KubeCon | CloudNativeCon

North America 2023

Cloud Native Storage: Storage TAG Intro, Projects, Landscape & Technology

Alex Chircop, Akamai
Xing Yang, VMware
Raffaele Spazzoli, Red Hat

Agenda

- Overview of the TAG
  - How to join and how to help
  - Overview of storage projects in the CNCF
- What is Cloud Native Storage
  - Why it is important
  - What's New
- Overview of the CNCF Storage Landscape document
- Overview of Data on Kubernetes document
- Overview of the Performance and Benchmarking document
- Overview of the Cloud Native DR document
- Community

Meetings

TAG Storage

CNCF SIGs were renamed TAGs (Technical Advisory Groups)

Meetings are on the 2nd and 4th Wednesday of every month at 8AM PT (USA Pacific)

- **Home:** https://github.com/cncf/tag-storage
- **Conf call:** http://bit.ly/cncf-storage-tag-call
- **Agenda:** http://bit.ly/cncf-storage-tag-minutes
- **Mail list:** https://lists.cncf.io/g/cncf-tag-storage

Our calls and membership are open!

Who we are

- We are a diverse set of users & developers of Cloud Native technologies with a storage focus
- We are leaders & early adopters

Co-Chairs	Tech Leads	TOC Liaisons
■Alex Chircop	■Luis Pabón	■Nikhita Raghunat
■Xing Yang	■Sheng Yang	■Matt Farina
■Raffaele Spazzoli	■Nick Connolly	

Questions? Reach out and feel free to connect on our mailing list, and CNCF Slack!

What we do

“Scale contributions by the CNCF technical and user community, while retaining integrity and increasing quality in support of the CNCF mission (to make cloud native computing ubiquitous).”

What we do

“Scale contributions by the CNCF technical and user community, while retaining integrity and increasing quality in support of the CNCF mission (to make cloud native computing ubiquitous).”

...this means we

What we do

“Scale contributions by the CNCF technical and user community, while retaining integrity and increasing quality in support of the CNCF mission (to make cloud native computing ubiquitous).”

...this means we

- **Educate**
- **Review Projects**
- **Engage with the user community**
- **Provide subject matter expertise**

CNCF Storage Projects

Graduated

ROOK

Vitess

HARBOR

etcd

TiKV

Incubating

Dragonfly

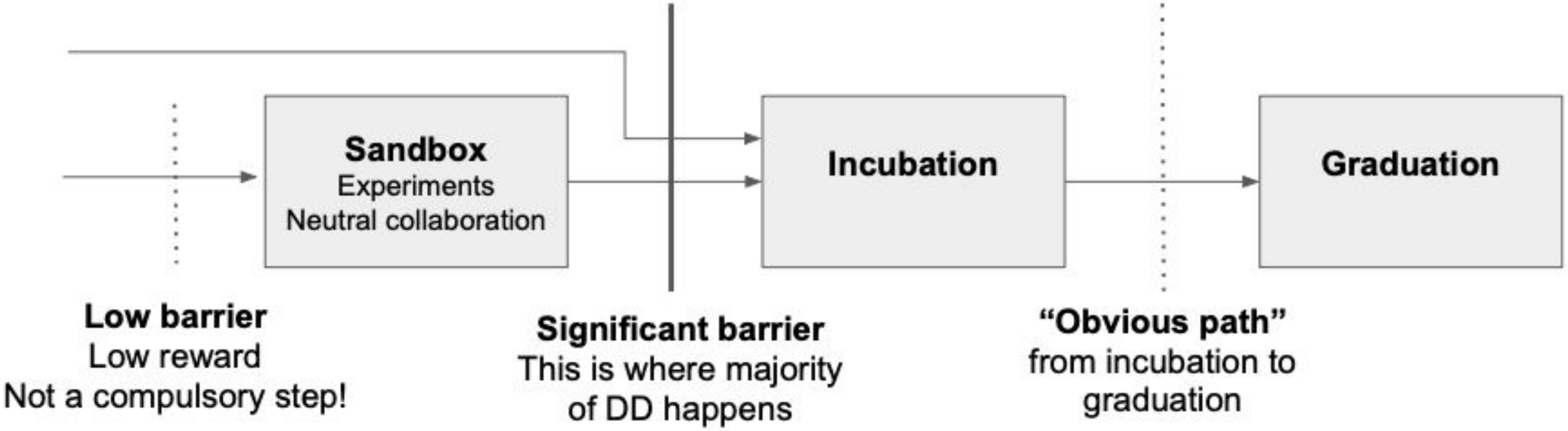
CubeFS

LONGHORN

CNCF Projects: https://www.cncf.io/projects/
Sandbox Projects: https://www.cncf.io/sandbox-projects/

CNCF Storage Projects

https://www.cncf.io/projects/



Sandbox
Experiments
Neutral collaboration

Incubation

Graduation

Low barrier
Low reward
Not a compulsory step!

Significant barrier
This is where majority
of DD happens

"Obvious path"
from incubation to
graduation

Sandbox	Incubation	Graduation
■Experiments ■IP Policy ■Build Community	■Used successfully in production ■Healthy number of committers ■Project metrics	■Mainstream production use ■Security audits ■Committers from multiple organisations

Cloud Native Storage

Why should you think about this?

Cloud Native Storage

Why should you think about this?

THERE IS **NO SUCH THING** AS A "**STATELESS**" ARCHITECTURE

IT'S JUST **SOMEONE ELSE'S PROBLEM**

Cloud Native Storage

Why should you think about this?

There is no such thing as a stateless application …

… all applications store state somewhere!

Cloud Native Storage is Here!

Move Stateful Workloads to K8s

- Automation
- Scale
- Performance
- Failover

➪ **Broad ecosystem and CSI support**

➪ **Operators for databases, message queues, *and many more!***

CNCF Storage Whitepaper

Whitepaper: https://bit.ly/cncf-storage-whitepaperV2

1. Definition of the attributes of a storage system

2. Definition of the layers in a storage solution with a focus on terminology and how they impact the attributes

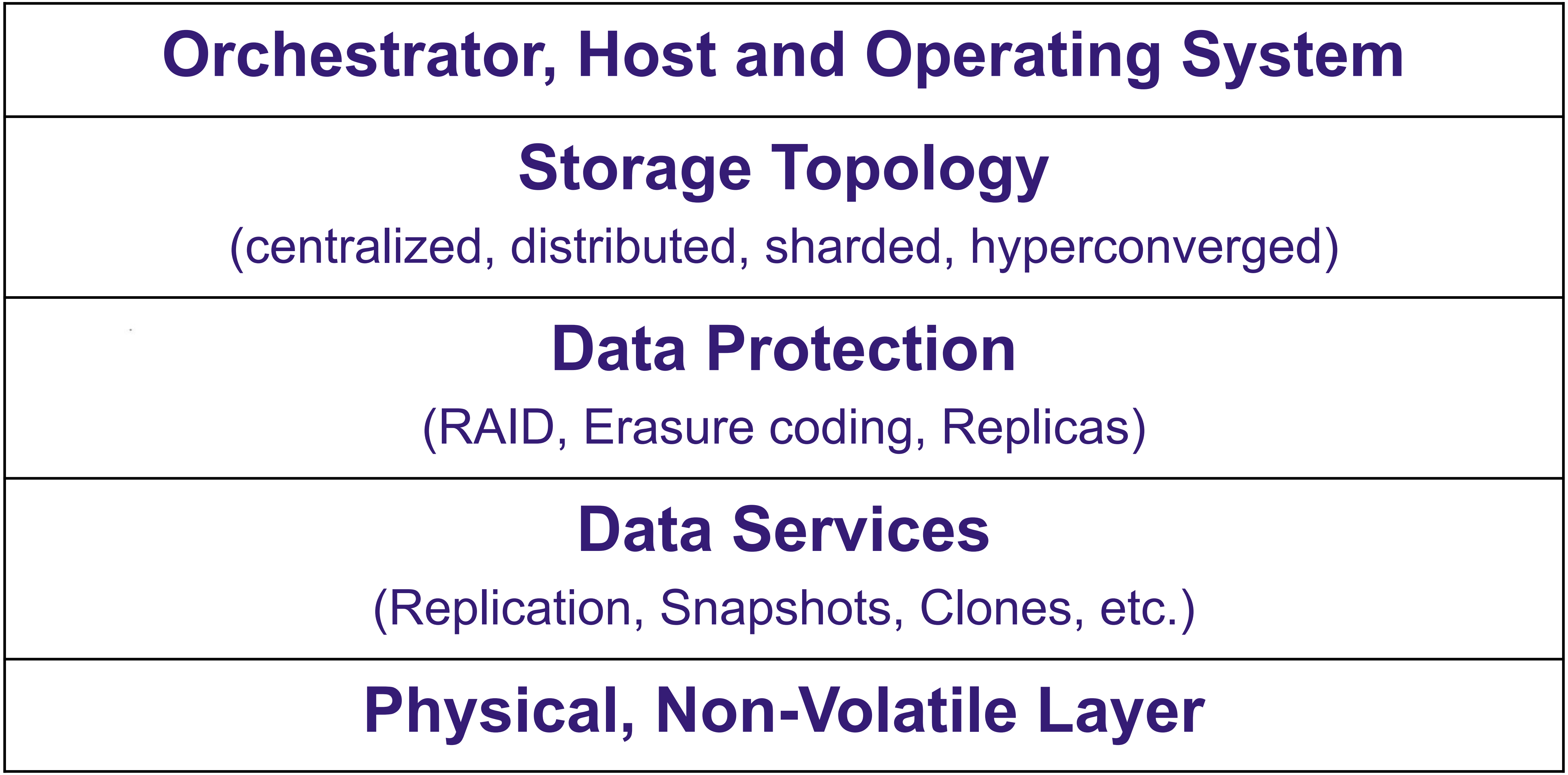
3. Definition of the data access interfaces in terms of volumes and application APIs

4. Definition of the management interfaces

Storage Attributes

Availability	Scalability	Performance	Consistency	Durability
Failover	Clients	Latency	Delay to access correct data after a commit	Data protection
Moving access between nodes	Operations	Operations		Redundancy
Redundancy	Throughput	Throughput	Delay between commit and data being committed to non-volatile store	Bit-Rot
Data Protection	Components			

Storage Layers



Orchestrator, Host and Operating System

Storage Topology

(centralized, distributed, sharded, hyperconverged)

Data Protection

(RAID, Erasure coding, Replicas)

Data Services

(Replication, Snapshots, Clones, etc.)

Physical, Non-Volatile Layer

Why this matters …

Let’s take a look at a couple of different use cases and deployments:

- Hyperconverged
- Block Volumes
- Shared Filesystems
- Object Stores

Why this matters …

Let’s take a look at a couple of different use cases and deployments:

- **Hyperconverged**
  - availability: converged fault and change management domains
  - performance: shared network and compute

Why this matters …

Let’s take a look at a couple of different use cases and deployments:

- **Block Volumes**
  - useful to disaggregate compute and storage
  - availability: ability to move volumes between nodes
  - performance: typically lower latency, but needs good connectivity between compute and storage nodes

Why this matters …

Let’s take a look at a couple of different use cases and deployments:

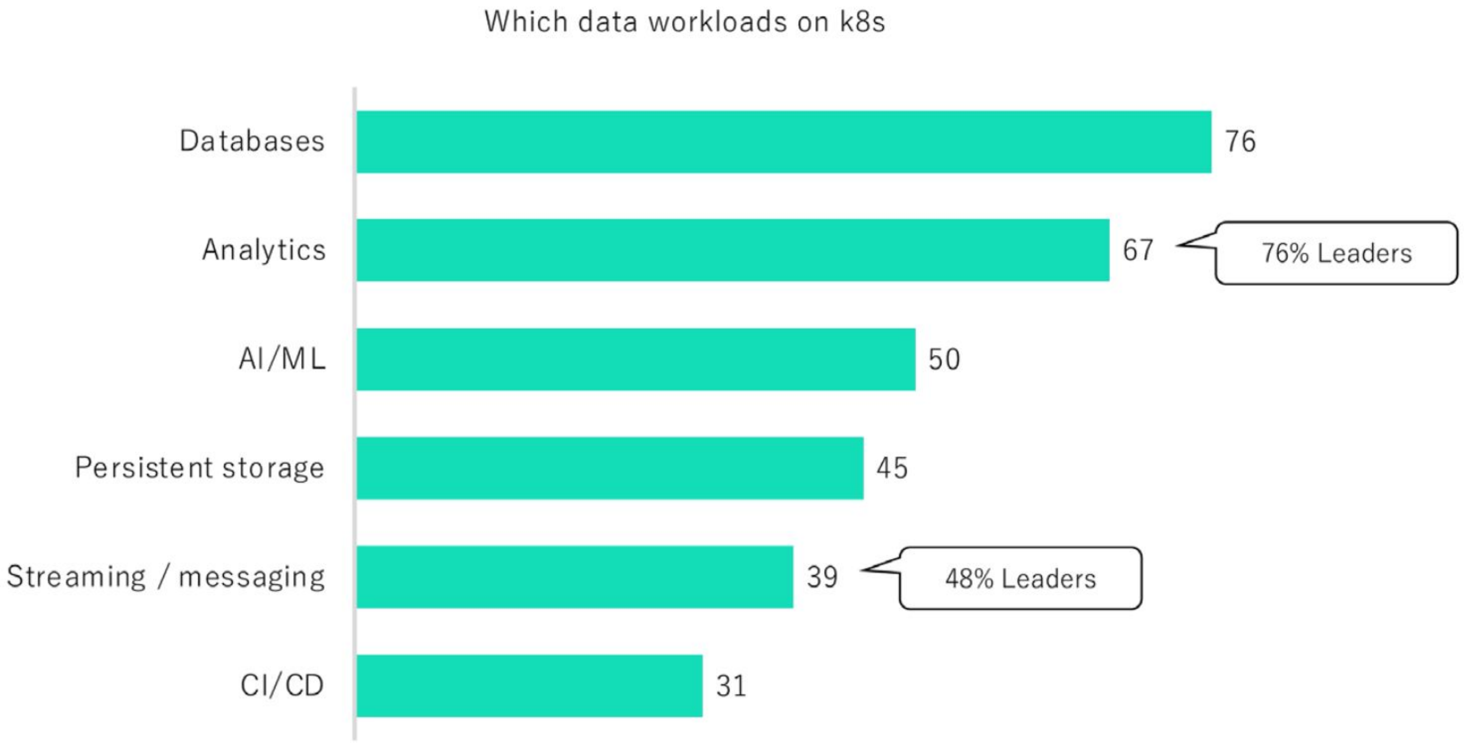
- **Shared Filesystems**
  - can be used by multiple nodes at the same time
  - consistency: distributed locks, cache coherency is hard
  - layers: could be built on block, object stores etc … and that determines many attributes

Why this matters …

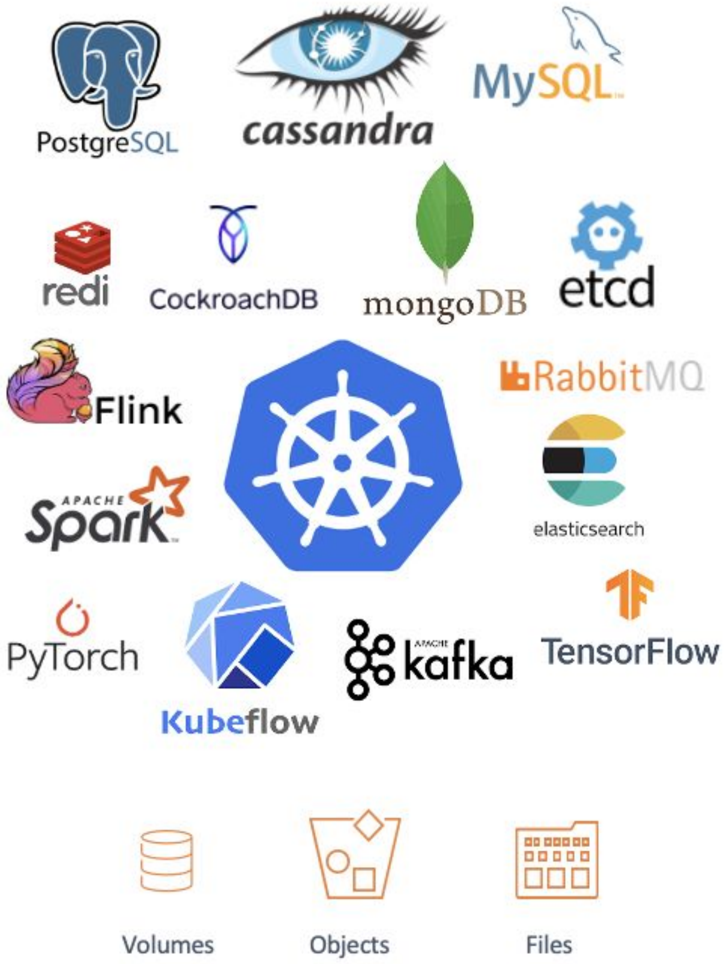
Let’s take a look at a couple of different use cases and deployments:

- **Object Stores**
  - scale: almost infinite for capacity, and throughput
  - latency: higher than data on a volume
  - performance: RPS is often the determining factor

Data Workloads on Kubernetes



Data on Kubernetes Community 2022 Survey



Data on Kubernetes Whitepaper

https://bit.ly/cncf-storage-dok-whitepaper

- Describe patterns of running data on Kubernetes
- Collaborating with Data on Kubernetes Community (DoKC)
- Focusing on databases in v1
- Paper layout
  - Attributes of a storage system and how they affect running data in Kubernetes
  - Running data inside vs outside of Kubernetes
  - Common Kubernetes patterns and features used when running data on Kubernetes
  - Observability
  - Security
  - Day 2 operations

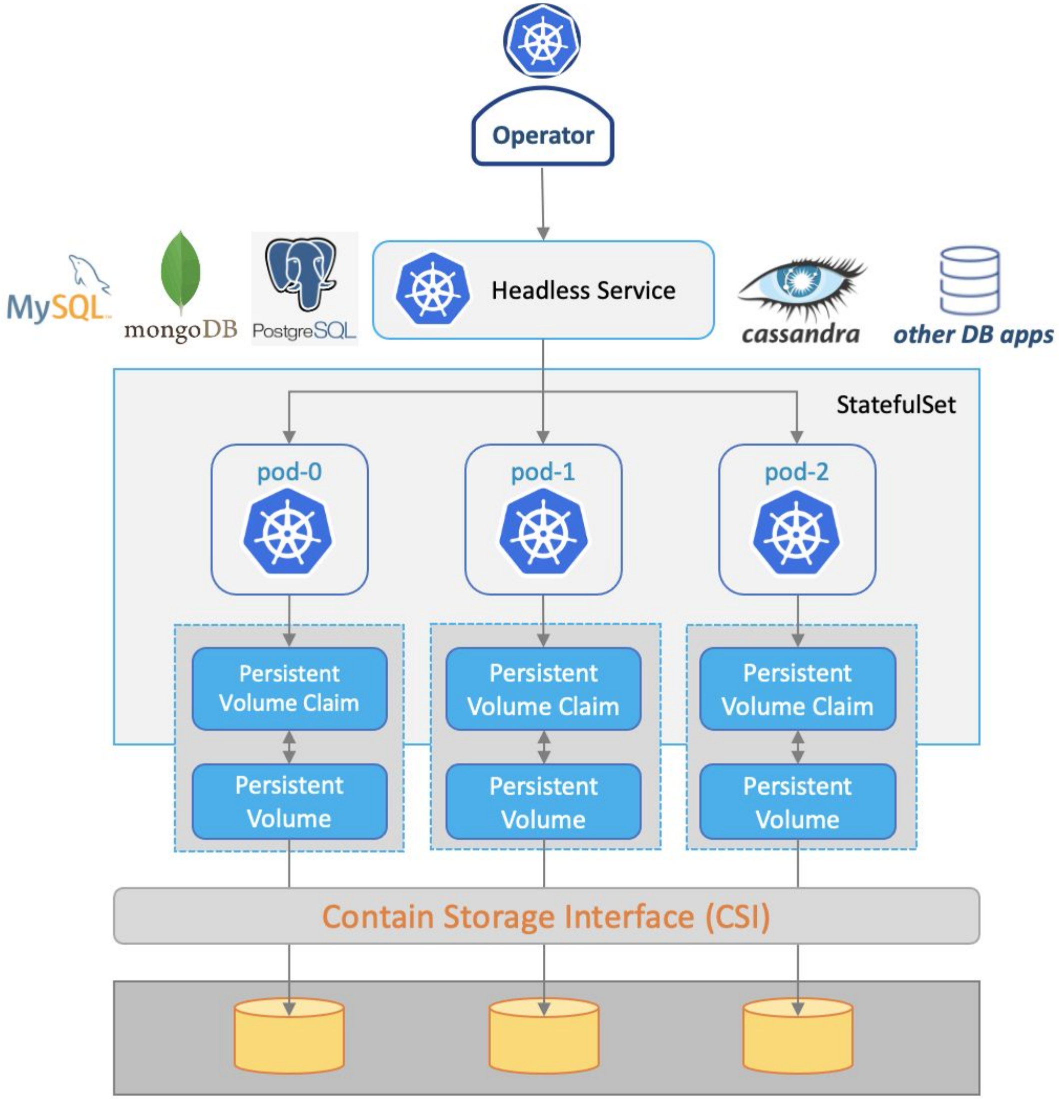
Storage Attributes and Running Data in K8s

- Storage System Attributes
  - Attributes
    - Availability
    - Consistency
    - Durability
    - Scalability
    - Performance
    - Observability
    - Elasticity
  - Storage Stacks
    - Stacks/Layers
    - Disaster Recovery

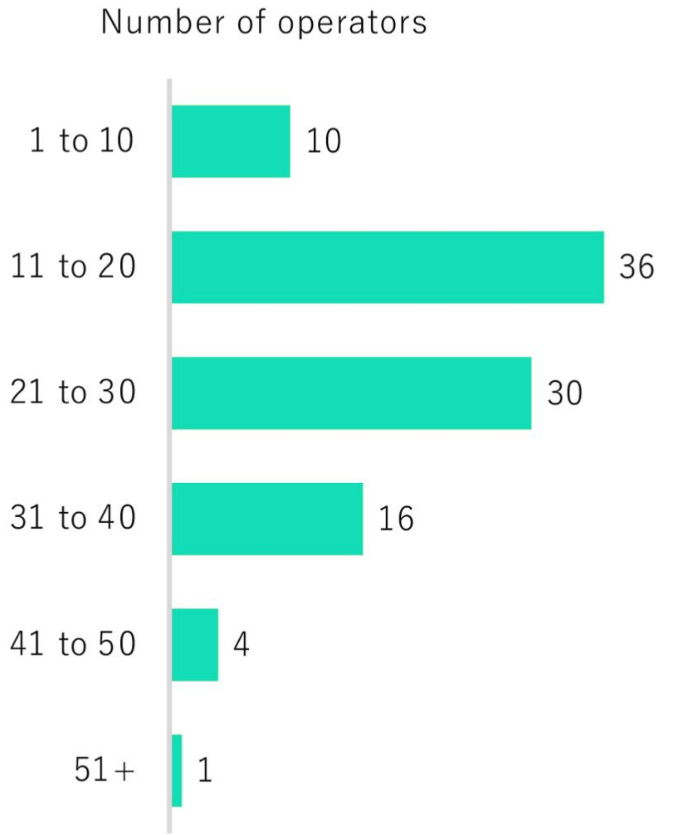
Running Data inside vs outside of K8s

- Managed database services, provided by most cloud providers
- Running data inside K8s with operators
    - Declarative approach
    - Automate “Day 2 Operations”
    - Externalize database functionalities such as monitoring, cert management to third parties

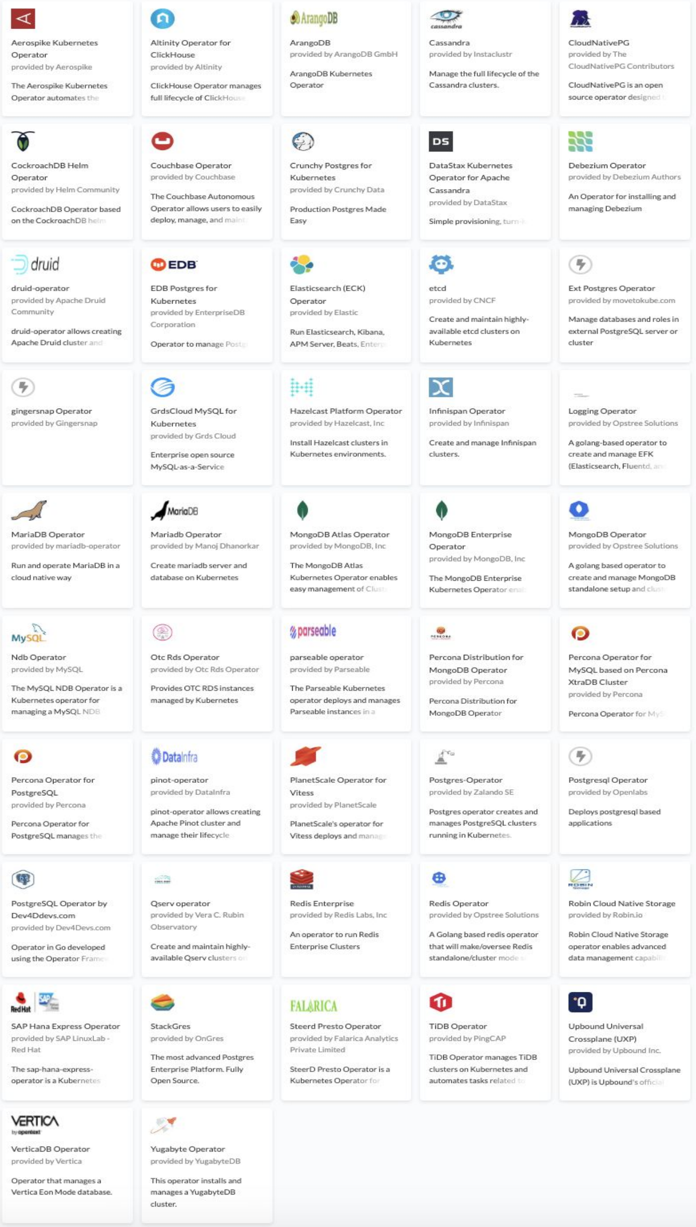
Kubernetes Operators



Kubernetes Operators (cont.)



Data on Kubernetes Community 2022 Survey



Aerospike Kubernetes Operator
provided by Aerospike
The Aerospike Kubernetes Operator automates the

Altinity Operator for ClickHouse
provided by Altinity
ClickHouse Operator manages full lifecycle of ClickHouse

ArangoDB
provided by ArangoDB GmbH
ArangoDB Kubernetes Operator

Cassandra
provided by Instaclustr
Manage the full lifecycle of the Cassandra clusters.

CloudNativePG
provided by The CloudNativePG Contributors
CloudNativePG is an open source operator designed

CockroachDB Helm Operator
provided by Helm Community
CockroachDB Operator based on the CockroachDB helm

Couchbase Operator
provided by Couchbase
The Couchbase Autonomous Operator allows users to easily deploy, manage, and maint

Crunchy Postgres for Kubernetes
provided by Crunchy Data
Production Postgres Made Easy

DataStax Kubernetes Operator for Apache Cassandra
provided by DataStax
Simple provisioning, turn-k

Debezium Operator
provided by Debezium Authors
An Operator for installing and managing Debezium

druid-operator
provided by Apache Druid Community
druid-operator allows creating Apache Druid cluster and

EDB Postgres for Kubernetes
provided by EnterpriseDB Corporation
Operator to manage Postg

Elasticsearch (ECK) Operator
provided by Elastic
Run Elasticsearch, Kibana, APM Server, Beats, Enterp

etcd
provided by CNCF
Create and maintain highly-available etcd clusters on Kubernetes

Ext Postgres Operator
provided by movetokube.com
Manage databases and roles in external PostgreSQL server or cluster

gingersnap Operator
provided by Gingersnap

GrdsCloud MySQL for Kubernetes
provided by Grds Cloud
Enterprise open source MySQL-as-a-Service

Hazelcast Platform Operator
provided by Hazelcast, Inc
Install Hazelcast clusters in Kubernetes environments.

Infinispan Operator
provided by Infinispan
Create and manage Infinispan clusters.

Logging Operator
provided by Opstree Solutions
A golang-based operator to create and manage EFK (Elasticsearch, Fluentd, an

MariaDB Operator
provided by mariadb-operator
Run and operate MariaDB in a cloud native way

Mariadb Operator
provided by Manoj Dhanorkar
Create mariadb server and database on Kubernetes

MongoDB Atlas Operator
provided by MongoDB, Inc
The MongoDB Atlas Kubernetes Operator enables easy management of Clus

MongoDB Enterprise Operator
provided by MongoDB, Inc
The MongoDB Enterprise Kubernetes Operator ena

MongoDB Operator
provided by Opstree Solutions
A golang based operator to create and manage MongoDB standalone setup and clus

Ndb Operator
provided by MySQL
The MySQL NDB Operator is a Kubernetes operator for managing a MySQL NDB

Otc Rds Operator
provided by Otc Rds Operator
Provides OTC RDS instances managed by Kubernetes

parseable operator
provided by Parseable
The Parseable Kubernetes operator deploys and manages Parseable instances in a

Percona Distribution for MongoDB Operator
provided by Percona
Percona Distribution for MongoDB Operator

Percona Operator for MySQL based on Percona XtraDB Cluster
provided by Percona
Percona Operator for MyS

Percona Operator for PostgreSQL
provided by Percona
Percona Operator for PostgreSQL manages the

pinot-operator
provided by DataInfra
pinot-operator allows creating Apache Pinot cluster and manage their lifecycle

PlanetScale Operator for Vitess
provided by PlanetScale
PlanetScale's operator for Vitess deploys and manag

Postgres-Operator
provided by Zalando SE
Postgres operator creates and manages PostgreSQL clusters running in Kubernetes.

Postgresql Operator
provided by Openlabs
Deploys postgresql based applications

PostgreSQL Operator by Dev4Ddevs.com
provided by Dev4Devs.com
Operator in Go developed using the Operator Frame

Qserv operator
provided by Vera C. Rubin Observatory
Create and maintain highly-available Qserv clusters on

Redis Enterprise
provided by Redis Labs, Inc
An operator to run Redis Enterprise Clusters

Redis Operator
provided by Opstree Solutions
A Golang based redis operator that will make/oversee Redis standalone/cluster mode s

Robin Cloud Native Storage
provided by Robin.io
Robin Cloud Native Storage operator enables advanced data management capabi

SAP Hana Express Operator
provided by SAP LinuxLab - Red Hat
The sap-hana-express-operator is a Kubernetes

StackGres
provided by OnGres
The most advanced Postgres Enterprise Platform. Fully Open Source.

Steerd Presto Operator
provided by Falarica Analytics Private Limited
SteerD Presto Operator is a Kubernetes Operator for

TiDB Operator
provided by PingCAP
TiDB Operator manages TiDB clusters on Kubernetes and automates tasks related to

Upbound Universal Crossplane (UXP)
provided by Upbound Inc.
Upbound Universal Crossplane (UXP) is Upbound's officia

VerticaDB Operator
provided by Vertica
Operator that manages a Vertica Eon Mode database.

Yugabyte Operator
provided by YugabyteDB
This operator installs and manages a YugabyteDB cluster.

- https://operatorhub.io/
  - 331 operators
  - 47 database operators, including etcd and Vitess
  - 9 PostgreSQL operators, including CloudNativePG
- Other operators not listed in operatorhub
- https://github.com/dokc/operator-feature-matrix

Common K8s Patterns and Features

- Kubernetes Operators
- Container Storage Interface (CSI)
- Kubernetes Workload APIs
- Topology Aware Scheduling
- Pod Disruption Budget
- Resource Management
- Separation of Control Plane and Data Plane
- Default Secure

Performance Whitepaper

Whitepaper:
https://bit.ly/cncf-tag-storage-performance-benchmarking

- Definition of common concepts for measuring performance and benchmarking for volumes and databases
- Definition of common pitfalls and considerations
  - Operations vs Throughput
  - Topology, Data Protection, Data Reduction, Encryption matters …
  - But Latency often matters more …
  - Concurrency for queues, clients and backends
  - Caching happens at multiple layers
  - Be critical and beware of results that are too good to be true!

Performance Whitepaper

Whitepaper:
https://bit.ly/cncf-tag-storage-performance-benchmarking

TL;DR - important takeaway:

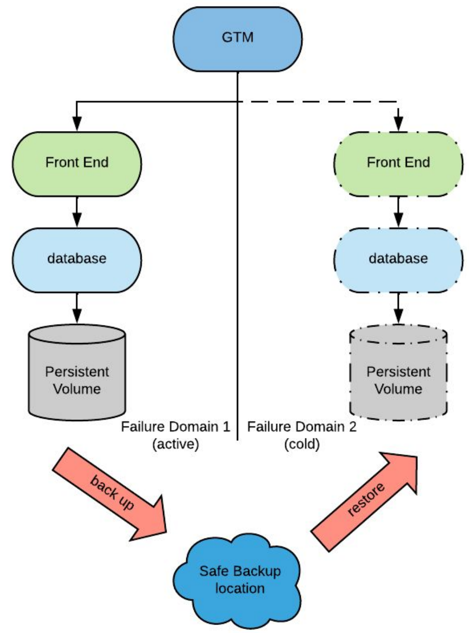
published results are not useful for making comparisons - it is hard to compare published results without a deep understanding of the test conditions, so it is always important to run your own test, on your own environment with your own applications

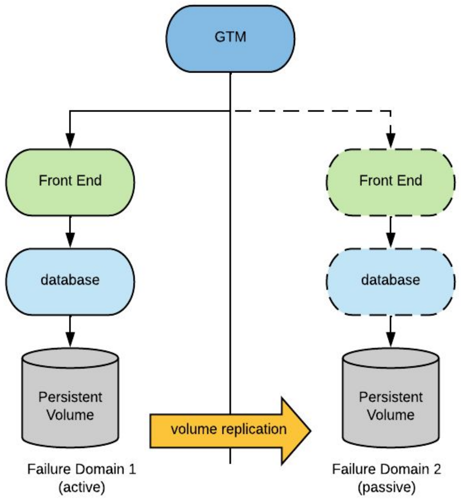
High-level DR Approaches

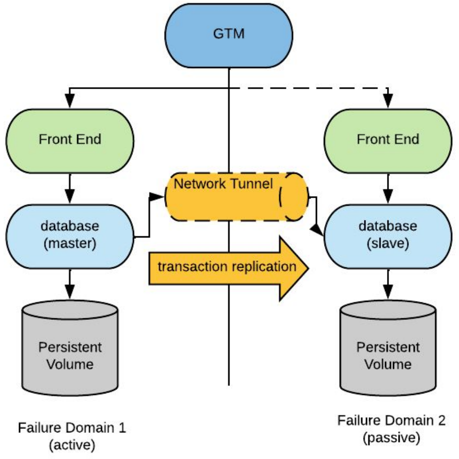
Failure Domain is either a data center or a cloud region

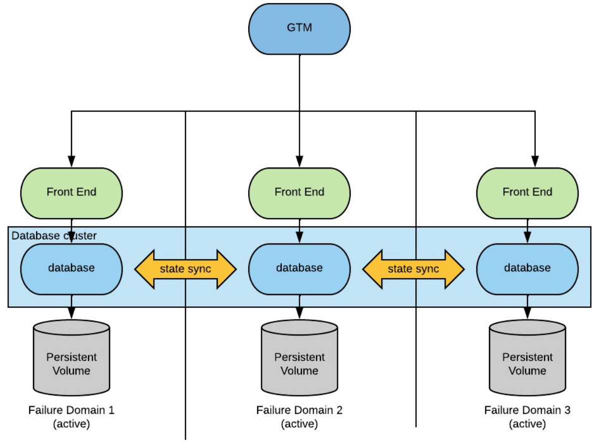
Active/Passive

Active/Active









Backup / Restore	Volume Replication	Transaction Replication	Distributed Stateful Workloads
Storage Capabilities			
❑ backup/restore	❑ sync/async repl		
Network Capabilities			
❑ global load balancer	❑ global load balancer	❑ global load balancer ❑ east-west path	❑ global load balancer ❑ east-west path
Workload Capabilities			
		❑ primary/secondary	❑ distributed stateful workload

Capabilities and Products

Backup & Restore

OADP

kasten by Veeam

TRILIOVAULT

COHESITY

portworx by Pure Storage

VELERO

k8up

Global Load Balancing

+ DNS

K8GB

Primary/ Secondary enabled middleware

MySQL Development Services

Cloud SQL

Amazon Aurora

SQL Azure SQL

Volume Replication

Red Hat OpenShift Data Foundation

portworx by Pure Storage

LONGHORN

ROOK

East-West path

SUBMARINER

CALICO

cilium

Distributed middleware

HashiCorp Vault

yugabyteDB

mongoDB.

cassandra

CockroachDB

kafka

Azure Cosmos DB

Google Cloud Spanner

NATS

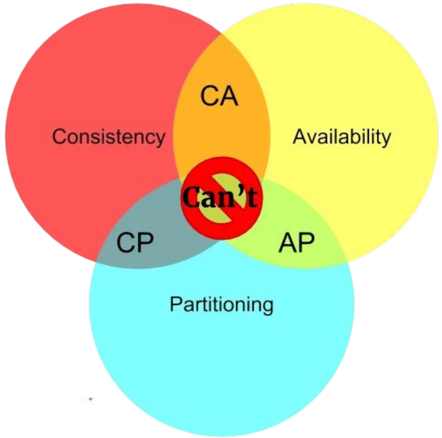
PULSAR

Cloud Native Disaster Recovery

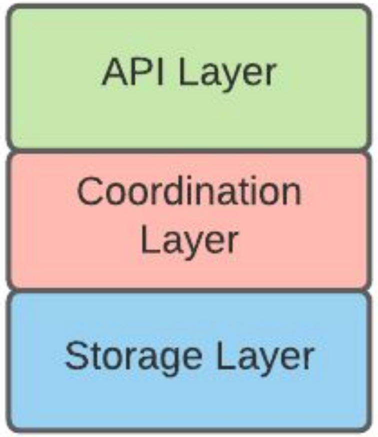
Whitepaper: http://bit.ly/cncf-cloud-native-DR

Concern	Traditional DR	Cloud Native DR
Type of deployment	active/passive, rarely active/active	Active / active
Disaster Detection and Recovery Trigger	Human	Autonomous
Disaster Recovery Procedure execution	Mix of manual and automated tasks	Automated
Recovery Time Objective (RTO)	From close to zero to hours	Close to zero
Recovery Point Objective (RPO)	From zero to hours	Exactly zero for strongly consistent deployments. Theoretically unbounded, practically close to zero for eventual consistent deployments.
DR Process Owner	Often the Storage Team	Application Team
Capabilities needed for DR	From storage (backup/restore, volume replication)	From networking (east-west communication, global load balancer)

Anatomy of a Distributed Stateful Workload

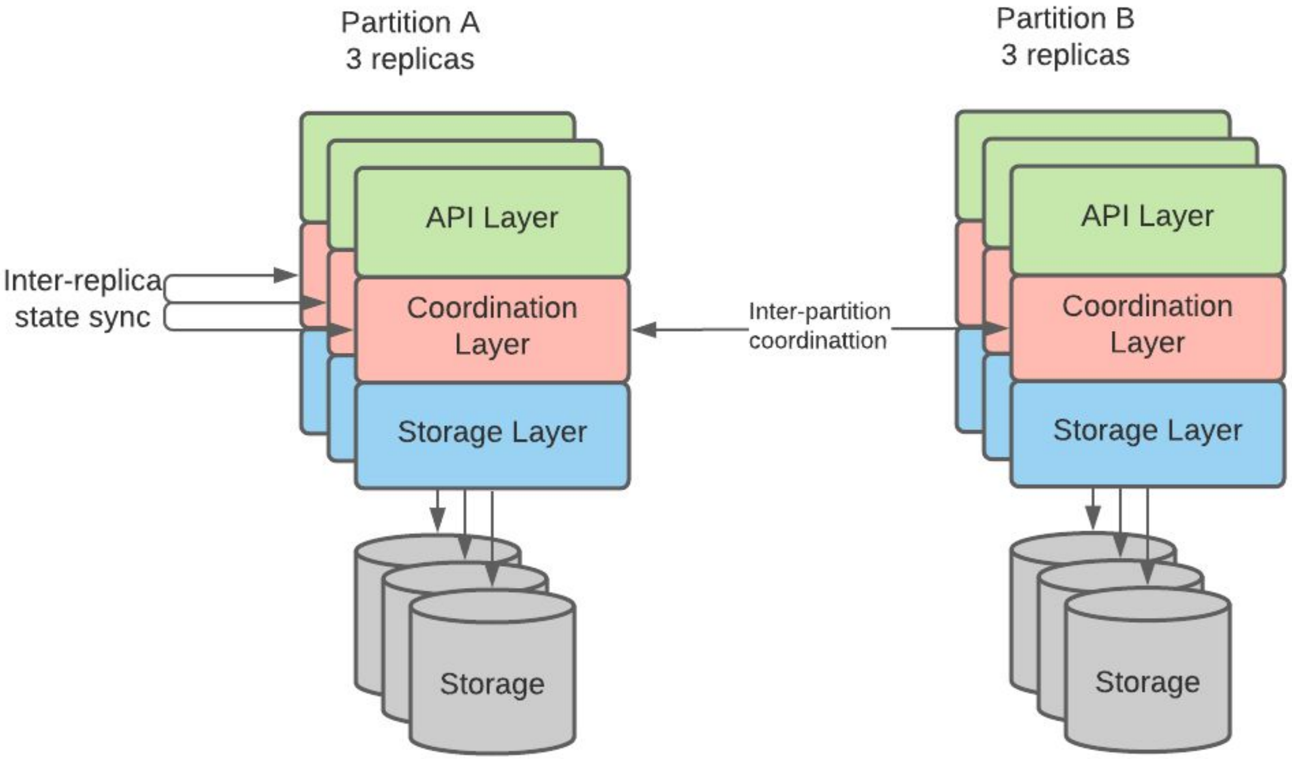


CAP Theorem



Stateful Workload Logical Tiers

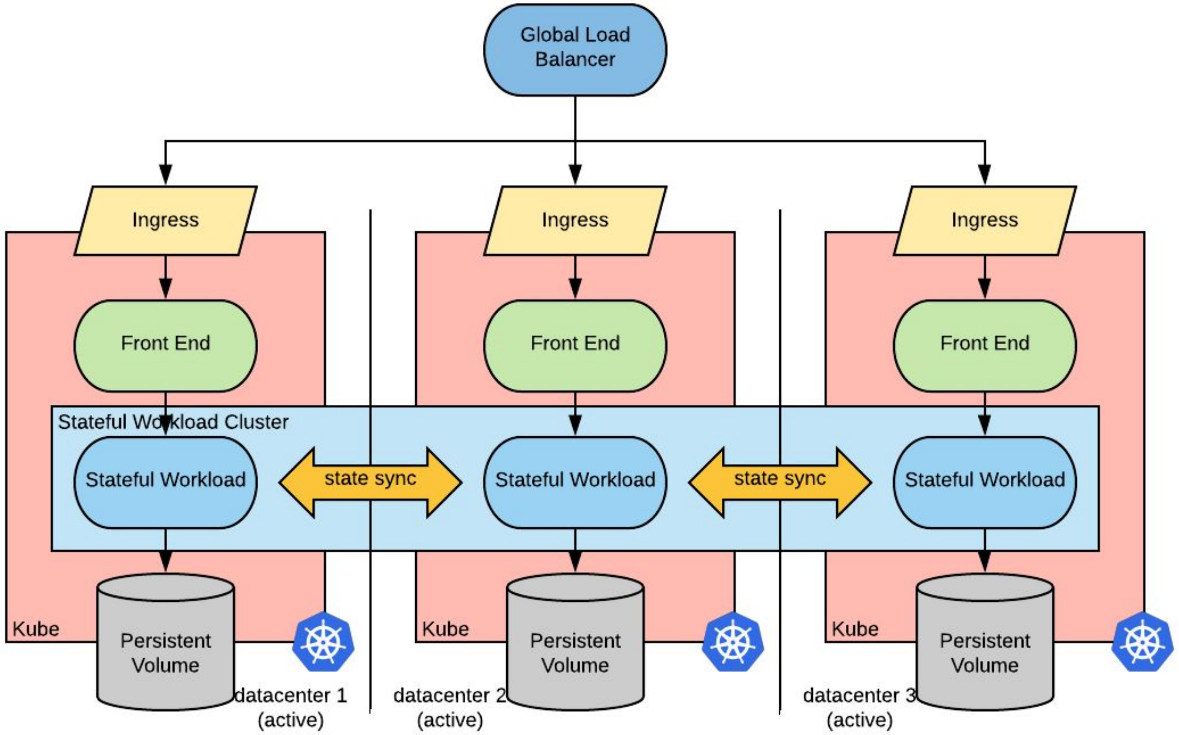
Replicas & Partitions

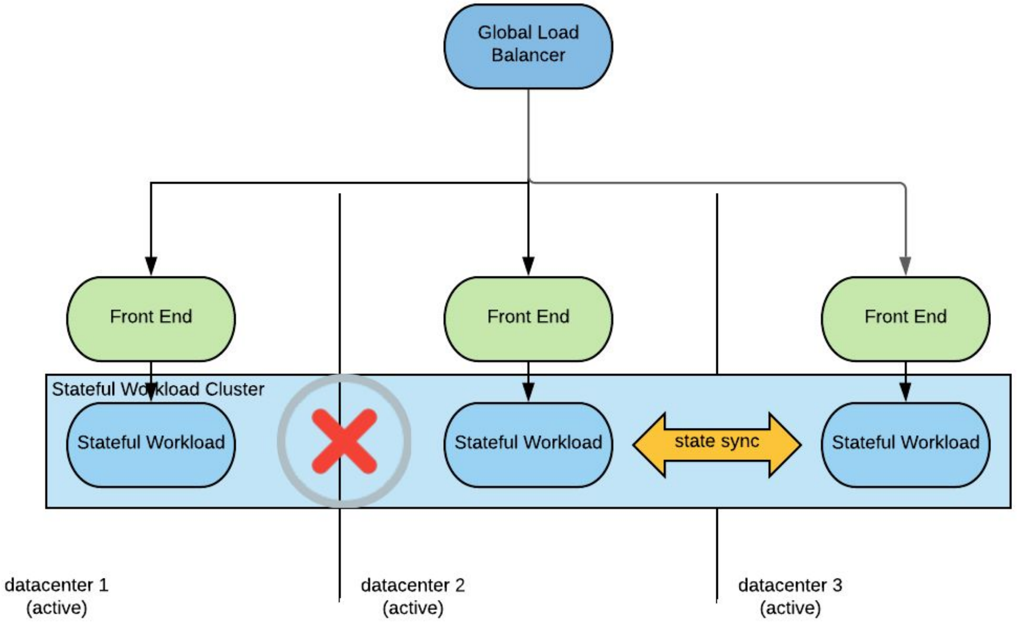


Examples of Consensus Protocol choices

Product	Replica consensus protocol	Shard consensus protocol
Etcd	Raft	N/A (no support for shards)
Consul	Raft	N/A (no support for shards)
Zookeeper	Atomic Broadcast (a derivative of Paxos)	N/A (no support for shards)
ElasticSearch	Paxos	N/A (No support for transactions)
Cassandra	Paxos	Supported, but details are not available.
MongoDB	Paxos	Homegrown protocol.
CockroachDB	Raft	2PC
YugabyteDB	Raft	2PC
TiKV	Raft	Percolator
Spanner	Raft	2PC+high-precision time service
Kafka	A custom derivative of PacificA	Custom Implementation of 2PC

CNDR -- Strong Consistency - Kubernetes Reference Architecture





Community

- How you can get involved?
    - Join our meeting
        - 2nd & 4th Wednesday each month
    - Submit and help review projects for consideration
- We value community presentations of projects in the cloud native storage space including, but not limited to: *management frameworks, block stores, filesystems, object stores, key-value stores and databases* -> learn and work with the community
- Consider a role in the TAG!
- Contribute to TAG projects and help the community!

Please scan the QR Code above
to leave feedback on this session