RED HAT® STORAGE

Andreas Koll
Solution Architect Storage
akoll@redhat.com

# *AGENDA*

- Motivation für Open & Software Defined Storage
- Red Hat Storage Overview
  - Red Hat Gluster Storage
  - Red Hat Ceph Storage
- Use Cases

# *AGENDA*

- Motivation für Open & Software Defined Storage

## THE RED HAT STORAGE MISSION

To offer a unified, open software-defined storage portfolio that delivers a range of data services for next generation workloads thereby accelerating the transition to modern IT infrastructures.

# New wave of storage disruption and innovation

- Hybrid-flash arrays
- Public cloud
- Software-defined storage
- Open source

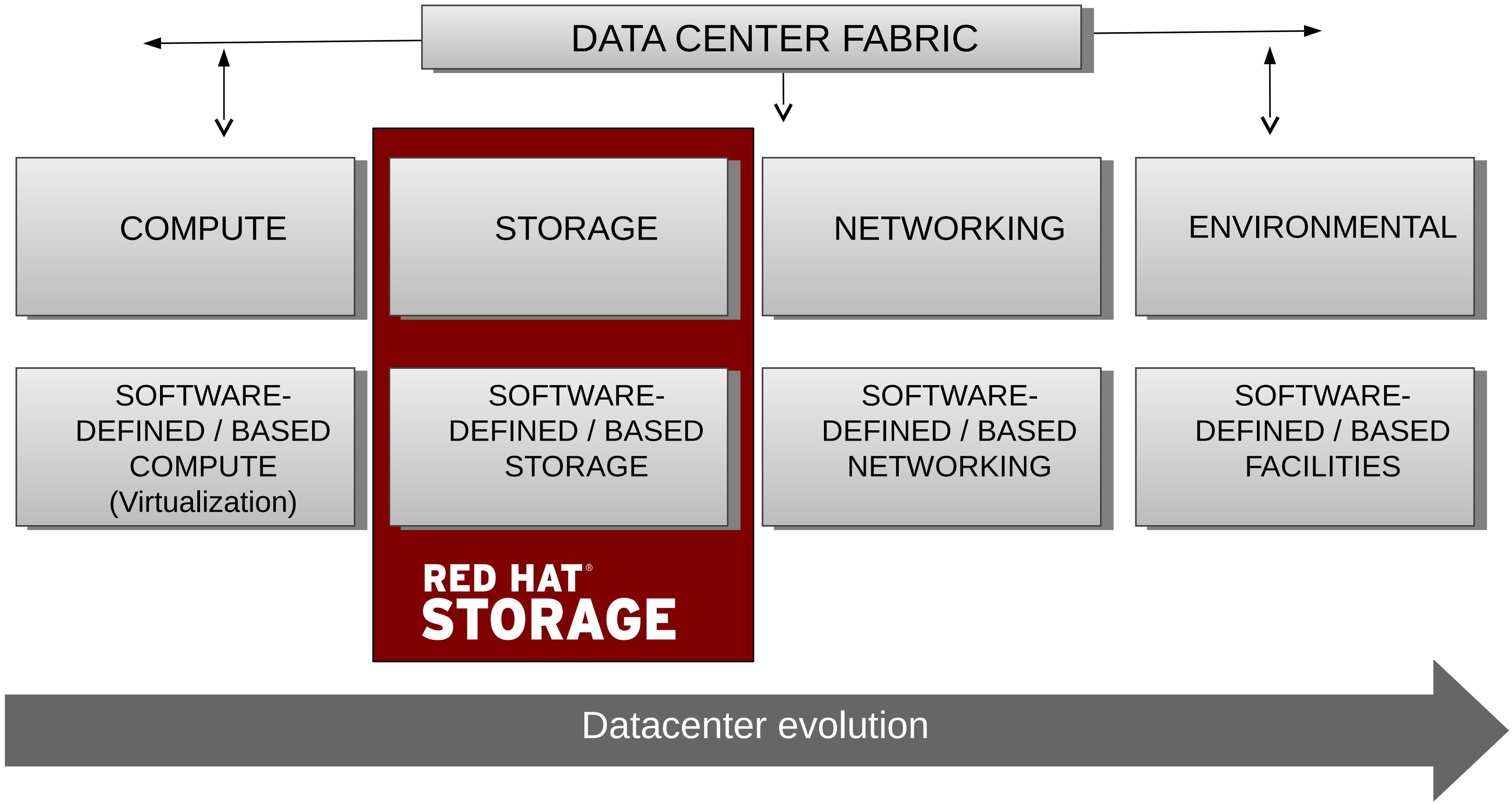
Source: Gartner Top 10 Tech Trends 2014

# Top tech trends for 2014 IT needs to have a plan in place by 2016

1. Software-defined networking
2. Software-defined storage
3. Hybrid cloud services
4. Integrated systems
5. Application acceleration
6. The internet of things
7. Open compute project
8. Intelligent datacenter
9. IT Demand
10. Organizational entrenchment and disruption

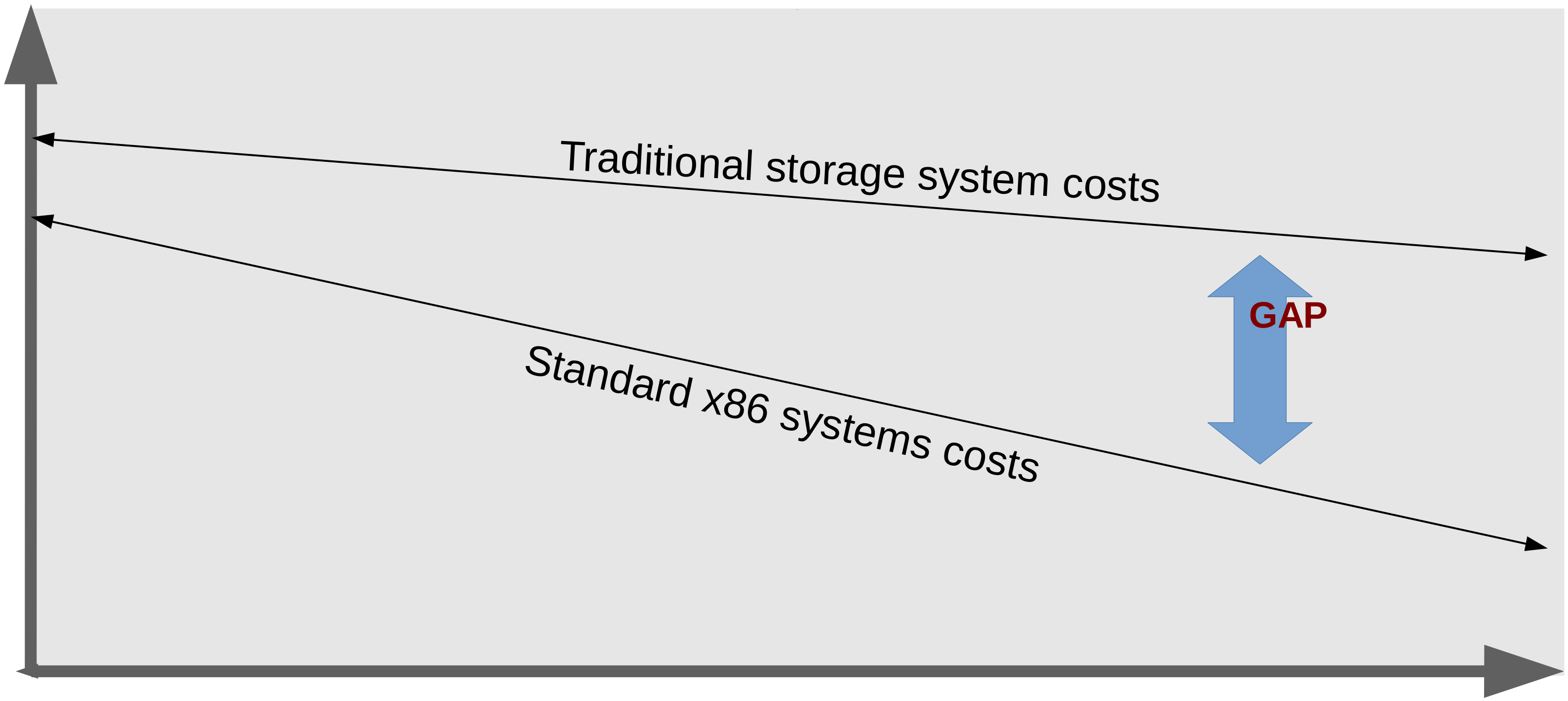
Source: Gartner Top 10 Tech Trends 2014

# Cornerstone of the software-defined datacenter

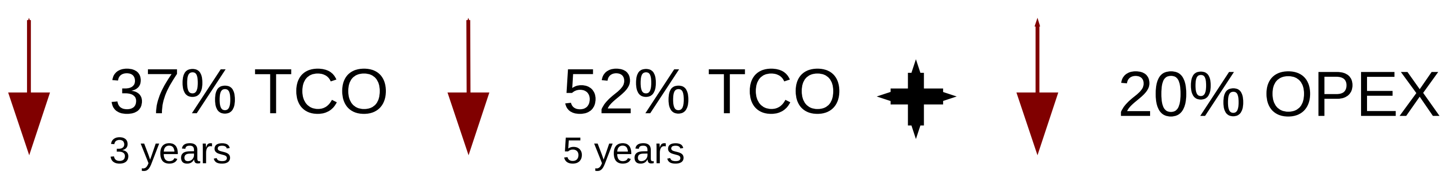



Source: **IDC Report** – Taxonomy for Software-Defined/Based Storage

# Up to 1/3 the cost of traditional Storage




In addition, you save up to:




Source: **IDC** research report: “The economics of software-based storage”

# *AGENDA*

- Motivation für Open & Software Defined Storage
- Red Hat Storage Overview
  - Red Hat Gluster Storage
  - Red Hat Ceph Storage
- Use Cases
- TCO & Customer example
- Q&A

# 2 Produkte.....

Red Hat **Gluster** Storage

Red Hat **Ceph** Storage

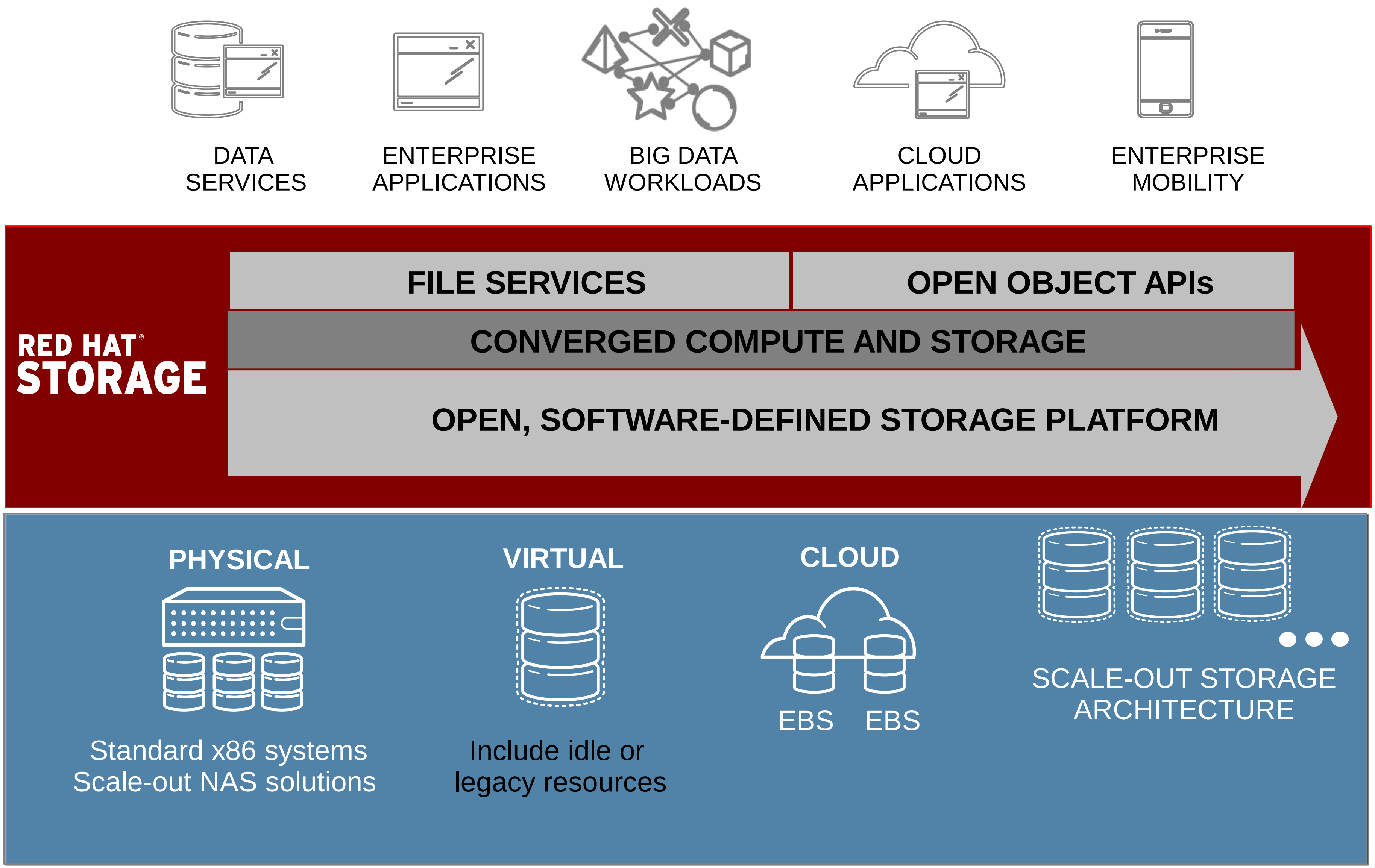
# Red Hat Gluster Storage

| | |
|---|---|
| **OPEN**<br>Open, software-defined distributed file and object storage system | • based on GlusterFS open source community project<br>• uses proven local file system ( XFS : fast, performand, scalable )<br>• data is stored in native format |
| **SCALABLE**<br>No metadata server | • uses an elastic hashing algorithm for data placement<br>• uses local file system’s extended attributes to store metadata<br>• shared nothing scale-out architecture |
| **ACCESSIBLE**<br>Multi-protocol to the same data | • NFS, SMB, object, HDFS, Gluster native protocol<br>• Posix compliant |
| **MODULAR**<br>No kernel dependencies | • GlusterFS is based on file-system in user space (FUSE)<br>• modular stackable architecture allows easy addition of features without being tied to any kernel version |
| **ALWAYS-ON**<br>High availability across data systems and applications | • synchronous replication with self-healing for server failure<br>• asynchronous geo-replication for site failure |

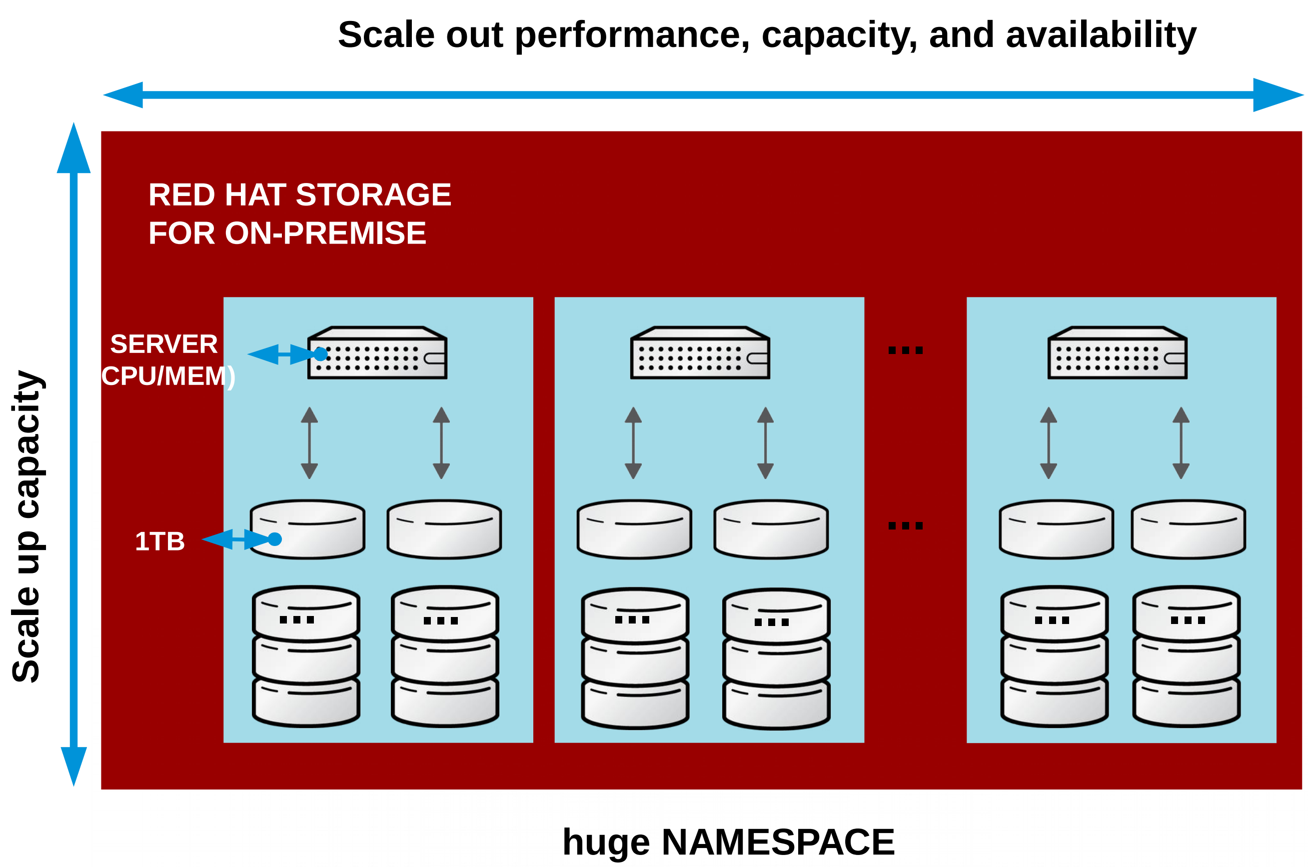
# Red Hat Storage Technology Highlights

- huge namespace
- software only, Open Source Solution
- runs in User Space
- simple and extremely easy to set up
- unified file- and object storage
- modular stackable architecture (scale-out)
- basefilesystem XFS in native format
- no MetaData server due DHT (elastic hash algorithm)

# INCREASE DATA, APPLICATION AND INFRASTRUCTURE AGILITY

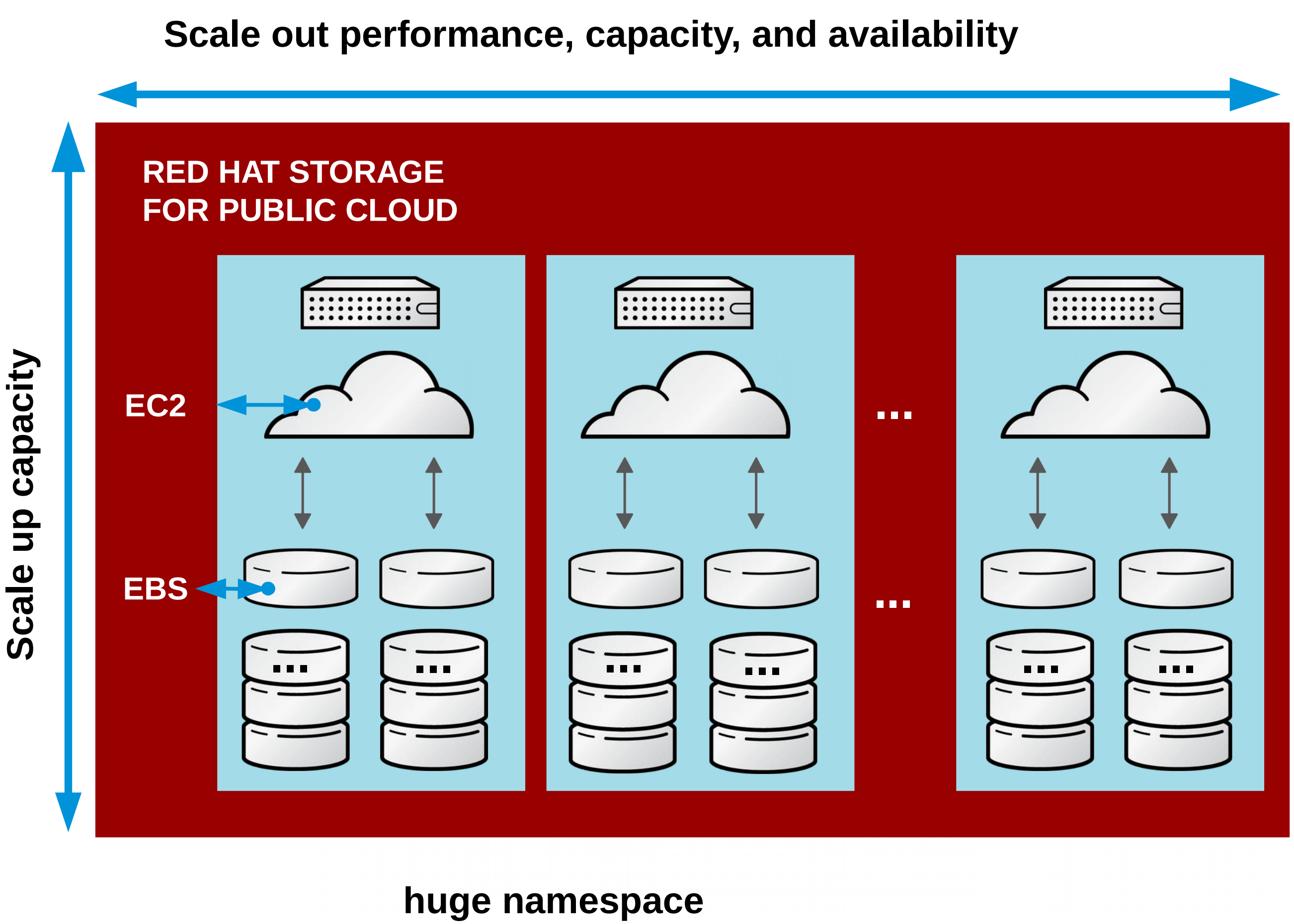


# RED HAT STORAGE DEPLOYMENT ON-PREMISE



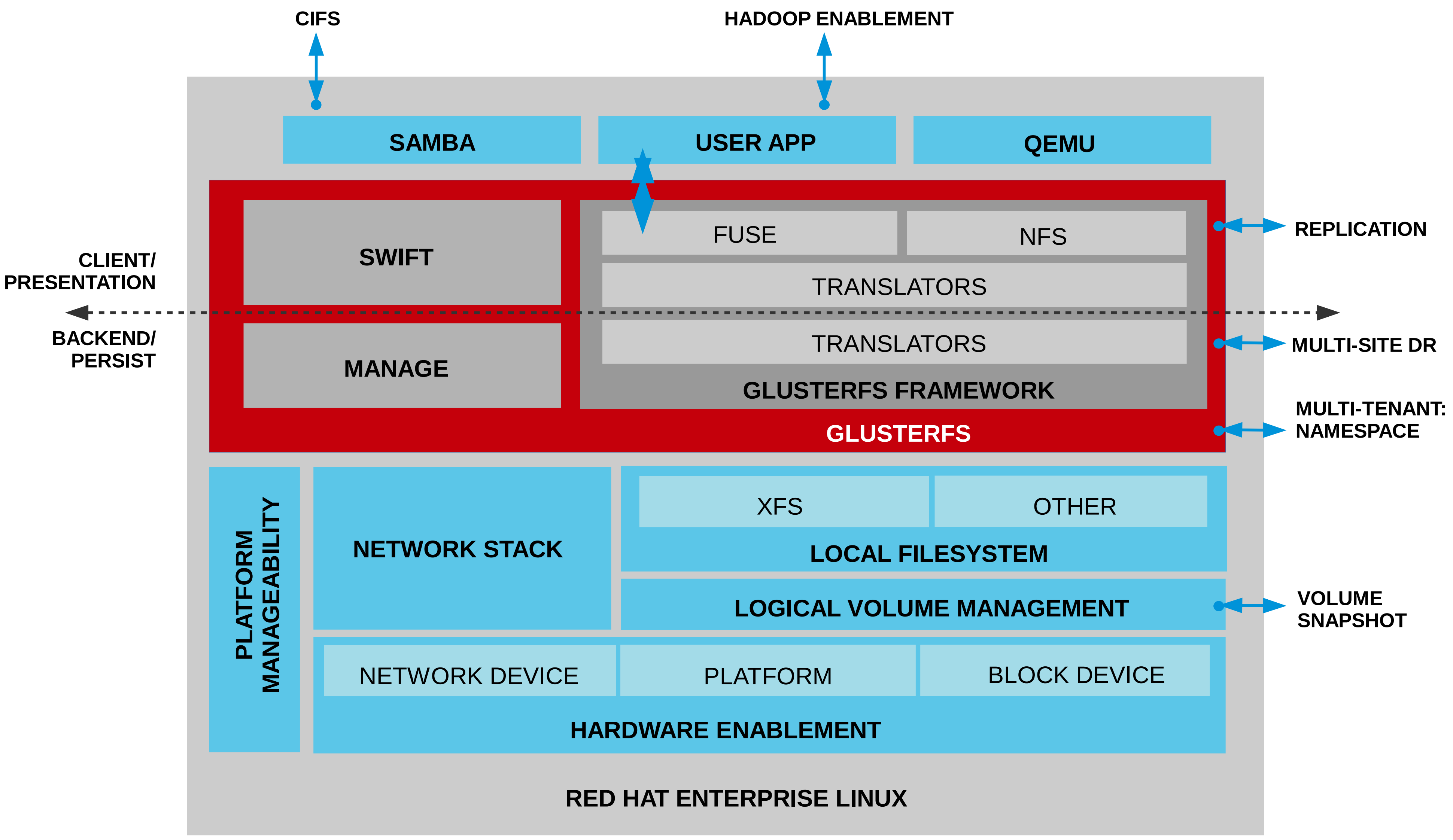

- global namespace
- aggregates CPU, memory, network capacity.
- deployed on Red Hat-supported servers and underlying storage: DAS, JBOD
- scale out performance and capacity as needed.
- replicate synchronously and asynchronously.

# RED HAT STORAGE DEPLOYMENT ON AMAZON CLOUD

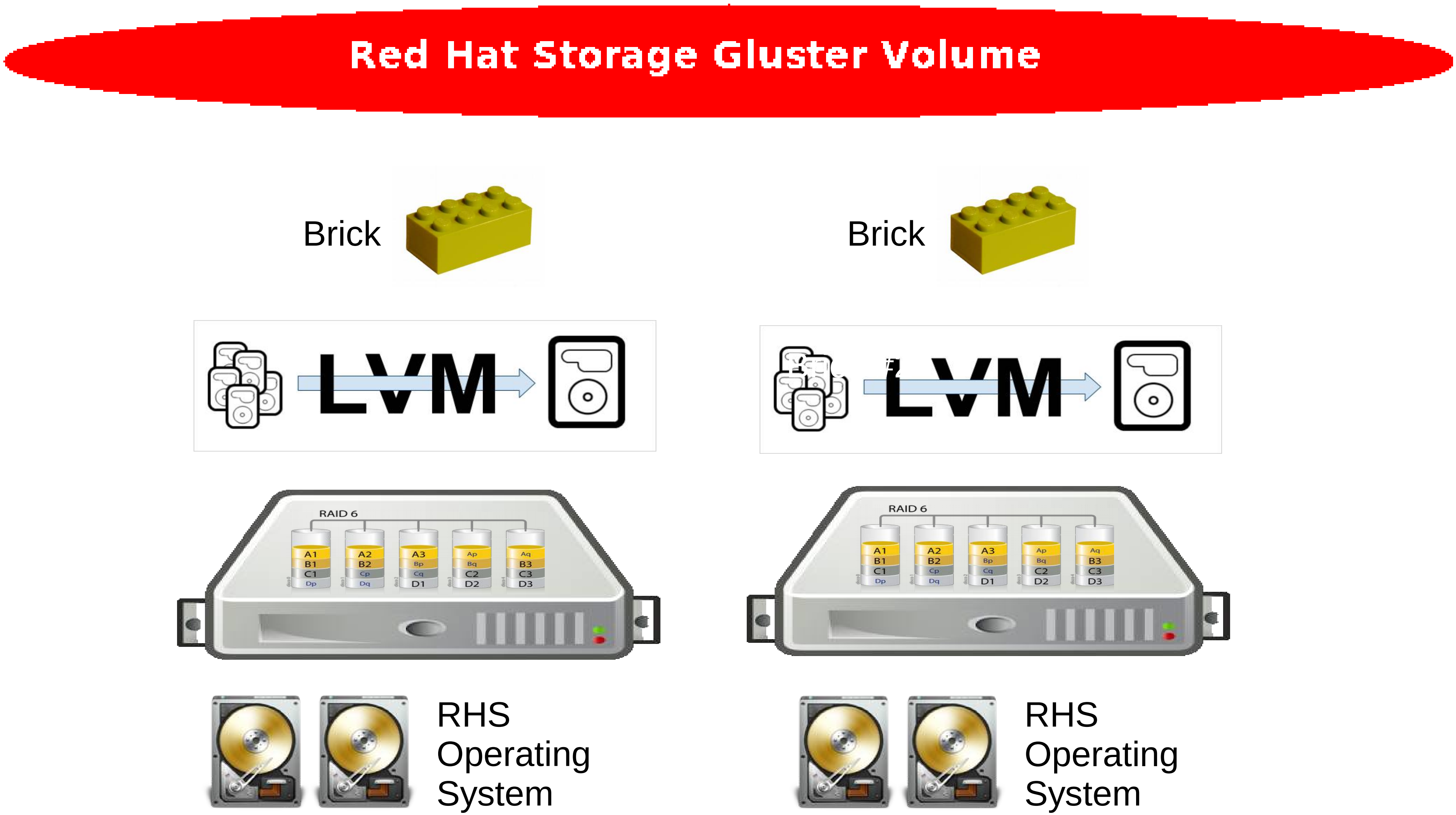



- GlusterFS Amazon Machine Images (AMIs)
- The only way to achieve high availability of Elastic Block Storage (EBS)
- Multiple EBS devices pooled
- POSIX compatible (no application to rewrite required to run on Amazon EC2)
- scale out capacity and performance as needed

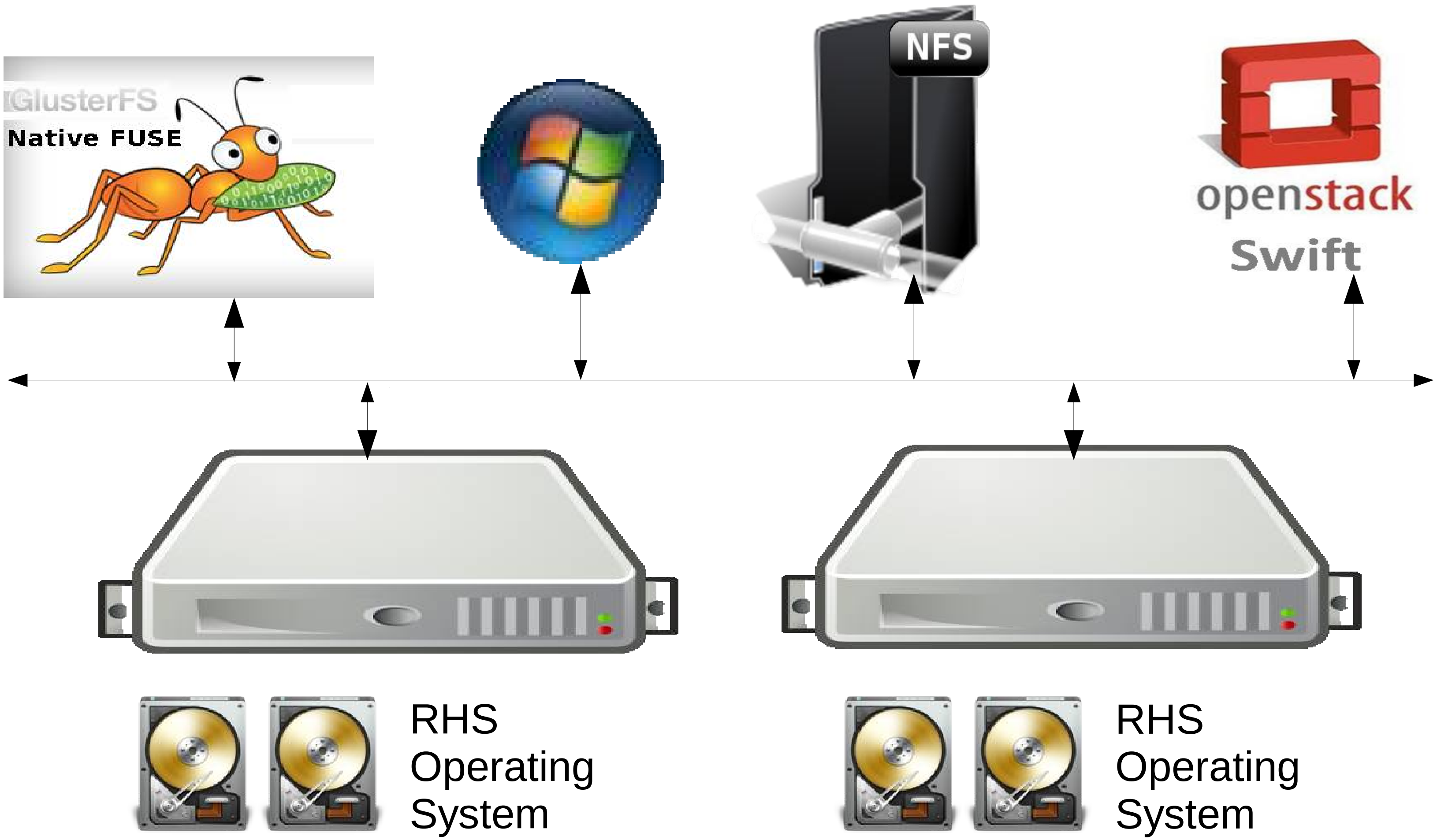
# RED HAT STORAGE TECHNOLOGY STACK



# Red Hat Storage volume setup anatomy

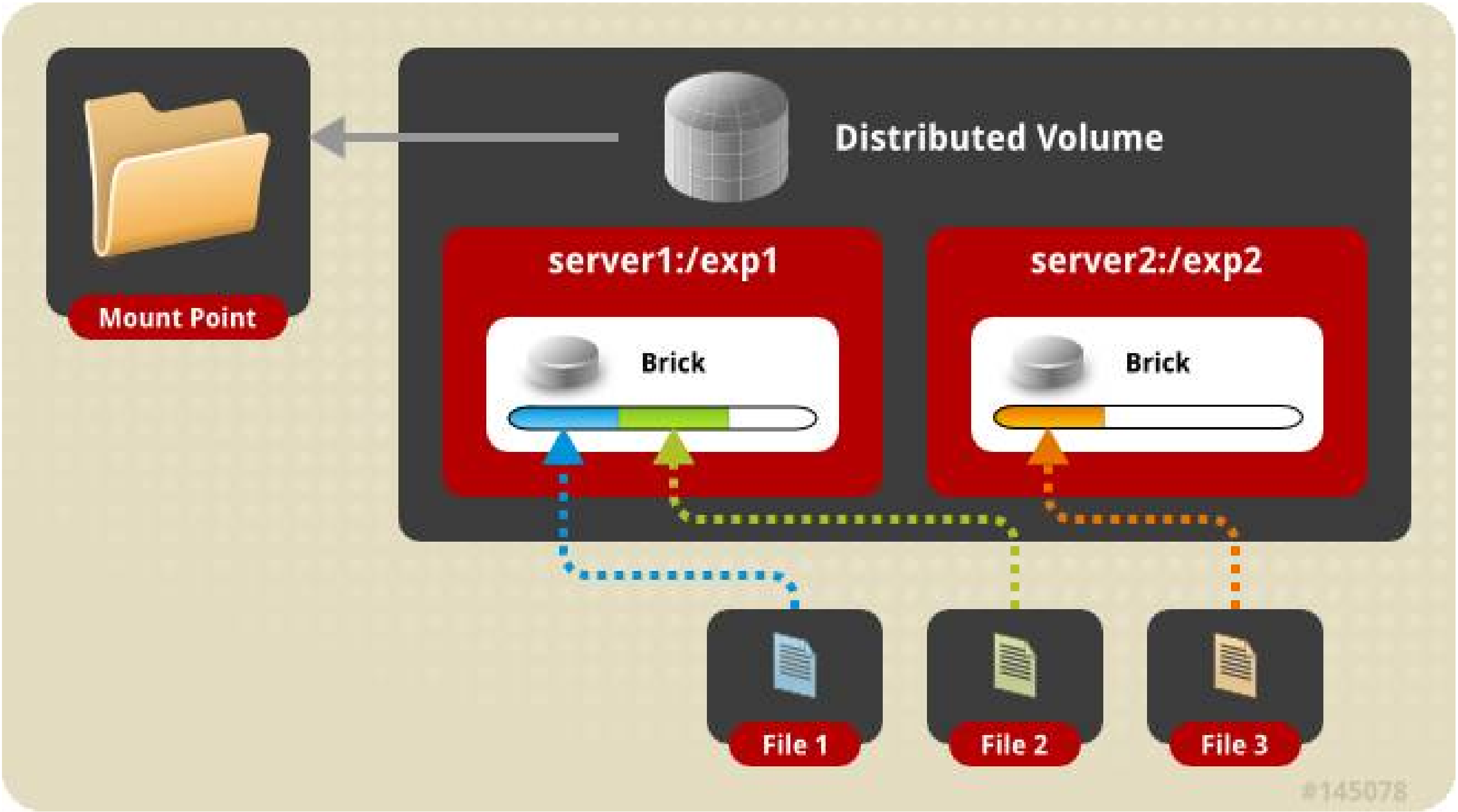


# Red Hat Storage volume connect options



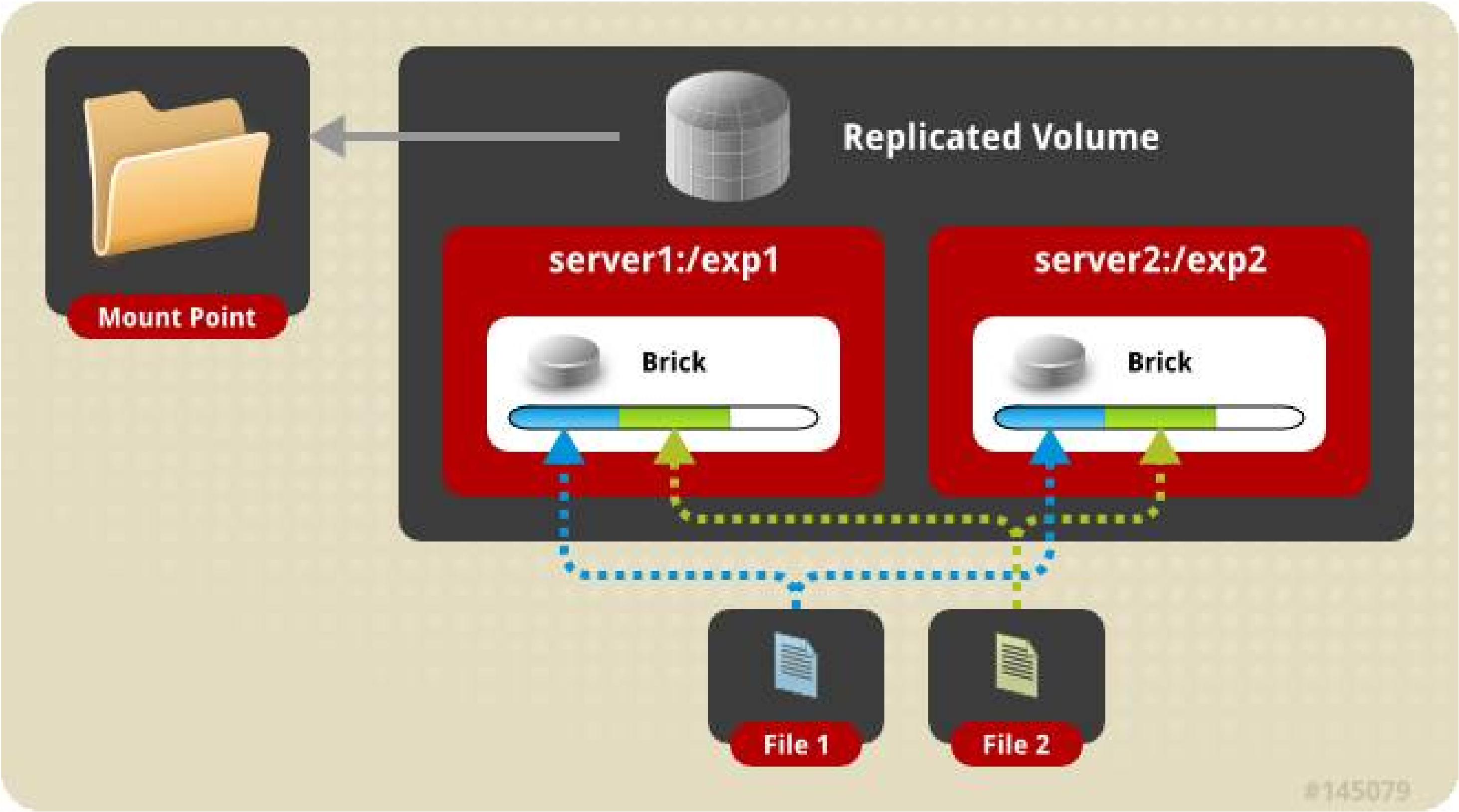
# Distributed Volume

- basic volume-type
- NO high availability
- distributing files across available bricks



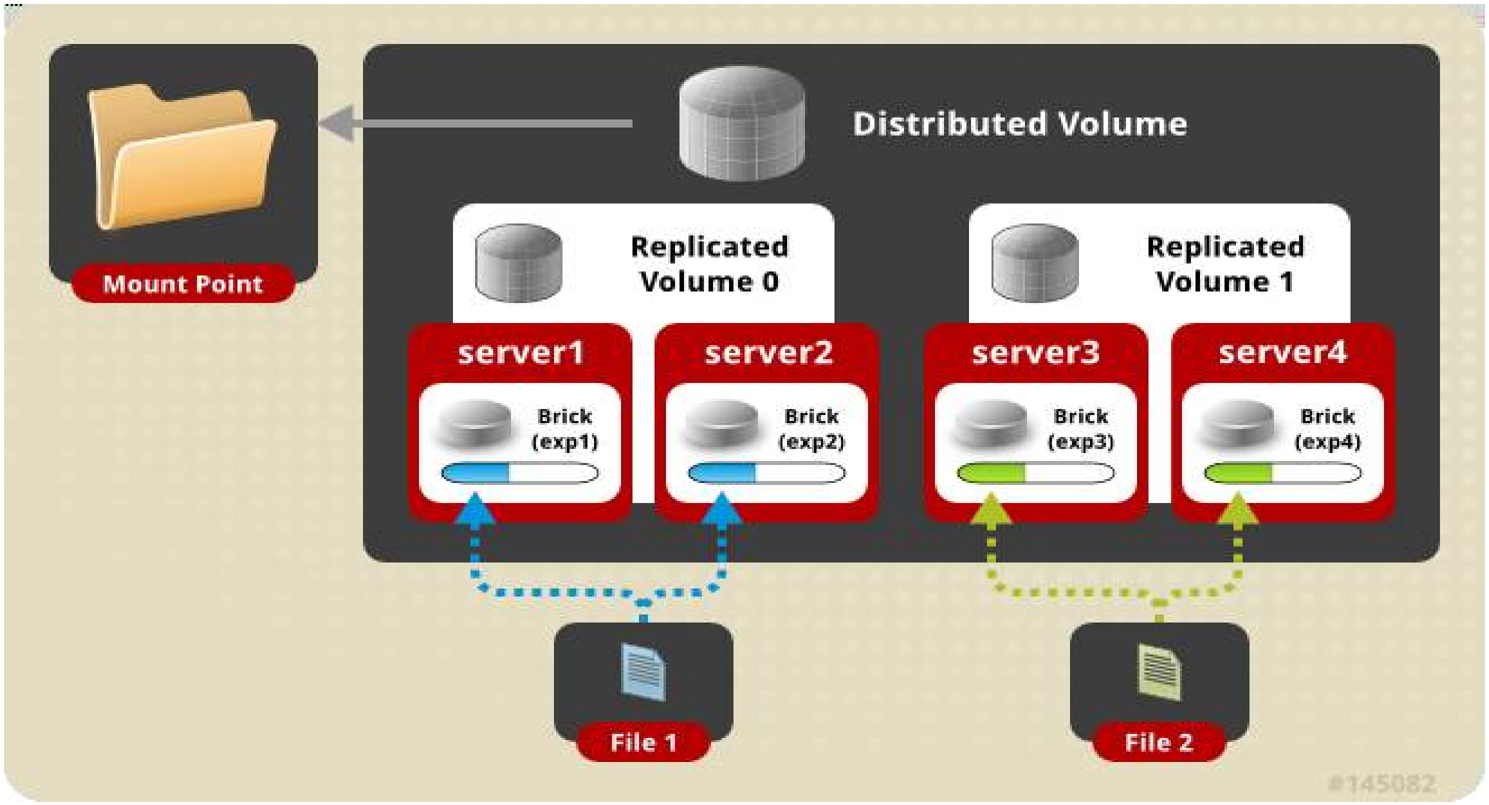
# Replicated Volume

- basic volume-type
- high availability (up to 3 copies)
- distributing files across available bricks



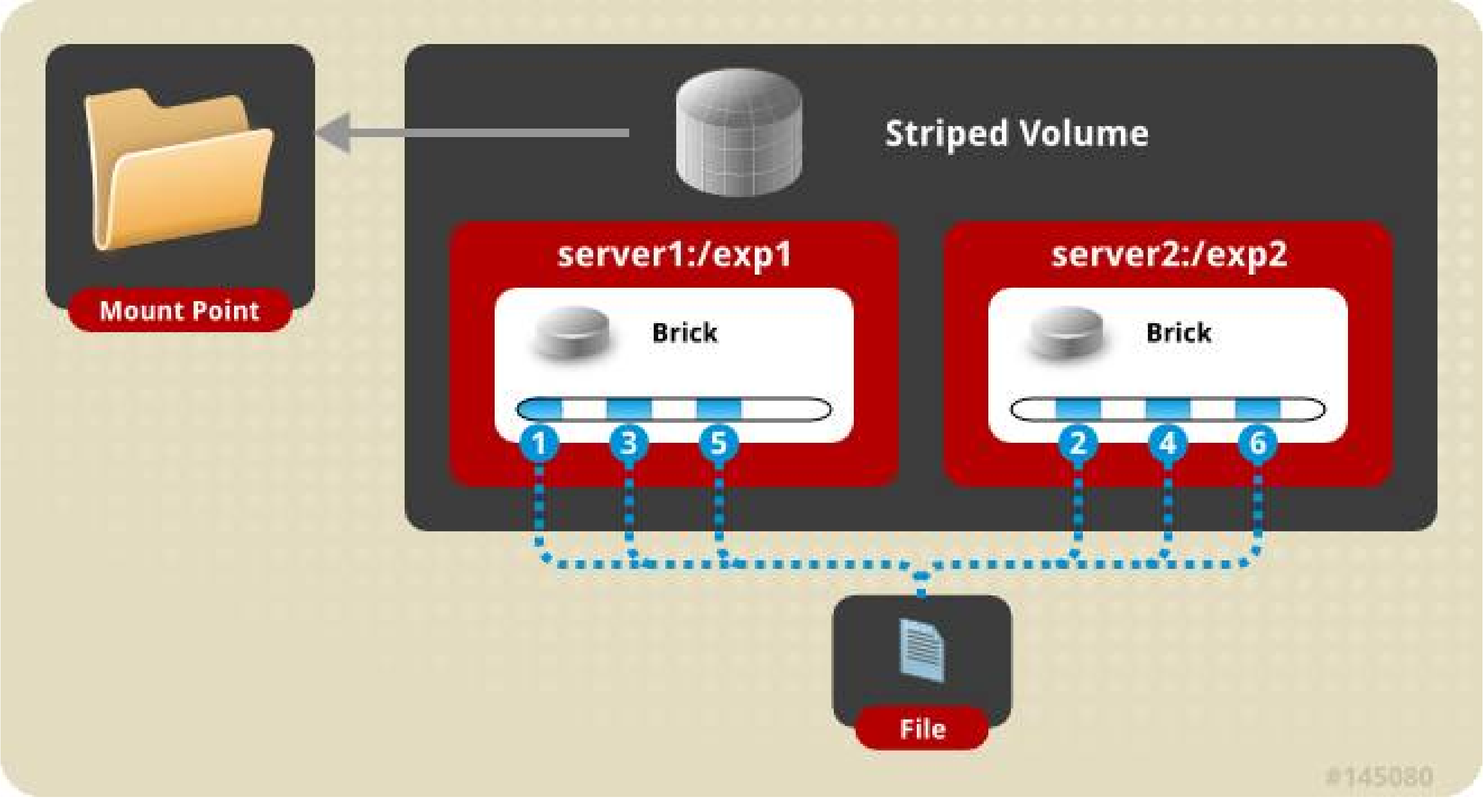
# Distributed-Replicated Volume

- combined volume-type
- high availability (up to 3 copies)
- High performance
- distributing files across available bricks

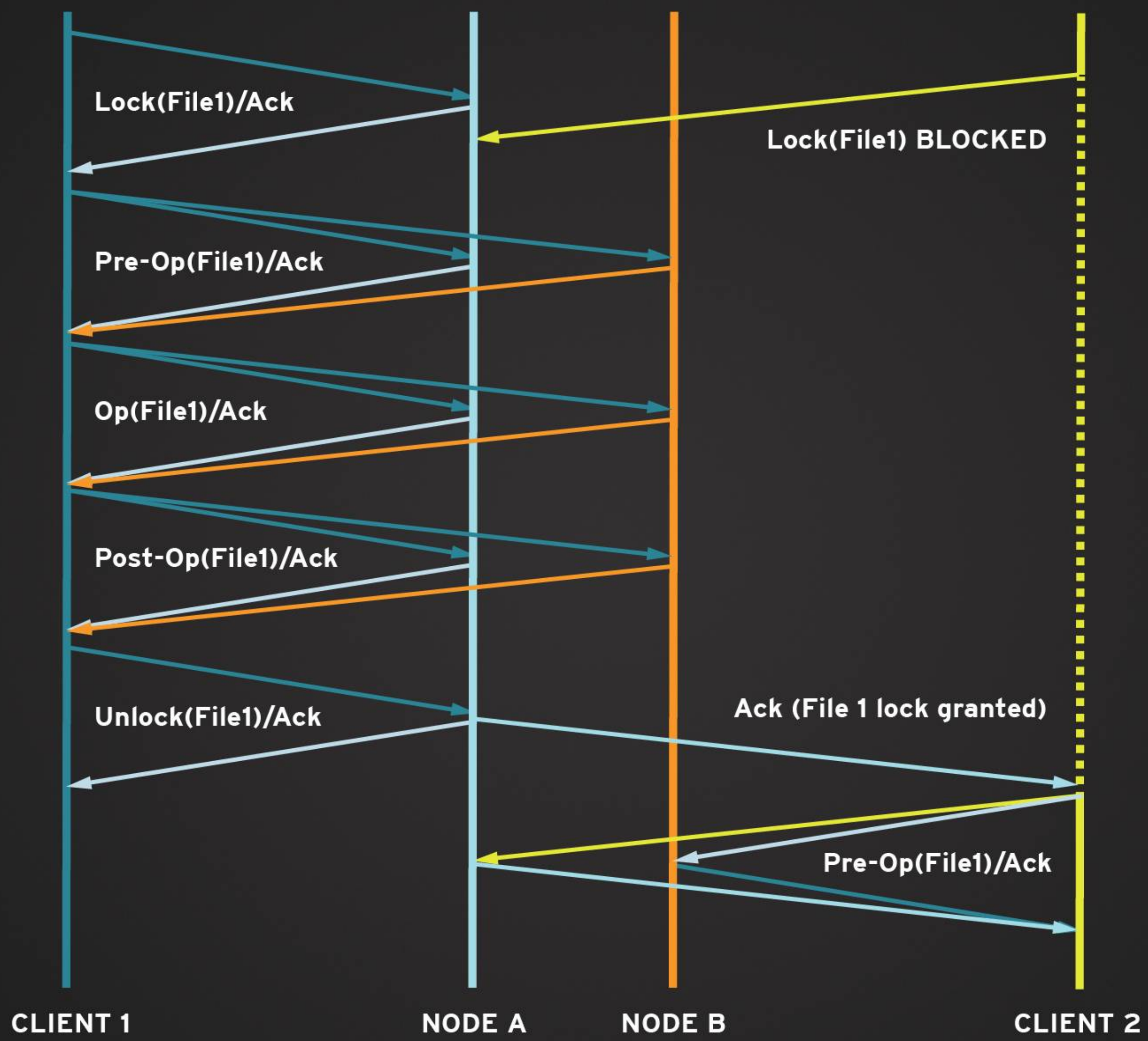


# Striped Volume (tech. preview)

- datafiles striped across bricks
- use in high concurrency environments w/ large files
- # of bricks should equal stripe count

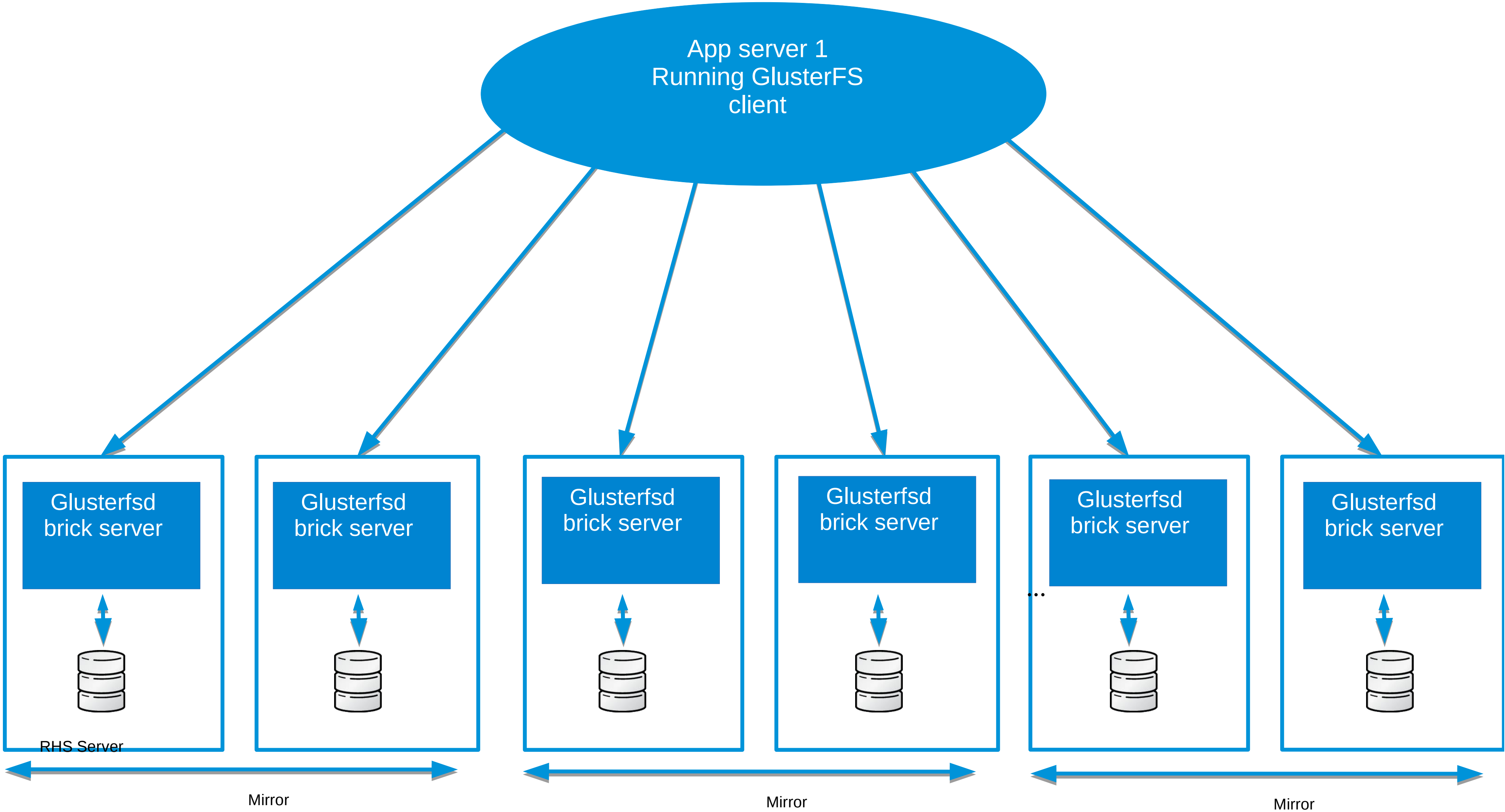


# HOW DOES REPLICATION ACTUALLY WORK?



# GlusterFS native client – data flow

**Clients talk directly to the data bricks based on elastic hash**



# NFS

Accessibility from UNIX and Linux systems

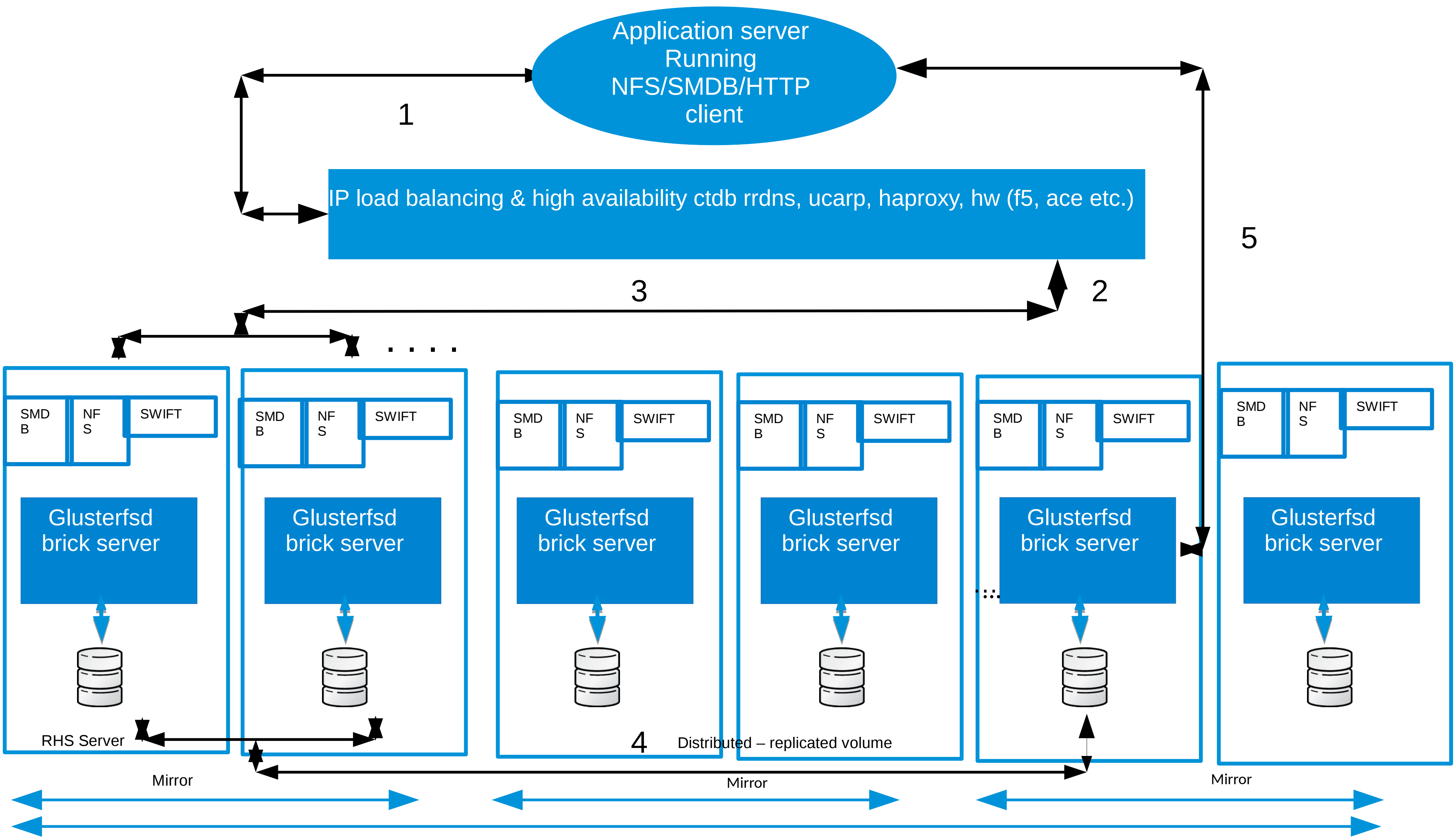
- Standard NFS v3 clients connect to GlusterFS NFS server process (user space) on storage node
- Mount GlusterFS volume from any storage node
- GlusterFS NFS server includes network lock manager (NLM) to synchronize locks across clients
- Better performance for reading many small files from a single client
- Load balancing must be managed externally
- Standard automounter is supported

# SMB/CIFS

Accessibility from Windows systems

- Storage node uses Samba with winbind to connect with Active Directory environments
- Samba uses Libgfapi library to communicate directly with GlusterFS server process without going through FUSE
- SMB clients can connect to any storage node running Samba
- SMB version 2.0 supported
- Load balancing must be managed externally
- CTDB is required for Samba clustering

# NFS, CIFS & OBJECT – data flow

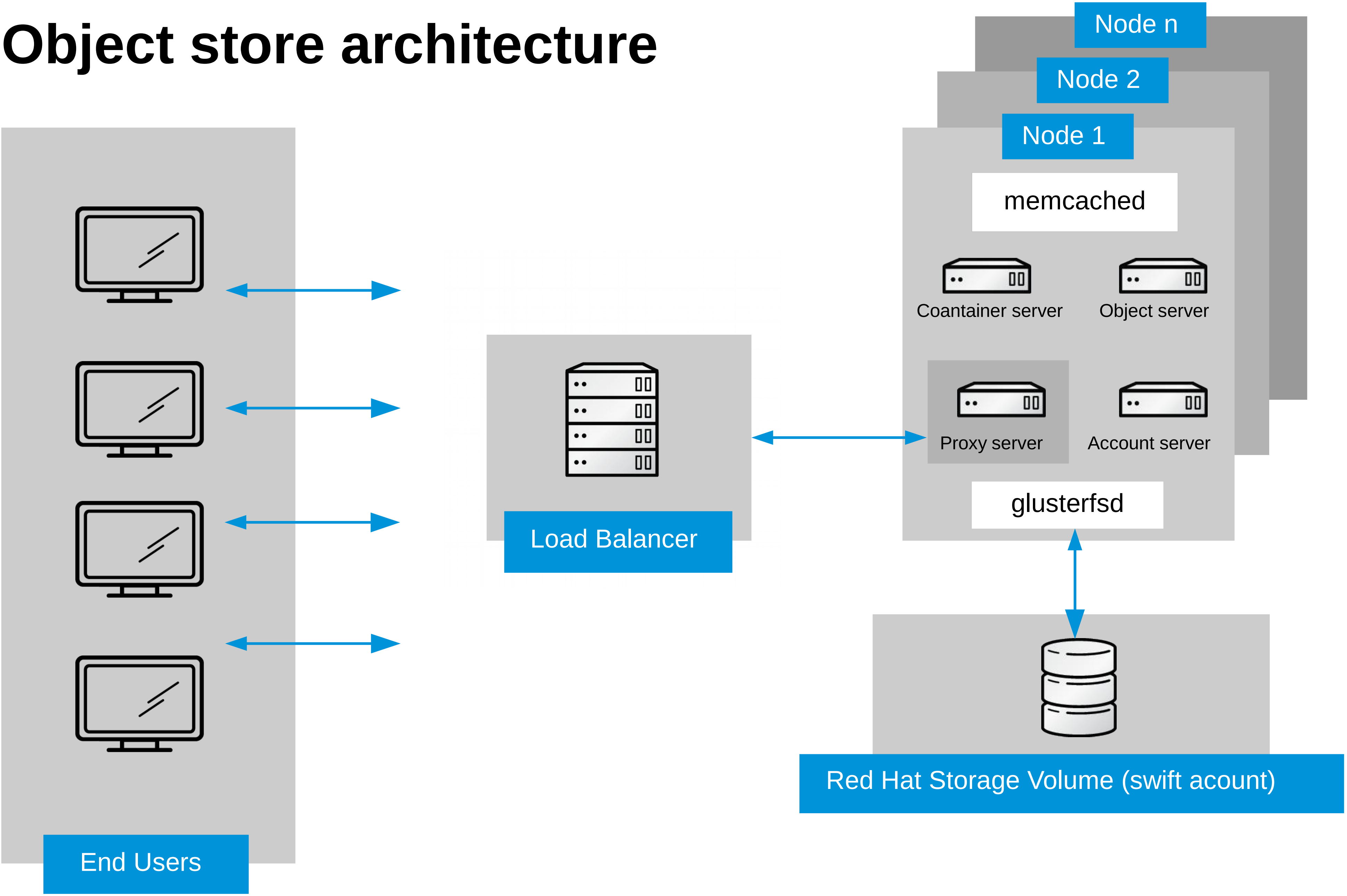



**C**lients first talk to the mounted storage node and are then directed to the data bricks. Non-native protocol adds an additional network hop.

# Object access of GlusterFS volume

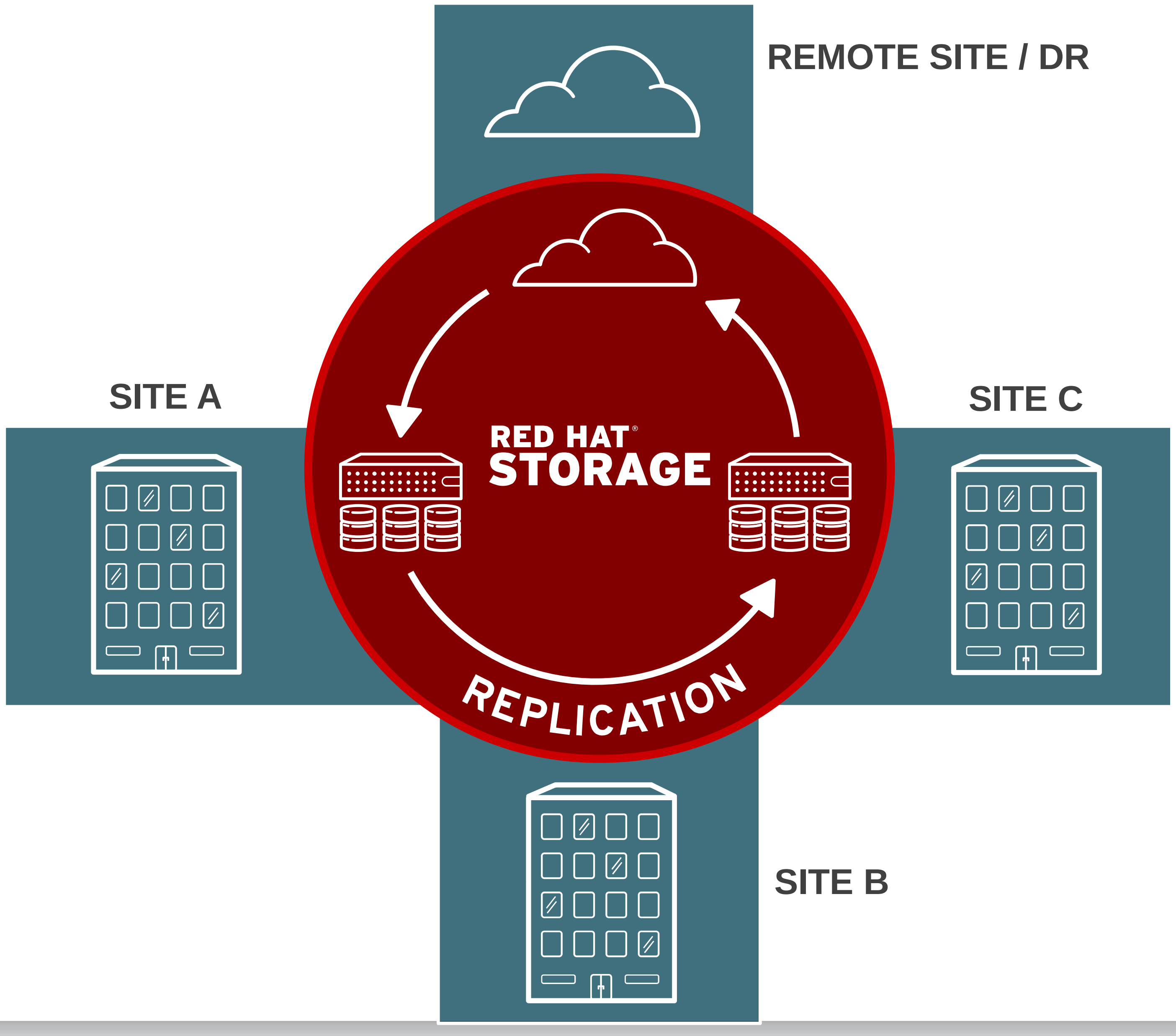
- Built upon OpenStack's Swift object storage
- GlusterFS is the back-end file system for Swift
- Implements objects as files and directories under the container
- Accounts are implemented as GlusterFS volumes
- Store and retrieve files using the REST interface
- Support integration with SWAuth and Keystone authentication service

# Object store architecture

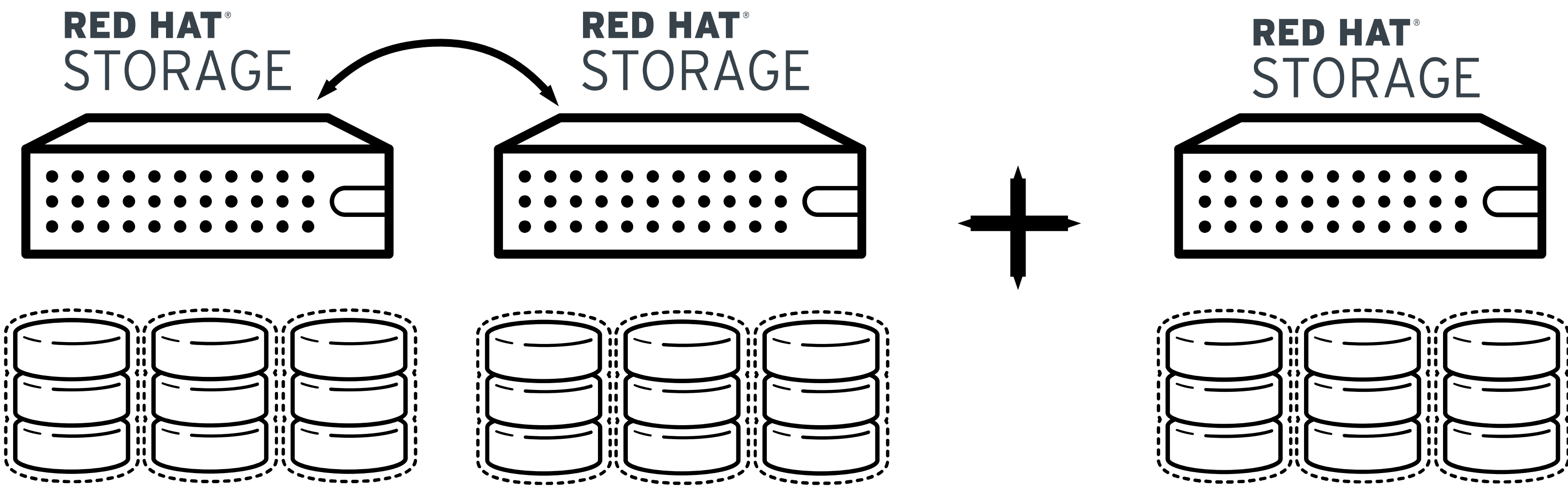


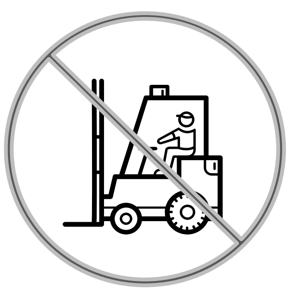
# Geo-replication : global data protection and availability

**Remote site / disaster recovery**



# Built-in re-balancing frees up IT budget



- Moves data to new node
- Upgrade without disruption
- Easily de-commission systems

**Forklift upgrades**

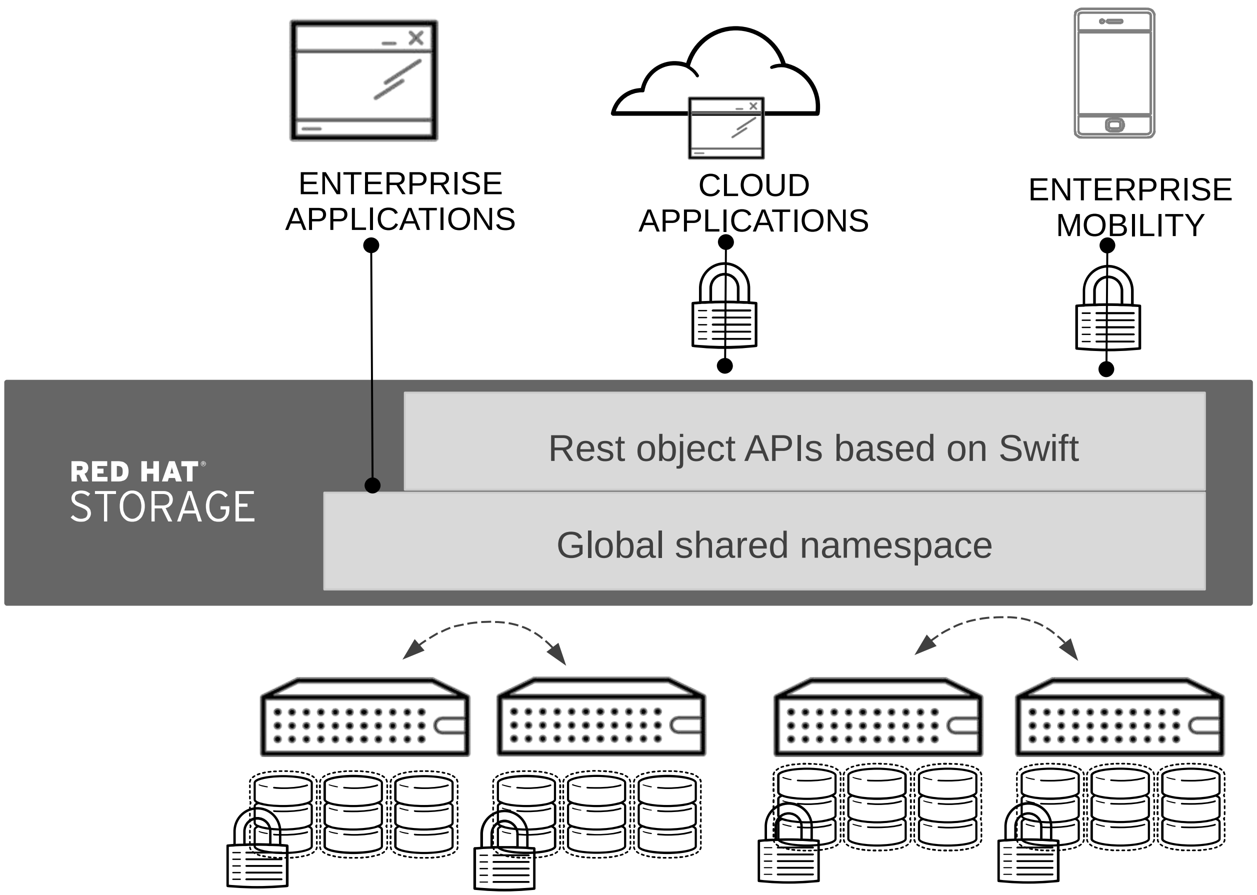
**Migration costs**

# HA for NFS and CIFS

- **Any IP failover tool can work for NFS**
  - Appliance based load balancers with heartbeat such as F5
  - Linux heartbeat, ucarp, CTDB
  - Not all failover works for CIFS as that requires some session handling
- **CTDB (Cluster Trivial Database) is what we use**
  - It is very simple to configure
  - Works for NFS
  - Works for CIFS
  - Is very robust and configurable
- **Round robin DNS for load balancing**
  - You can use any load balancer you want
  - RRDNS is simple to configure and works well
  - Prevents hot spots of activity

# Simultaneous, secure object access to file data

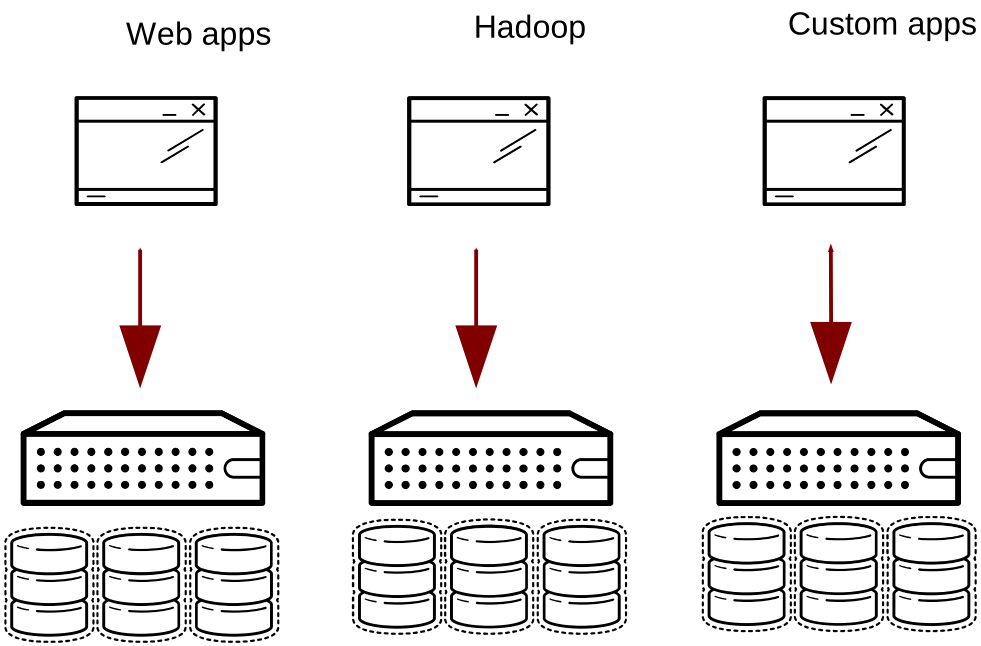
## Unified file and object storage




- Open rest API based on OpenStack® Swift
- Https encrypted data over the wire
- Encrypt data at rest with DMCRYPT
- Identity and authentication support keystone, Kerberos, and Red Hat IDM

# Storage co-resident applications

## Converging compute and storage

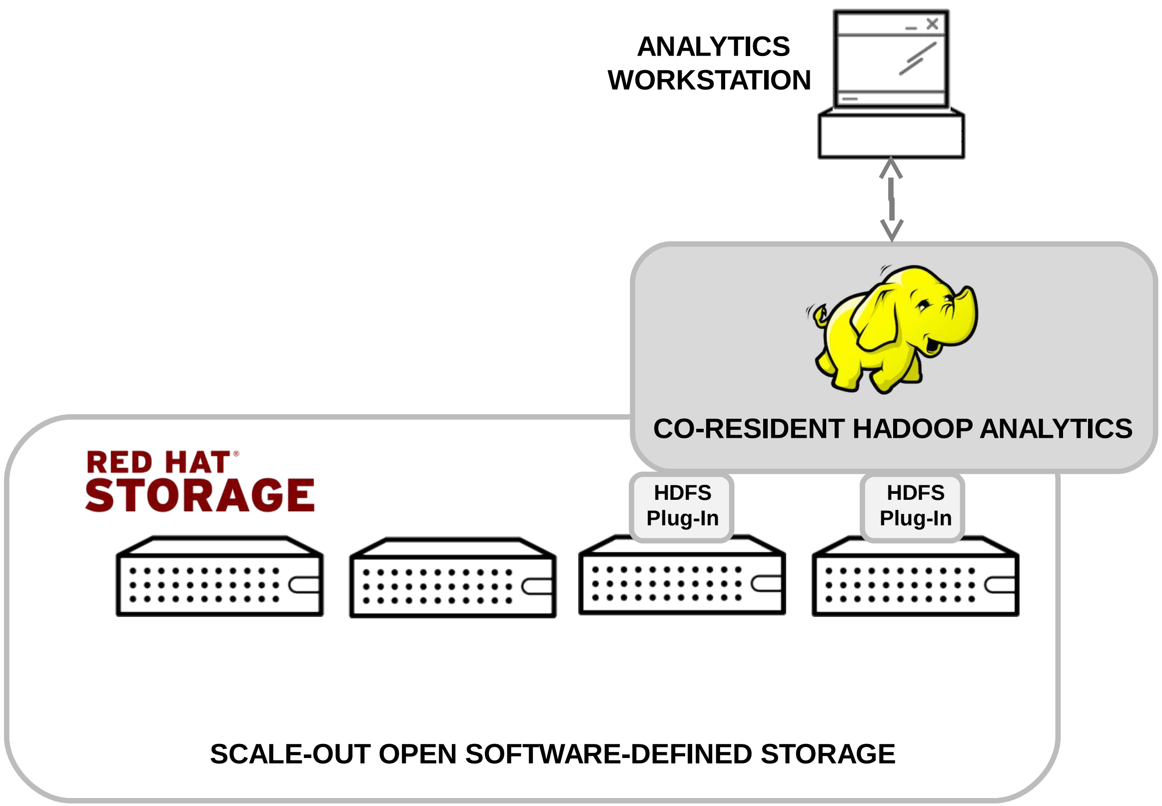



- Run native Red Hat Enterprise Linux applications
- Process data locally
- Eliminate an entire tier of hardware
- Use c-groups to manage resources
- Use Red Hat Enterprise Linux KVM

RED HAT® STORAGE

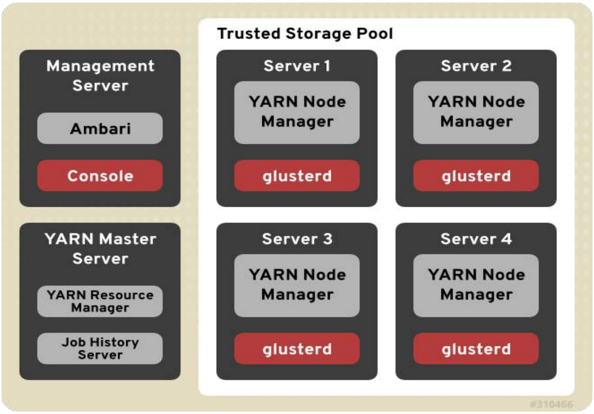
# Red Hat Storage Server Hadoop connector

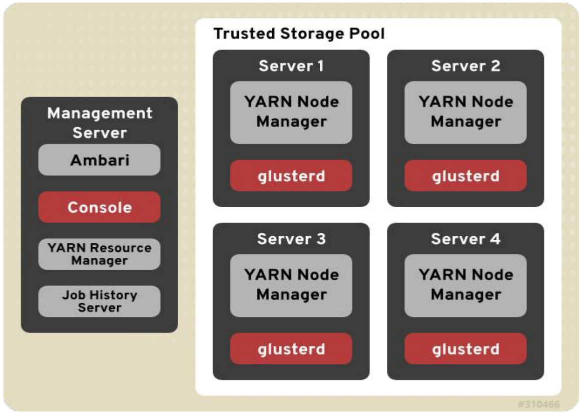
## Apply Hadoop analytics directly on production data

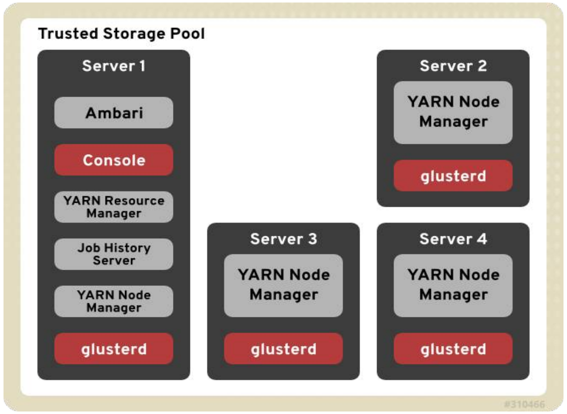



- Direct data access
- Eliminate HDFS ingestion overhead
- Included with your Red Hat Storage Server subscription
- Alongside or instead of HDFS

# Red Hat Storage Server Hadoop connector




2 additional Server

1 additional Server

0 additional Server

FOR TEST ONLY

# Simplified and unified storage management

**Single management console for converged storage and compute**

Console storage operations

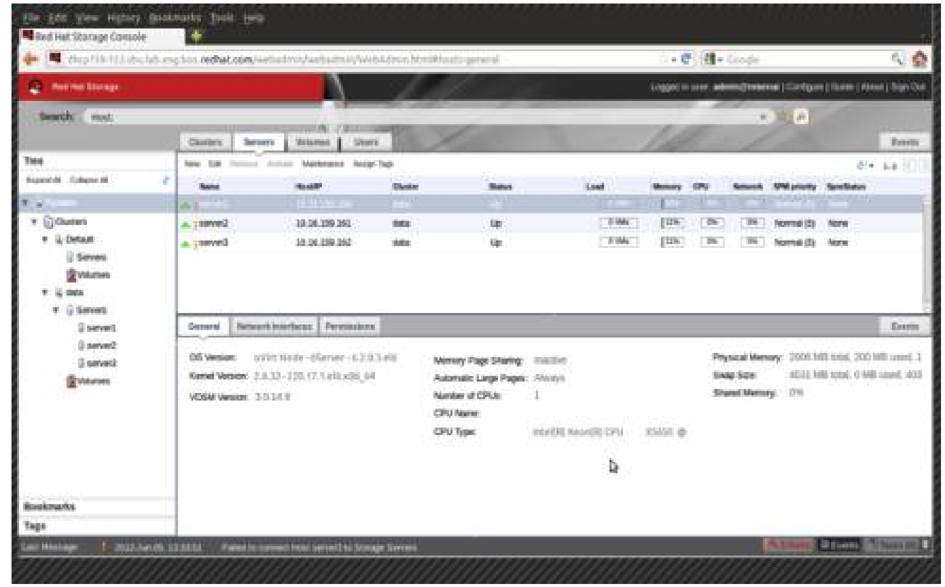
- Intuitive user interface
- Volume management
- On-premise and public cloud
- Rest-based API

Provisionable with Satellite

- Life cycle management
- Familiar Red Hat Enterprise Linux tools

Virtualization and storage

- Shared management with RHEV-M



MANAGEMENT TOOLS & FRAMEWORK

**Red Hat Storage Server Console**
Storage

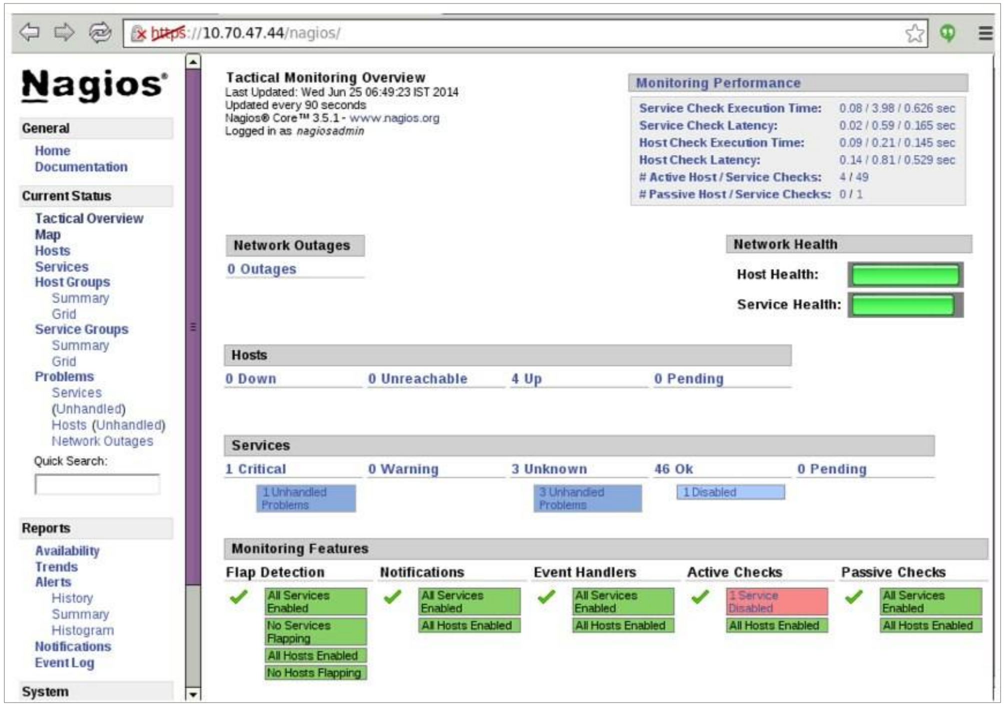
**RHEV Manager**
Virtualization

**Red Hat Satellite**
Provisioning and life cycle management

# Comprehensive, Integrated Monitoring

## SNMP, extensible Nagios framework - integrates with existing tooling

- Monitoring logical entities and physical resources
- Alerts, reports, trending and capacity planning graphs
- Non-disruptive upgrades
- Works with Red Hat Storage Console
- Can be set up to run in standalone mode








redhat.

# Red Hat Storage Server subscription options

Red Hat Storage Server is available in 1 and 3 year subscriptions

- Red Hat Storage Server for On-Premise
  - Designed as a software appliance for deployment within your datacenter
- Red Hat Storage Server Module for On-Premise
  - Designed for deployment on existing Red Hat Enterprise Linux installs
- Red Hat Storage Server for Public Cloud
  - Designed for use in the public cloud (amazon currently supported)
- Red Hat Storage Server for Hybrid Cloud
  - Combines subscriptions for on-premise and public-cloud

# What is in the Red Hat Storage Server subscription?

Red Hat Storage Server for On-Premise is an all-inclusive .iso

- Red Hat Enterprise Linux tuned for Red Hat Storage
- GlusterFS, XFS file system
- Support for all standard protocols
- HDFS plugin for Hadoop workloads
- Red Hat Storage console

Red Hat Storage Server Module for On-Premise is a collection of RPMs

- Deployable on existing Red Hat Enterprise Linux Installations
- GlusterFS, XFS file system
- Support for all standard protocols
- HDFS plugin for Hadoop workloads
- Red Hat Storage console

Red Hat Storage Server for Public Cloud is an all-inclusive Amazon AMI

- Red Hat Enterprise Linux tuned for Red Hat Storage
- GlusterFS, XFS file system
- Support for all standard protocols
- HDFS plugin for Hadoop workloads
- Does not include Red Hat Storage console

# Hardware (I)

Generic Requirements

- Must be in the Red Hat Hardware Compatibility List for Storage for Red Hat Enterprise Linux 6.0 and newer
- 2-socket (with 4-core, 6-core, or 8-core) servers are recommended
- Reliable backplane RAID controller shipped by server vendors or from OEM manufacturers
- RAID 6 and RAID 1+0 Support in hardware RAID controller (must be flash-backed or battery-backed)
- 1X 50 GB SAS disks for RHSS installation if a separate partition is created for /var (otherwise 200GB)
- 1 X 10 GigE NIC for data traffic is recommended. It's advisable to use NIC bonding with 2 X 10 GigE for increasing throughput and resiliency. 1 GigE NIC may also be used
- Redundant power supply
- Out of band management card to manage and monitor RHS nodes even when the server is down. The same interface may be used for data and management traffic, but it's recommended to have separate networks for data and management.

# Hardware (II)

- **High Performance**

  - 2u/24 (JBODs may be attached)
  - 15000 RPM 900GB drives(2.5" inch SAS) OR Solid state disks.
  - Minimum RAM 48 GB

- **General Purpose File Serving use-case**

  2u/12 (JBODs may be attached)

  7200 or 10000 RPM, up to 6 TB drives (3.5" SAS or SATA)

  Minimum RAM 32 GB

- **Archival use-case**

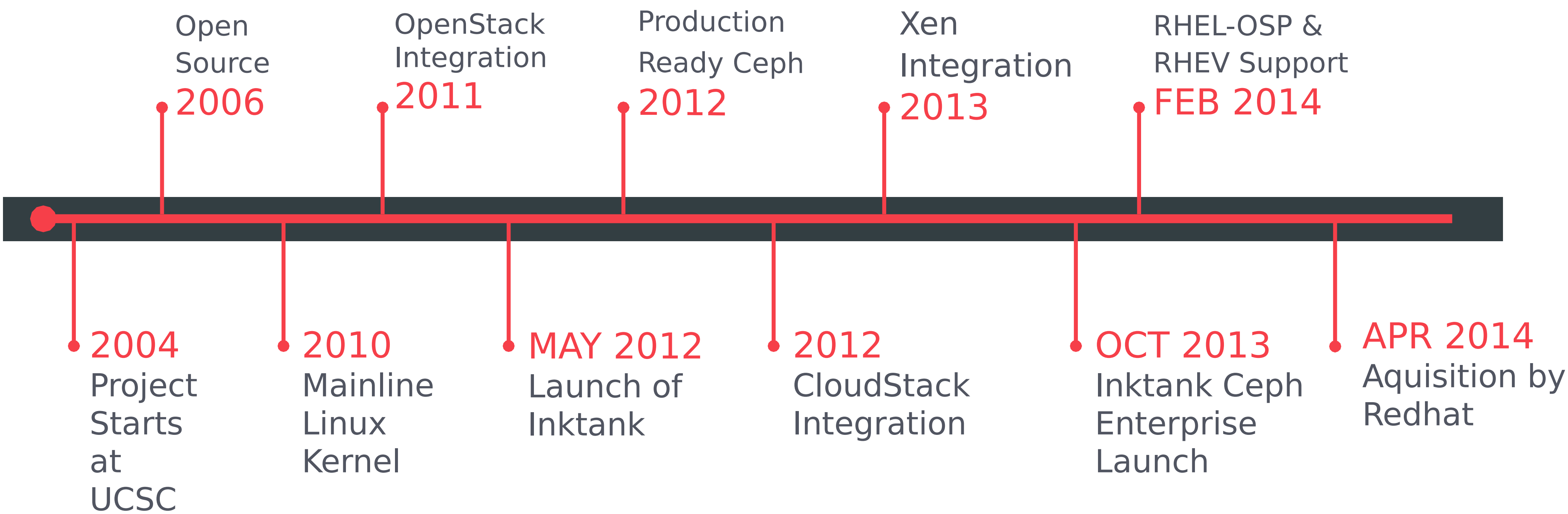
- - 4u/60 (JBODs may be attached)
  - 7200 or 10000 RPM, up to 6 TB drives (3.5" SAS or SATA)
  - Minimum RAM 16 GB

# 2 Produkte.....

Red Hat **Gluster** Storage

Red Hat **Ceph** Storage

# History of CEPH

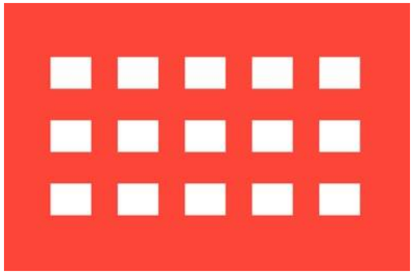


# TRADITIONAL STORAGE VS. CEPH

TRADITIONAL ENTERPRISE STORAGE

| | |
|---|---|
| Single Purpose | Multi-Purpose, Unified |
| Hardware | Distributed Software |
| Single Vendor Lock-in | Open |
| Hard Scale Limit | Exabyte Scale |

# CEPH unified storage



OBJECT STORAGE

- S3 & Swift
- Multi-tenant
- Keystone
- Geo-Replication
- Native API

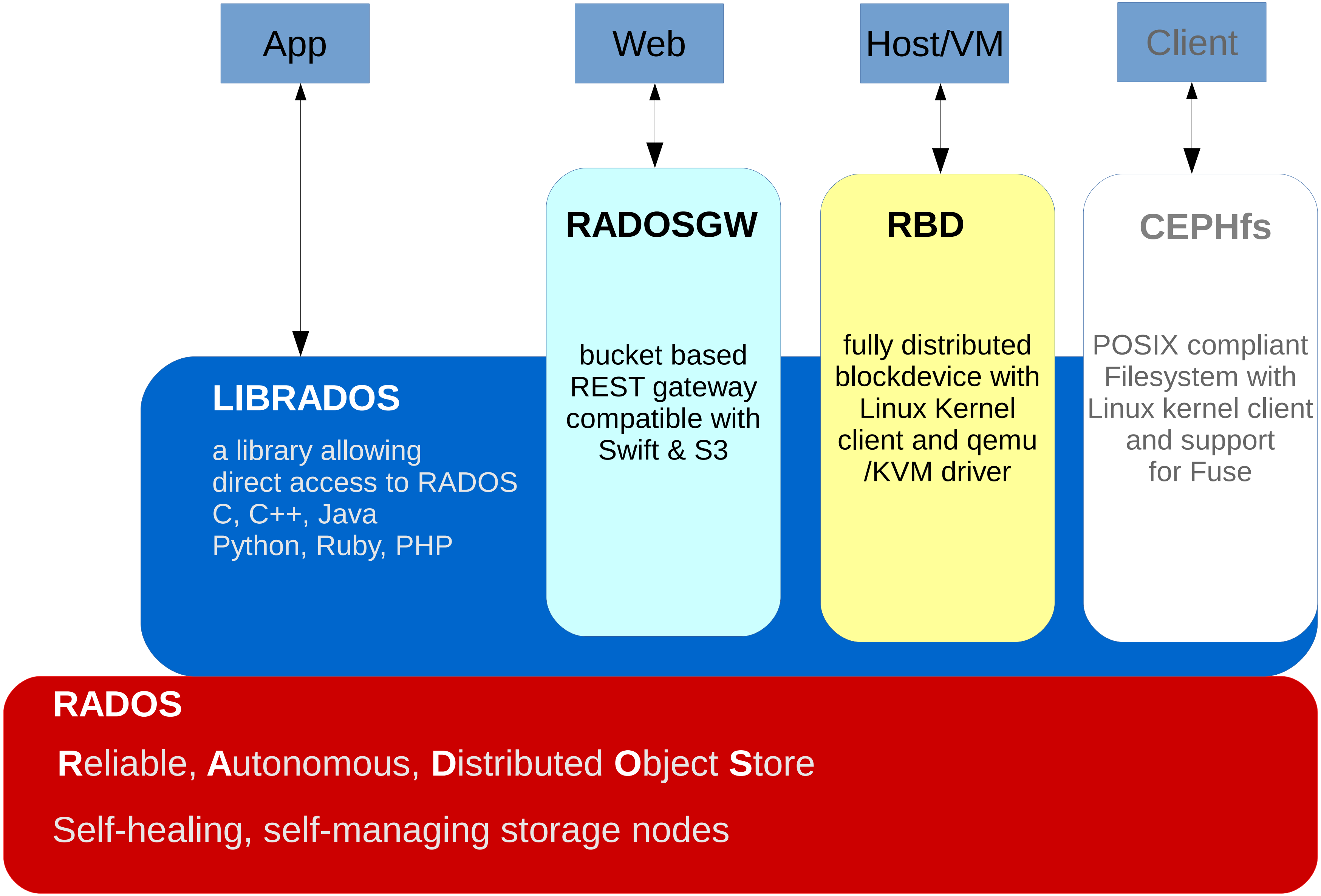
BLOCK STORAGE

- Snapshots
- Clones
- OpenStack
- Linux Kernel
- iSCSI

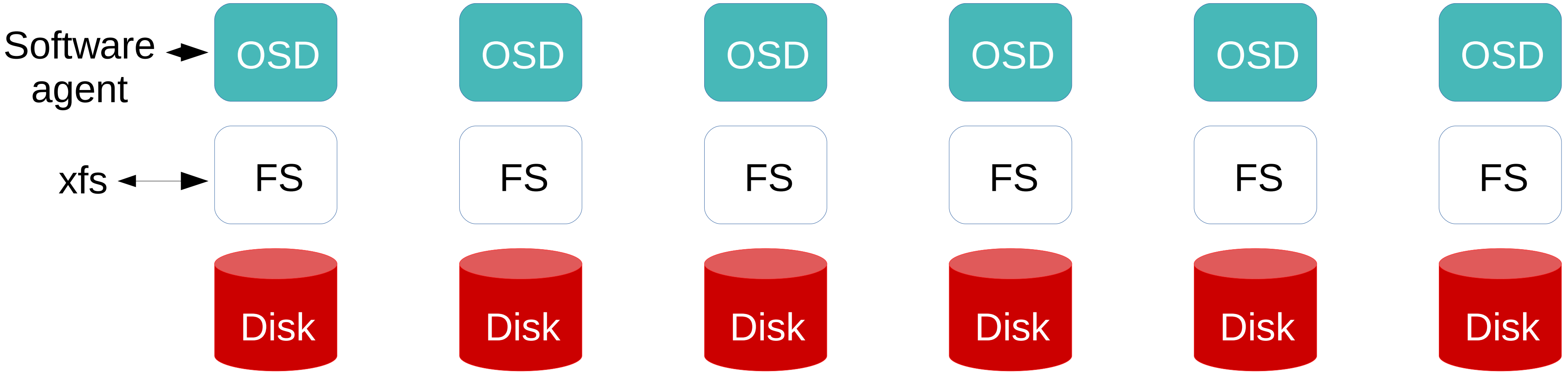
FILE SYSTEM

- POSIX
- Linux Kernel
- CIFS/NFS
- HDFS
- Distributed Metadata

# CEPH : build for exabyte scale



# CEPH : OSD‘s (Object Storage Daemons)




**RADOS**

**R**eliable, **A**utonomous, **D**istributed **O**bject **S**tore

Self-healing, self-managing storage nodes

# CEPH : OSD's & Monitors

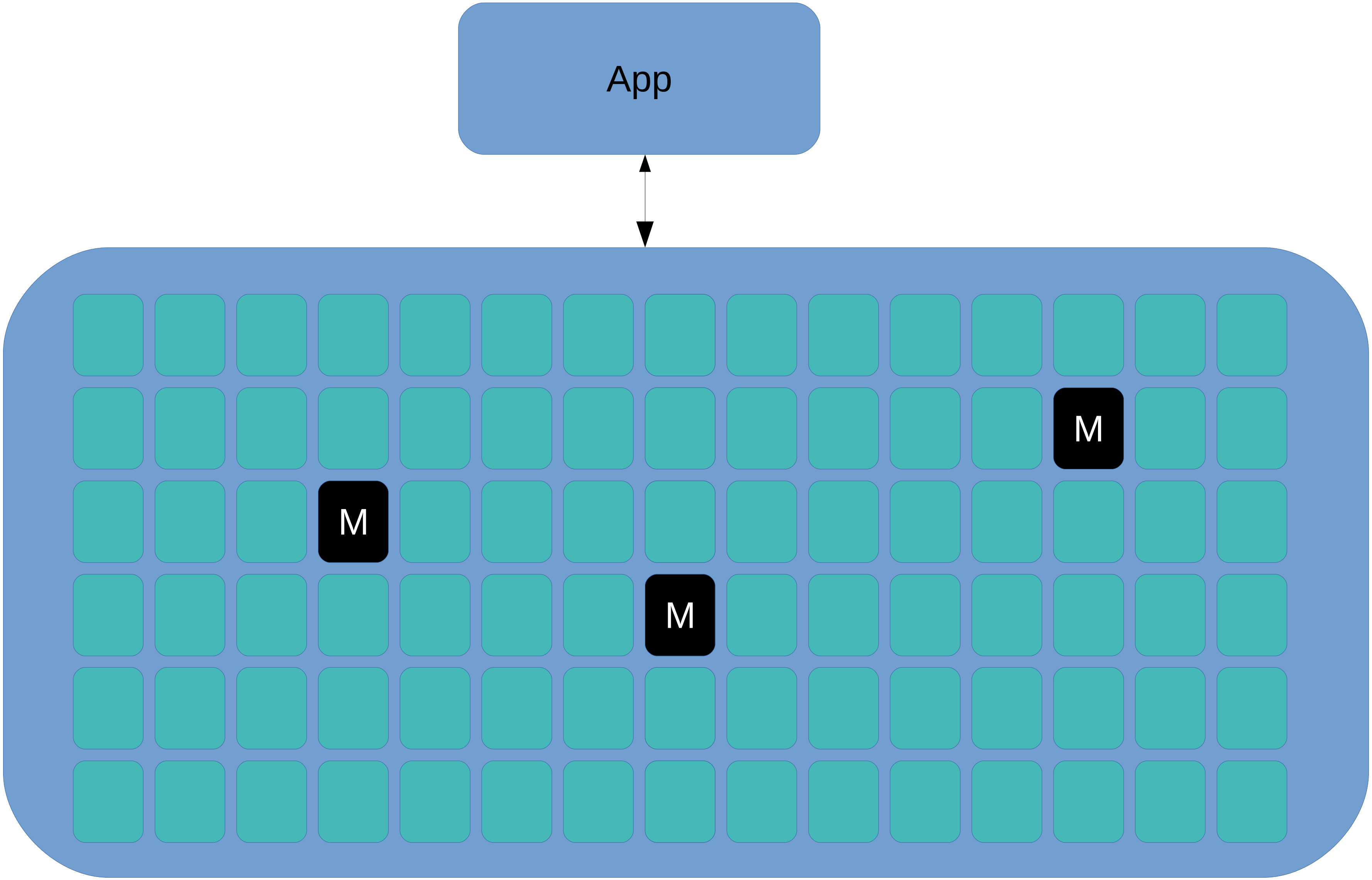
OSD

- 10 - 10.00 OSD's possible
- one / disk, RAID group, SSD
- serves stored objects to clients
- replication / recovery tasks

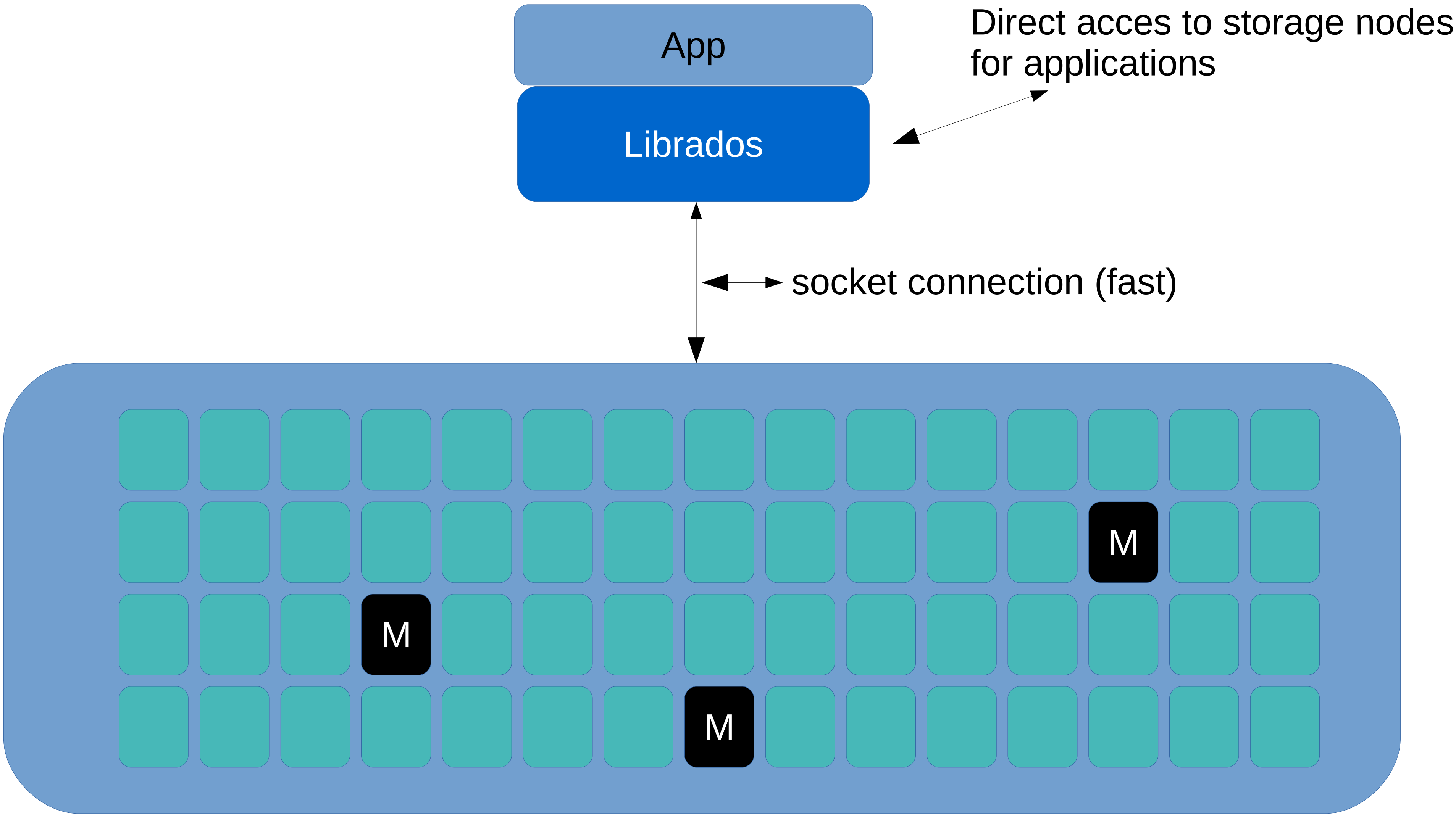
M

- maintains cluster membership & state of OSD's
- they vote for existing resources
- they don't serve objects to clients !
- provides concensus for distributed decision making
- few (3-5) per cluster

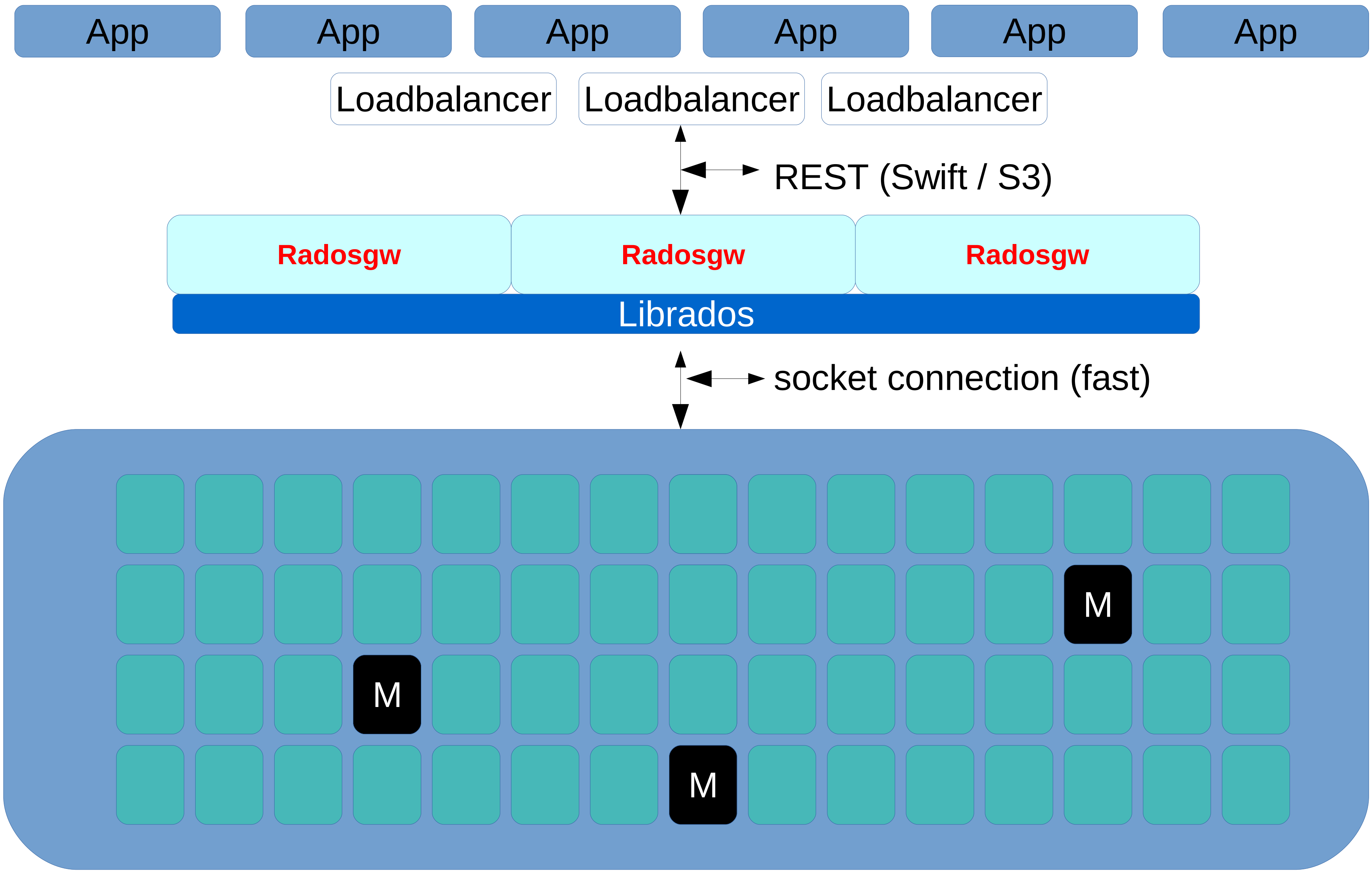
# CEPH : OSD's & Monitors



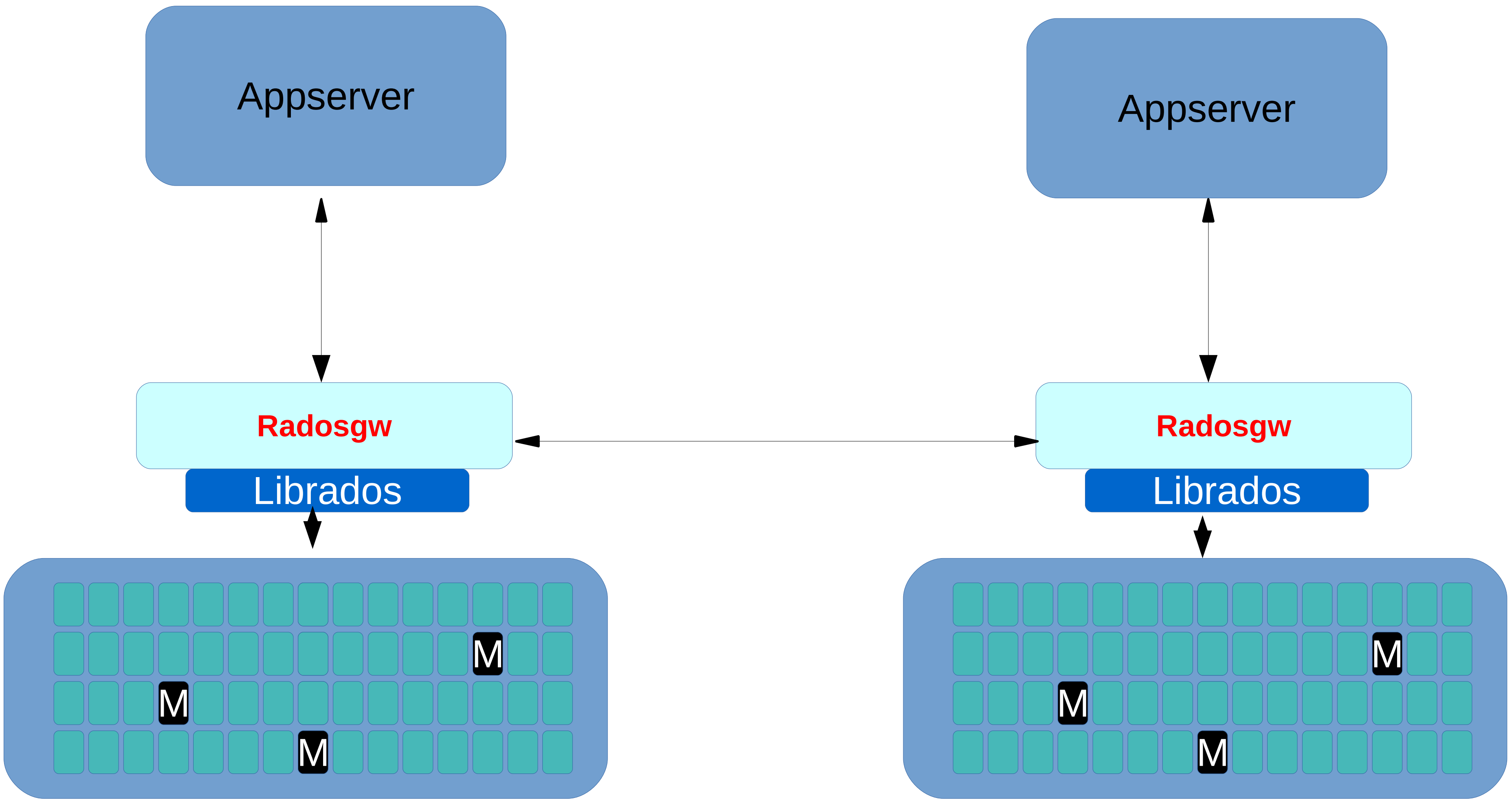
# CEPH : Application to Rados



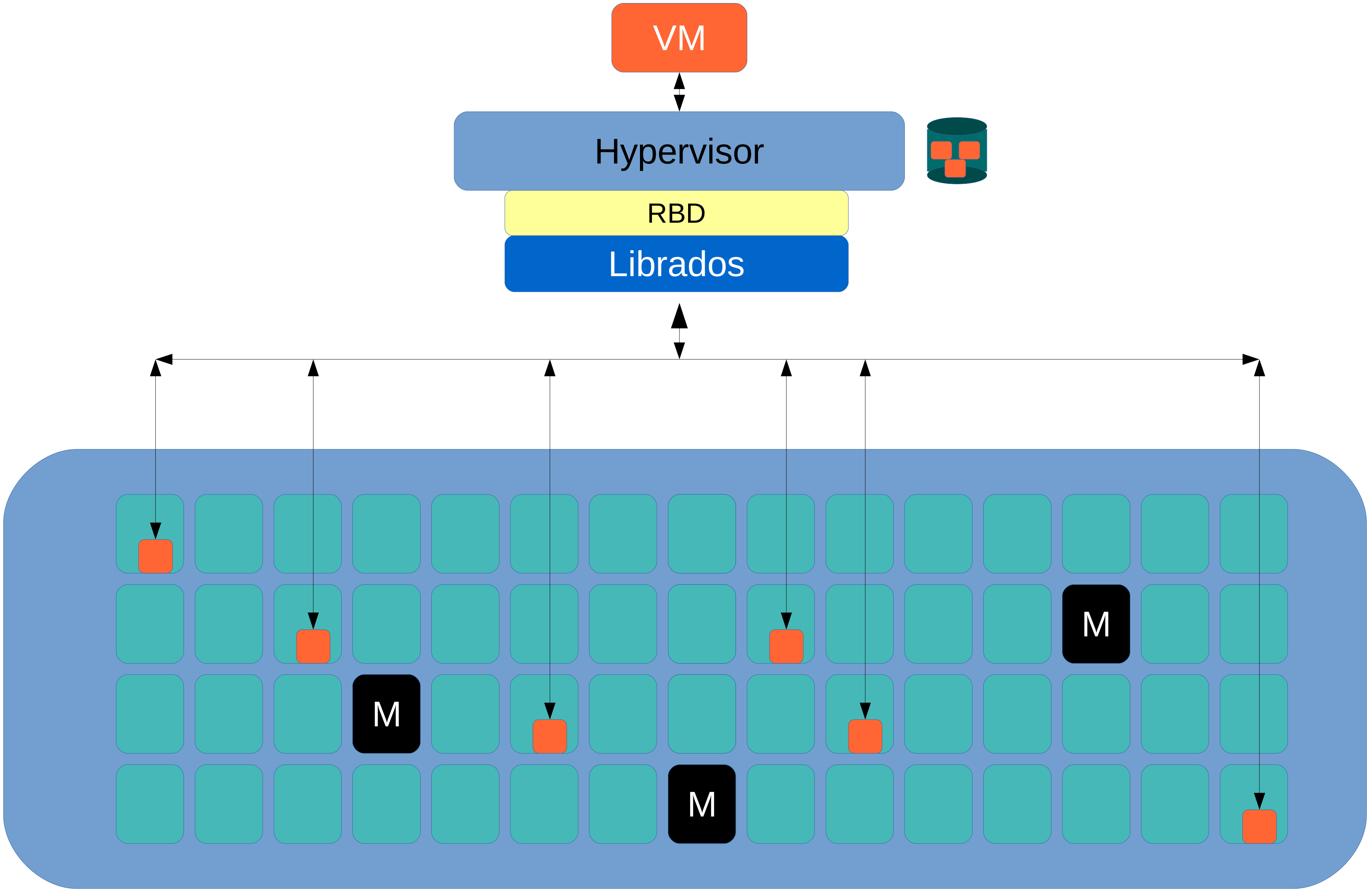
# CEPH : REST to Rados



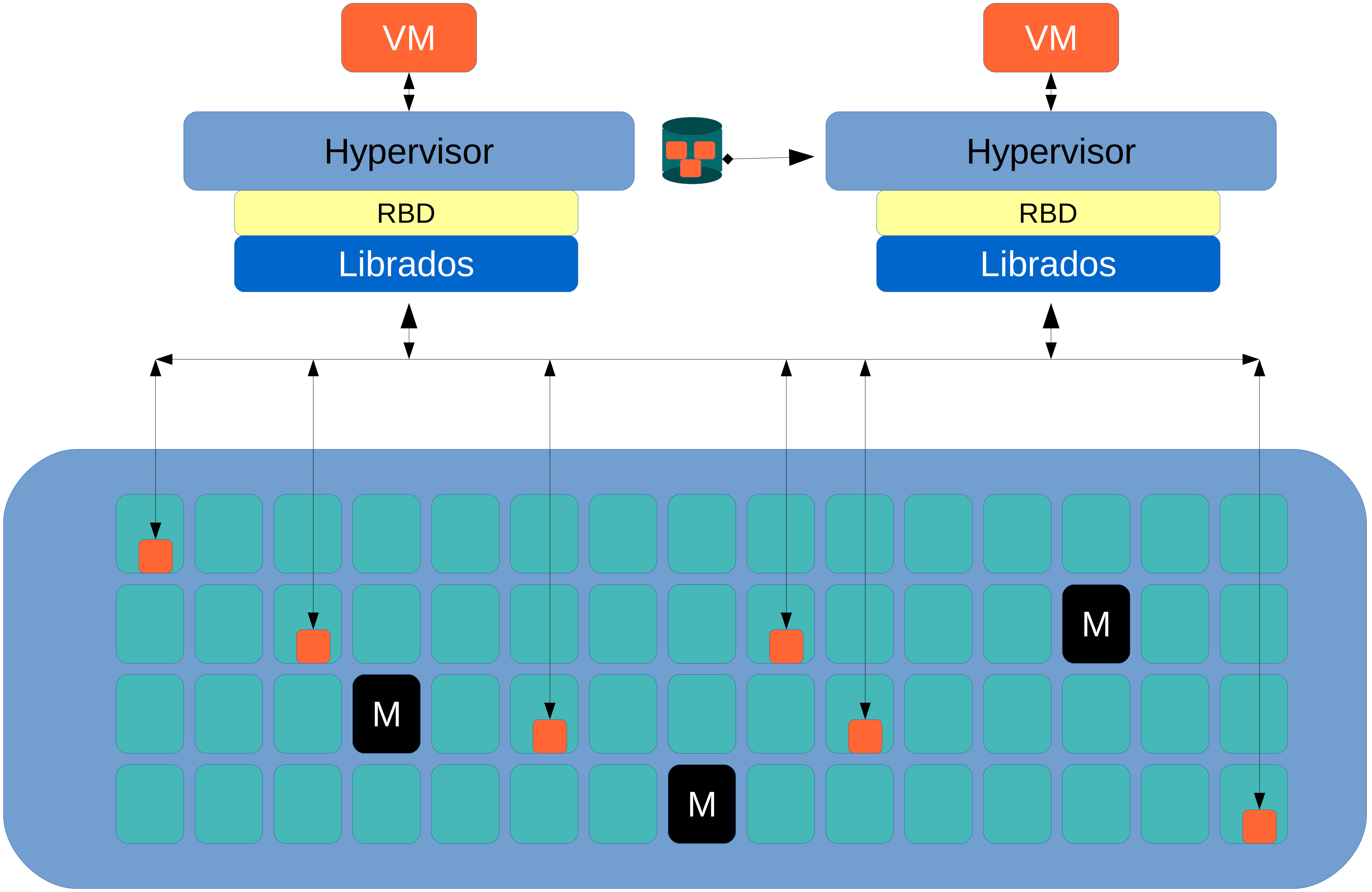
# CEPH : Multisite Object Store



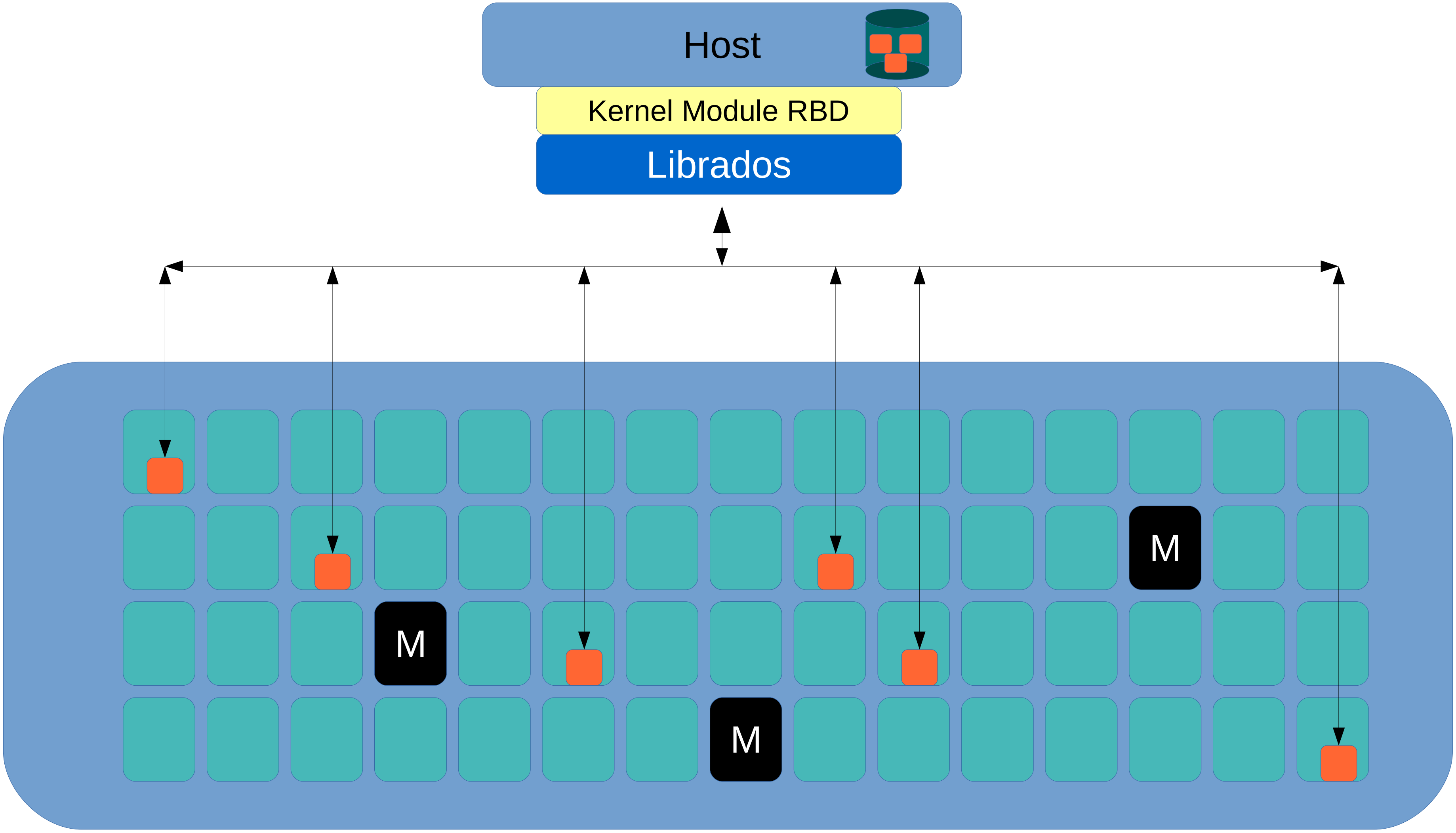
# CEPH : RBD (RADOS Block Device) to Rados



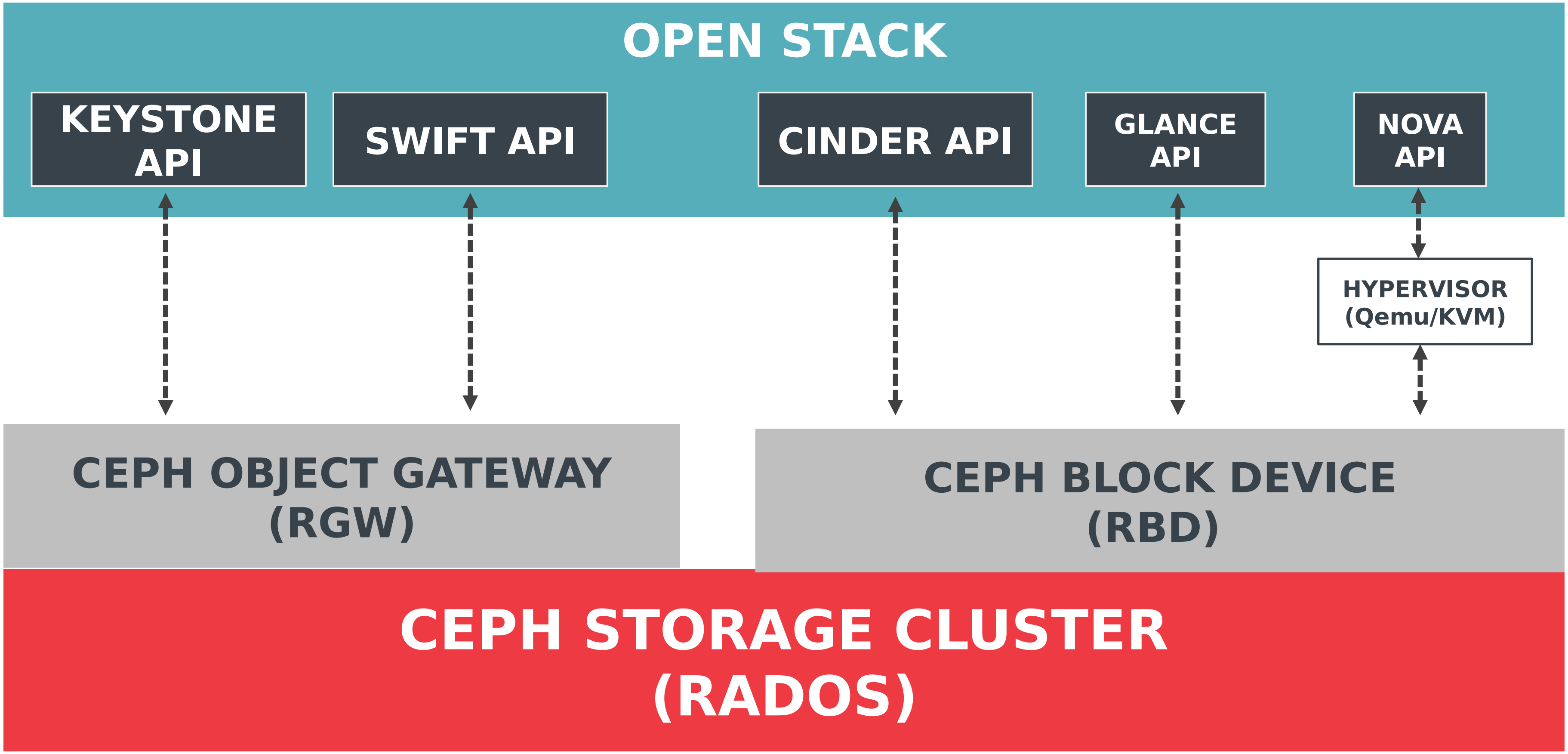
# CEPH : Migration of VM



# CEPH : RBD Device on Host



# CEPH integration with OPENSTACK

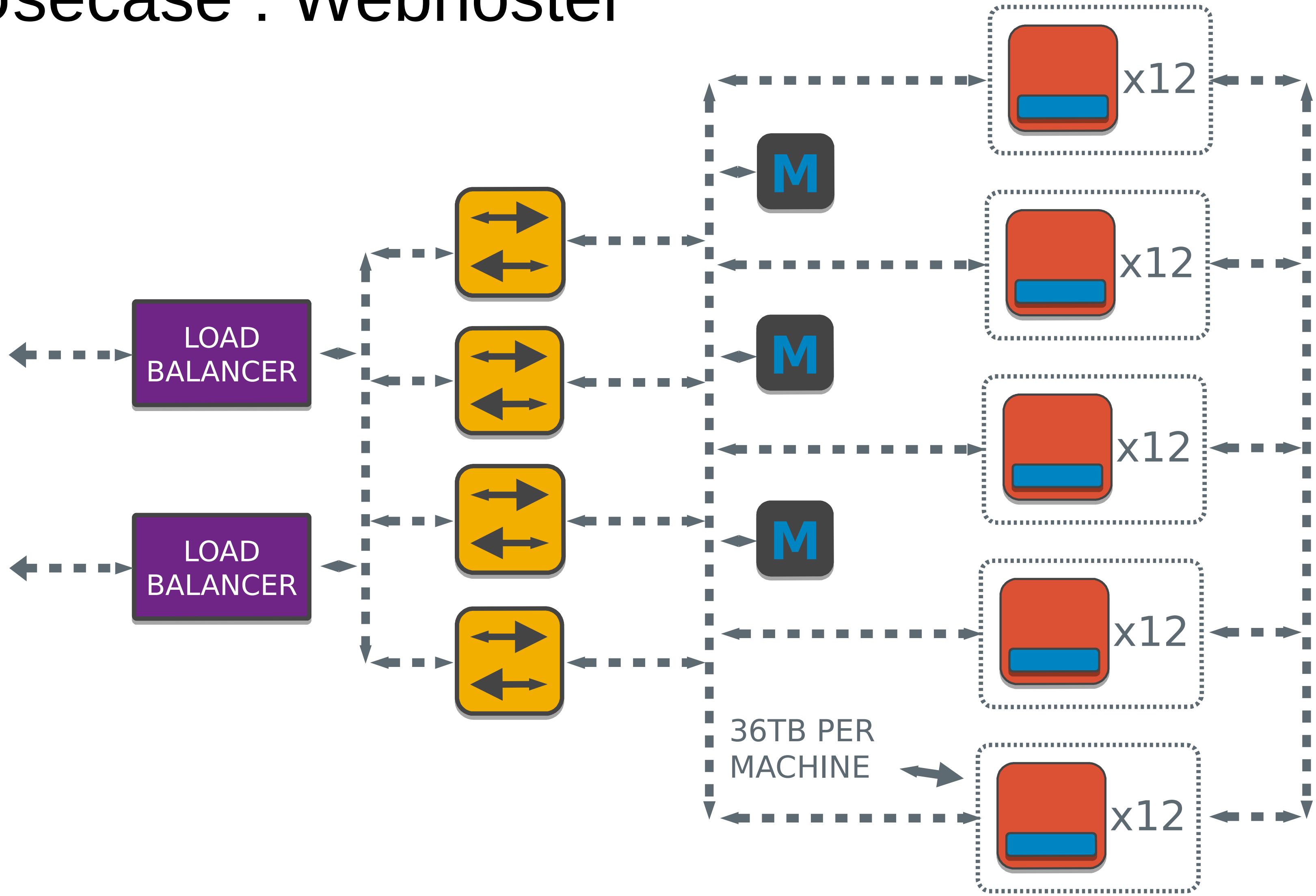


# Usecase : Webhoster

DreamHost®

- 15 years creating and deploying services
- Over 340,000 entrepreneur and developer customers
- Open source obsessed
  - Hosting over 500,000 WordPress sites
  - Contributing Ceph, Ceilometer, Akanda
  - OpenStack innovator & contributor

# Usecase : Webhoster



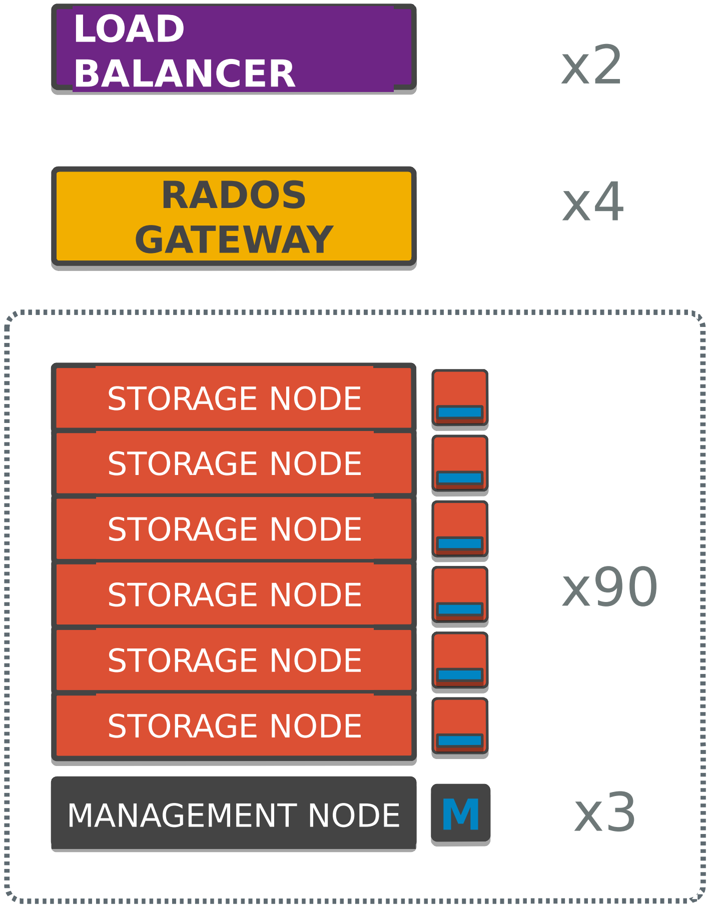

# Usecase : Webhoster

**STORAGE NODE**

Dell PowerEdge R515
6 core AMD CPU, 32GB RAM
2x 300GB SAS drives (OS)
12x 3TB SATA drives
2x 10GbE, 1x 1GbE, IPMI

**MANAGEMENT NODE**

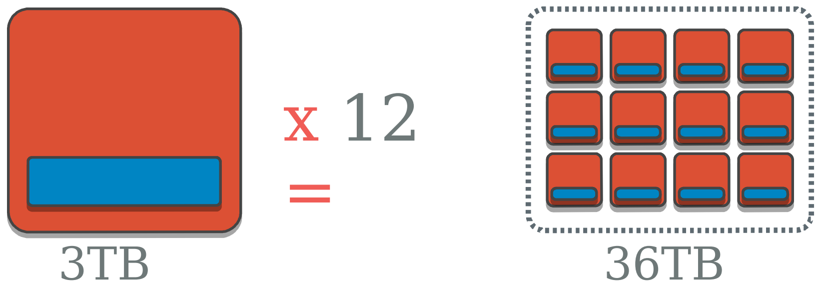
Dell PowerEdge R415
2x 1TB SATA
1x 10GbE



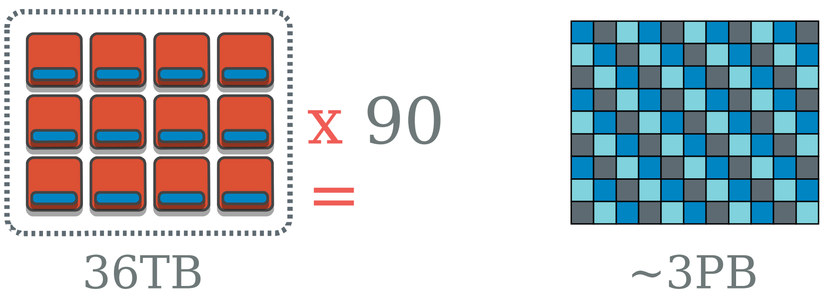


Andreas Koll

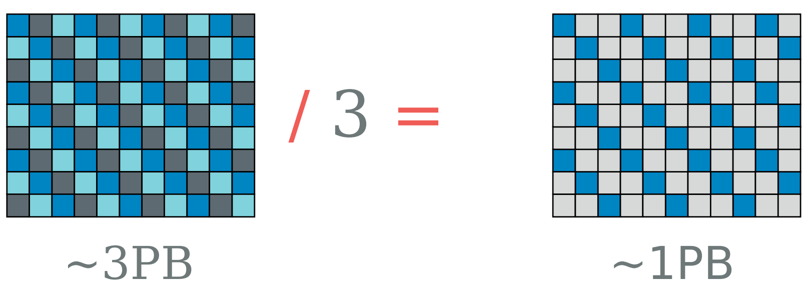
# Usecase : Webhoster




3TB per OSD
x 12 OSDs per node
= 36TB per node




36TB per node
x 90 nodes
= ~3PB total capacity




~3PB total capacity
/ 3 replicas per object
= ~1PB usable capacity



Andreas Koll

redhat.

# RED HAT STORAGE FUTURE WORKLOADS

# USE CASES: TODAY AND FUTURE

| ANALYTICS | CURRENT USE CASES | TARGET USE CASES |
| --- | --- | --- |
| | **Big Data analytics**<br>• Storage plug-in for Hortonworks Data Platform | **Big Data analytics**<br>• Persistent back-end for Spark |
| | **Machine data analytics**<br>• Online cold storage for IT operations data with Splunk | **Machine data analytics**<br>• Storage for ELK, Solr |

# USE CASES: TODAY AND FUTURE

OPENSTACK

## CURRENT USE CASES

### Virtual machine storage

- Virtual machine volume storage with Cinder, Nova and Glance

### Object storage for tenant applications

- Swift-compatible storage for cloud applications

## TARGET USE CASES

### Database storage

- Storage for relational databases with Trove

### Storage back-end for Manila

- Shared file system-as-a-service for tenants

# USE CASES: TODAY AND FUTURE

## ENTERPRISE SHARING

| | CURRENT USE CASES | TARGET USE CASES |
|---|---|---|
| ENTERPRISE SHARING | **Scale out file store**<br>• Storage for active archives, media streaming, content repositories, VM images, and general-purpose file shares | **Compliant archives**<br>• Scalable, cost-effective storage for compliance and regulatory needs |
| | **Enterprise file sync and share**<br>• Storage for Dropbox-style enterprise shared folders | **File services for containers**<br>• File storage services for containers and pods |

# USE CASES: TODAY AND FUTURE

**CLOUD STORAGE**

| | CURRENT USE CASES | TARGET USE CASES |
|---|---|---|
| CLOUD STORAGE | **S3-based object storage for apps**<br>• Cost-effective, S3-compatible, on-premise object store | **Enterprise sync and share**<br>• Storage for shared folders (object backend) |

# USE CASES: TODAY AND FUTURE

## ENTERPRISE VIRTUALIZATION

### CURRENT USE CASES

**Conventional virtualization storage**

- Integrated storage for Red Hat Enterprise Virtualization
- (with separate compute and storage clusters)

### TARGET USE CASES

**Hyper-converged architectures**

- Hyper-converged architectures

# THANK YOU