lessons

1. Things die. Design for HA from the beginning
2. Use persistent volumes wisely. They can help, but they come at a cost.
3. Limits and Requests help make things stable.
4. Monitoring is essential

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Do I recommend OKE? Yes!

What does OKE add over Kubernetes?

- Fully managed Kubernetes Control Plane
- Future feature to also have a fully managed Kubernetes Data Plane

What does Kubernetes add over Virtual Machines?

- Managed deployment of Docker containers
- All tools for liveness, readiness, capacity limits, scaling, HA, etc.

What do Virtual Machines add over Bare Metal?

- More variety in VM shapes and sizes
- Better support for HA / DR (e.g. network attached block volumes)

The higher up the stack, the better!

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Resources

1. https://developer.oracle.com
2. https://www.oracle.com/cloud/free/
3. https://docs.cloud.oracle.com/en-us/iaas/Content/ContEng/Concepts/contengoverview.htm
4. https://www.oracle.com/cloud-native/
5. https://docs.cloud.oracle.com/en-us/iaas/Content/Resources/Assets/whitepapers/oci-security.pdf
6. https://www.oracle.com/a/ocom/docs/cloud/oracle-cloud-infrastructure-platform-overview-wp.pdf

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

Thank you

Copyright © 2020, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted