
KubeCon
CloudNativeCon
North America 2022
BUILDING FOR THE ROAD AHEAD
DETROIT 2022
TiKV
TiKV
HELM

BUILDING FOR THE ROAD AHEAD

DETROIT 2022

Longhorn: Intro, Deep Dive, Q & A

David Ko
Senior Engineering Manager, SUSE

Joshua Moody
Staff Software Engineer, SUSE

Agenda

- What is Longhorn
- Feature List
- Momentum, Community, Story/Roadmap
- Releases
- How Current Longhorn Works
  - Control Plane
  - Data Plane
  - Snapshot, Backup, Replica Rebuilding
  - Disaster Recovery
  - Volume Live Migration
- What is Next for Longhorn?

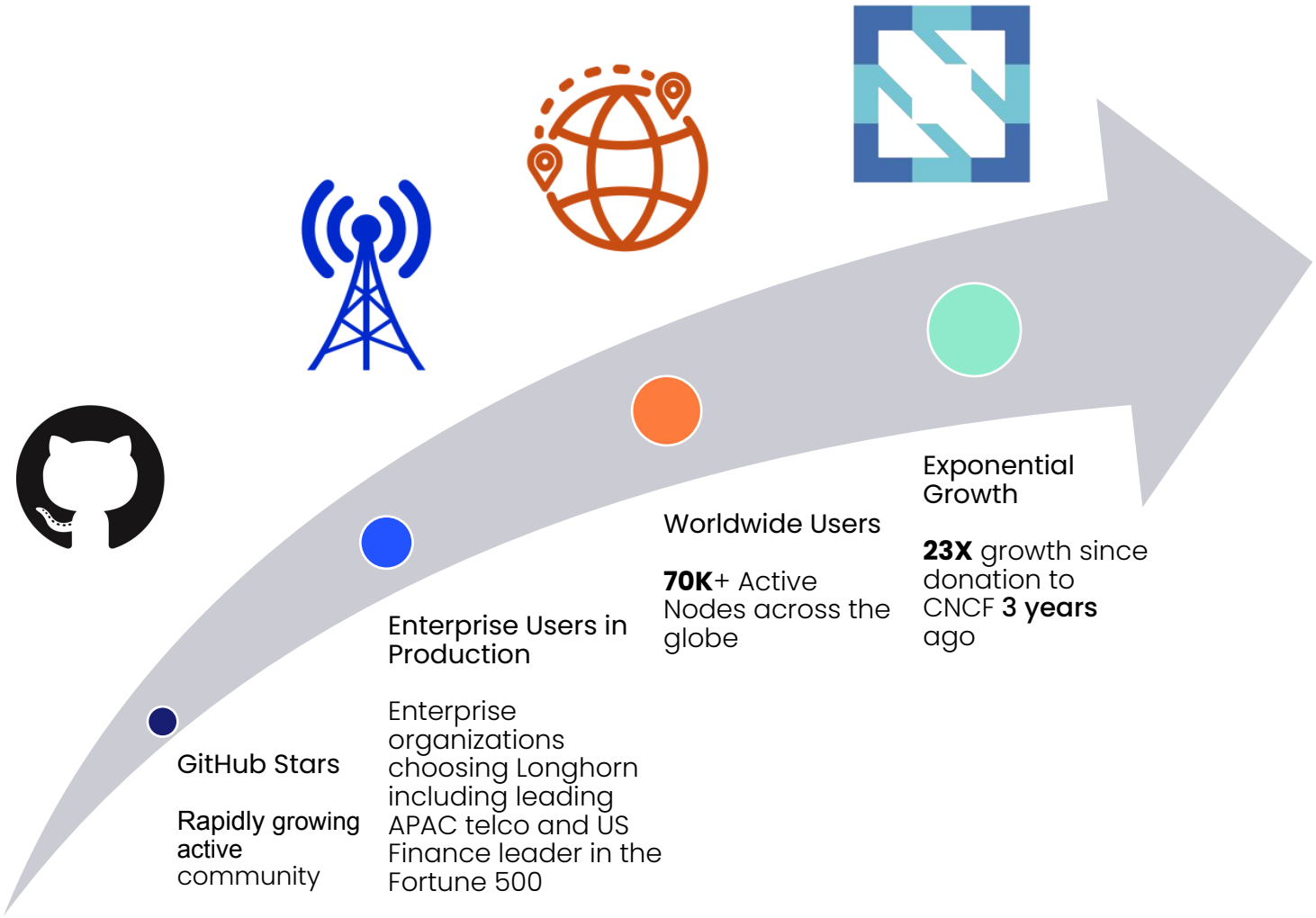
What is Longhorn

- Highly available, software-defined persistent block storage for Kubernetes
- Lightweight, reliable, and easy-to-use
- Deploy Longhorn from Rancher with just one-click
- Adds persistent volume support to any certified K8s cluster.
- Storage Agnostic – any ext4/xfs filesystem can be added to a Longhorn cluster
- NFS and S3 compatible (backup storage)
- Kubernetes-first design implemented in CRDs and controller pattern
- Open source and owned by the CNCF

Longhorn Feature List

- Enterprise-grade distributed block storage software for Kubernetes
- Market leading performance
- Volume thin-provisioning
- Volume snapshots
- Volume backup and restore
- Volume clone, expansion
- Cross-zone replica scheduling
- Auto replica rebalancing
- Encryption at-rest and in-transit
- Storage Tag for node and disk selection
- Cross-cluster disaster recovery volume with defined RTO and RPO
- RWX Support
- Live upgrade of Longhorn software without impacting running volumes
- Policy-based recurring jobs for snapshot/backup
- Intuitive UI

Longhorn Momentum



GitHub Stars

Rapidly growing active community

Enterprise Users in Production

Enterprise organizations choosing Longhorn including leading APAC telco and US Finance leader in the Fortune 500

Worldwide Users

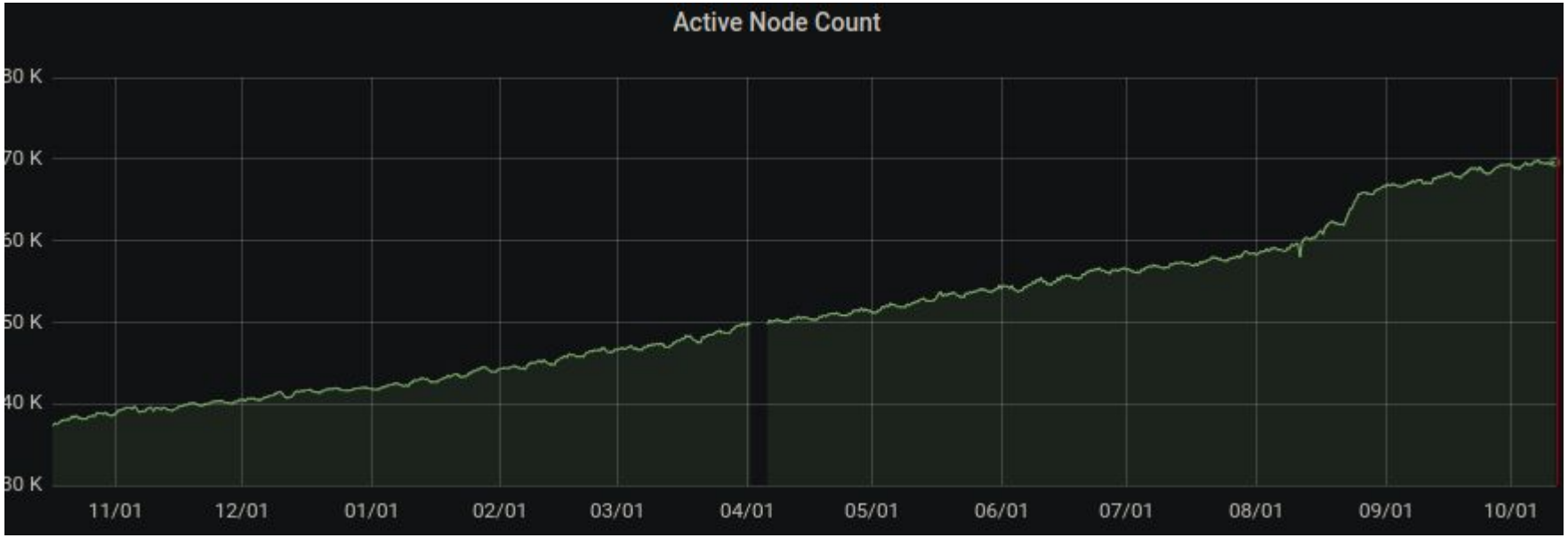
70K+ Active Nodes across the globe

Exponential Growth

23X growth since donation to CNCF **3 years** ago

Longhorn Community

- Active node count world-wide: 70k (+92% YoY)
- GitHub Star: 4.2k



Longhorn Story, road to 1.3 & 1.4*

KubeCon CloudNativeCon North America 2022

'14

'16

Open sourced

Re-written in Go,
built for Docker

'18

Longhorn 0.3

Support S3 for backup
Basic K8s support

'19

Longhorn 0.6

DR Volume
Helm chart

CNCF Sandbox

'20

Longhorn 1.0

Volume resizing
Automatic recover faulted volume
Air gap
First GA Release

'21

Longhorn 1.1

RWX support
ARM64 Support
Prometheus support

'21

Longhorn 1.2

Encryption at Rest and In Transit
Policy Based Recurring Job

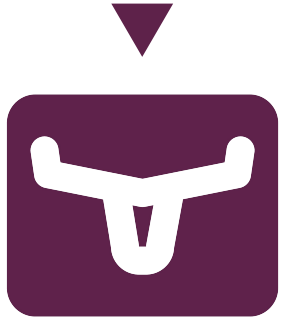
CNCF Incubating

Kubernetes Native API support (v1beta2) with Admission and Conversion Webhooks

Storage Network

Public Cloud Managed Kubernetes Cluster Support

Resource-Efficient Snapshot Purge



Longhorn 1.3
June 2022

CSI Snapshot for Longhorn Snapshot

Backing Image Download

Secured Communication among Control/Data Plane components

Upgrade Path Enhancement via Version Filtering

Longhorn Story, road to 1.4*

'14

'16

Open sourced

Re-written in Go,
built for Docker

'18

Longhorn 0.3

Support S3 for backup
Basic K8s support

'19

Longhorn 0.6

DR Volume
Helm chart

CNCF Sandbox

'20

Longhorn 1.0

Volume resizing
Automatic recover faulted volume
Air gap
First GA Release

'21

Longhorn 1.1

RWX support
ARM64 Support
Prometheus support

'21

Longhorn 1.2

Encryption at Rest and In Transit
Policy Based Recurring Job

CNCF Incubating

Reclaim/Shrink/Trim Volume Support

RWX Volume GA

Longhorn System Backup & Restore

Bit-Rot Protection

ARM64 GA

Longhorn 1.4
December 2022

Snapshot Checksum - Data Consistency

K8s 1.25 Support

Support Bundle Enhancement

Online Volume Expansion

Longhorn Releases

Active Maintained Branches

- 1.2 and 1.3

Upcoming Releases

- 1.3.2 - *10/14*
- 1.2.6 - *11/04*
- 1.4.0 - *End of December*

How Current Longhorn Works

Volume Elements

- Volume Frontend (iSCSI)
- Volume (Engine)
- Volume replica (Replica)

Volume Lifecycle

- CSI
- PVC/PV

Data Placement

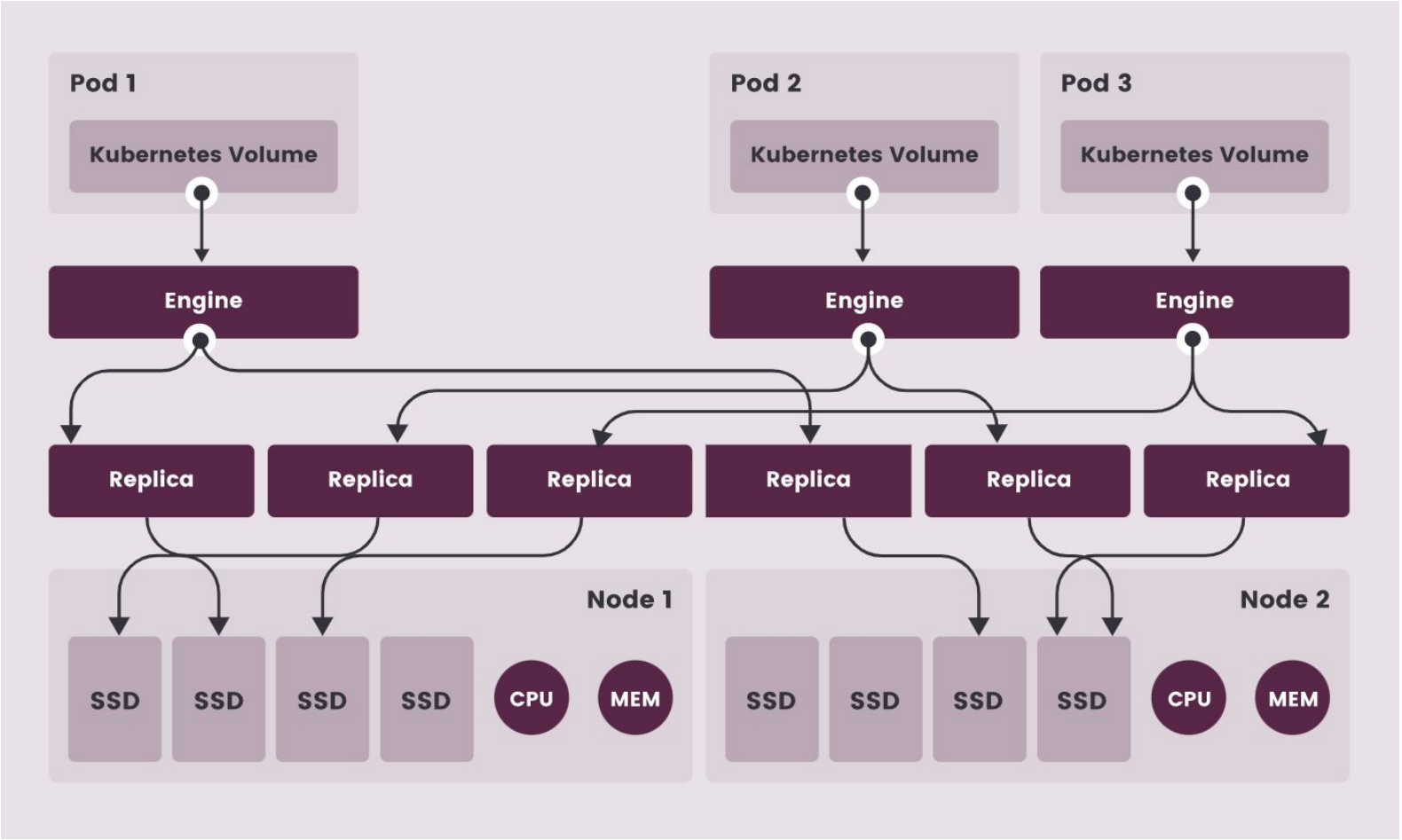
- Longhorn disk (FS on host)

Deployment

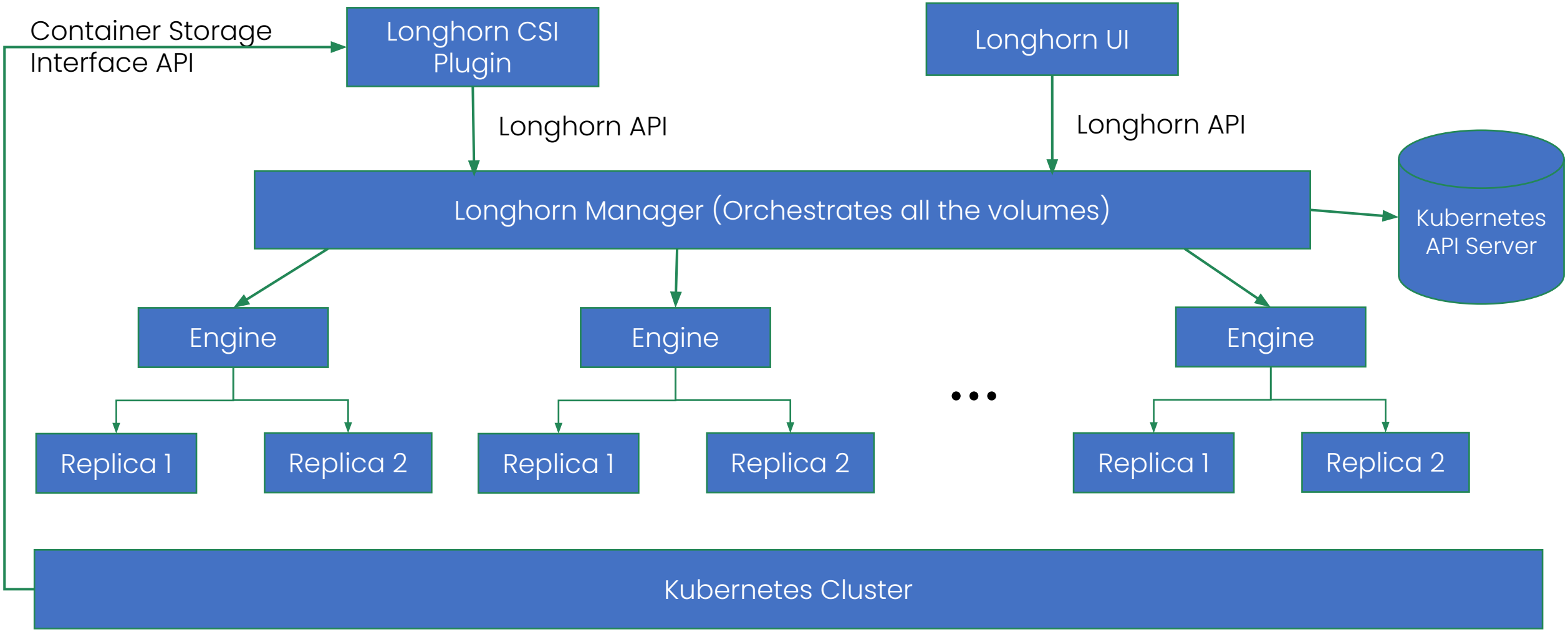
- Segregated Microservice

Control Plane

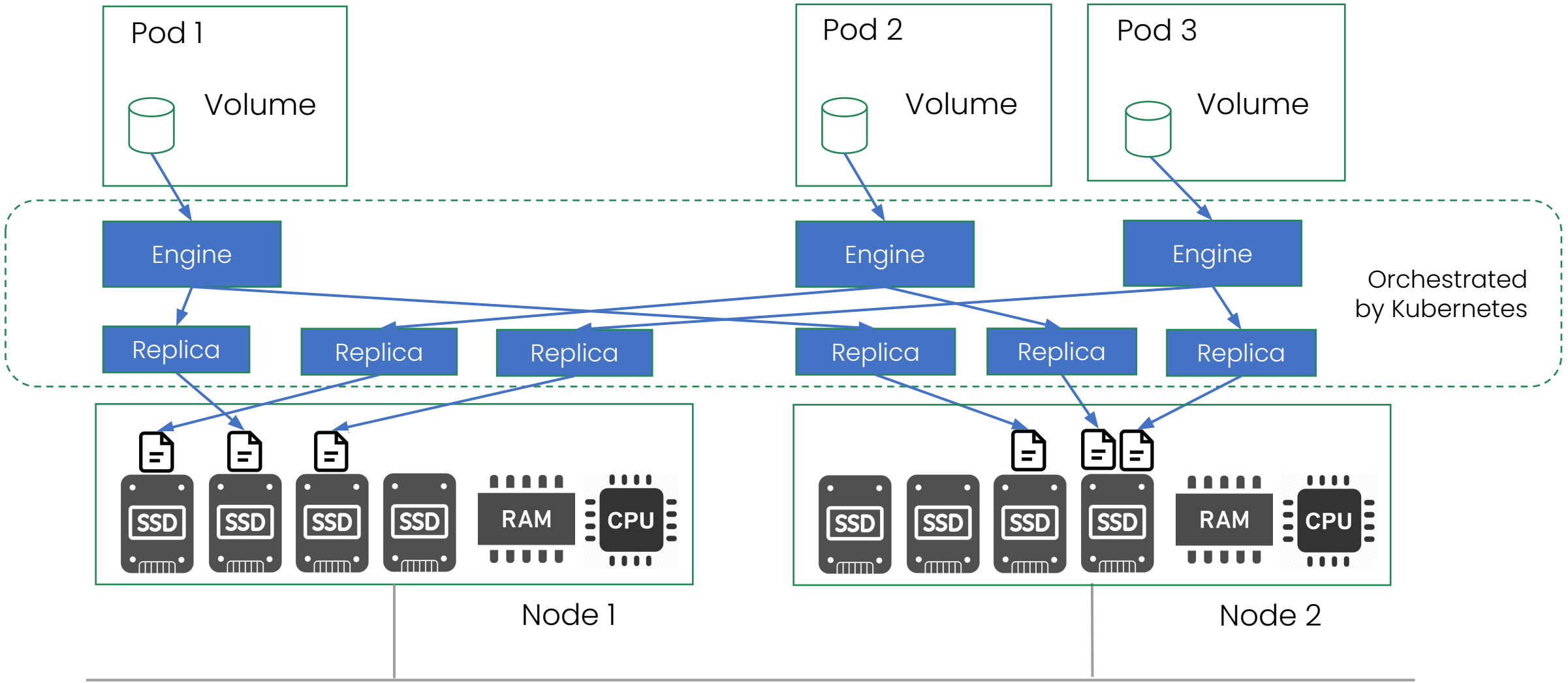
- Kubernetes Controller + CR



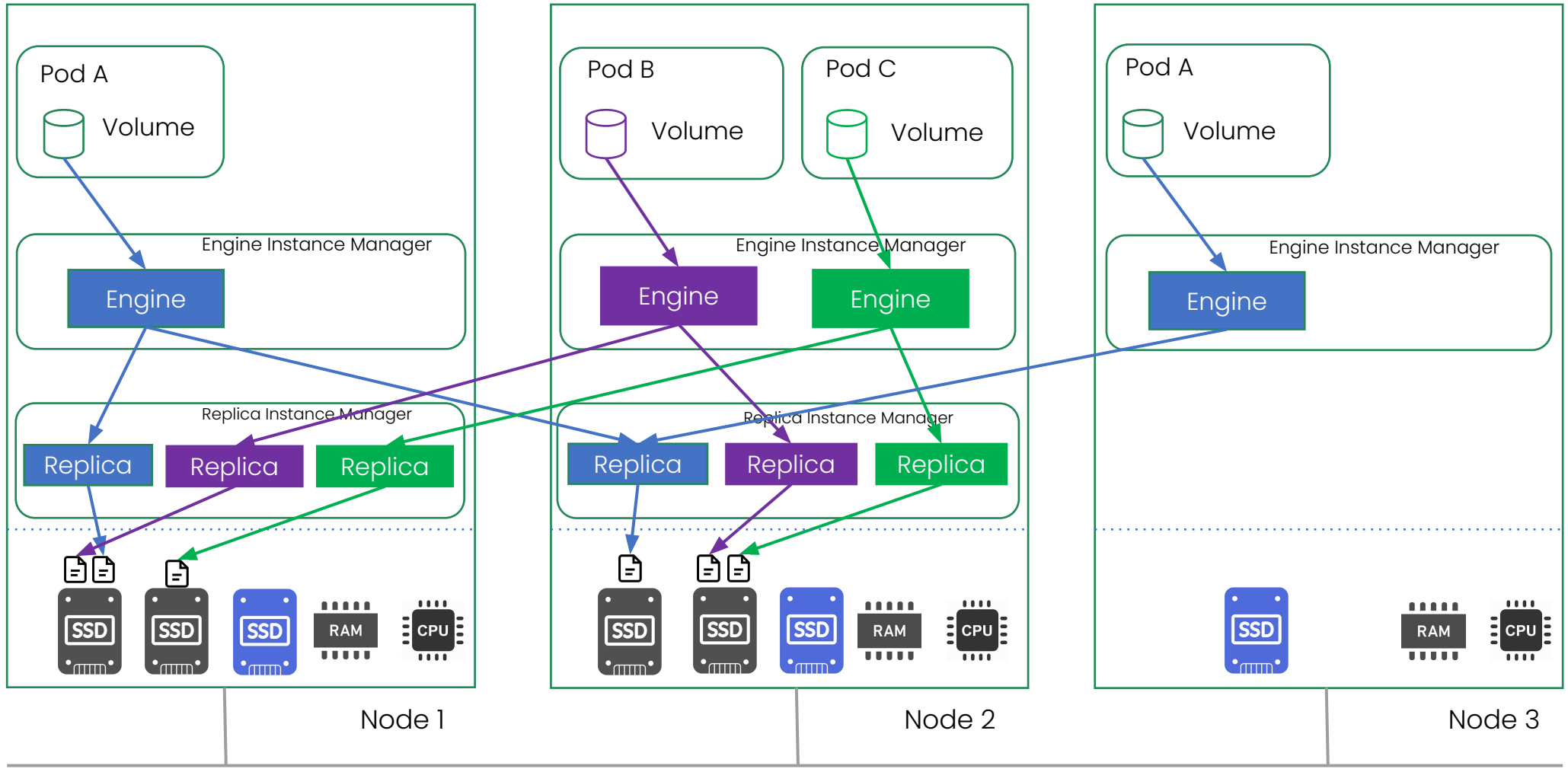
Longhorn Manager - Control Plane



Longhorn Engine & Replica - DP, Volume Segregation

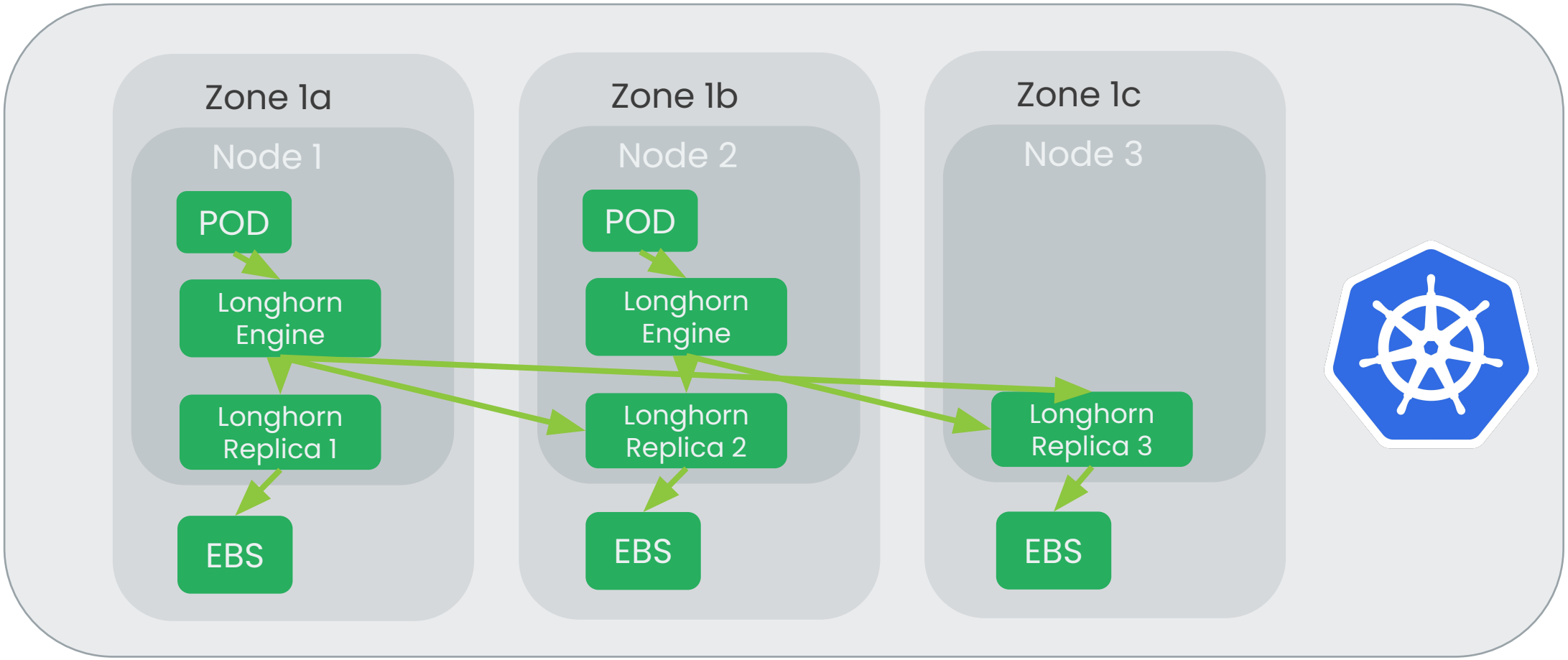


Longhorn Volume Failover



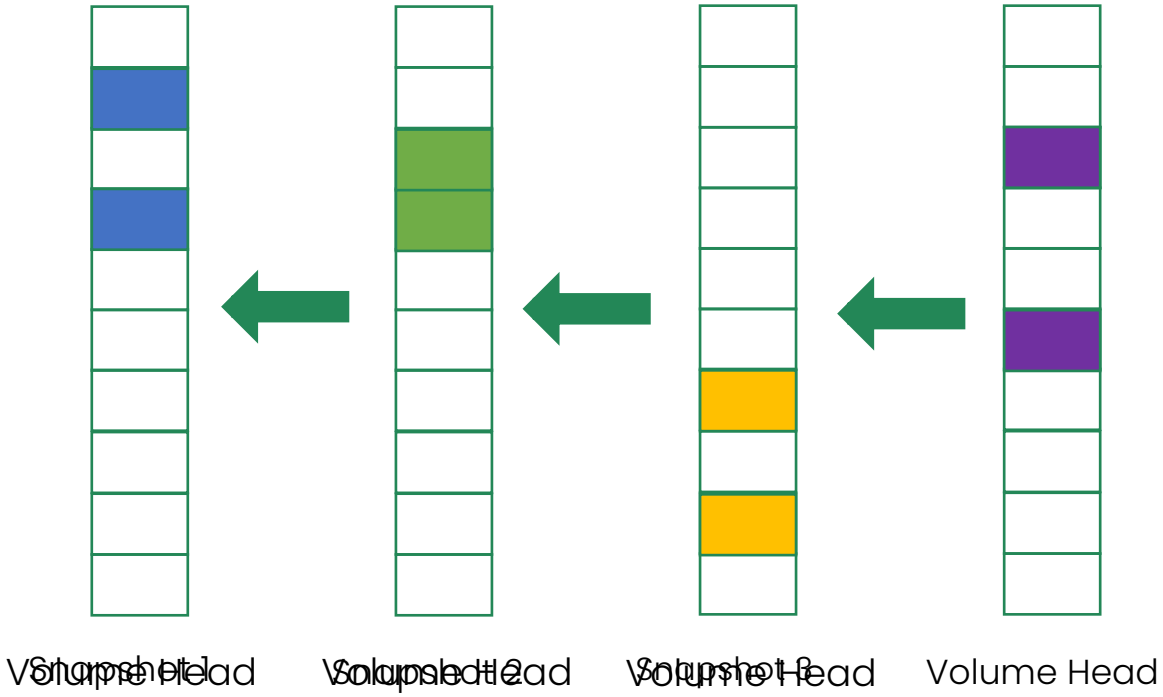
Longhorn Volume HA

Longhorn provides high availability block device across the availability zone

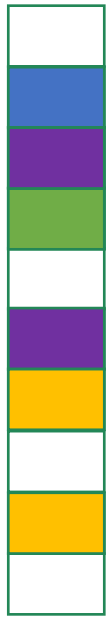


Longhorn Volume Snapshot

Snapshot Chain



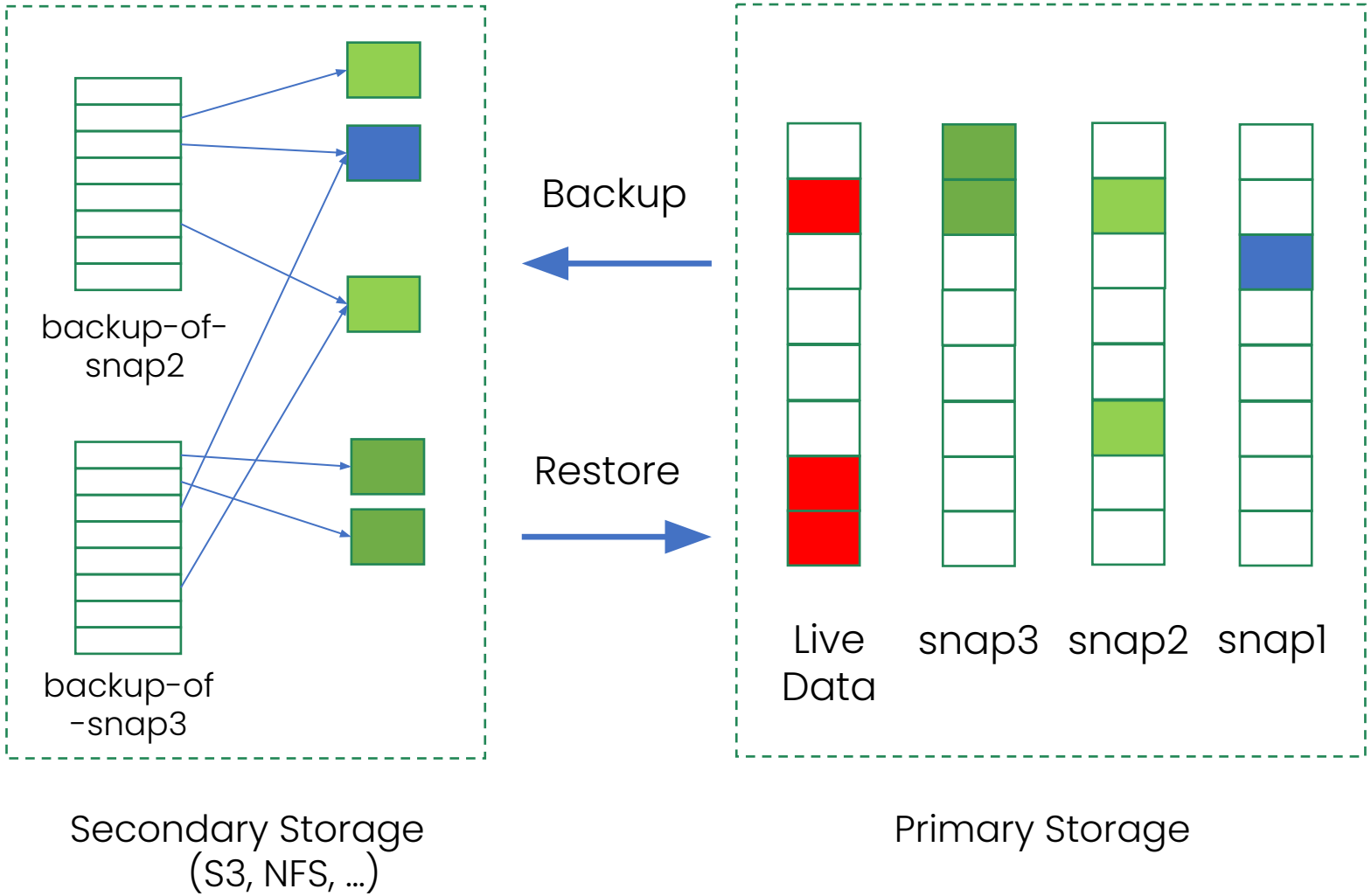
Volume Content



- Block Size: 4k
- Based on Linux Sparse File
- Read: lazily fill up a read index
- Write: always to the volume head, update read index as well

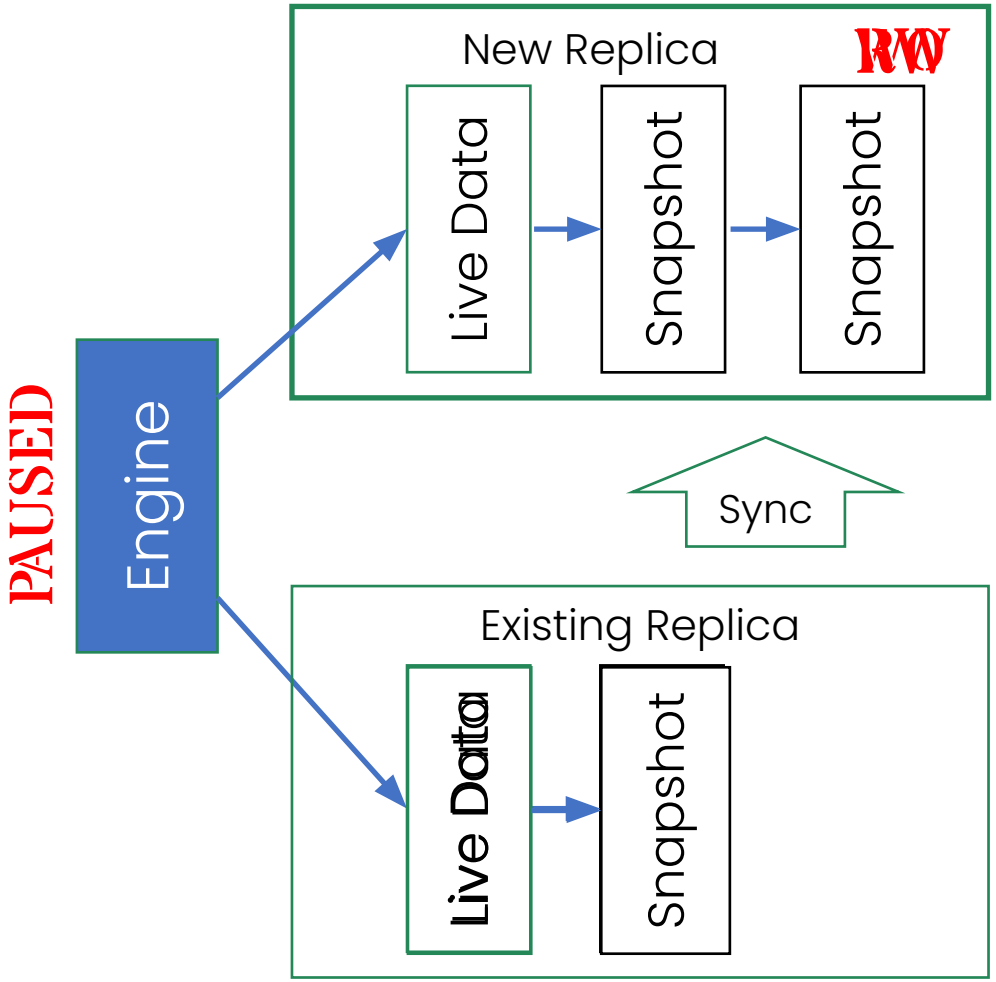
Longhorn Volume Backup

- AWS EBS-style backup
- Only changed blocks are copied
- 2M block size

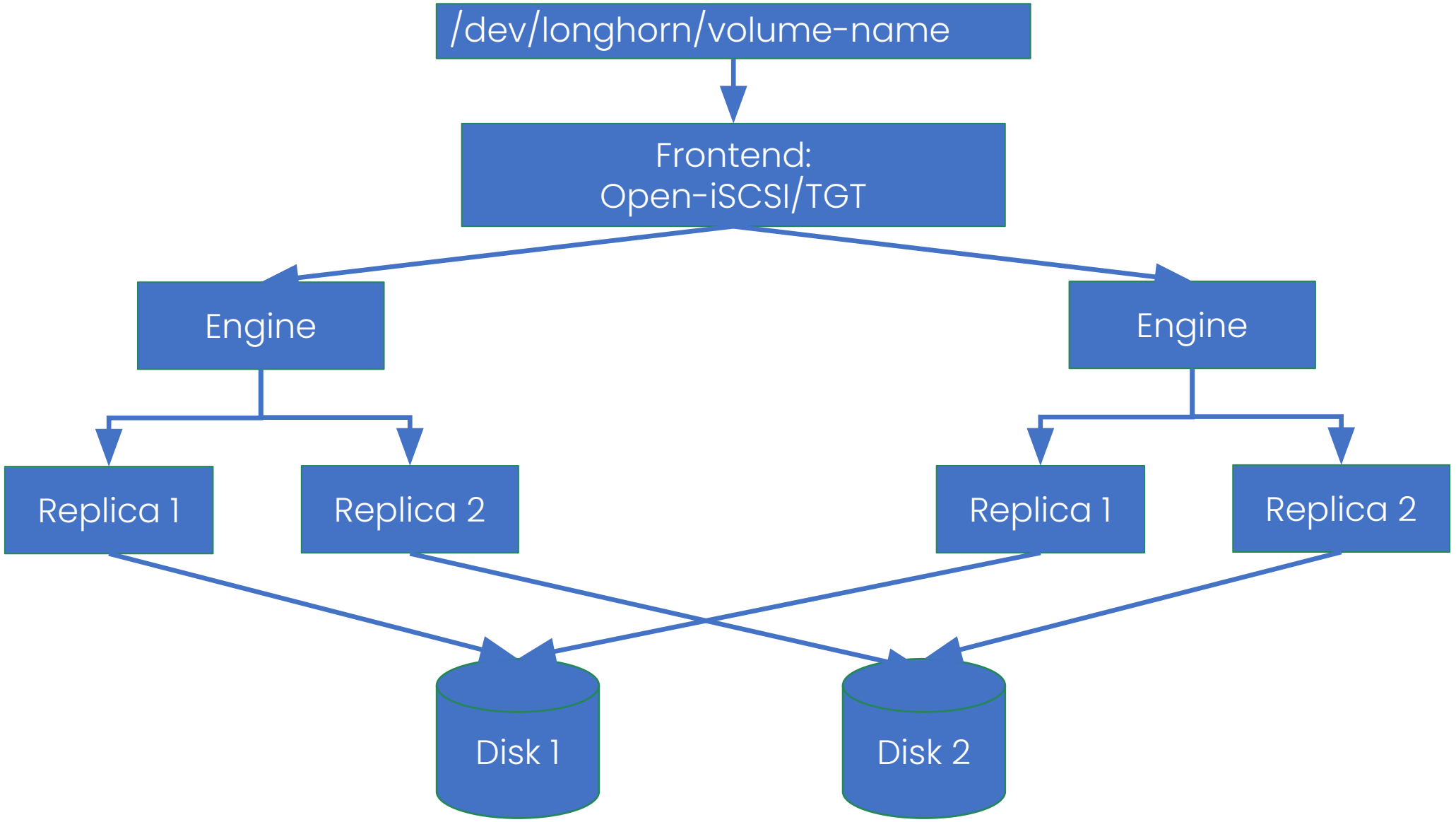


Longhorn Volume Replica Rebuilding

1. Pause engine
2. Take snapshot of existing replica
3. Add new replica in WO mode
4. Unpause engine
5. Sync snapshots
6. Set new replica to RW

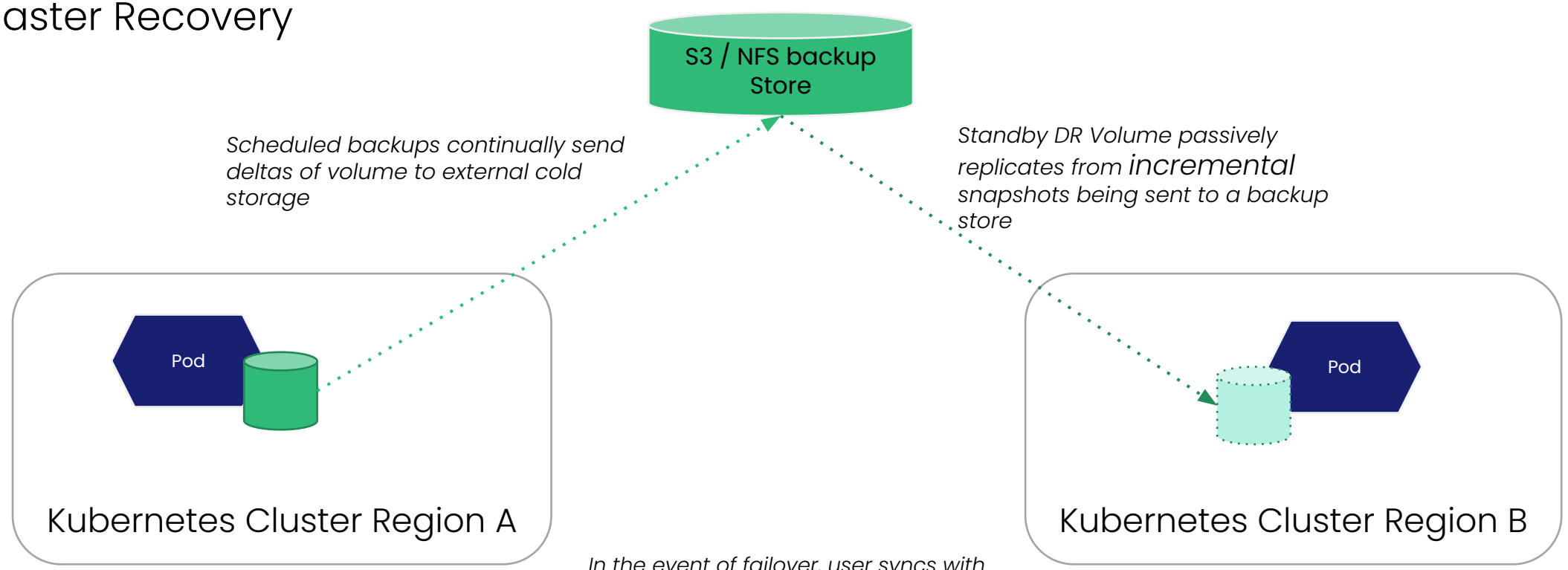


Longhorn Volume Live Migration



Longhorn Disaster Recovery

Multi-Cluster, Multi-site Disaster Recovery



In the event of failover, user syncs with the latest backup and becomes active for workloads in the cluster. Time to recovery is short because most of the delta is already replicated

What is Next? Longhorn SPDK

Goals

- Space Efficiency
- Resource Optimization
- IO Performance

Please check “Improving Longhorn Performance With SPDK” session to know more about the details

How

- Longhorn SPDK, new data engine
  - Adopt SPDK to implementate new data engine via (virtual) bdev, logical volume and NVMe-OF target
  - Leverage logical volume thin-provisioning and snapshot for feature parity
  - Drivers on userspace, zero-copy w/o context switch
  - Polling instead of interrupts to lower IO latency
  - Avoid locks in the I/O path, instead relying on message passing (event driven)

Thank You 🙏

Q & A 🙋

KubeCon | CloudNativeCon
North America 2022

BUILDING FOR THE ROAD AHEAD

DETROIT 2022

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