



**SDC** 

STORAGE DEVELOPER CONFERENCE

SNIA  SANTA CLARA, 2017

# **Volumes as a Micro-Service**

## **Distributed Block Storage Enabled by Docker**

**Sheng Yang**  
**Rancher Labs**

# About me - Sheng Yang

- ❑ Principle Engineer at Rancher Labs
- ❑ Joined Rancher Labs since 2015
- ❑ Worked at Citrix since 2011, focus on CloudStack
- ❑ Before that, worked at Intel, focus on KVM and Linux kernel
- ❑ Email: [sheng.yang@rancher.com](mailto:sheng.yang@rancher.com)
- ❑ Twitter: @yasker



# Agenda

- ❑ What's the deal about Docker
- ❑ Traditional scalable storage cluster
- ❑ Longhorn overview
- ❑ Under the hood
- ❑ Questions

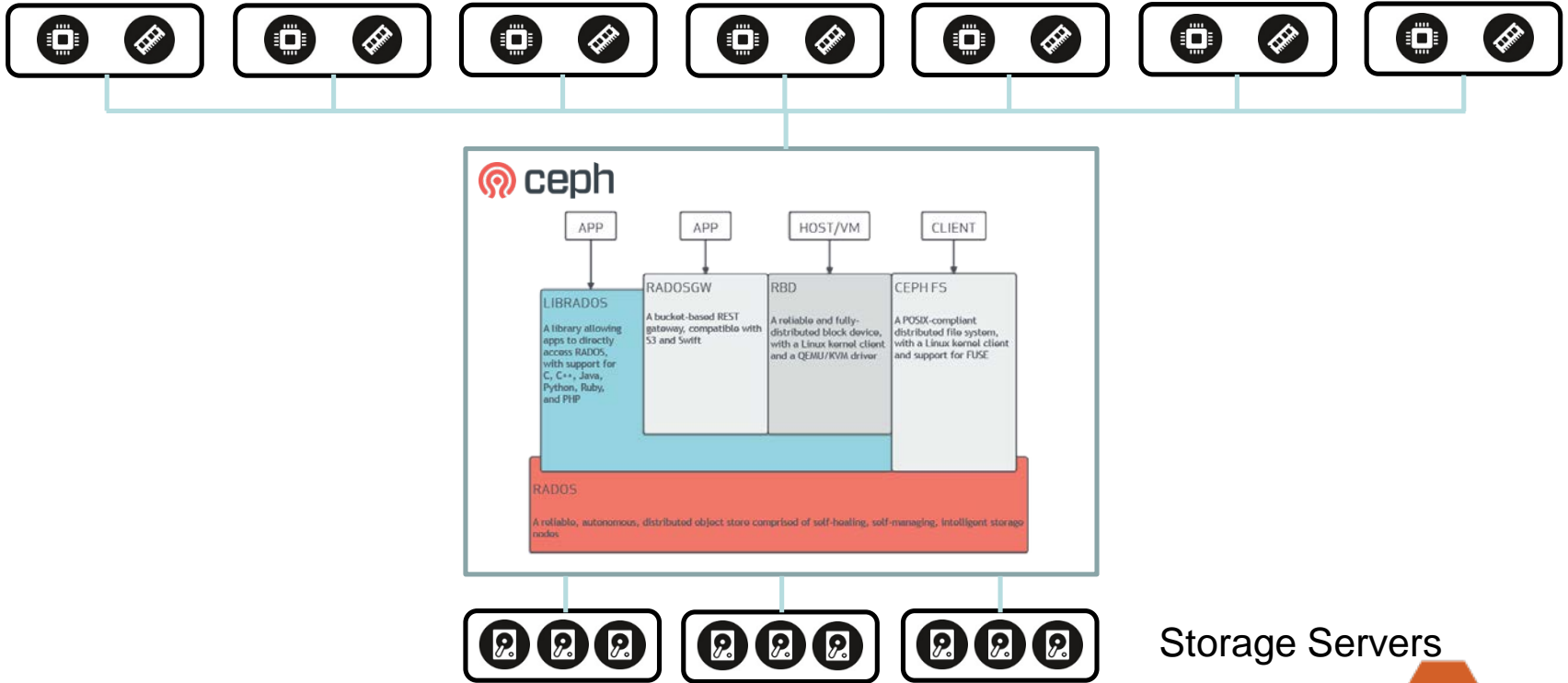


# What's the deal about Docker

- ❑ Docker is the best way to deliver a piece of software
  - ❑ Great portability
  - ❑ Minimal overhead
  - ❑ Small footprint
- ❑ It's easy to deploy micro-services with Docker



# Traditional scalable storage cluster



**One controller for 100 volumes**

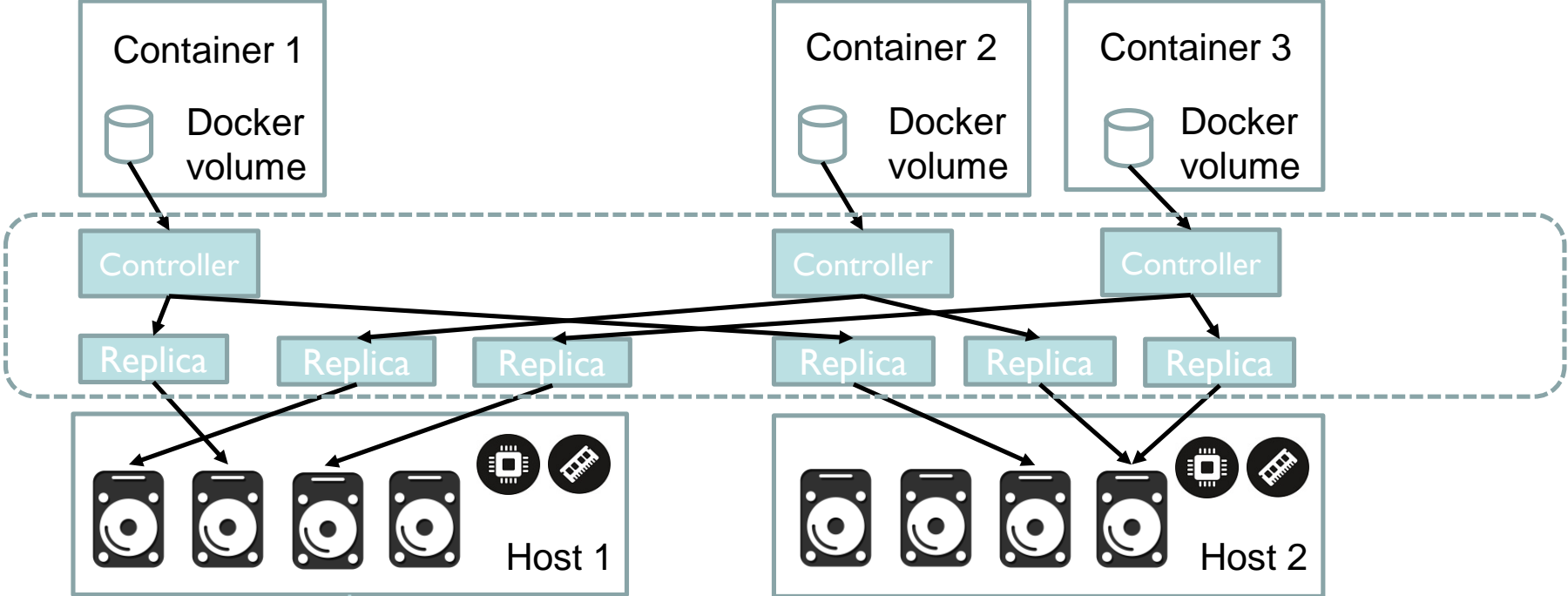


**A separate controller for each of the 100 volumes**

**Use Docker to manage these controllers**



# Longhorn Overview



# Focus on simplicity, reliability, and performance

## Controller

- Mirroring
- Rebuild
- Encryption

## Replica

- Snapshot
- Backup
- QoS

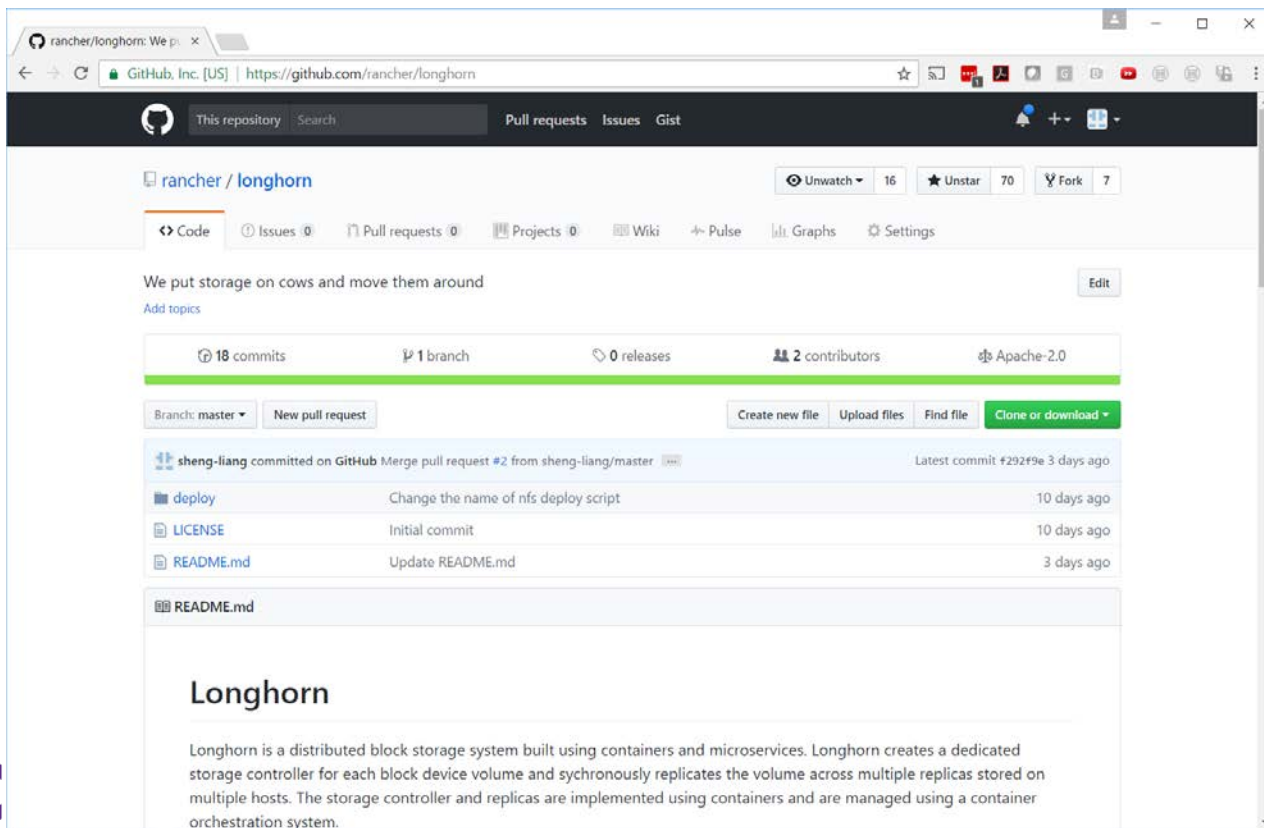
## Does Not Support

- Controller HA
- Tiering
- Striping
- Dedup
- Compression





# Project Longhorn: distributed block storage system built using containers and microservices



The screenshot shows the GitHub repository page for `rancher/longhorn`. The repository is described as "We put storage on cows and move them around". It has 18 commits, 1 branch, 0 releases, 2 contributors, and is licensed under Apache-2.0. The latest commit by sheng-liang is a merge pull request #2 from sheng-liang/master, committed 3 days ago. The commit history includes a deployment script change, the initial commit, and a README update. The README content is partially visible, starting with the title "Longhorn" and a description of the system.

rancher / longhorn

Unwatch 16 Unstar 70 Fork 7

Code Issues 0 Pull requests 0 Projects 0 Wiki Pulse Graphs Settings

We put storage on cows and move them around

Add topics

18 commits 1 branch 0 releases 2 contributors Apache-2.0

Branch: master New pull request Create new file Upload files Find file Clone or download

sheng-liang committed on GitHub Merge pull request #2 from sheng-liang/master Latest commit #292f9e 3 days ago

deploy	Change the name of nfs deploy script	10 days ago
LICENSE	Initial commit	10 days ago
README.md	Update README.md	3 days ago

README.md

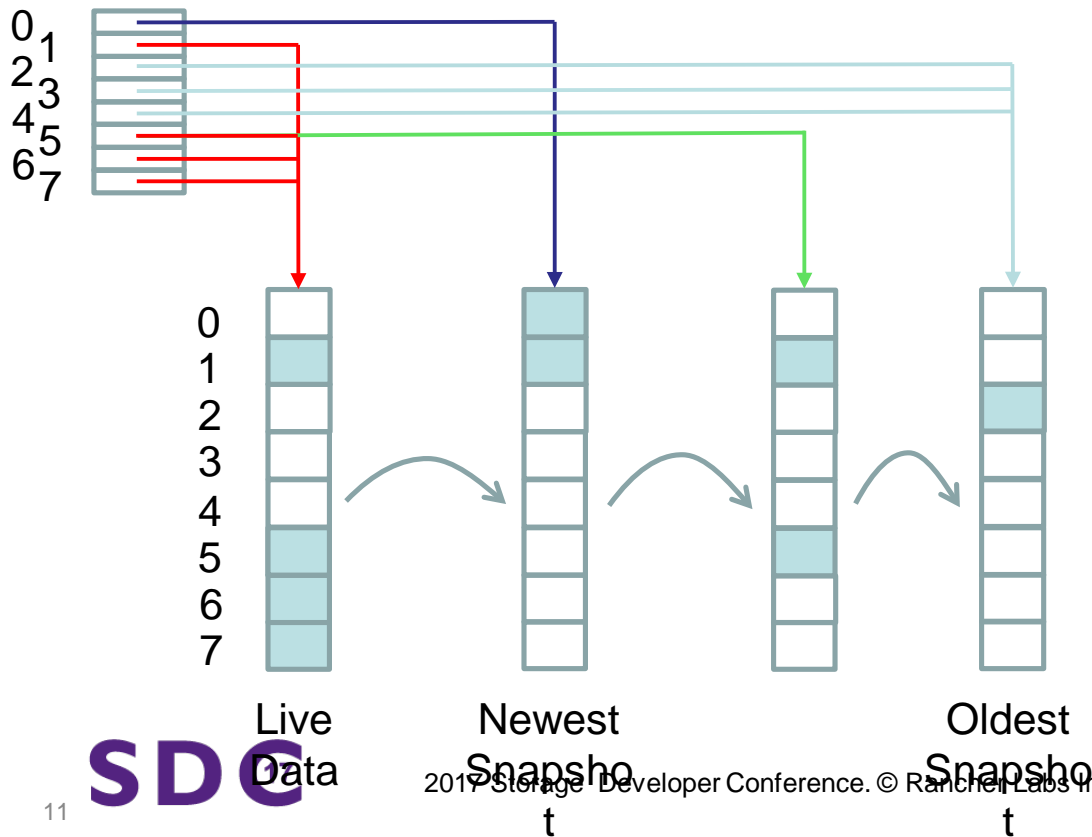
## Longhorn

Longhorn is a distributed block storage system built using containers and microservices. Longhorn creates a dedicated storage controller for each block device volume and synchronously replicates the volume across multiple replicas stored on multiple hosts. The storage controller and replicas are implemented using containers and are managed using a container orchestration system.

# Longhorn Under the Hood



# Read index



Use Linux sparse files to store differencing disks

4K block size

Read: lazily fill up a read index

Write: always to live data, update read index if needed

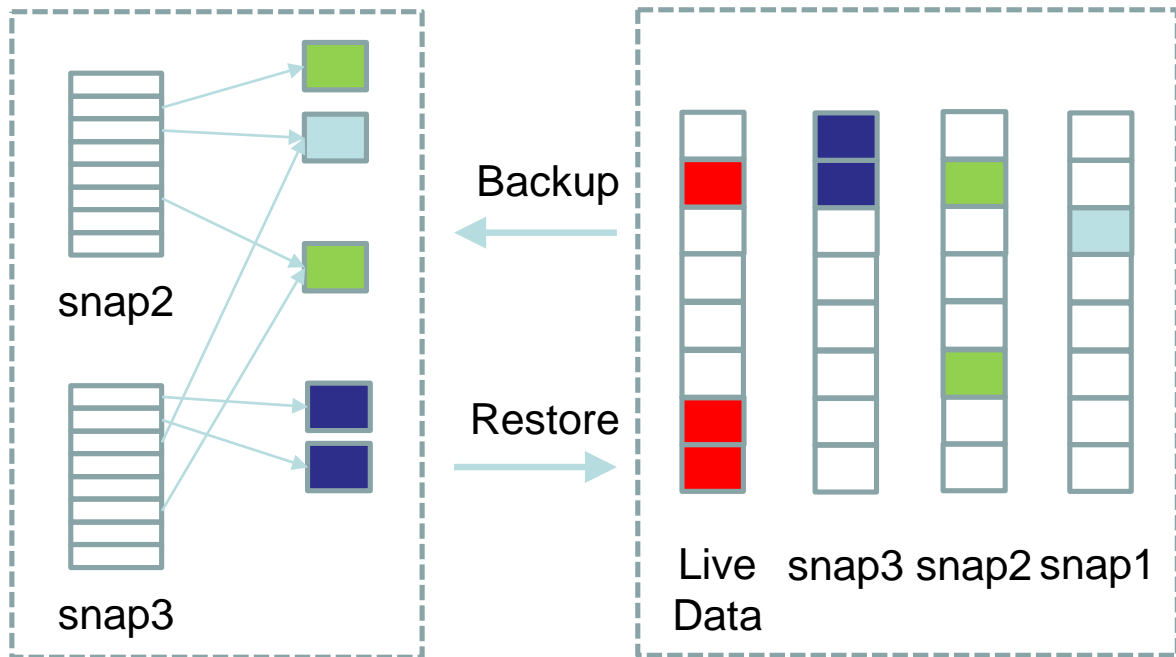


# Backup

AWS EBS-style backup

Only changed blocks are copied

2M block size



Secondary Storage

Primary Storage

# How backups are stored

volume.cfg

backups/

    snap2.cfg

    snap3.cfg

blocks/

c0facb6ba3102d29e8d847f32982a030028369020fd5ab6dfc99e63f8a1af903.blk

f1af6a6aa6410a1eea5a1ba2a8856cc7bb01b302483e819f3ff4ca46bb17bb16.blk

21935af9e15f5c32c843fbfb6fa01369cc7c0aa0c589f7d1e930bf351f8650c7.blk

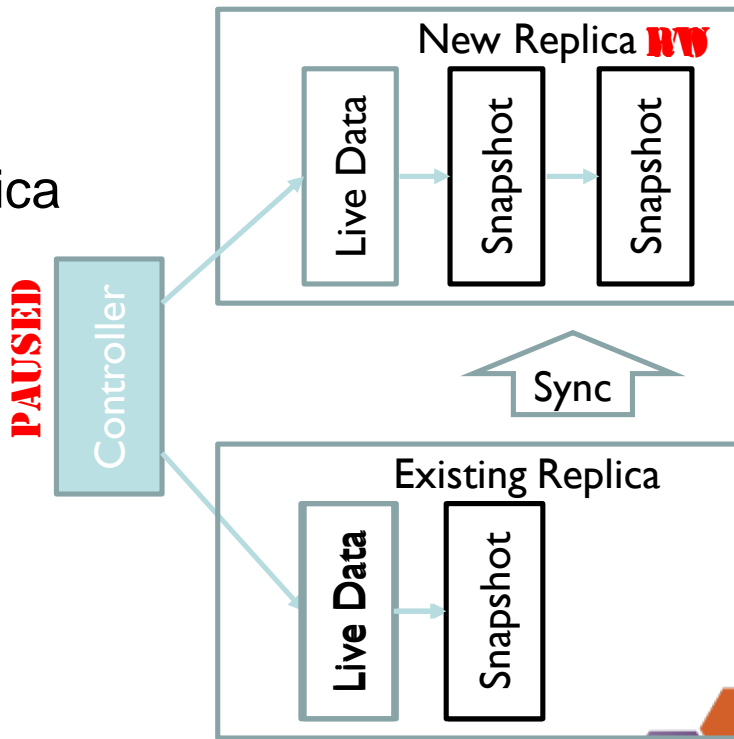
731859029215873fdac1c9f2f8bd25a334abf0f3a9e1b057cf2cacc2826d86b0.blk

965b2b6871ebb1b57d1bad2c087aeebc3f7052487b38fac939d655a493b49d06.blk

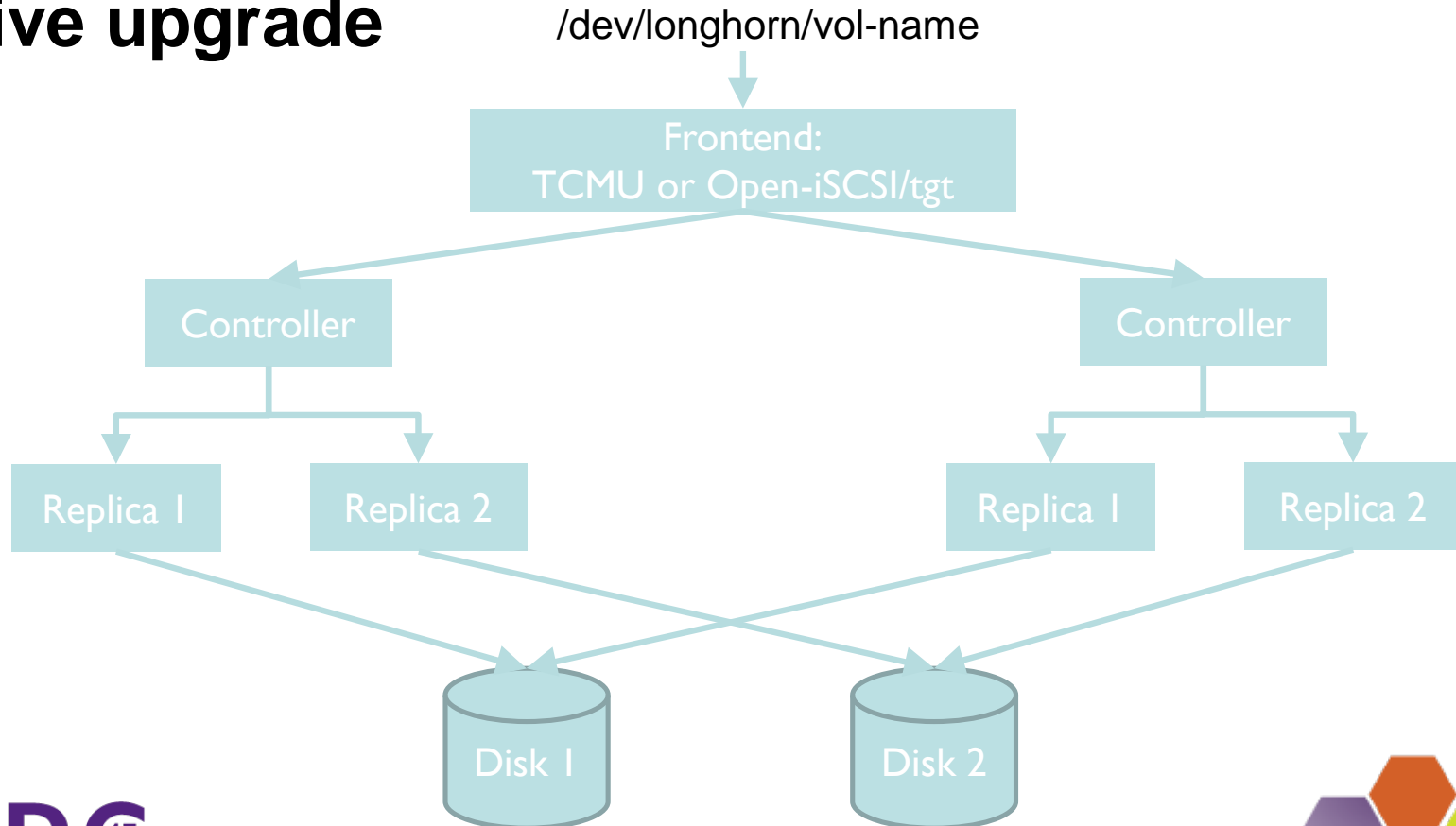


# Add a new replica (replica rebuild)

- Pause controller
- Take snapshot of existing replica
- Add new replica in WO mode
- Unpause controller
- Sync snapshots
- Set new replica to RW

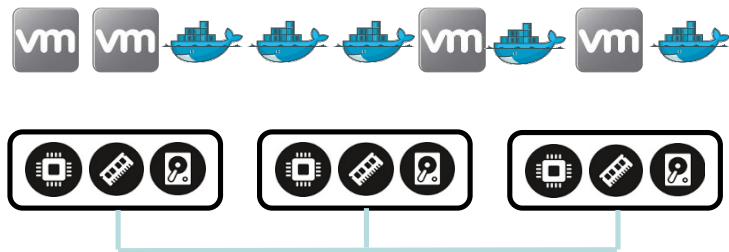


# Live upgrade



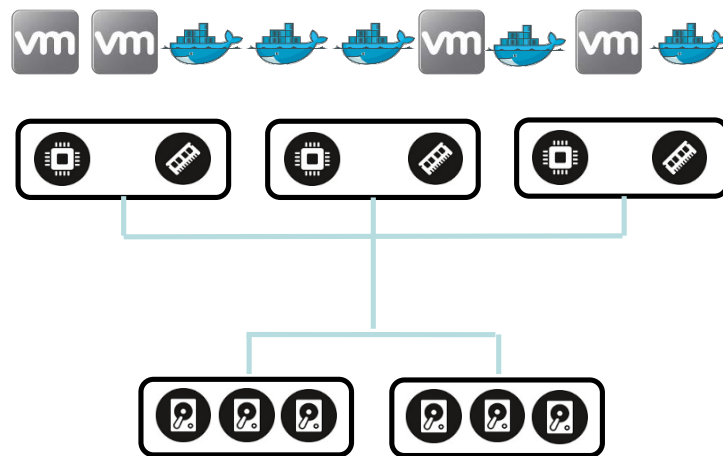
# Two deployment models

Schedule replicas on the same set of hosts as controllers



Hyper-Converged

Schedule replicas on dedicated storage servers



Dedicated Storage Servers





# What works now

1. Distributed volumes on a Docker Swarm cluster
2. Fault detection and replica rebuild
3. Snapshots, backups, and recurring snapshots and backups
4. UI and API



# Upcoming work

1. Kubernetes FlexVolume driver
2. Deploy Longhorn clusters from Rancher catalog
3. Controller and replica live upgrade
4. Event log for Longhorn orchestration activities (e.g., replica rebuild)
5. Ability to backup to S3
6. Replica scheduling based on disk capacity and IOPS
7. Multiple disks on the same host
8. Volume stats, including throughput and IOPS
9. Authentication and user management of the Longhorn UI and API
10. Volume encryption
11. Performance tuning



# Questions?



# Thank you!

