Kubernetes for Storage. An overview

- Container Attached Storage
- Built *on* Kubernetes *for* Kubernetes
- A complete solution for the use of Kubernetes as a data plane

CNCF Webinar

July 16, 2020

Join us for KubeCon + CloudNativeCon Virtual

Event dates: August 17-20, 2020

Schedule: Now available!

Cost: $75

Register now!

Before we begin...

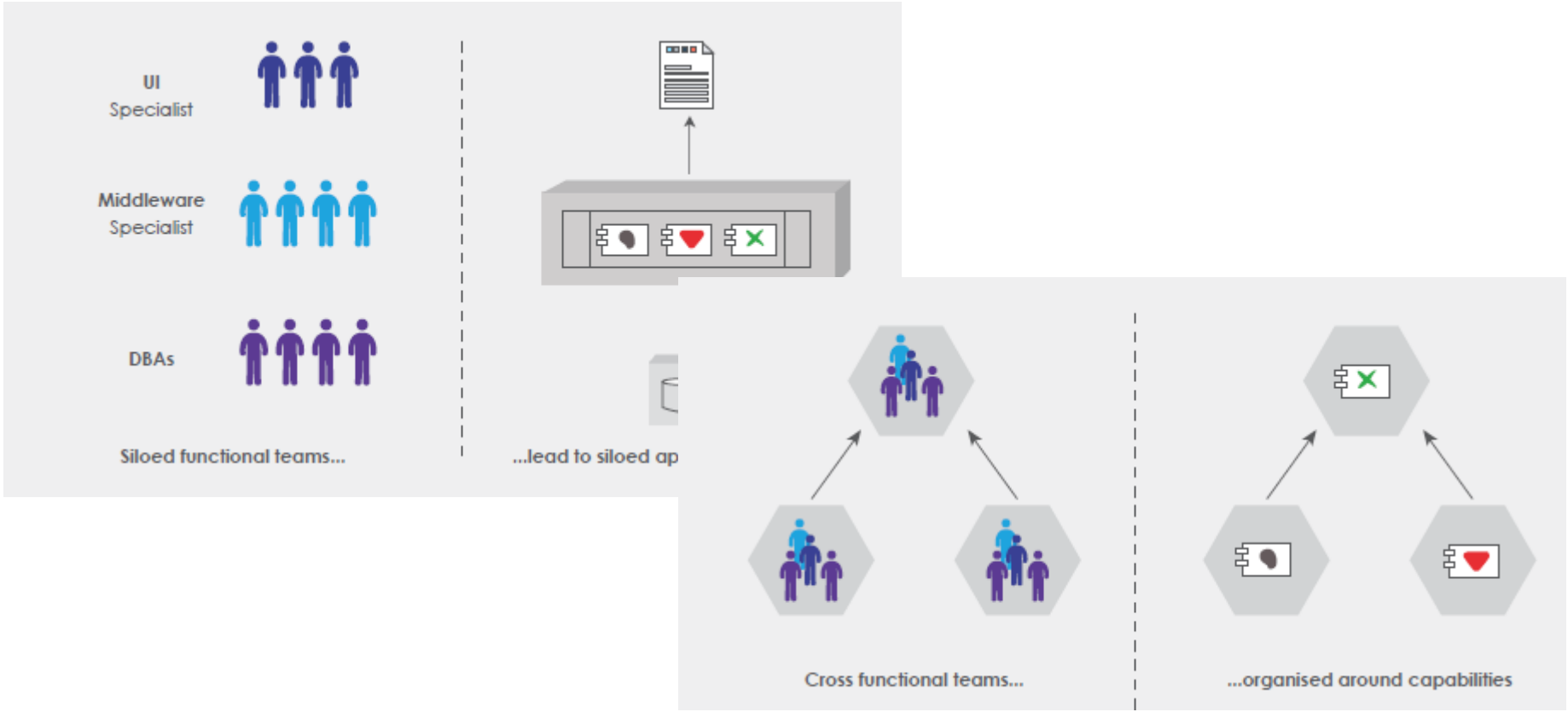
They say, It's(CAS) is a sign

Varys tells Tyrion that on the streets it is called the **Red** Messenger.

The **comet** is interpreted differently by different people in Westeros and Essos.

To some, like Melisandre and Old Nan , it signals the coming of dragons.

Conway's Law for Software Development



MayaData

Your Presenters:

Kiran Mova

Chief Architect
Co-Founder
MayaData Inc

Murat Karslioglu

VP of Products
MayaData Inc

Brian Matheson

Developer Advocate
MayaData Inc

@kiranmova

kiranmova

@muratkarslioglu

murat

@brian_matheson

Brian Matheson

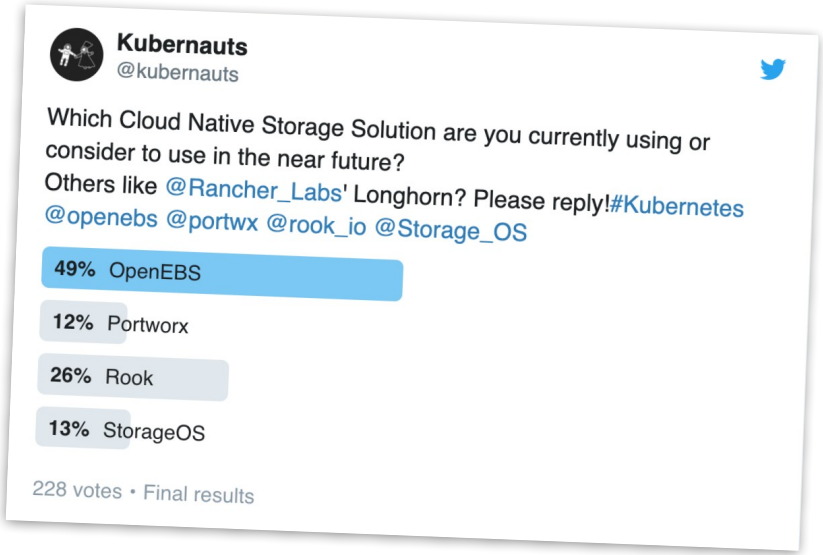
Who is MayaData? Popularity of OpenEBS?

Company	Rank for PRs to CNCF
Google	1st
Red Hat	2nd
VMware	3rd
Microsoft	4th
MayaData	5th

Source: All CNCF PRs authors

- Code *is* marketing
- Contributing in CNCF ecosystem
- 19 CKAs & growing

- 4X yr/yr growth in container pulls
- #1 CNS in trial per CNCF survey
- Rapidly becoming the defacto standard for stateful workloads on Kubernetes



MayaData

Data On Kubernetes Community (DOKC)

DOKCs will be an openly governed and self-organizing group of curious and experienced operators and engineers ***concerned with running data-intensive workloads on Kubernetes***.

The first DOKC talk will be held as a virtual meet-up **July 21st** and will feature Patrick McFadin, VP Developer Relations, ***DataStax***.

Other companies that have volunteered to participate - *Confluent, Arista, Yugabyte, Optoro, 2nd Quadrant*

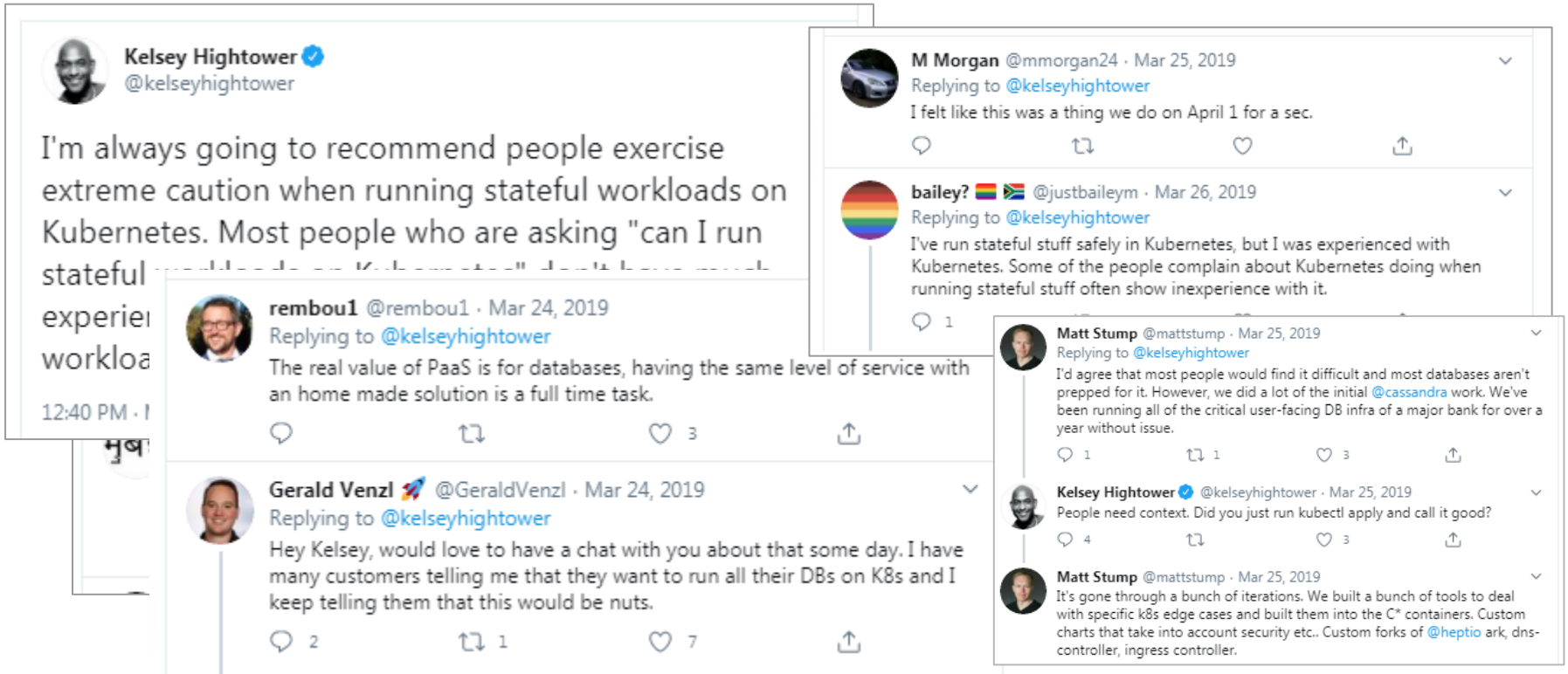
Demetrios Brinkmann

https://dok.community/

Agenda

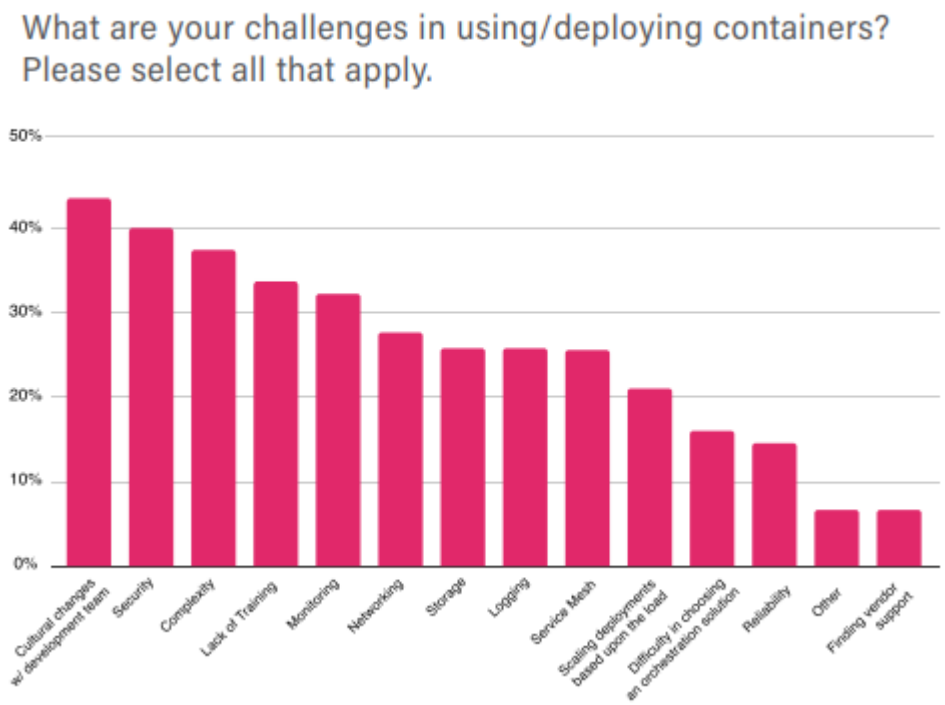
- K8s for Stateful
- Container Attached Storage (CAS)
- Benchmarking Stateful workloads and Storage in K8s
  - And some resources you can use to do it yourself
- K8s as Data Layer - Challenges and Best practices
- Q and A

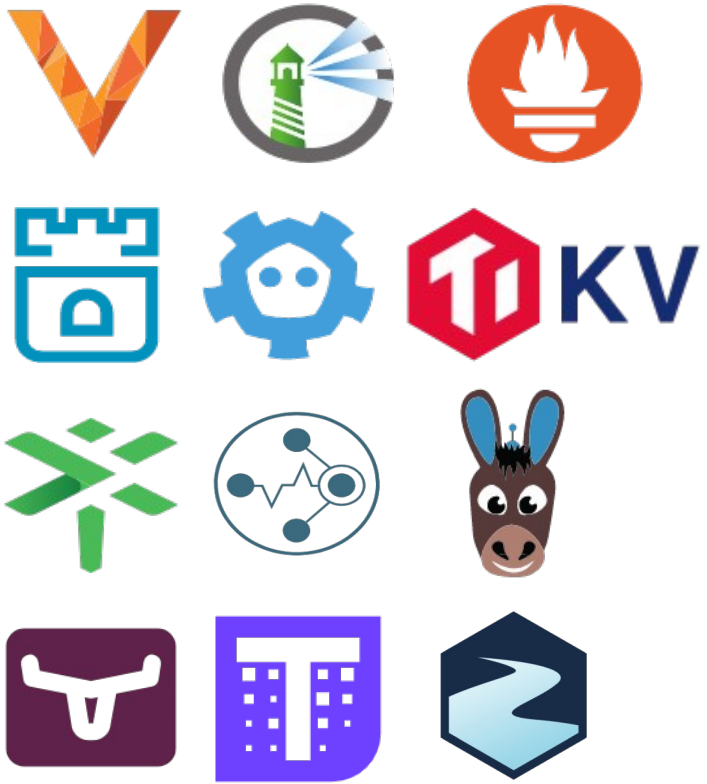
K8s for Stateful: A polarizing topic



MayaData

K8s for Stateful : A cultural shift





MayaData

PostgreSQL on OpenEBS Local PV

Home / Blog / 2ndQuadrant / Local Persistent Volumes and PostgreSQL usage in Kubernetes

Local Persistent Volumes and PostgreSQL usage in Kubernetes

June 22, 2020 / in 2ndQuadrant, Cloud Native / by Gabriele Bartolini

Can I use PostgreSQL in Kubernetes and expect to achieve performance results of the storage that are comparable to traditional installations on bare metal or VMs? In this article I go through the benchmarks we did in our own Private Cloud based on Kubernetes 1.17 to test the performance of local persistent volumes using OpenEBS Local PV.

MayaData

K8s for Stateful : More on Local PV

- Postgres
- MySQL
- Kafka
- Redis
- ElasticSearch
- Prometheus
- Thanos

The vast majority of applications are able to better handle failover and replication than a block level device.

Instead of introducing another distributed system into an already complex environment, OpenEBS's Local PVs allow us to leverage fast local storage.

Additionally, by leveraging ZFS we are able to have encryption at rest for all of our workloads, compression, and the peace of mind of a COW based file system. OpenEBS has allowed us to **not introduce a complicated distributed system** into our platform.

The adoption has been smooth and completely transparent to our end users.

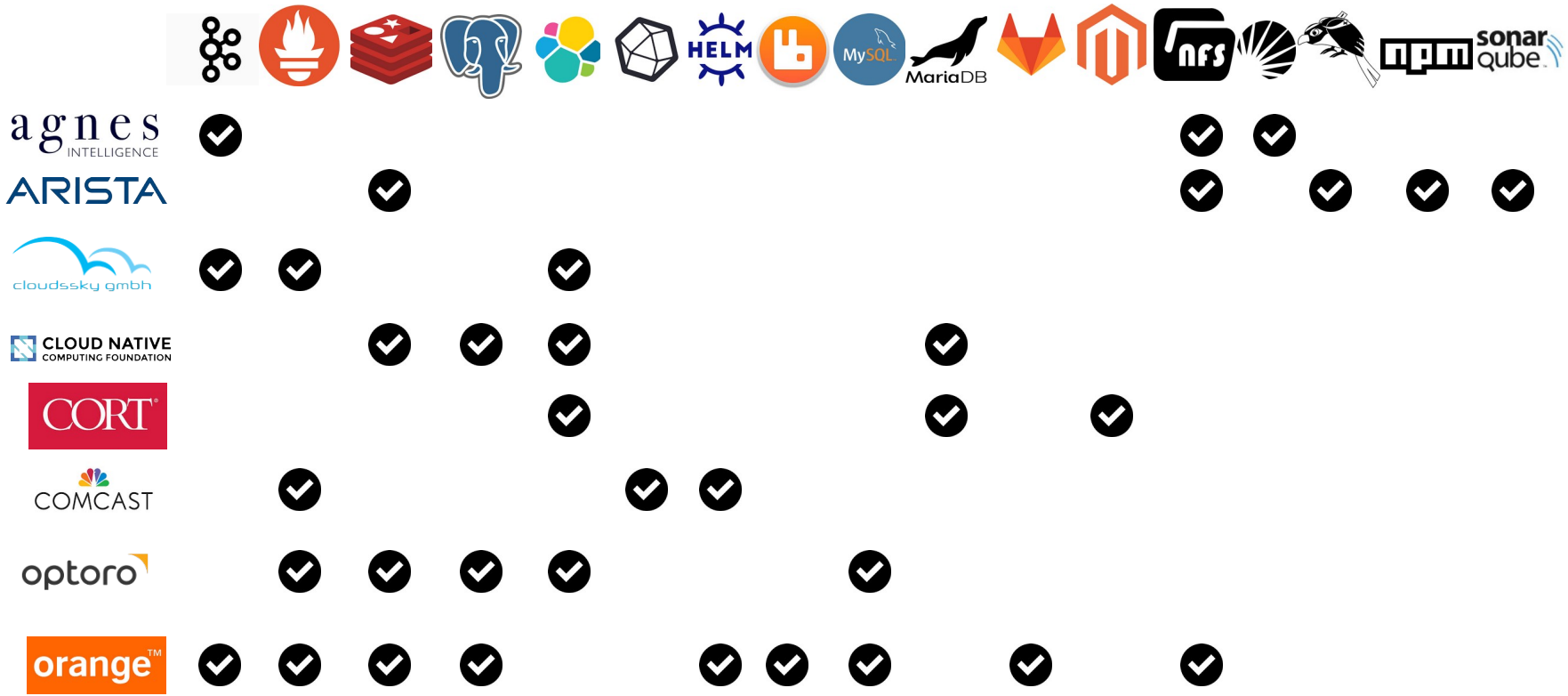
K8s for Stateful : Resistance is Futile

OpenEBS Adopters

This is the list of organizations and users that publicly shared details of how they are using OpenEBS for running their Stateful workloads.

Organization	Stateful Workloads	Success Story
Agnes Intelligence	Apache Kafka, Apache Solr, NFS	English
Arista Networks	Gerrit (multiple flavors), NPM, Maven, Redis, NFS, Sonarqube, Internal tools	English
CLEW Medical	PostgreSQL, Keycloak, RabbitMQ	English
Clouds Sky GmbH	Confluent Kafka, Strimzi Kafka, Elasticsearch, Prometheus	English
CNCF, The Linux Foundation	PostgreSQL, MariaDB, ElasticSearch, Redis, DevStats	English
Code Wave	Bitwarden, Bookstack, Allegros Ralph, Limesurvey, Grafana, Hackmd/Codimd, Minio, Nextcloud, Percona XtraDB Cluster Operator, Nextcloud, Sonarqube, Sentry, Jupyterhub	English
Comcast	Prometheus, Alertmanager, Influxdb, Helm Chartmuseum	English
CORT	Magento, Elasticsearch, MariaDB	English
DISID	Minio, DataStax, Greenplum, Gridgain, mongoDB, Qlickhouse, PosgtreSQL	English

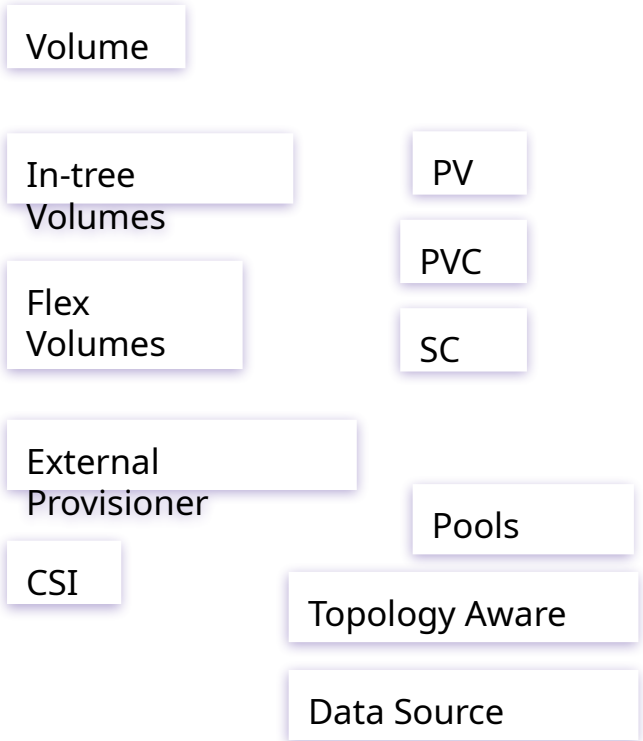
Example references and workloads



MayaData

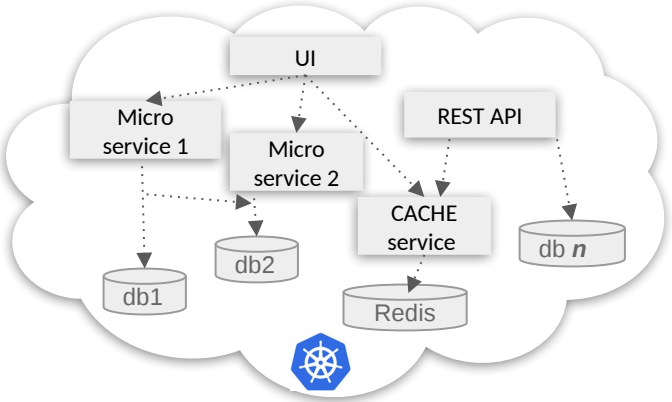
K8s for Stateful : The primitives

- Native interfaces for connecting workloads (Pods) to Persistent Volumes (PVs).
- Dynamic provisioning of PV via Persistent Volume Claim (PVC) and Storage Class (SC).
- More abstraction through community efforts around Persistent Volumes (PV) and Persistent Volume Claims (PVC) and Container Storage Interface (CSI)
- CSI to handle vendor specific needs and avoid wildfire of “volume plugins” or “drivers” in K8s main repo



MayaData

K8s for Stateful: Can’t I just?



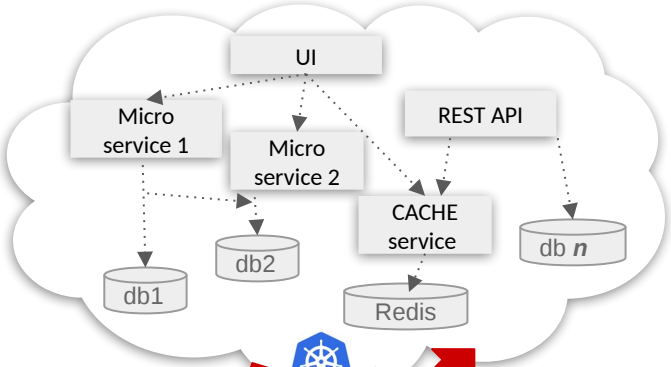
CSI

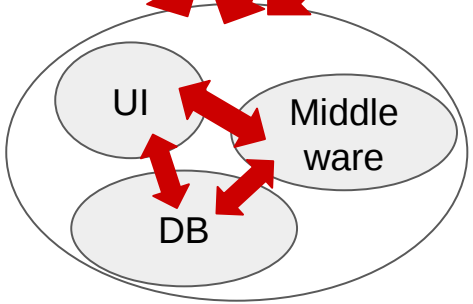
Of course you can. And you do. However you lose so many benefits of moving to Kubernetes.

Most workloads just use Direct Attached Storage instead.

MayaData

K8s for Stateful: What is inside that SAN?





A shared storage system is a complex monolithic distributed system built before

Kubernetes

These systems have DBs for metadata
They have provisioning systems
They have retry & other logic

They take all the IO, mix it together, and do their best

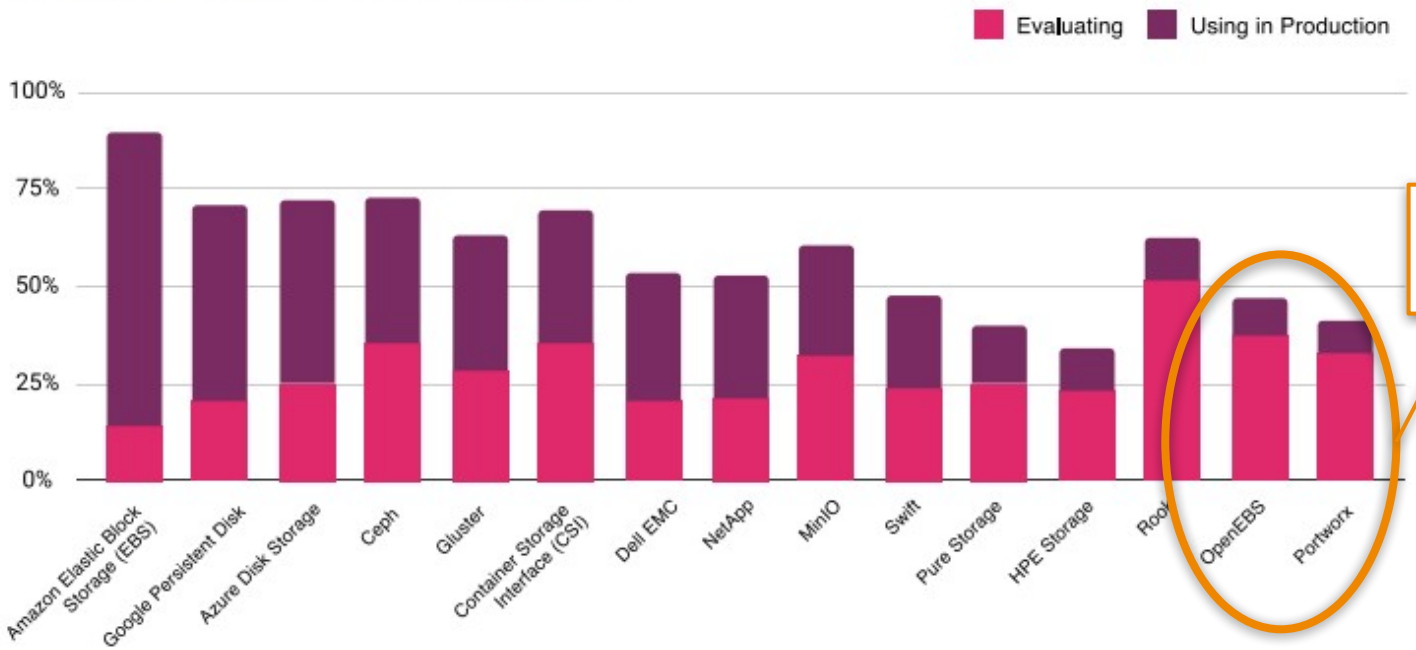
Designed when storage media was slow and apps were NOT resilient

MayaData

K8s for Stateful : CNCF Survey

Cloud Native Storage

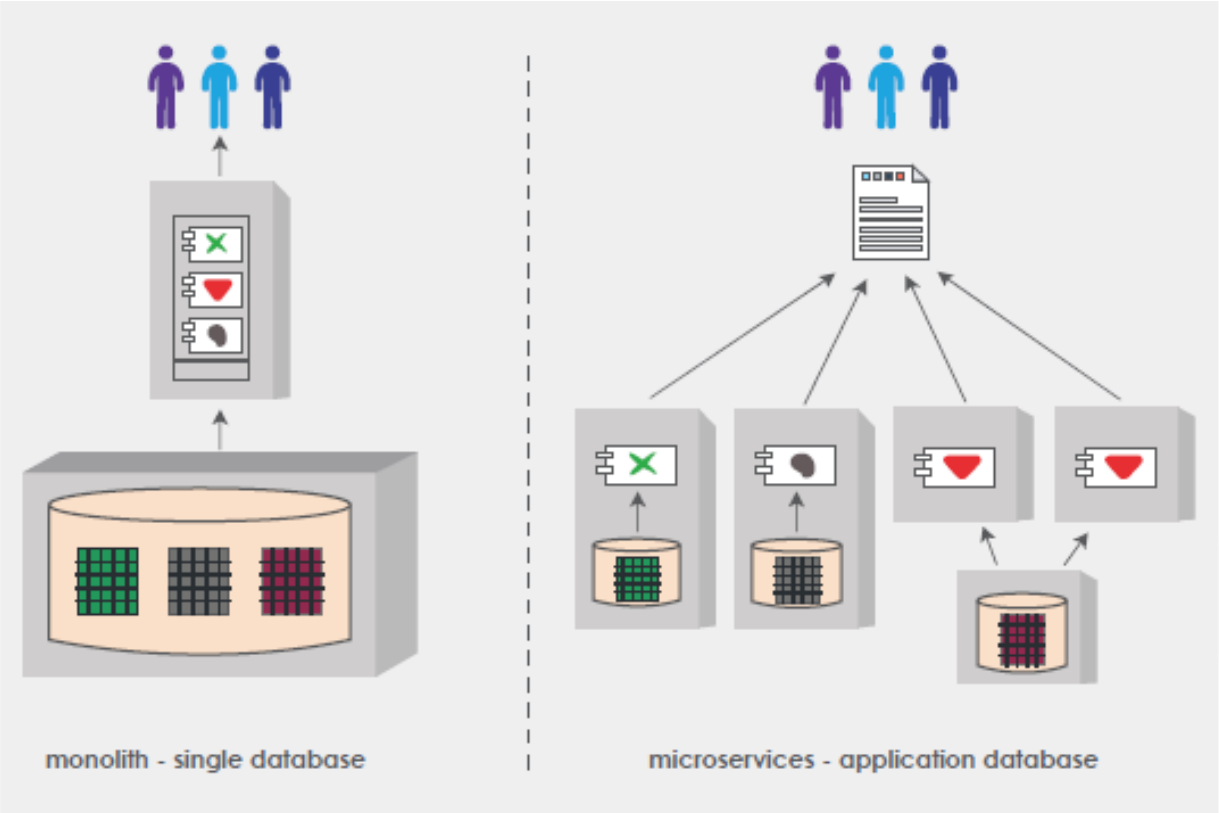
Given a considerable increase in the number of cloud native storage projects, we changed the storage question this year to include each storage project or product listed on the CNCF landscape. 14% of respondents are using storage projects in production, with another 27% evaluating storage projects. Only 5% of respondents indicated they were not planning on using or evaluating any storage projects.



Container Attached Storage most evaluated

MayaData

Conway's Law for Data Management



loosely coupled teams

loosely coupled applications

loosely coupled data

MayaData

CNCF end user

https://www.youtube.com/watch?v=0CEHN6ECaPs

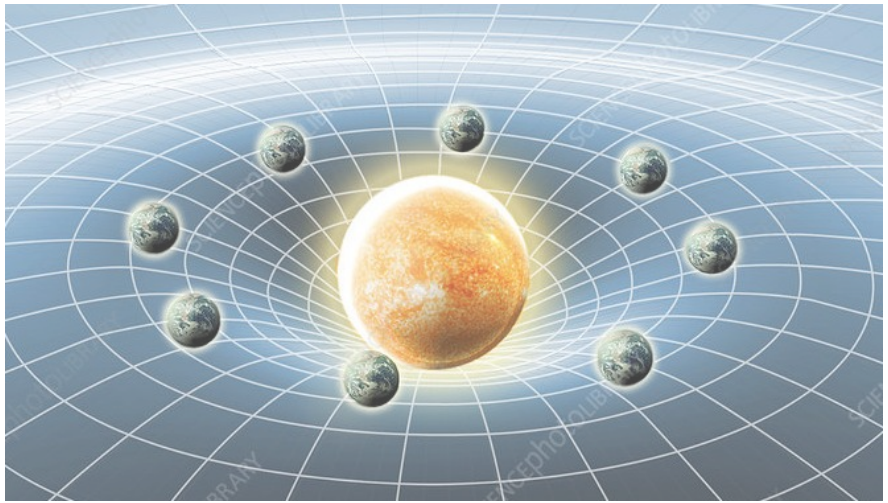
https://www.youtube.com/watch?v=z_LbRfDKPvE

Steven Bower at Bloomberg

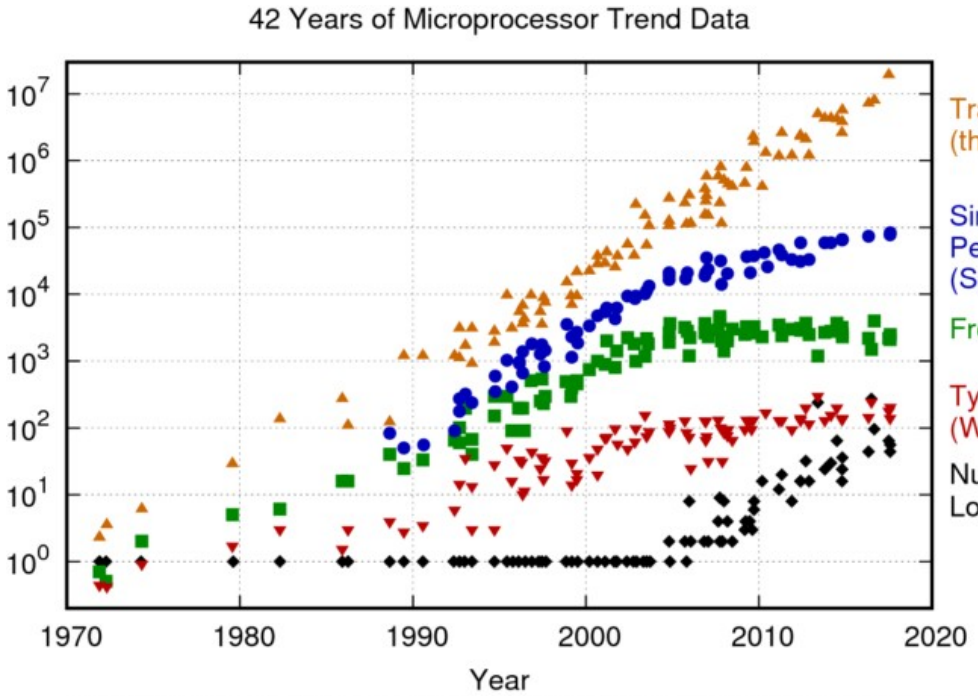
- Moved to Kubernetes in order to simplify and standardize their environments
- CNCF end user of the year 18/19
- Running dozens of different stateful workloads at scale
- Believes in open source
- Not about cost savings - about agility
- Everything loosely coupled
- Teams are autonomous and full stack
- Does not use shared storage
- Uses OpenEBS - different flavors

Data Gravity - Lockin / Costs

- As data grows — it has the tendency to pull applications towards it (gravity)
- Everything will evolve around the sun and it dominates the planets
  - Latency, throughput, IO blender
  - If the sun goes super nova — all your apps circling it will be gone instantly
- Some solutions involve replicating the sun towards some other location in the “space time continuum”
  - It works — but it exacerbates the problem

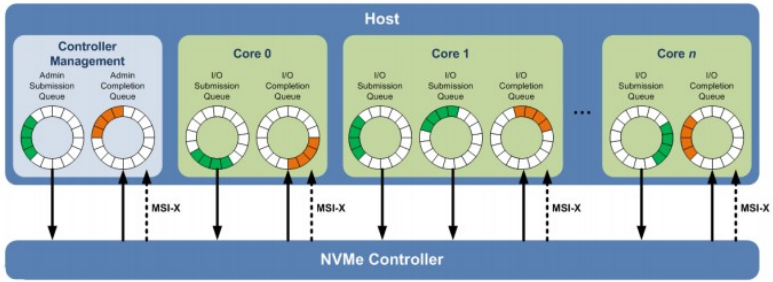


Under-utilization of Technology



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

https://github.com/karlrupp/microprocessor-trend-data

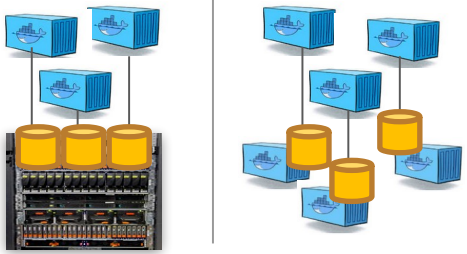


MayaData

Summary - challenges around K8s storage

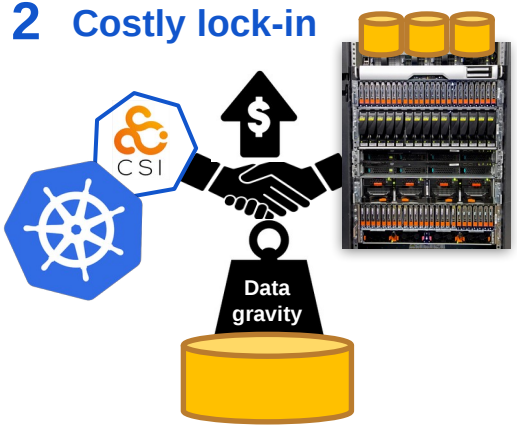
1 Conway's Law

Shared everything vs Per workload, per team



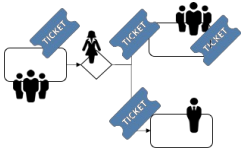
External storage — *Container native*

2 Costly lock-in



3 Process mismatch & 100x more dynamism

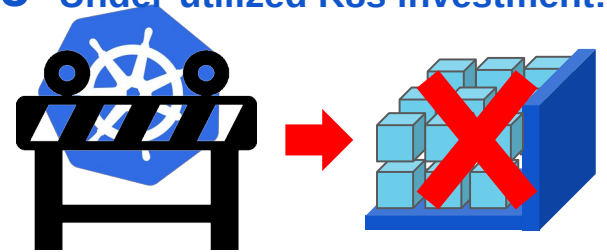
Traditional processes vs Automated Kube - Ops



GitOps

4 External know-how required for traditional storage

5 Under-utilized K8s investment!



MayaData

Container Attached Storage

K8s Storage done right.

CAS : Apps have changed ...

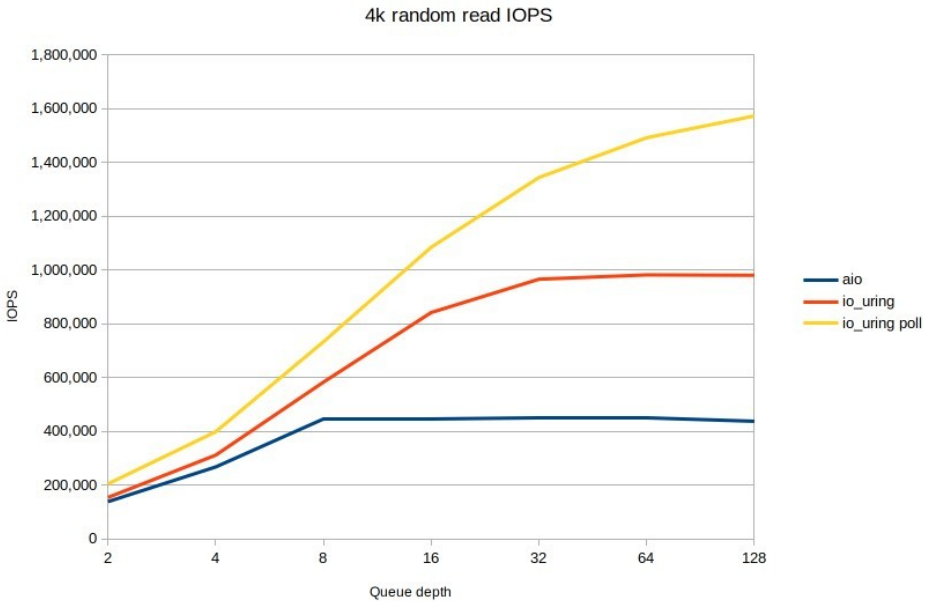
- Meta languages, Go, Rust,...
- Apps are often distributed systems themselves
  - Is a distributed storage system still needed?
- Designed to fail and expected to fail
  - Across racks, DC's, regions and providers, physical or virtual
- Scalability batteries included
  - HaProxy, Envoy, Nginx, Auto scaling
- Loosely Coupled. Agility:
  - releasing frequently - always changing

CAS : Built for Cloud Native IO demands …

- Datasets of individual containers relativity small in terms of IO and size
  - Prefer having a collection of small stars over a big sun?
  - Multiple smaller databases than large databases (Conway's Law)
- Hardware Trends are changing
  - Built using low latency technologies
  - NVMe, DPDK/SPDK

CAS: Ring based communication channels

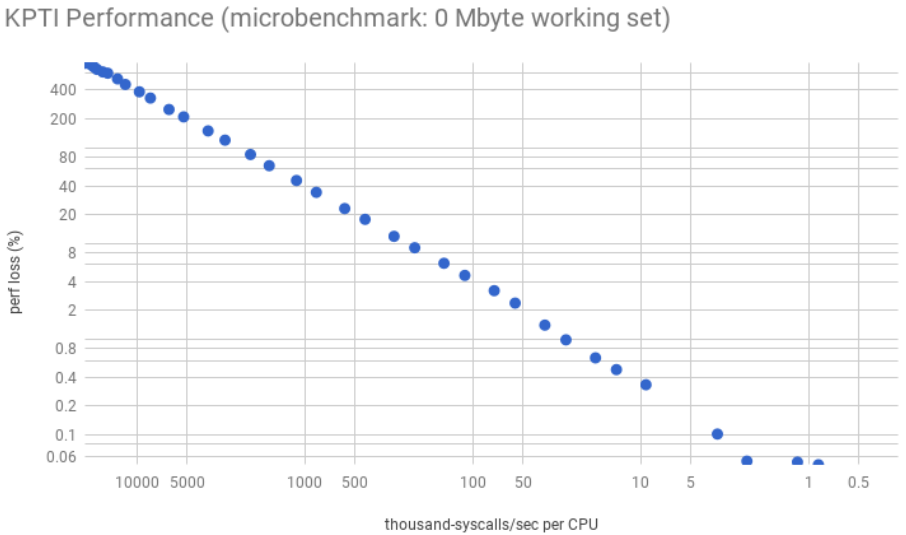
- One queue is used for submitting new requests
- A separate queue is used to store requests that have been completed
- Sometimes a third queue is used to submit admin commands
- io_uring a new interface added to the kernel to “catch up” with the high speed devices, poll mode FTW.



MayaData

CAS: Hardware bugs and their impact

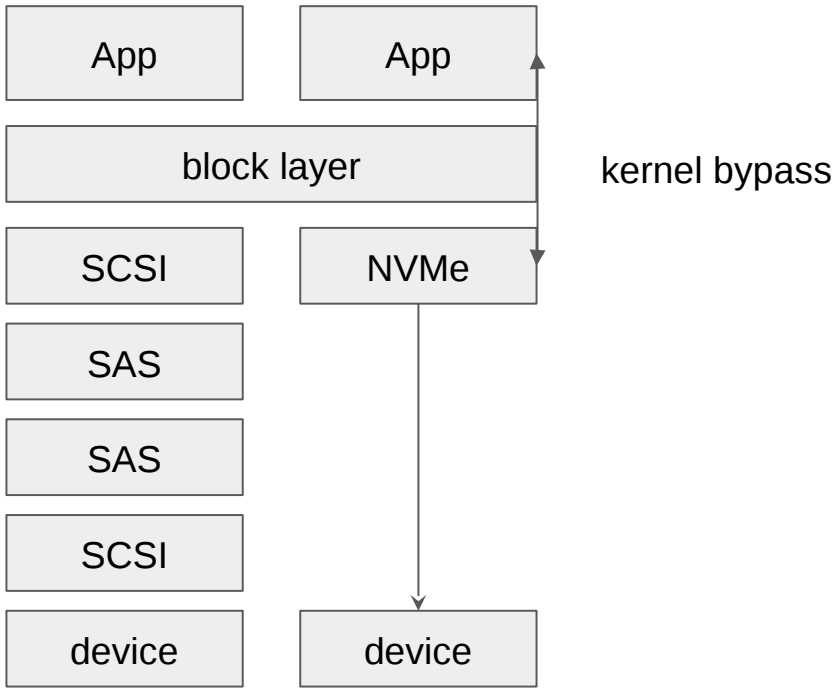
- High number of system calls have a huge impact on performance
- Two solutions to mitigate this:
  - Making use of huge pages
  - Try to do as much as possible in user space



MayaData

CAS: NVMe => Less is More

- NVMe is a protocol that dictates how bits are moved between the CPU/device but also -- between devices
  - Its origin can be found with Infi Band used in HPC for many years (1999)
- NVMe over Fabrics extends the protocol over TCP, RDMA, FC, virtio
- A complete replacement of the SCSI protocol which goes back all the way to 1978

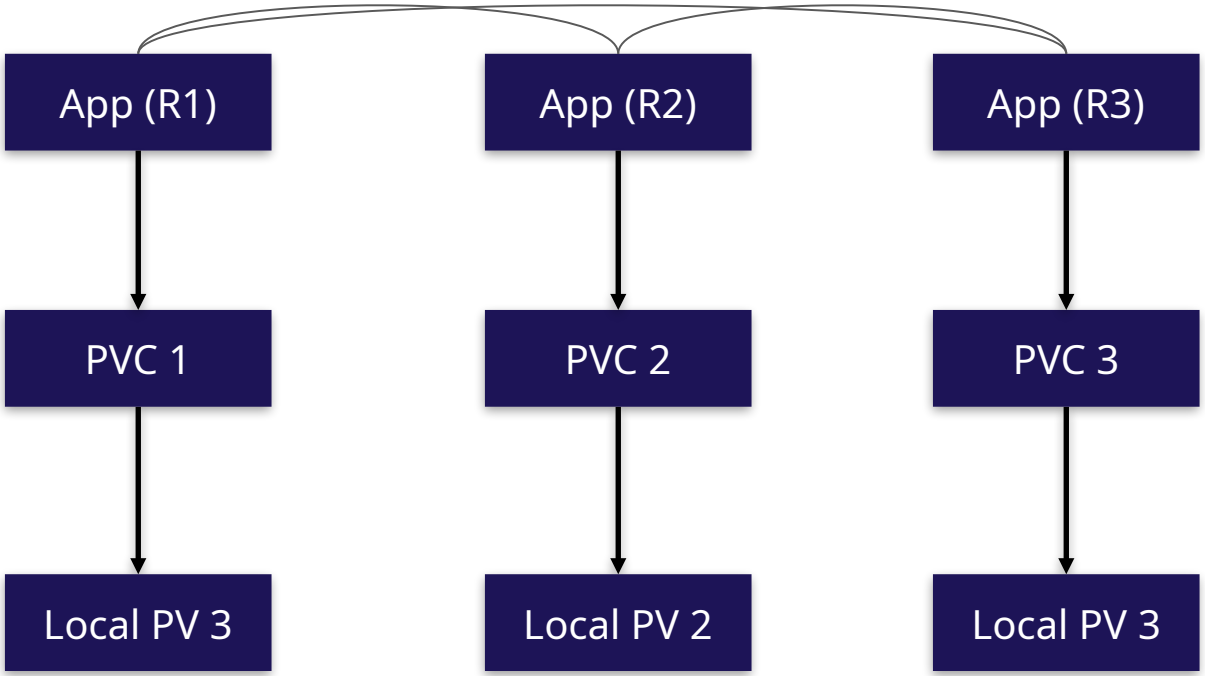


MayaData

CAS: Impacts of HW changes on the stack

- Packets come in at a very high rate, single CPU 100% how to scale?
  - CPU has ~67ns per packet @3GHz
- Solution: spread across multiple cores which requires locking
  - **Locks are expensive** and locks are in memory which is 70-40ns away?
- Amdahl's law starts to dominate the performance envelope
- Context switches and system calls have gotten far more expensive post spectere meltdown
- What we seem to need are lockless queues that scale per core
  - **Poll mode drivers**
- Partial rewrites are inevitable, the rewards are high
  - *ScyllaDB, VPP, Open vSwitch,*

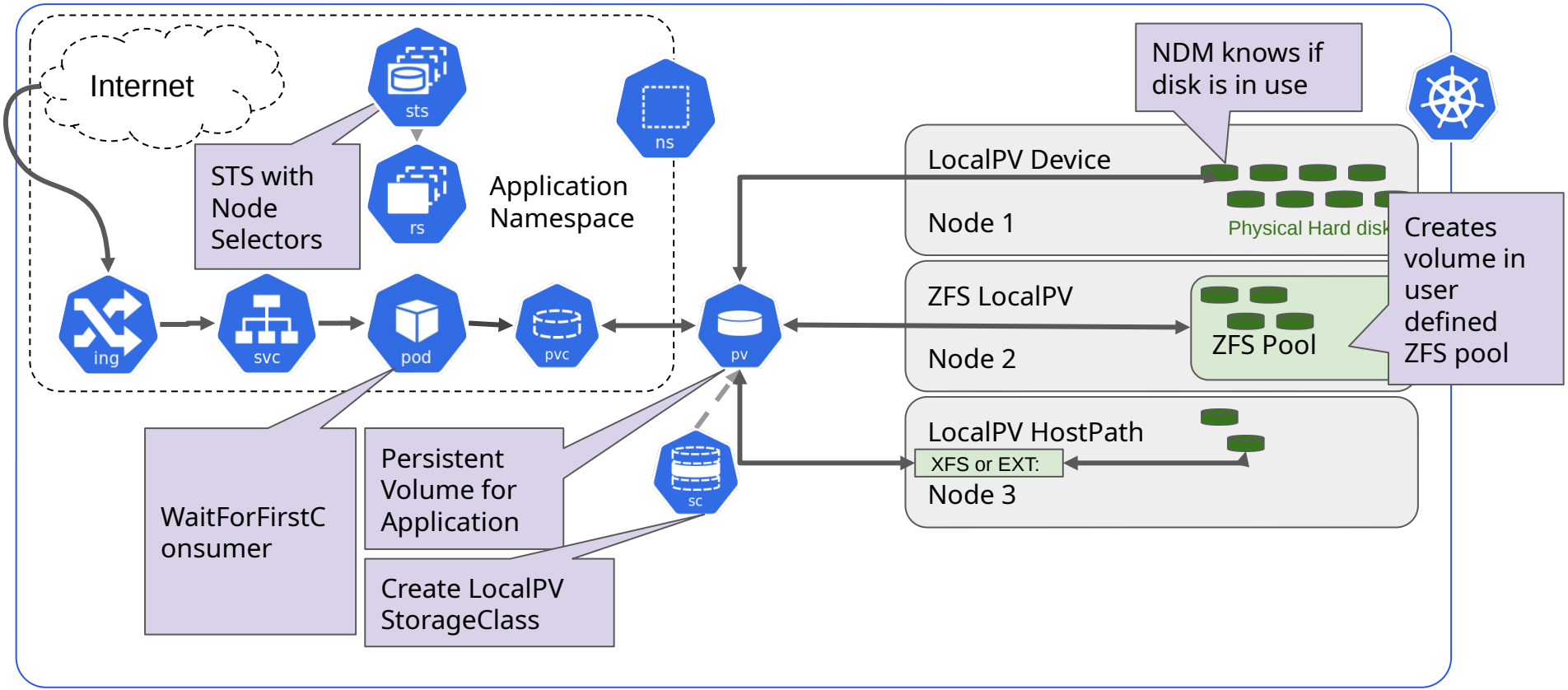
CAS : Pattern - Distributed Apps



- Hostpath
- Block Device
- Local ZFS / LVM

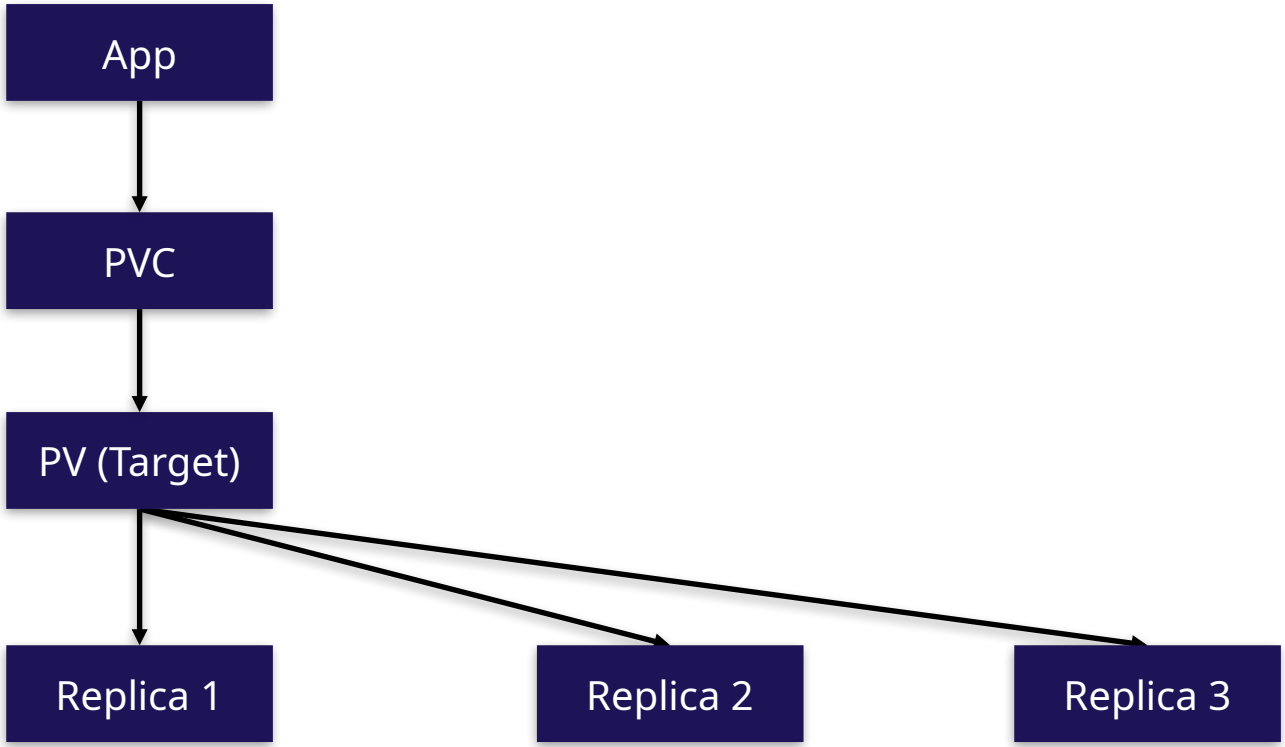
MayaData

Connect a Stateful App to OpenEBS LocalPV



MayaData

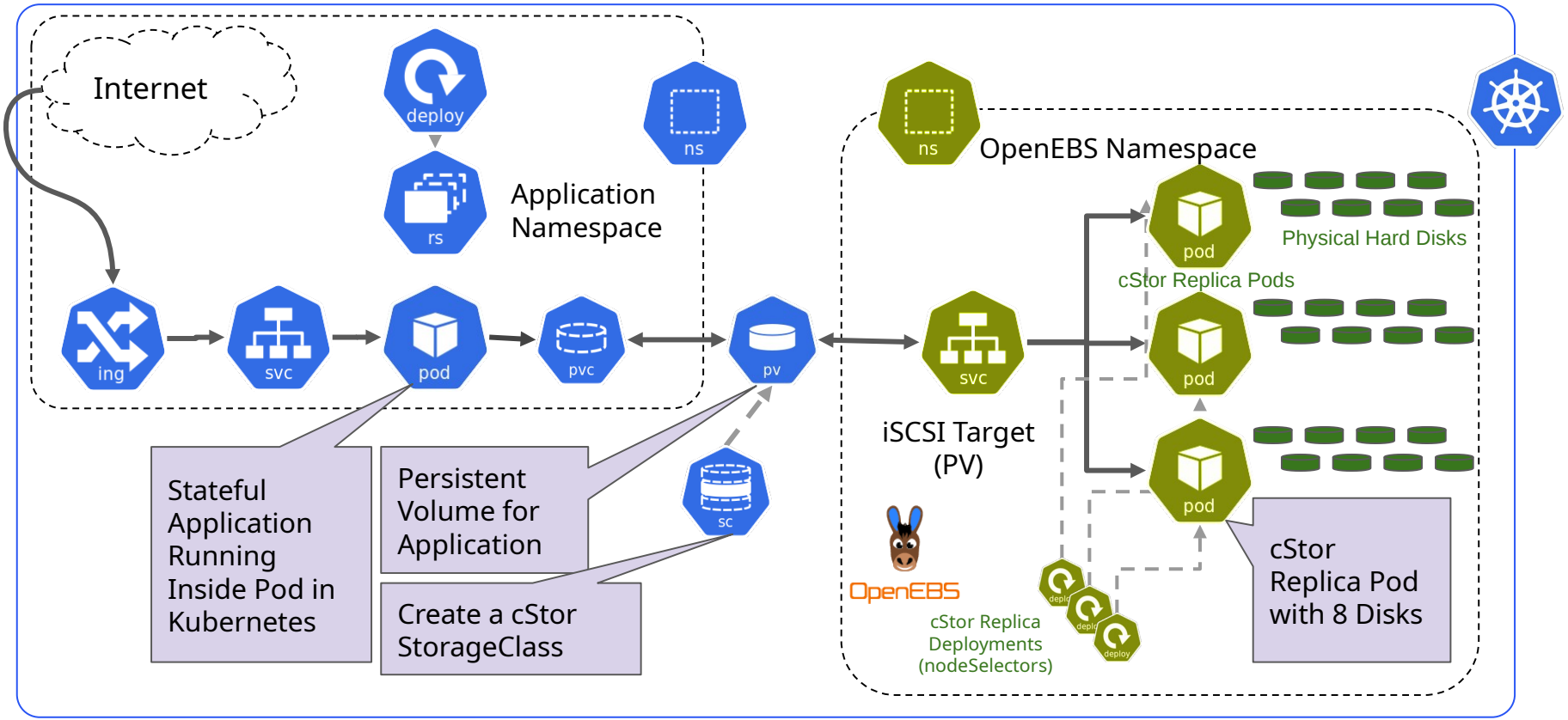
CAS : Pattern - Replicated and Isolated



- Jiva/Longhorn
- cStor
- Mayastor

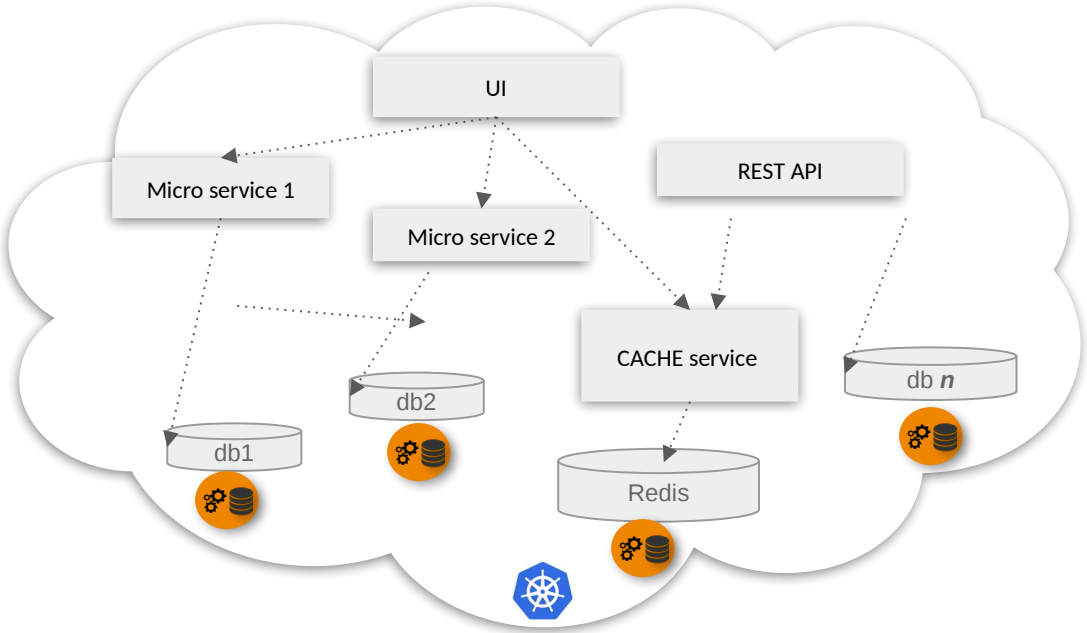
MayaData

Connect a Stateful App to OpenEBS cStor Storage



MayaData

CAS - using K8S as a data layer



Every workload & team its own system

Different engines for different workloads

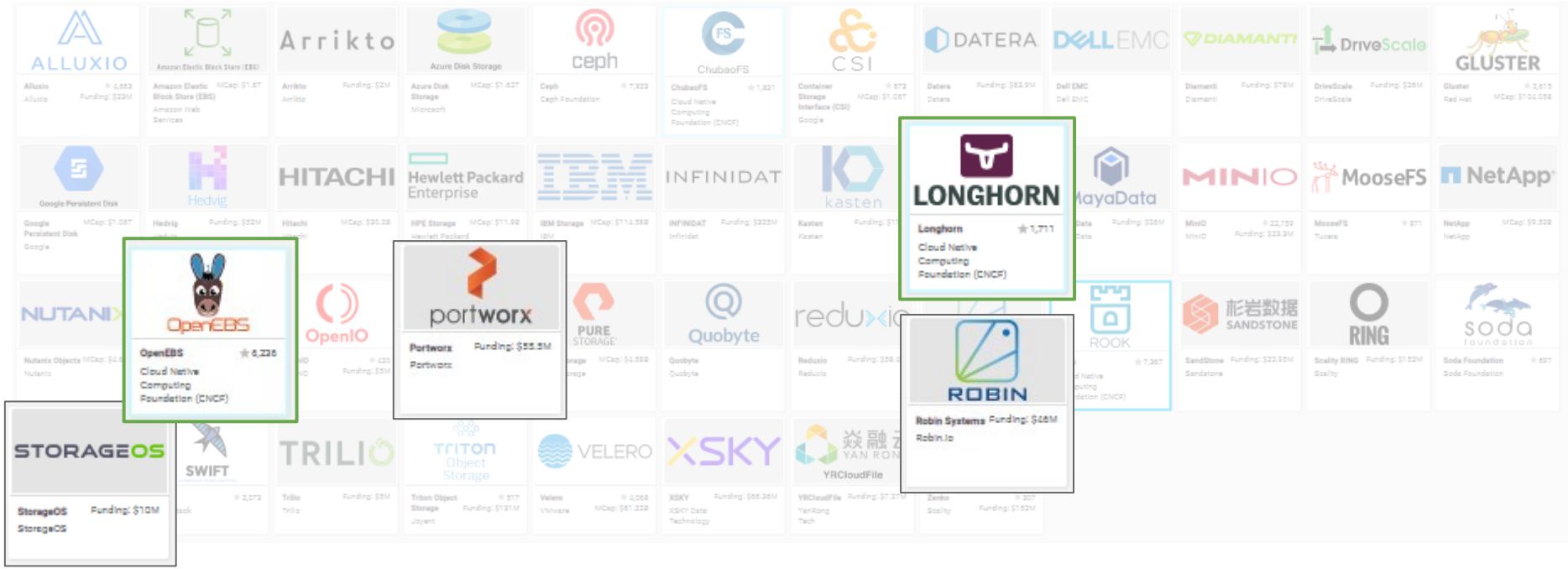
Built *on* Kubernetes *for* Kubernetes

Delivers the benefits o for data
- No lock-in
- Open source
- Runs consistently everywhere
  - Any underlying cloud or disk or SAN

AND the right architecture for NVMe

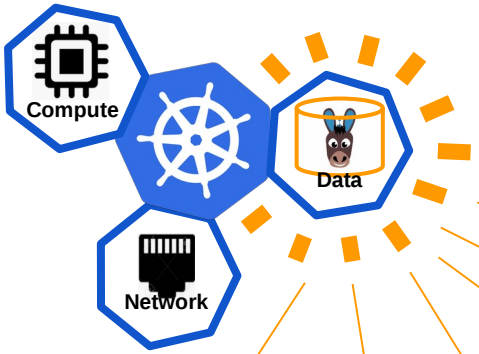
MayaData

CAS Examples on CNCF Landscape



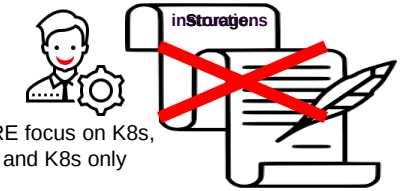
MayaData

Summary - Unleash the power of K8s with CAS



Based on OpenEBS, Open Source,
5th largest CNCF contributor

Built on Kubernetes for Kubernetes



SRE focus on K8s,
and K8s only

Container attached storage using disk within the K8s cluster.
Workload-down storage - ideal for small micro '2 pizza' teams

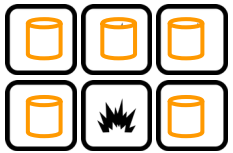
Avoid Lock-in

Cross Cloud visibility
Multi-platform abstraction

Data portability and
no gravity issues

50-70%
reduced
storage TCO

Native Hyperconvergence:
Truly Cloud Native Storage
for Kubernetes



High Availability -
limited blast radius

MayaData

CAS : What about performance?

For low latency workloads - that are already distributed in nature - use different flavors of OpenEBS Local PV.

cStor (though it is relatively slow due to its strict consistency checks), fares well in workloads that are not very IO intensive. Have users running 50+ workloads on cStor in 3 node cluster.

Mayastor - specifically targeted for low latency workloads operates at Local PV - low latency numbers - while maintaining strict consistency and offering storage services.

CAS: OpenEBS Mayastor 0.2.0 - ALPHA

https://www.youtube.com/watch?v=_5MfGMf8PG4

- 100% user space implementation
  - Crucially important to avoid cloud dependencies; ubuntu-GKE != ubuntu-AWS
  - Leverages poll mode drivers and auto detects uring support
- The Nexus supports several storage protocols and can be used with existing iSCSI, NVM-oF targets and local storage
  - Can do n-way mirrors i.e iscsi://<host>/iqn + nvmf://host/nqn + file:///dev/sdb
- Also API driven, i.e write to NVMe directly by passing the kernel

Container Attached Storage

Performance Benchmarking

Ben Hundley

https://getoctane.io/

Solving costs on Kubernetes

Performance Benchmarking - Test Details

Kubernetes Cluster Details

- m5ad.2xlarge on AWS (8 cores, 32GiB RAM, 300GiB NVMe SSD)
- Kubernetes 1.16.8
- Amazon Linux 2

FIO and pgbench

- Fio profiles for Postgres were generated with pgbench and blktrace.
- Fio command to replay the profile.

https://github.com/openebs/performance-benchmark/tree/master/benchmark-tool

Performance Benchmarking - CAS

- Tuning the nodes for performance - as part of your Terraform / Ansible
- Test for scale - number of workloads
- Day 2 Operations in Progress
- Noisy neighbour / Load
- Chaos

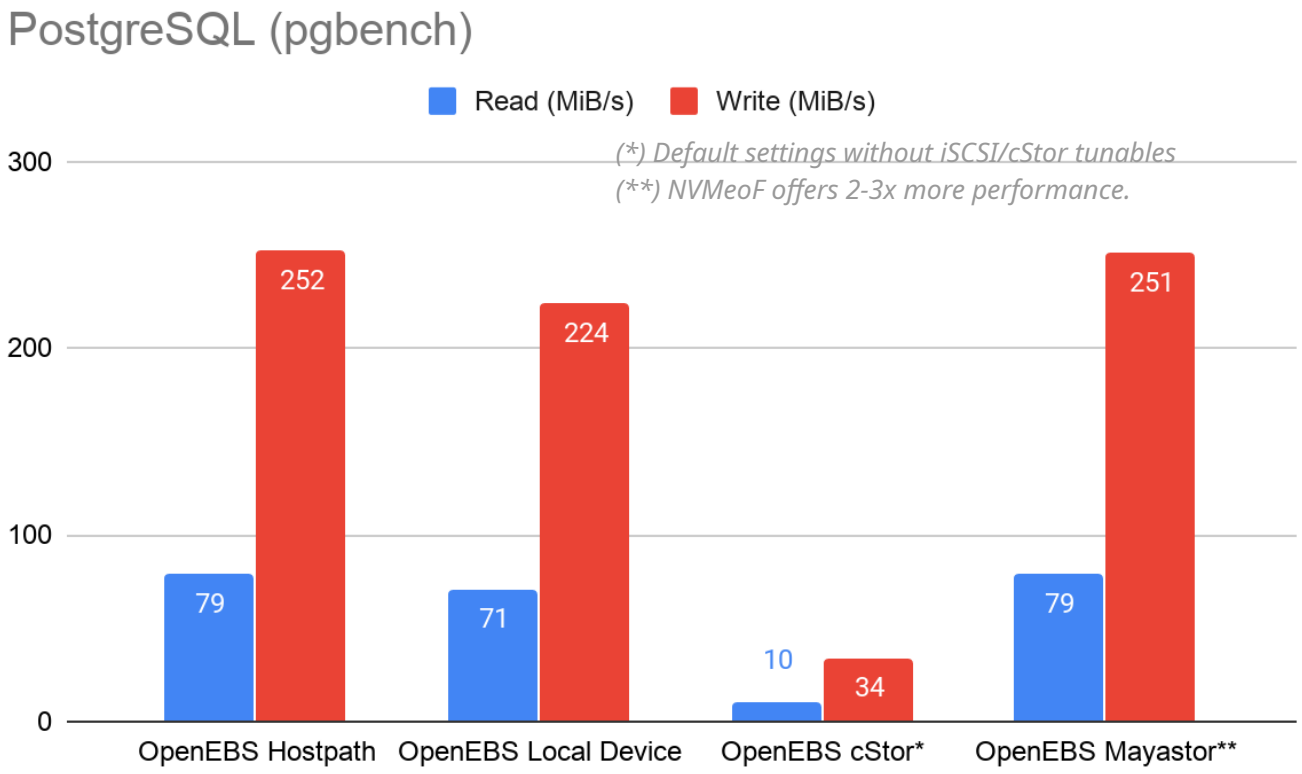
in /etc/iscsi/iscsid.conf and change:

```
node.session.cmds_max = 4096
node.session.queue_depth = 128
```

etc/sysctl.conf:

```
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_sack = 0
net.ipv4.tcp_rmem = 10000000 10000000 1000000
net.ipv4.tcp_wmem = 10000000 10000000 1000000
net.ipv4.tcp_mem = 10000000 10000000 10000000
net.core.rmem_default = 524287
net.core.wmem_default = 524287
net.core.rmem_max = 524287
net.core.wmem_max = 524287
net.core.optmem_max = 524287
net.core.netdev_max_backlog = 300000
```

Per workload Benchmarks



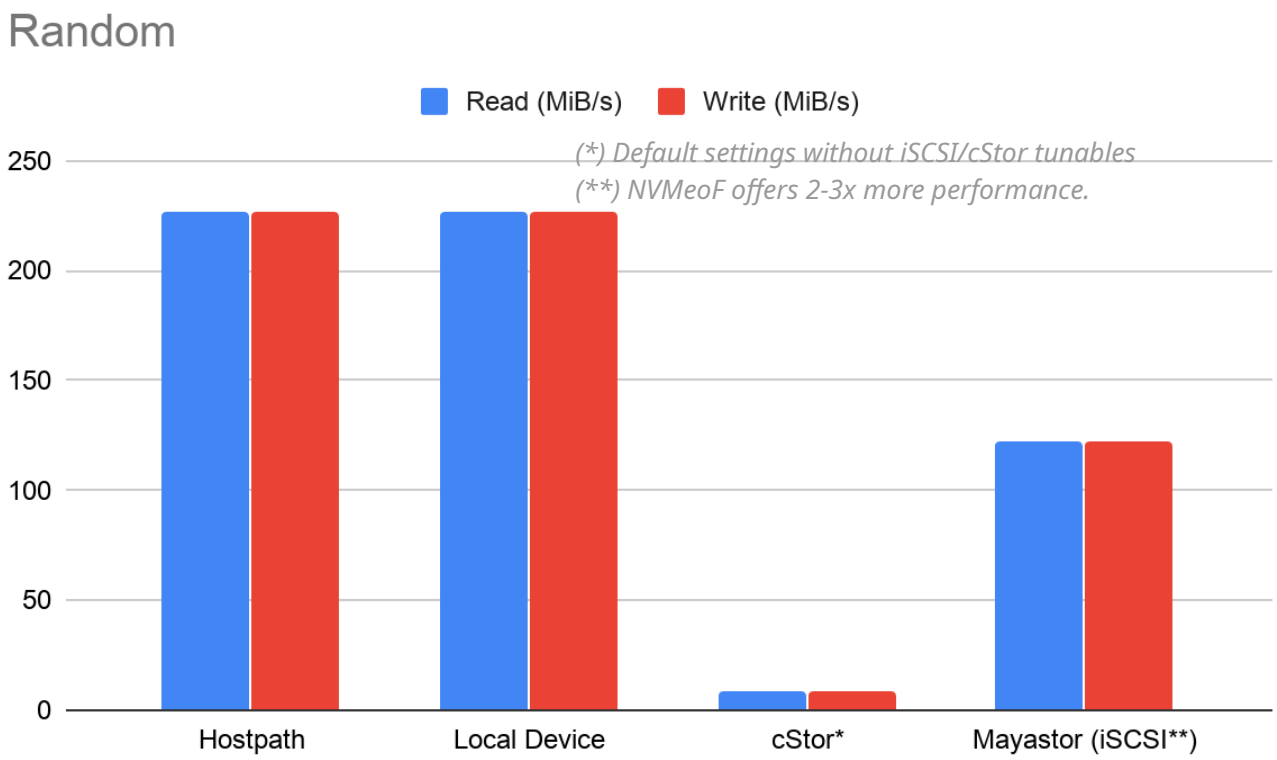
MayaData

Application Benchmarks

	IOPS		Throughput (MiB/s)	
	Read	Write	Read	Write
OpenEBS Hostpath	10200	16300	79.9	252
OpenEBS Device	9042	14500	71	224
OpenEBS cStor*	1383	2214	10.9	34.3
OpenEBS Mayastor**	10200	16200	79.5	251

(*) Default settings without iSCSI/cStor tunables
(**) NVMeoF offers 2-3x more performance.

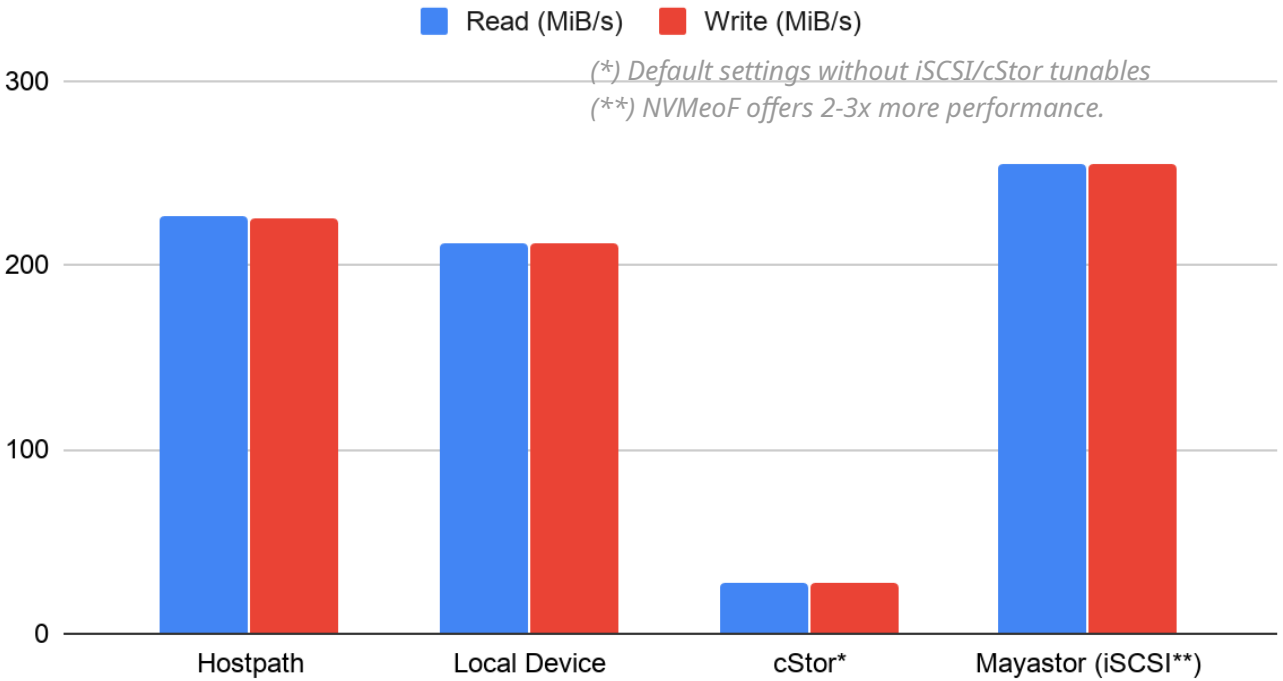
FIO



MayaData

FIO

Sequential

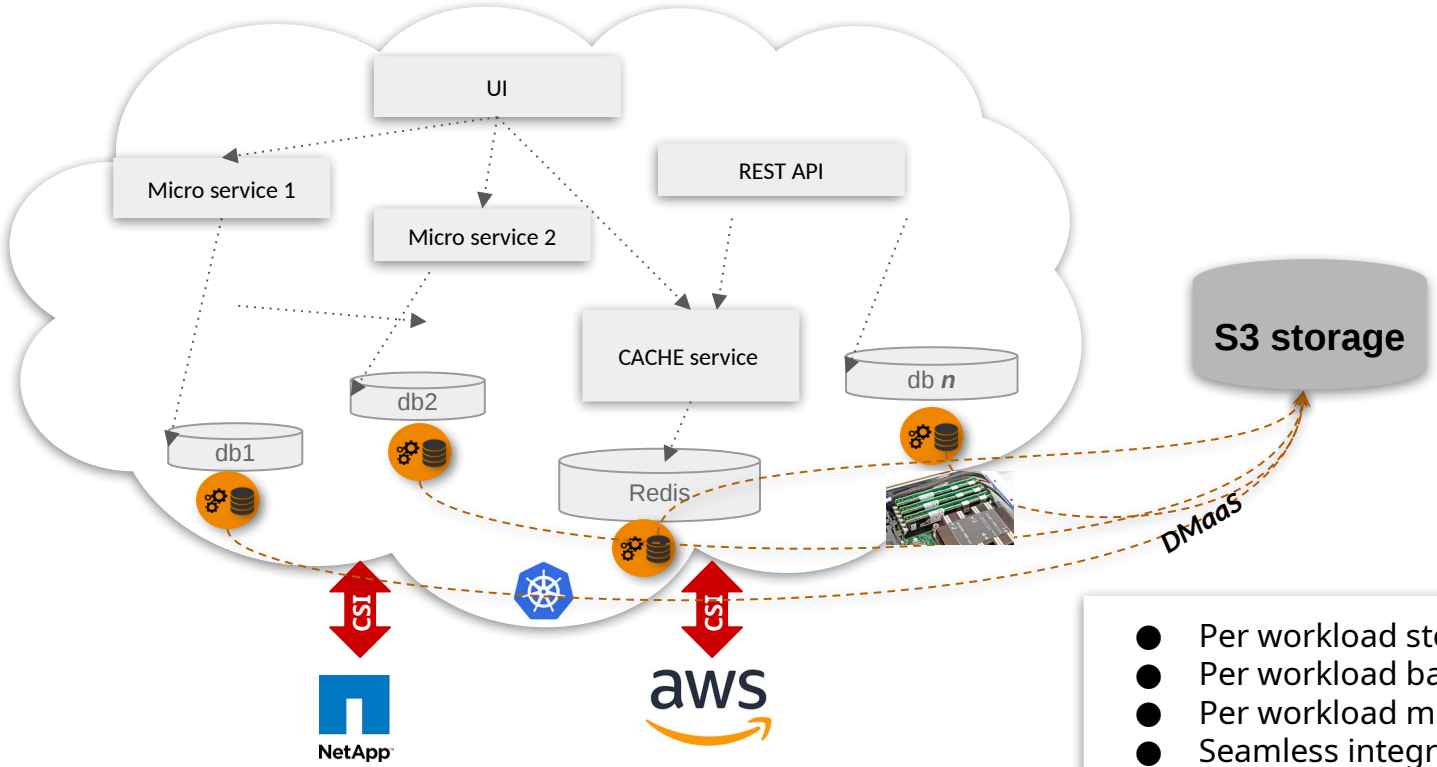


MayaData

Container Attached Storage

Data Agility

Data Protection



- Per workload storage
- Per workload backup
- Per workload management
- Seamless integration w Optane & bare metal & CSI provisioned clouds and legacy storage

MayaData

Kubera = Kubernetes as your data layer

Kubernetes

Kubernetes without OpenEBS & other extensions limited in data capabilities

Stateful Applications

Kubernetes APIs

Kubera equips you for the rest of it.

Policy based storage

Policy and environment based application granular placement of PVs

Backup and DR

Application and data availability.

Compliance & Governance

Management of data ACLs.

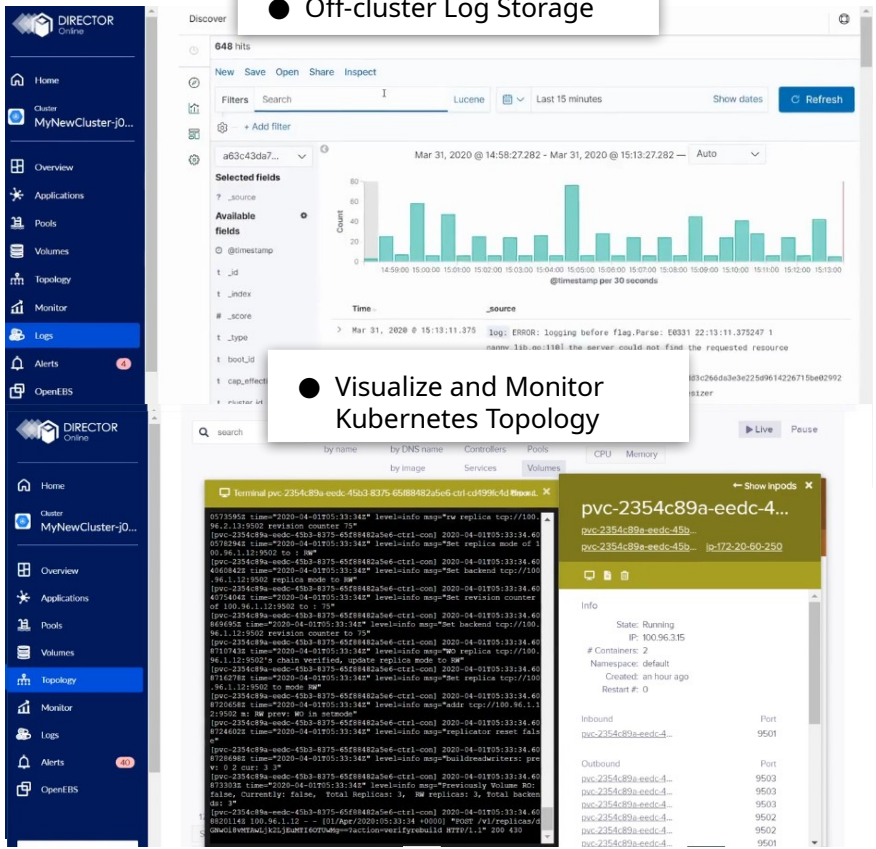
Monitoring & Actions

Anomaly detection and proactive config optimization optimized for data

MayaData

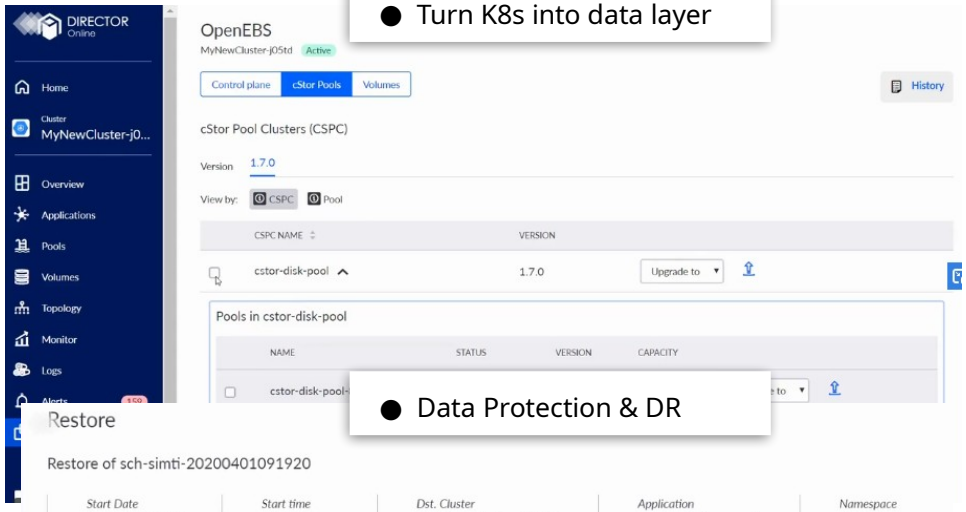
Kubera = Multicloud Data Agility toolset

- Off-cluster Log Storage

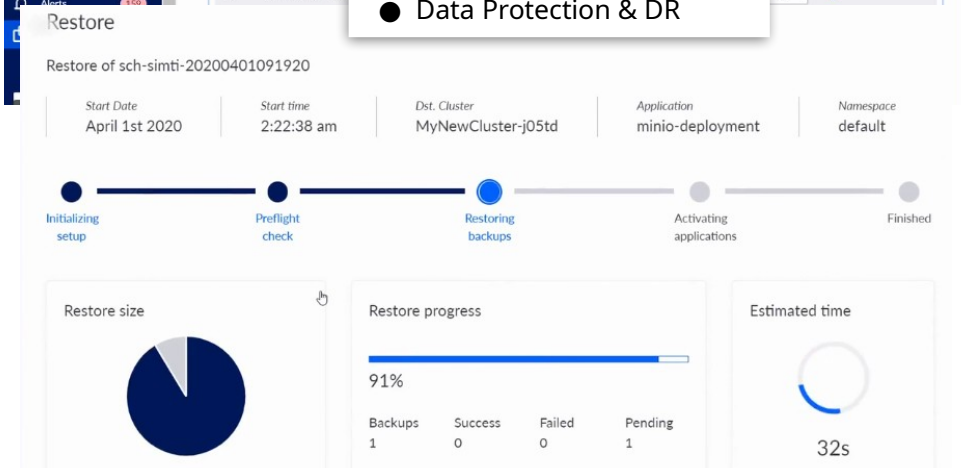


- Visualize and Monitor Kubernetes Topology

- Turn K8s into data layer



- Data Protection & DR



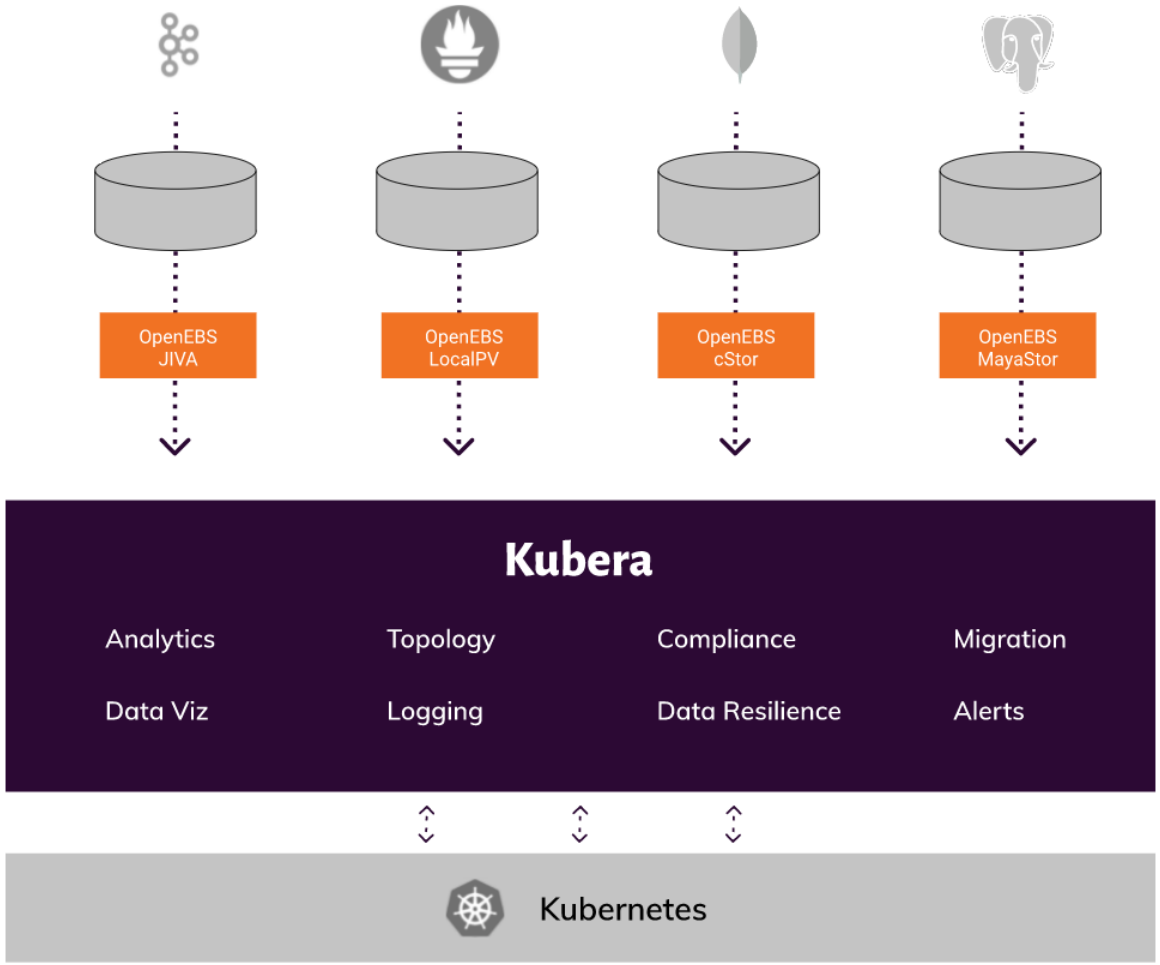
https://account.mayadata.io/login

MayaData

MayaData Kubera

- Open Source
- Kubera BASIC
- Kubera STANDARD
- Kubera ENTERPRISE

MayaData



- Conway's Law
  - Small teams
  - Small workloads
  - Loosely coupled
- Different engines per workload & per team

- Operations - automated
- K8s as data layer
- 24 / 7 support
- SaaS & on premises

- Any Kubernetes
- 100% user space

MayaData

Data On Kubernetes Community (DOKC)

DOKCs will be an openly governed and self-organizing group of curious and experienced operators and engineers ***concerned with running data-intensive workloads on Kubernetes***.

The first DOKC talk will be held as a virtual meet-up **July 21st** and will feature Patrick McFadin, VP Developer Relations, ***DataStax***.

Other companies that have volunteered to participate - *Confluent, Arista, Yugabyte, Optoro, 2nd Quadrant*

Demetrios Brinkmann

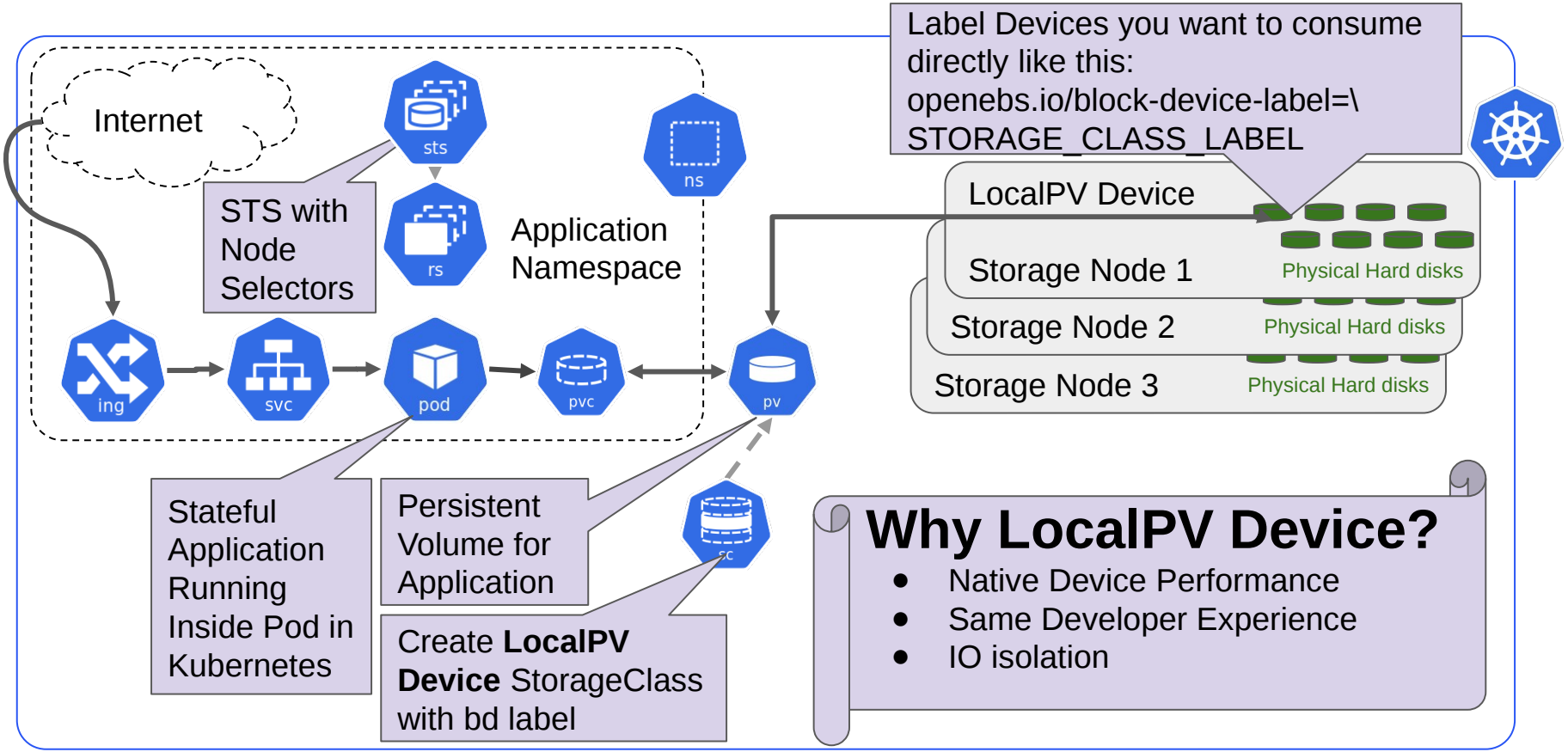
https://dok.community/

Q and A

Kubernetes Slack #openebs

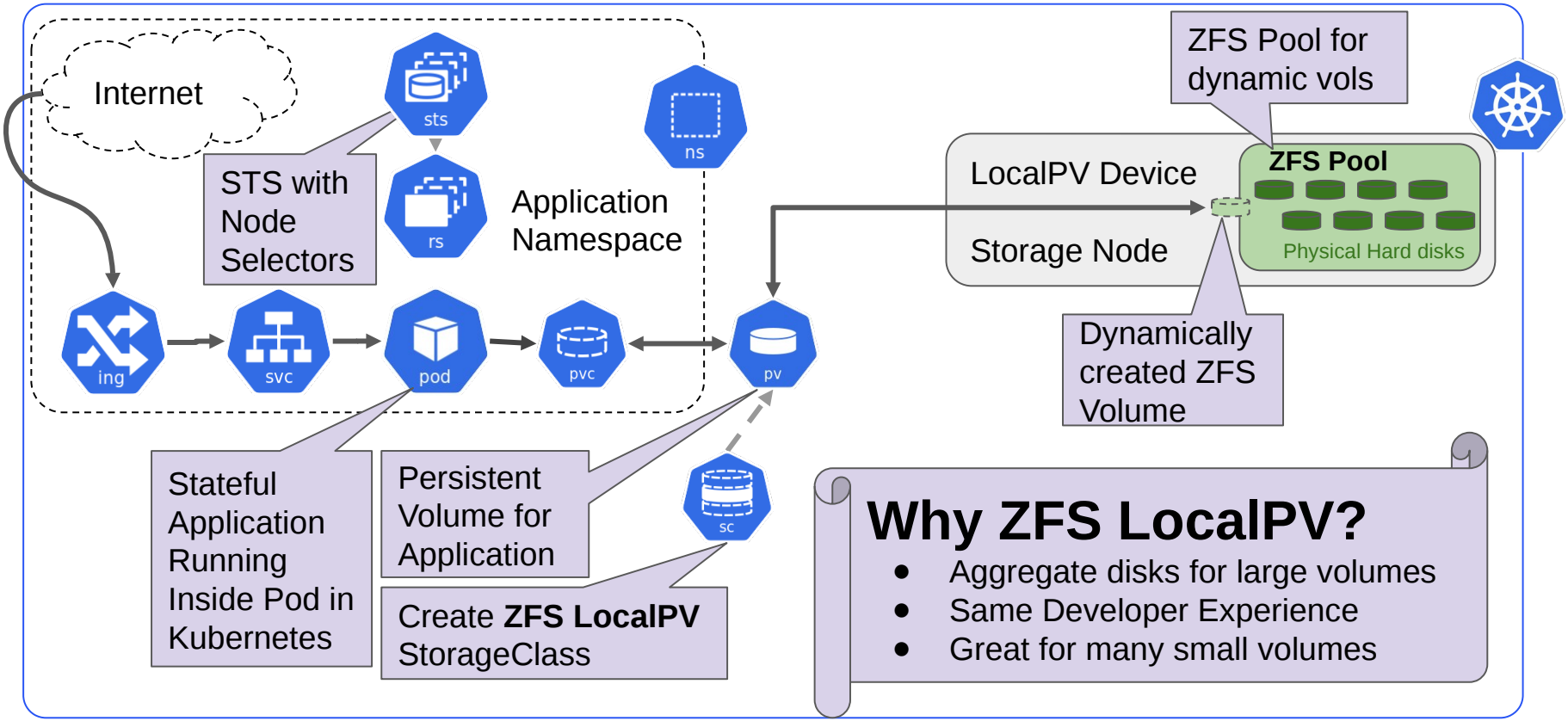
Storage Operations fades away!

LocalPV Device: Performance with Isolation



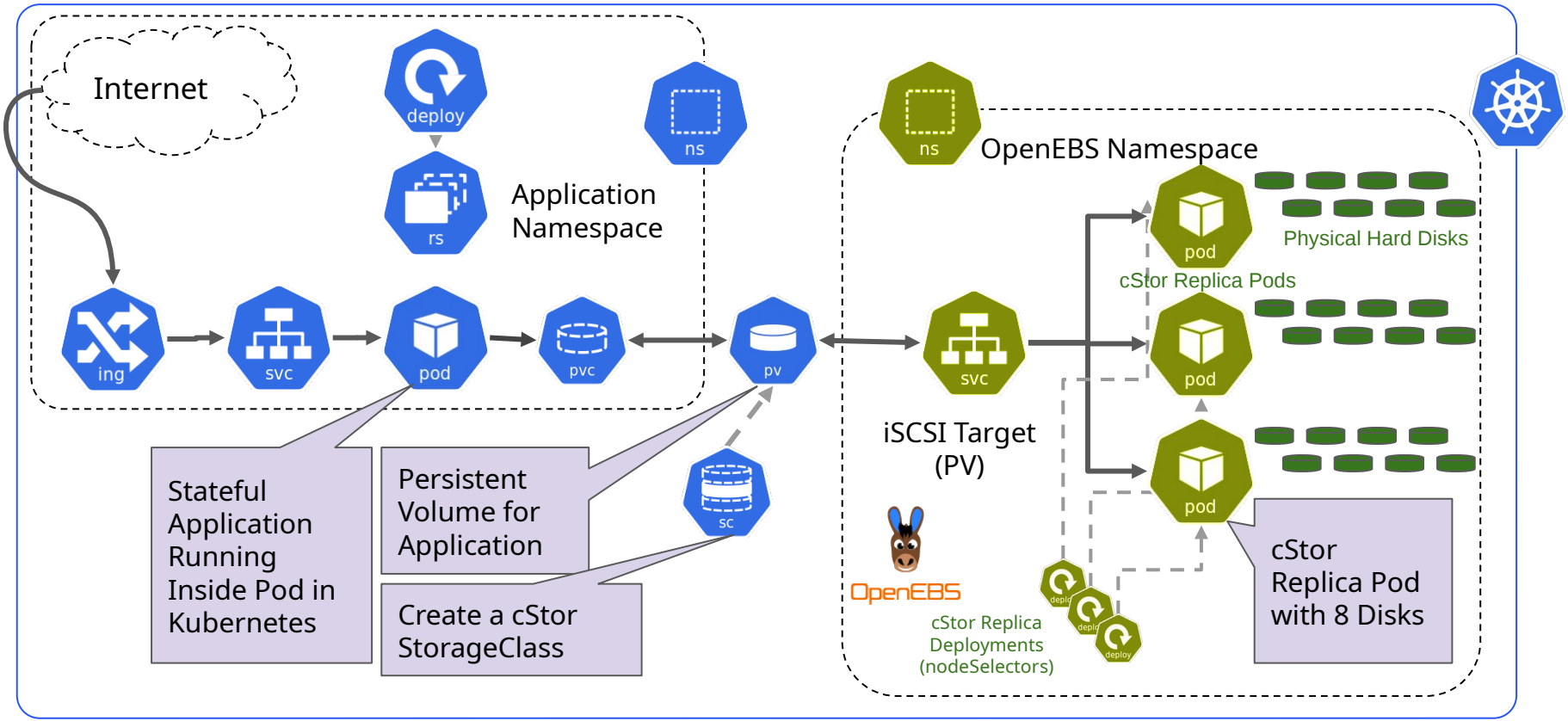
MayaData

ZFS LocalPV: Aggregation and Dynamic Sizing



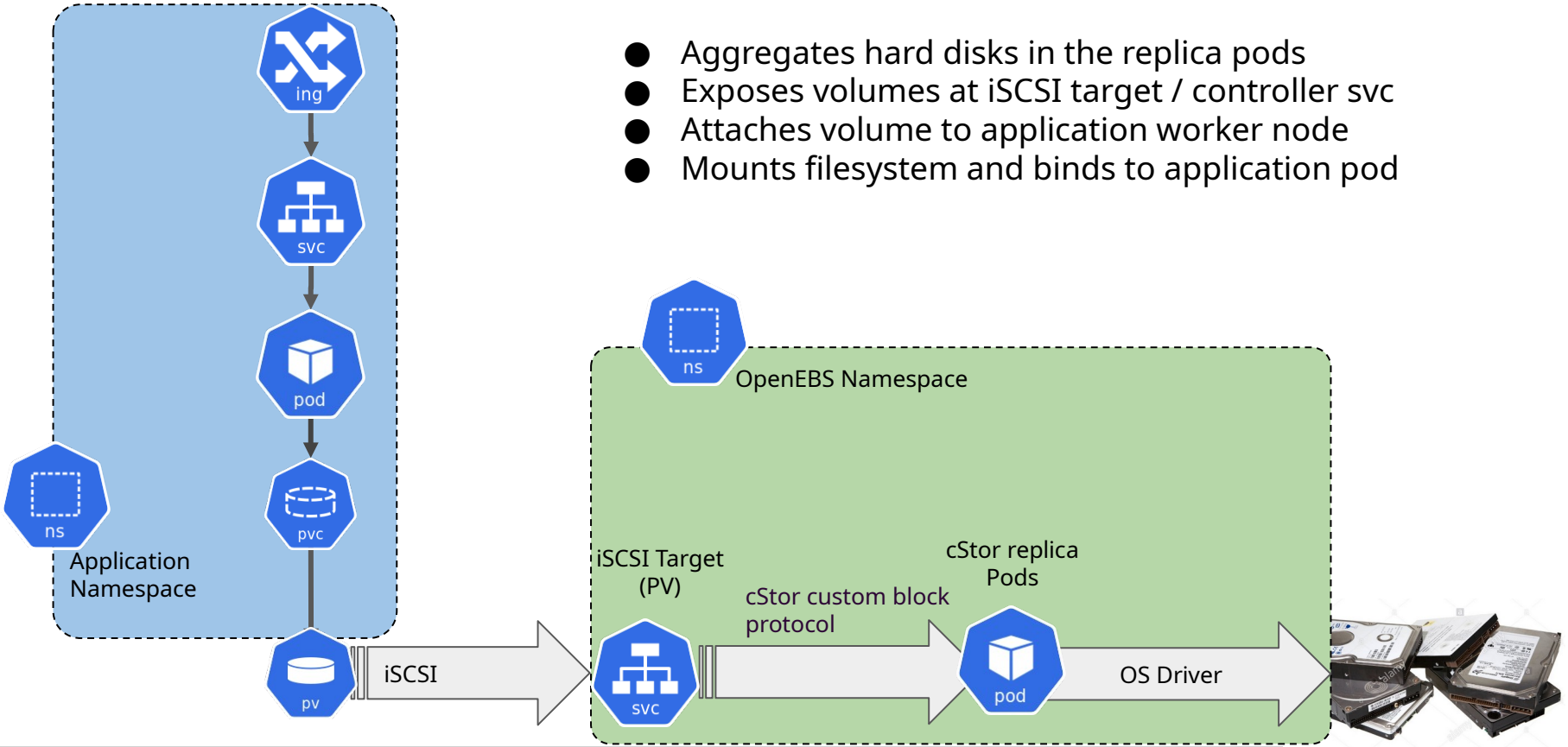
MayaData

Connect a Stateful App to OpenEBS cStor Storage



MayaData

cStor IO Path



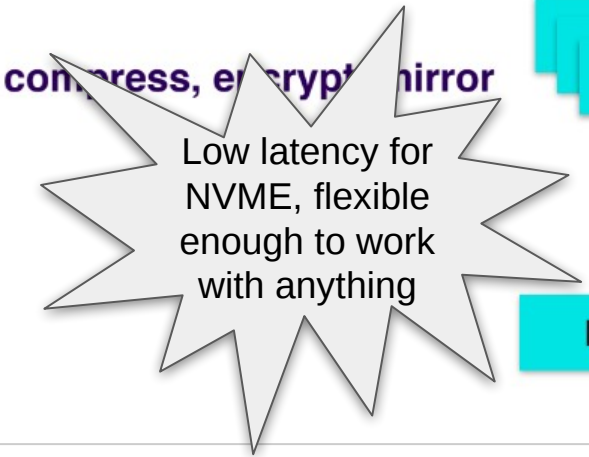
- Aggregates hard disks in the replica pods
- Exposes volumes at iSCSI target / controller svc
- Attaches volume to application worker node
- Mounts filesystem and binds to application pod

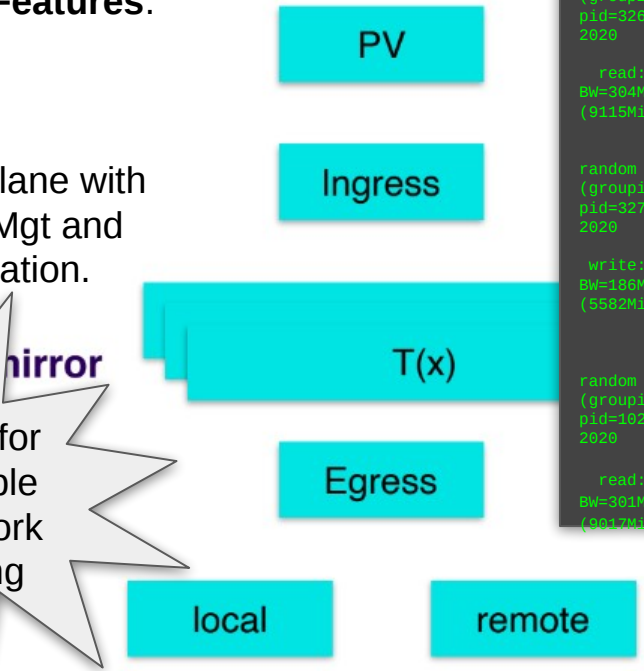
MayaData

cStor vs Mayastor IO path

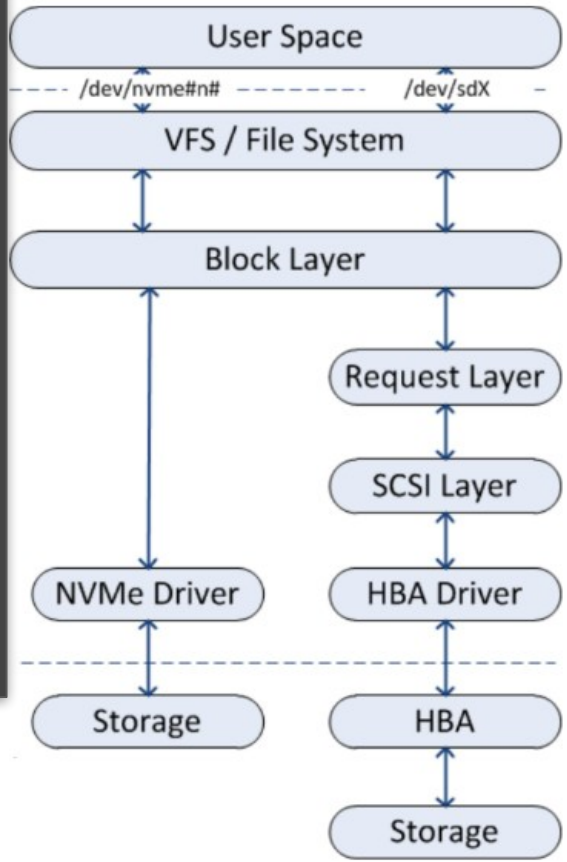
New
Storage Engine for
Performance with Features.

Same
Declarative,
Composable Data Plane with
Developer Friendly Mgt and
API-Driven Orchestration.






```
sequential read 2-way mirror:
(groupid=0, jobs=1): err= 0:
pid=32573: Thu Feb 13 21:23:41
2020

  read: IOPS=79.2k,
BW=309MiB/s (324MB/s)
(9280MiB/30002msec)


sequential write 2-way mirror:
(groupid=1, jobs=1): err= 0:
pid=32651: Thu Feb 13 21:23:41
2020

  read: IOPS=77.8k,
BW=304MiB/s (319MB/s)
(9115MiB/30002msec)


random write 2-way mirror:
(groupid=2, jobs=1): err= 0:
pid=32718: Thu Feb 13 21:23:41
2020

 write: IOPS=47.6k,
BW=186MiB/s (195MB/s)
(5582MiB/30003msec)


random read 2-way mirror:
(groupid=0, jobs=1): err= 0:
pid=1020: Thu Feb 13 21:23:41
2020

  read: IOPS=76.9k,
BW=301MiB/s (315MB/s)
(9017MiB/30002msec)
```



MayaData