KubeCon | CloudNativeCon

Europe 2022

WELCOME TO VALENCIA

KubeCon | CloudNativeCon
Europe 2022

Kubernetes Persistent Data Challenges – AZ, Region and Multi-Cloud Patterns

Chris Milsted, Ondat
Patrick McFadin, DataStax

Kubernetes Persistent Data Challenges – AZ, Region and Multi-Cloud Patterns

Chris Milsted
Solutions Architect,
Ondat

Patrick McFadin
VP Developer Relations
DataStax

Special thanks to….

For help, assistance and contributing lots of work throughout the talk and demo building process. This talk is what it is because of their help!

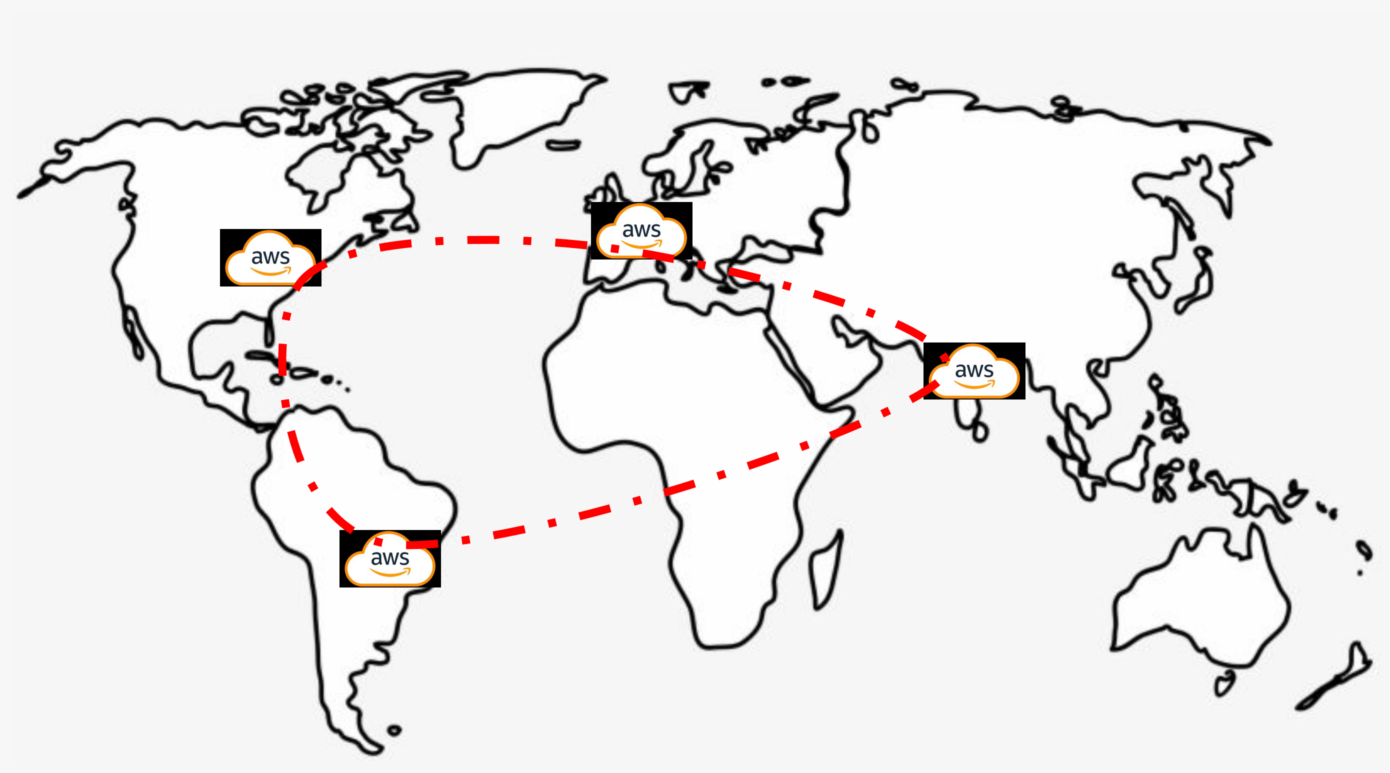
Alex Dejanovski
Software Engineer
Datastax

Raghavan "Rags" Srinivas
Developer Advocate
DataStax

Why Multi-AZ

Failure Type	SLA
Individual VM	99.9%
Hardware	99.95%
Entire Datacenter	99.99%

Why Multi-Region



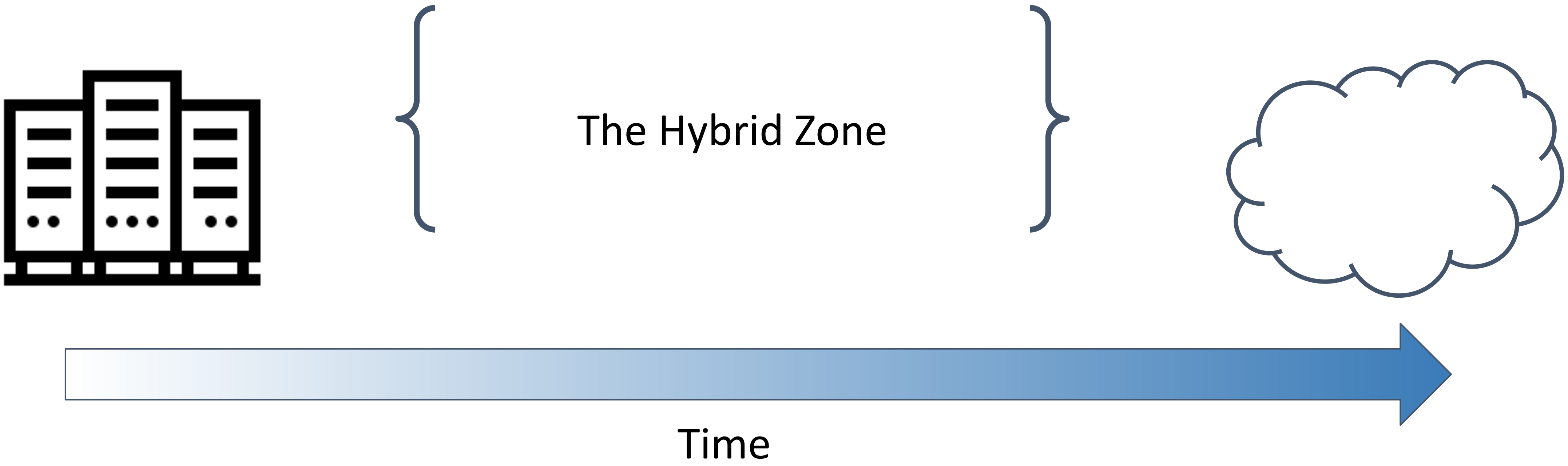
- Closer to your customers
- Regulatory requirements
- Maximum uptime

You have 0% chance for 100% uptime if you are in one region

Why Multi-Cloud

1. Acquisition
2. Migrating to a new provider
3. Another unit in your company doing their own thing

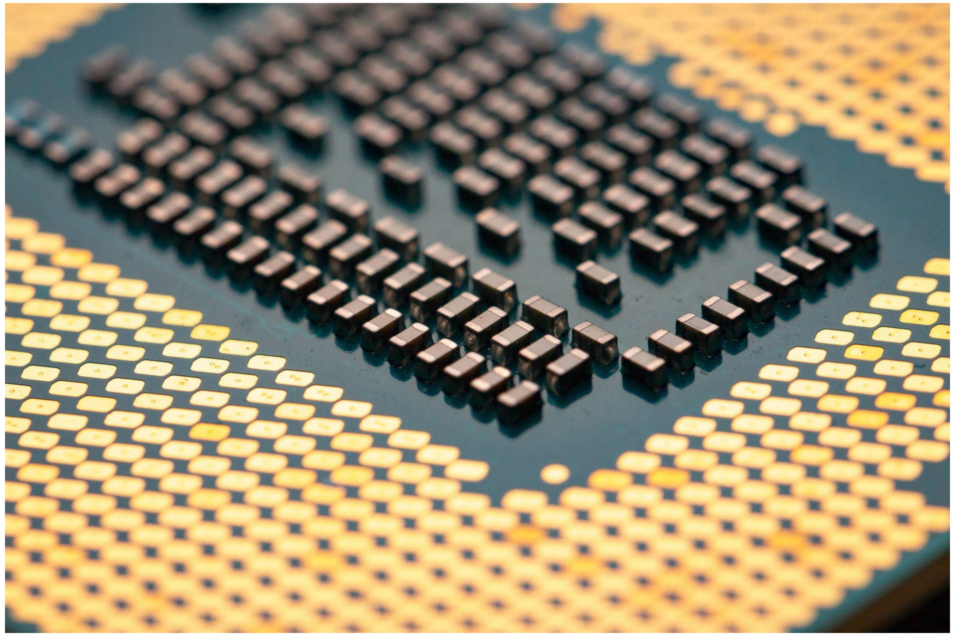
Hybrid?

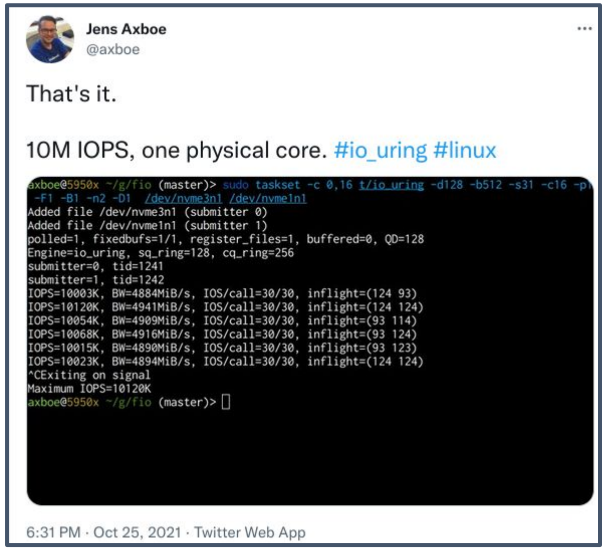


The Hybrid Zone

Time

History lessons - why data in kubernetes





Jens Axboe
@axboe

That's it.

10M IOPS, one physical core. #io_uring #linux

```
axboe@5950x ~/g/fio (master)> sudo taskset -c 0,16 t/io_uring -d128 -b512 -s31 -c16 -p
 -F1 -B1 -n2 -D1  /dev/nvme3n1 /dev/nvme1n1
Added file /dev/nvme3n1 (submitter 0)
Added file /dev/nvme1n1 (submitter 1)
polled=1, fixedbufs=1/1, register_files=1, buffered=0, QD=128
Engine=io_uring, sq_ring=128, cq_ring=256
submitter=0, tid=1241
submitter=1, tid=1242
IOPS=10003K, BW=4884MiB/s, IOS/call=30/30, inflight=(124 93)
IOPS=10120K, BW=4941MiB/s, IOS/call=30/30, inflight=(124 124)
IOPS=10054K, BW=4909MiB/s, IOS/call=30/30, inflight=(93 114)
IOPS=10068K, BW=4916MiB/s, IOS/call=30/30, inflight=(93 124)
IOPS=10015K, BW=4890MiB/s, IOS/call=30/30, inflight=(93 123)
IOPS=10023K, BW=4894MiB/s, IOS/call=30/30, inflight=(124 124)
^CExiting on signal
Maximum IOPS=10120K
axboe@5950x ~/g/fio (master)> 
```

6:31 PM · Oct 25, 2021 · Twitter Web App

People doing it are going faster

RESEARCH REPORT

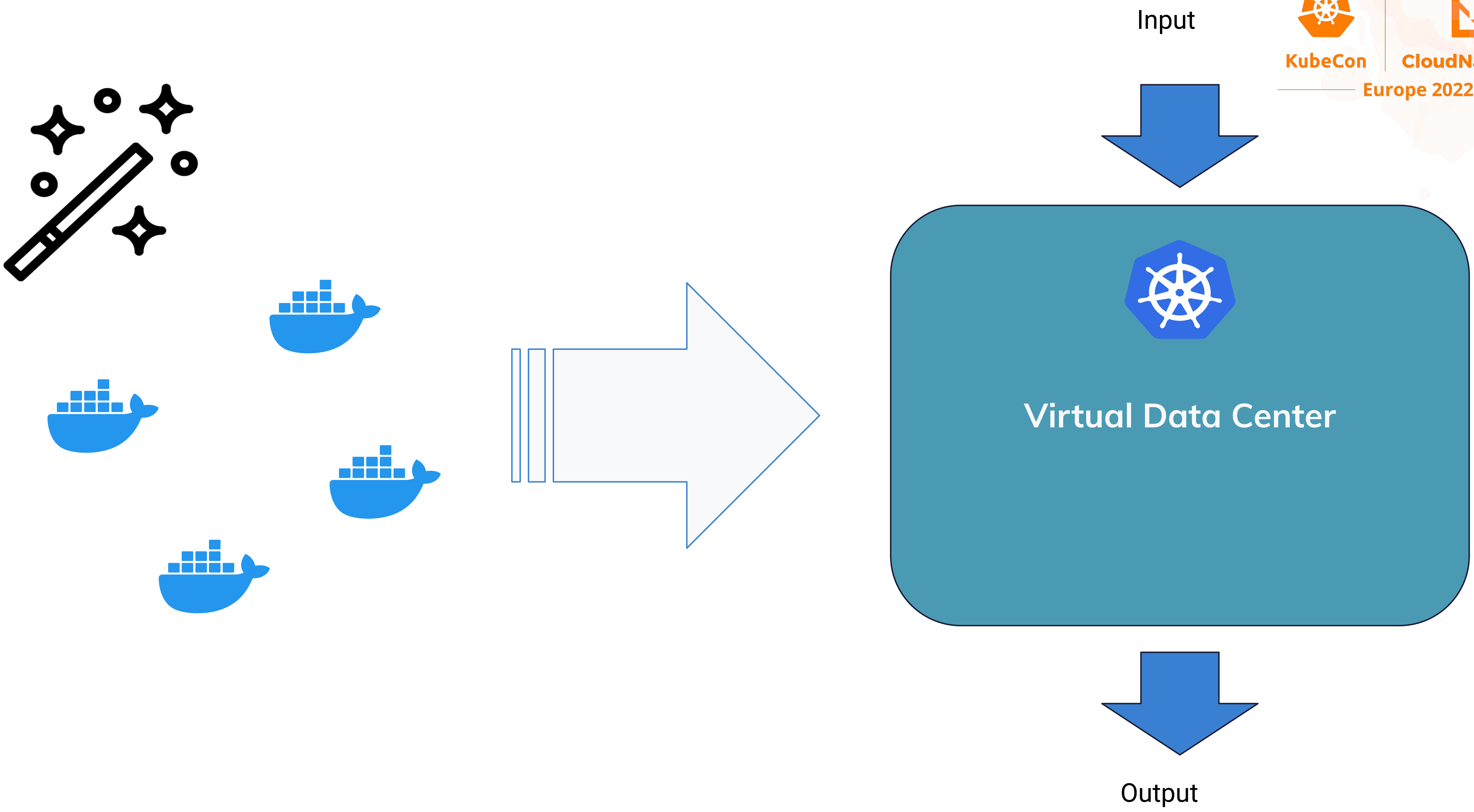
Data on Kubernetes 2021

Insights from over 500 executives and technology leaders on how Kubernetes is being used for data and the factors driving further adoption

Key Findings

- Kubernetes has become a core part of IT – half of the respondents are running 50% or more of their production workloads on it, and they are very satisfied and more productive as a result. The most advanced users report 2x or greater productivity gains.
- 90% believe it is ready for stateful workloads, and a large majority (70%) are running them in production with databases topping the list. Companies report significant benefits to standardization, consistency, and management as key drivers.
- Significant challenges remain. As they seek to expand their data on Kubernetes footprint, enterprises find a lack of integration and interoperability with existing tools and stacks; skilled staff; quality of Kubernetes operators; and trusted vendors.
- Business demands are creating pressures for further adoption. The increasing importance of real-time data to competitive advantage will sharpen companies' need to run data on Kubernetes. A majority believe standards will improve data management and that data should become declarative

https://dok.community/wp-content/uploads/2021/10/DoK_Report_2021.pdf



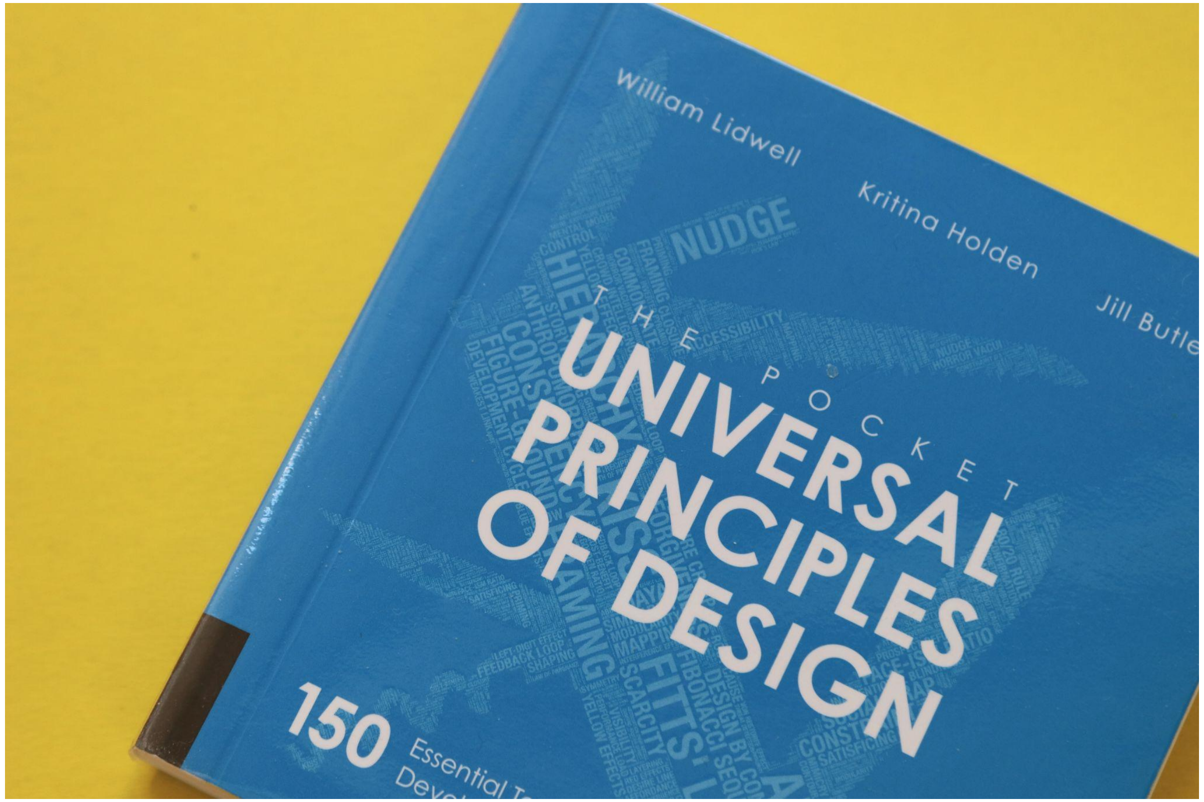
Input

Virtual Data Center

Output

magic by Mask Icon from the Noun Project

High level architectural pattern for k8s clusters



8 Fallacies

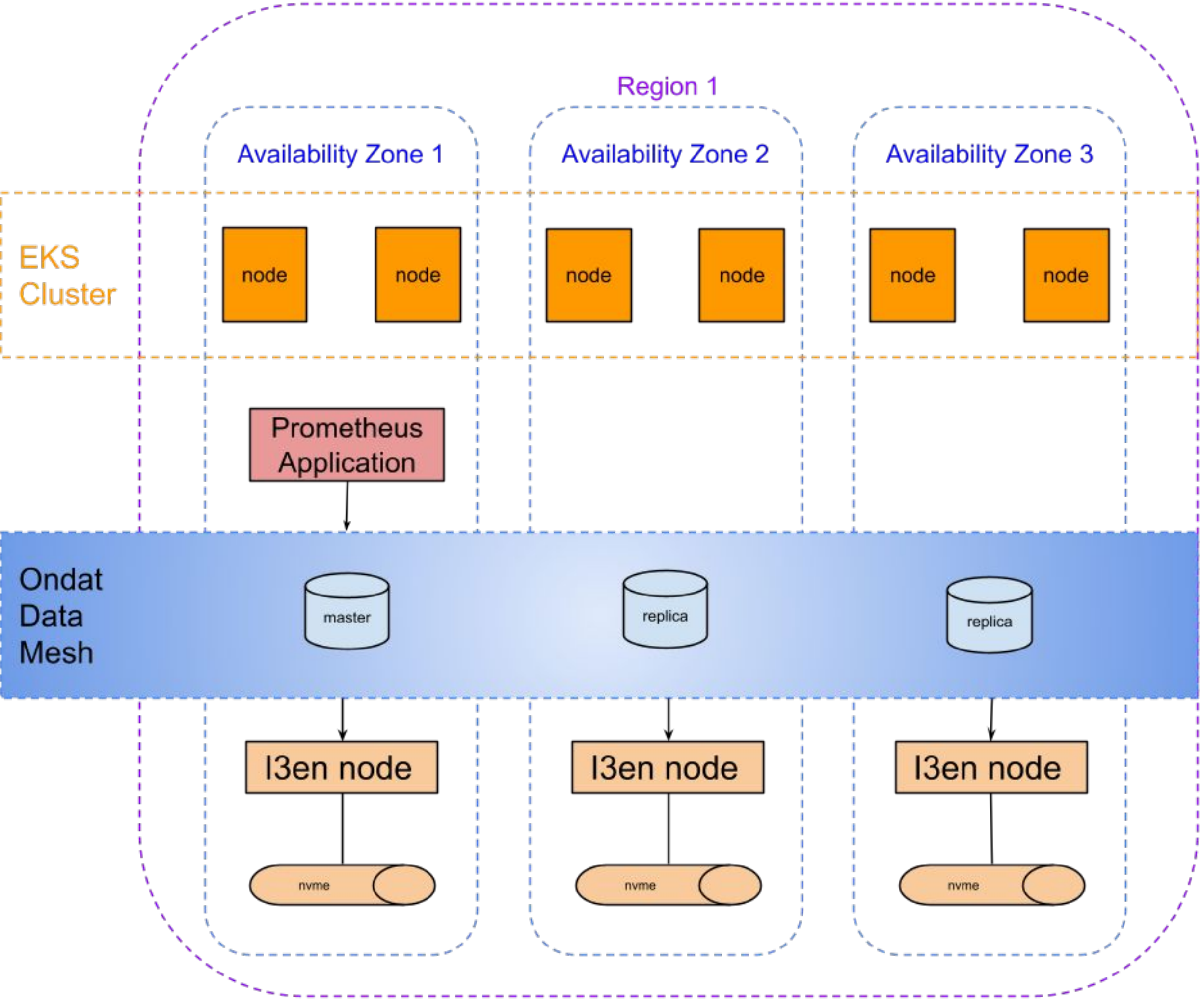
The network is **reliable**.
Latency is zero.
Bandwidth is **infinite**.
The network is **secure**.
Topology doesn't **change**.
There is one **administrator**.
Transport cost is **zero**.
The network is **homogeneous**.

Pick two out of three - CAP Theorem

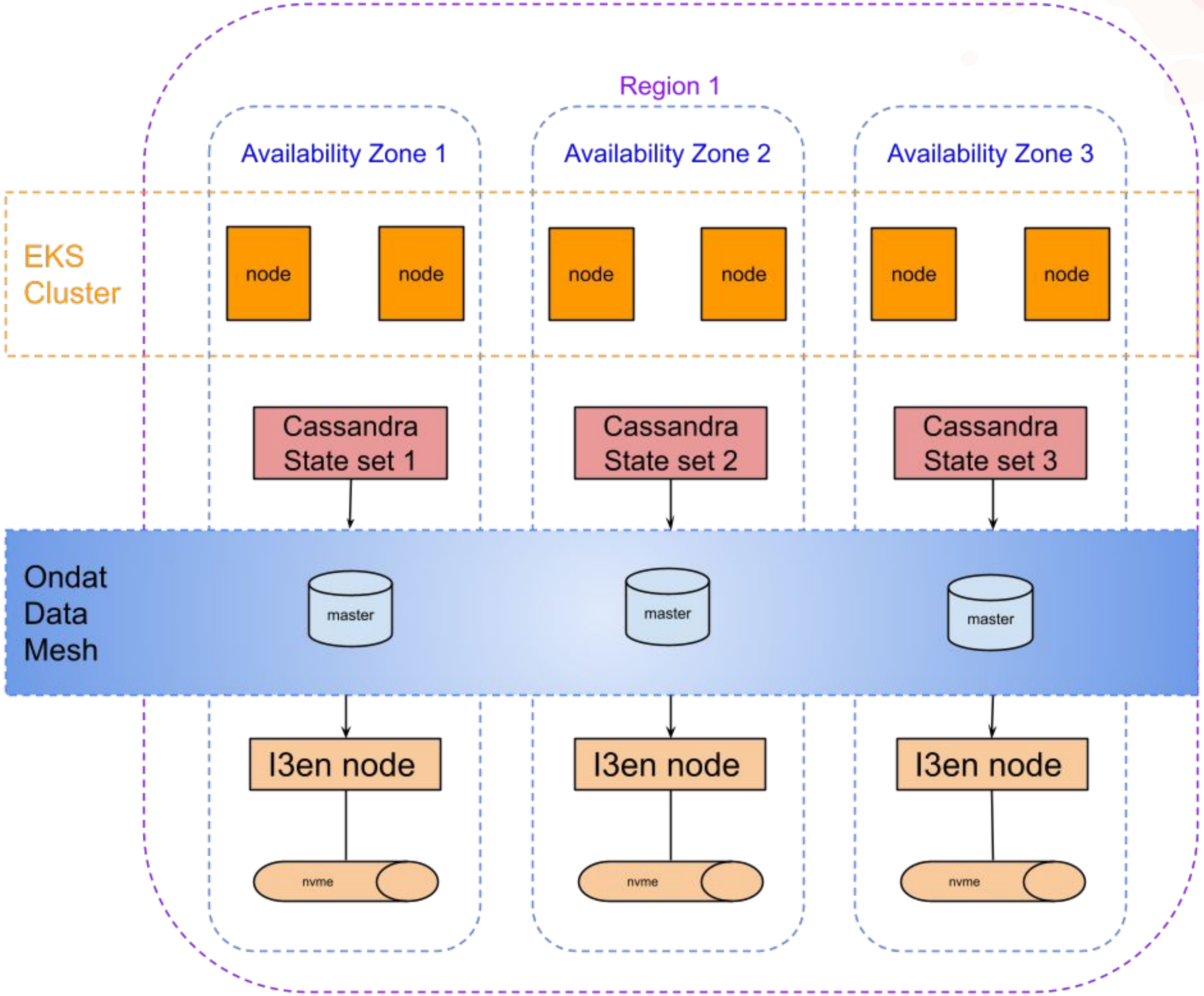
Consistency
Every read receives the most recent write or an error.
Availability
Every request receives a (non-error) response, without the guarantee that it contains the most recent write.
Partition tolerance
The system continues to operate despite an arbitrary number of messages being dropped (or delayed) by the network between nodes.

- https://github.com/cncf/tag-storage/blob/master/Cloud%20Native%20Disaster%20Recovery.pdf

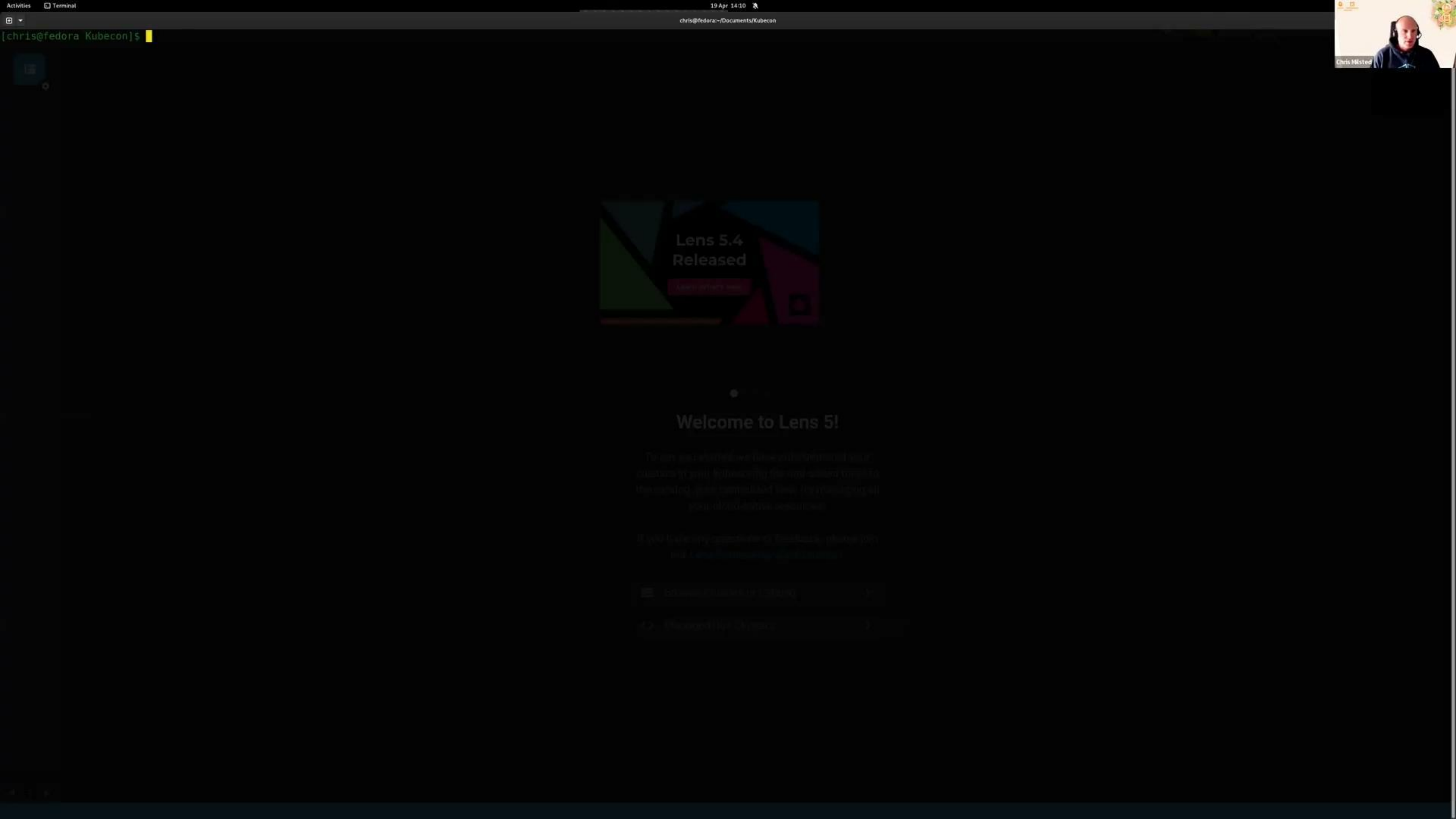
First patterns



Single cluster, single AZ, Single Region



Single cluster, multiple AZ, Single Region



```
[chris@fedora Kubecon]$ 
```

Welcome to Lens 5!

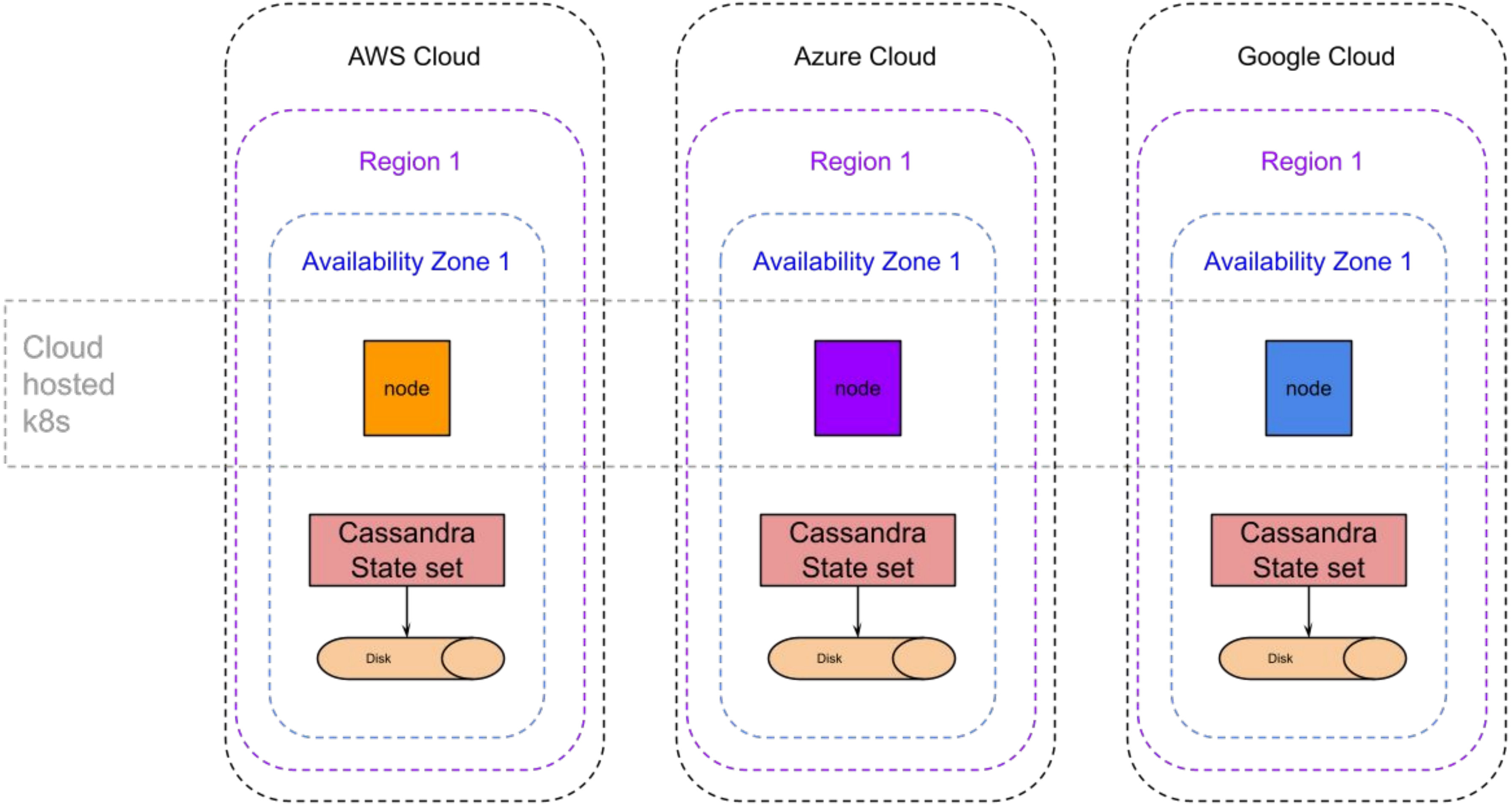
[illegible]

[illegible]

Takeaways from Demo

- Please do follow the Kubernetes design principles - 1 cluster per Region.
- Use a CSI plugin to automate your storage of choice
- Make sure your CSI plugin and application level storage controls complement each other. e.g. avoid double replication
- Follow these building blocks for no-to-low downtime even in a single cloud provider and region.

Second Patterns



KubeCon

Takeaways on Multi-Cloud

- Get the network right
- Mind your credentials cross cloud
- There is a difference between control plane and data plane

Conclusions

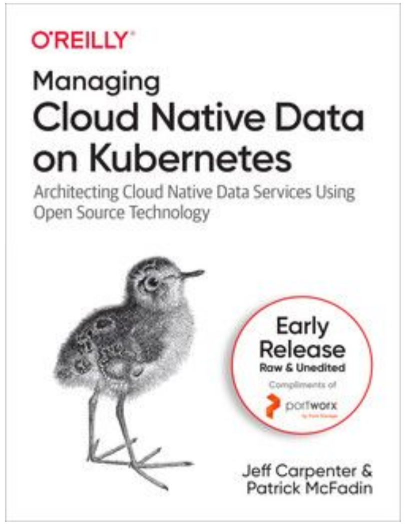
- Remember our lessons from history, co-locate data and compute in kubernetes and think about how to maximise the use of NVMe (or faster) devices (virtualise).
- Use a CSI plugin to orchestrate your storage.
- Always put a limit range to control size of PVCs and also a resource quota to limit either the total number of PVC’s or total requested storage size (to prevent denial of service).
- Pick a strategy for your Storage Classes and Publish this, e.g.
  - *Basic Storage class - use for sophisticated applications like Cassandra where replication and resilience are all controlled at the application level*
  - *Replicated, topology aware storage class where block level replication done at the storage level and will observe standard K8s topology keys for AZ placement.*
  - *Replicated, Topology aware and per Volume encrypted for workloads that need at rest encryption.*
  - *Custom - e.g. adding in storage layer features such as fencing to enable fast failover in case of node failure for stateful sets.*
- You can build patterns which span clouds and regions and zones and continents for ultimate availability. It will be at the cost of network and security and other challenges.

Links

- https://docs.ondat.io/docs/introduction/self-eval/
- https://k8ssandra.io/get-started/

- https://github.com/ragsns/avx-multicloud-k8s
- https://github.com/chris-milsted/kubecon-2022-valencia-demo

- https://dok.community/wp-content/uploads/2021/10/DoK_Report_2021.pdf
- https://github.com/cncf/tag-storage/blob/master/Cloud%20Native%20Disaster%20Recovery.pdf
- https://kubernetes.io/docs/tasks/administer-cluster/limit-storage-consumption/
- Managing Cloud Native Data on Kubernetes [Book]



O'REILLY

Managing
Cloud Native Data on Kubernetes

Architecting Cloud Native Data Services Using Open Source Technology

Early Release
Raw & Unedited

Compliments of
portworx

Jeff Carpenter & Patrick McFadin