Software Defined Storage with Gluster

HV Open

Patrick Ladd
Technical Account Manager, FSI
January 9th 2019

pladd@redhat.com https://people.redhat.com/pladd

Agenda

- Software Defined Storage
  - What is it?
  - Why?
- Red Hat Gluster Storage (RHGS)
  - Concepts
  - Architecture
  - Features
- Applications
  - General Applications
  - Container Native Storage
  - Red Hat Storage One
  - Sample Customers

Software Defined Storage

The Data Explosion

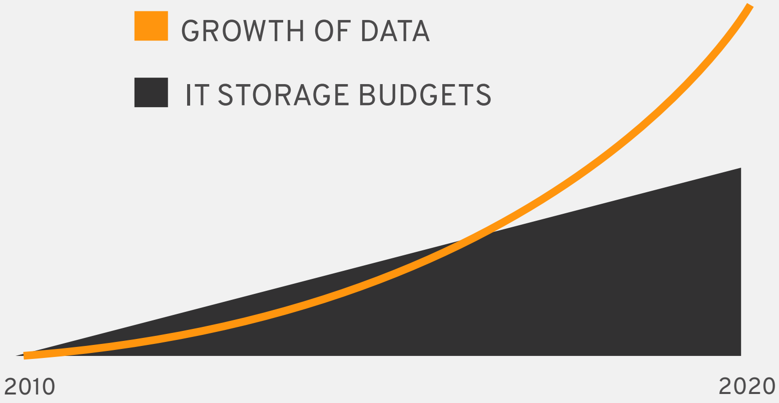
WEB, MOBILE, SOCIAL MEDIA, CLOUD
Our digital assets have grown exponentially due to web scale services like Facebook, Flickr, Snapchat, YouTube, and Netflix.

MEDIA AND ENTERTAINMENT INDUSTRIES
A staggering amount of content is created during today’s optimized production processes.

VIDEO ON-DEMAND SERVICES
Rapid growth of video on-demand has culminated in 50% of households using this service.

MEDICAL INDUSTRY
Medical imaging needs are vast, and regulatory requirements can be demanding.

The Data Storage Shortfall



Data stores are growing exponentially, while IT budgets are not

HDDs are becoming more dense, but $/GB decline is slowing

Software and hardware advances are needed to close the gap

The Datacenter is Changing

DEVELOPMENT MODEL	APPLICATION ARCHITECTURE	DEPLOYMENT AND PACKAGING	APPLICATION INFRASTRUCTURE	STORAGE
Waterfall	Monolithic	Bare Metal	Data Center	Scale Up
Agile	N-Tier	Virtual Services	Hosted	Scale Out
DEVOPS	MICROSERVICES	CONTAINERS	HYBRID CLOUD	SOFTWARE-DEFINED STORAGE

What is Software Defined Storage?

SERVER-BASED

CENTRALIZED CONTROL

OPEN ECOSYSTEM

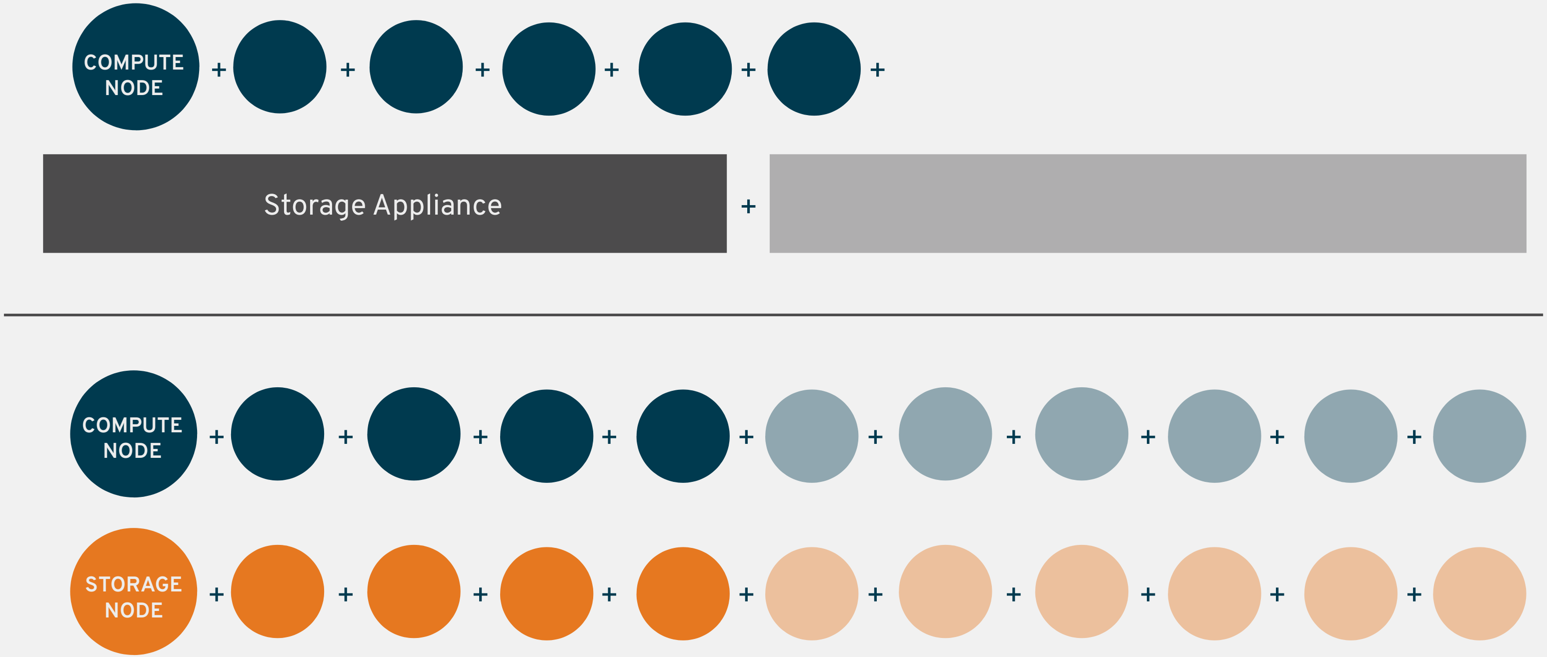
Industry Standard Hardware

Standardization makes storage more convenient

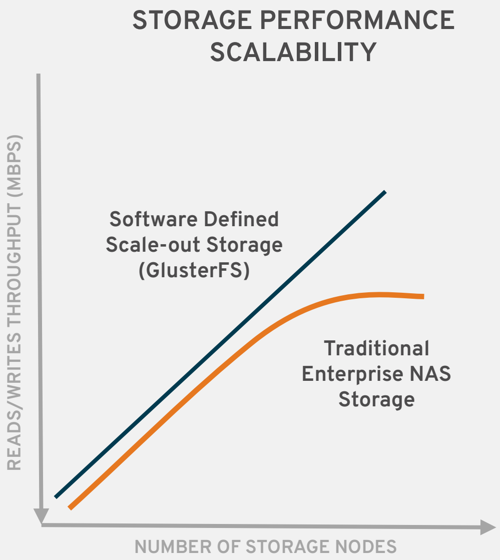
Customers can build clusters using standard hardware from existing vendors that’s perfect for their workload.

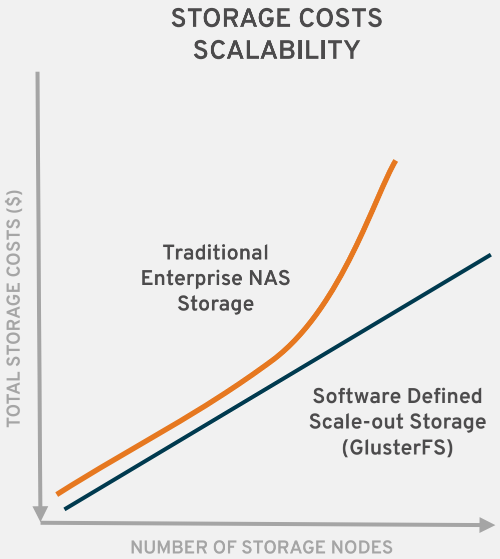
- Clusters can be performance-optimized, capacity-optimized, or throughput-optimized
- Need capacity? Add more disks. Too slow? Add more servers.
- Clusters can become larger or smaller with no downtime

Virtualized Storage Scales Better



Comparing Throughput and Costs at Scale



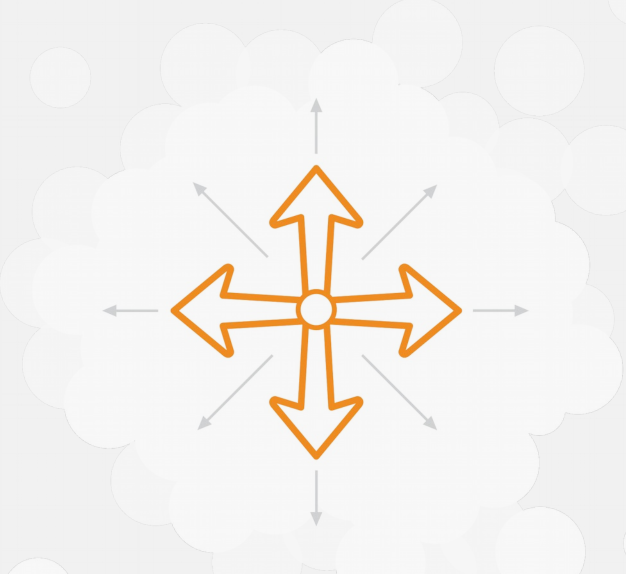


The Robustness of Software

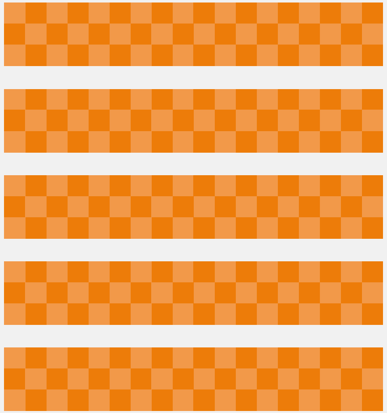
Software can do things hardware can’t

Storage services based on software are more flexible than hardware-based implementations

- Can be deployed on bare metal, inside containers, inside VMs, or in the public cloud
- Can deploy on a single server, or thousands, and can be upgraded and reconfigured on the fly
- Grows and shrinks programmatically to meet changing demands

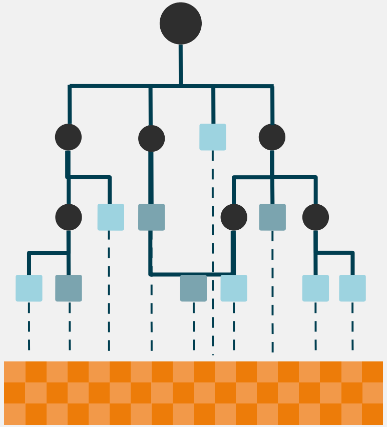


Different Kinds of Storage



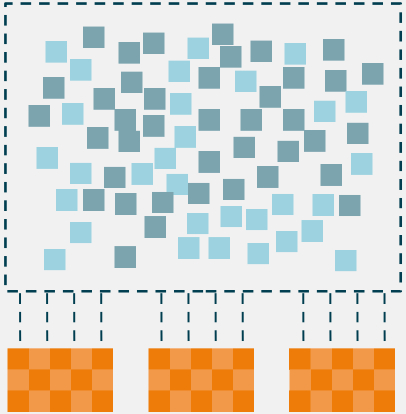
BLOCK STORAGE

Data as sequential uniform **blocks**



FILE STORAGE

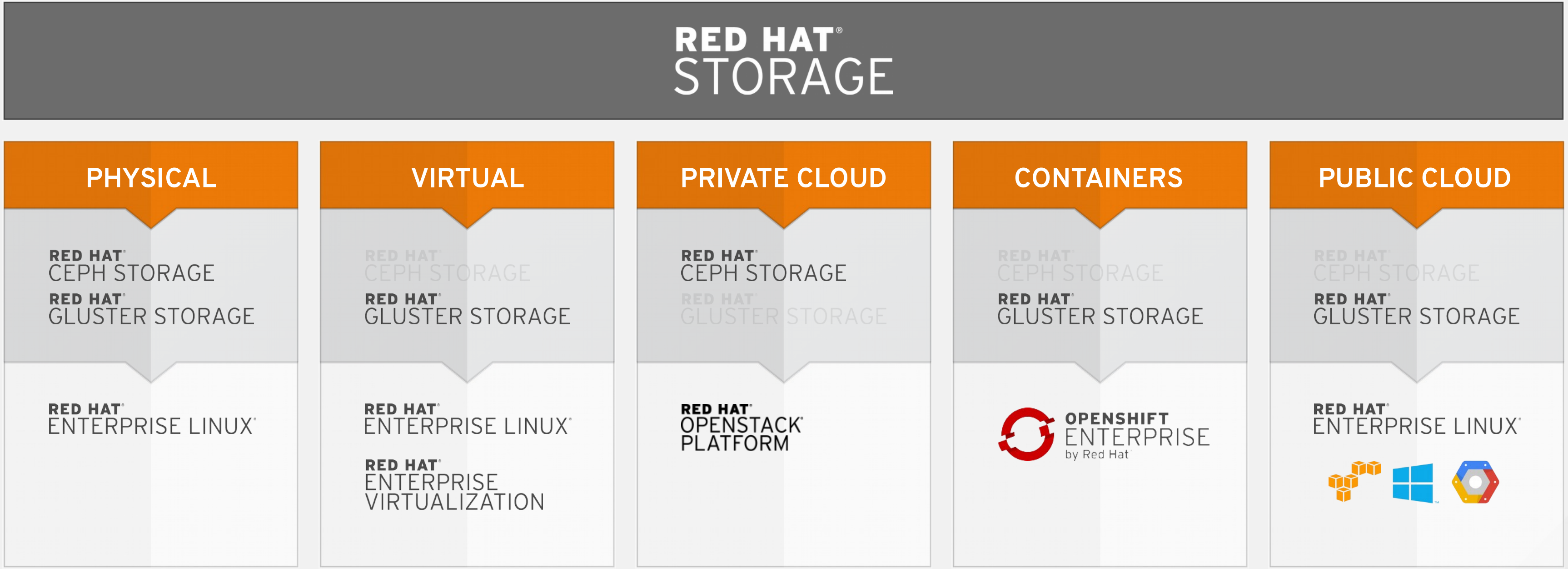
Data as buckets of hierarchical **folders and files**



OBJECT STORAGE

Data as a predictably mapped, loosely structured cluster of **objects**

How Storage Fits



RED HAT® STORAGE

PHYSICAL

RED HAT® CEPH STORAGE

RED HAT® GLUSTER STORAGE

RED HAT® ENTERPRISE LINUX®

VIRTUAL

RED HAT® CEPH STORAGE

RED HAT® GLUSTER STORAGE

RED HAT® ENTERPRISE LINUX®

RED HAT® ENTERPRISE VIRTUALIZATION

PRIVATE CLOUD

RED HAT® CEPH STORAGE

RED HAT® GLUSTER STORAGE

RED HAT® OPENSTACK® PLATFORM

CONTAINERS

RED HAT® CEPH STORAGE

RED HAT® GLUSTER STORAGE

OPENSHIFT ENTERPRISE by Red Hat

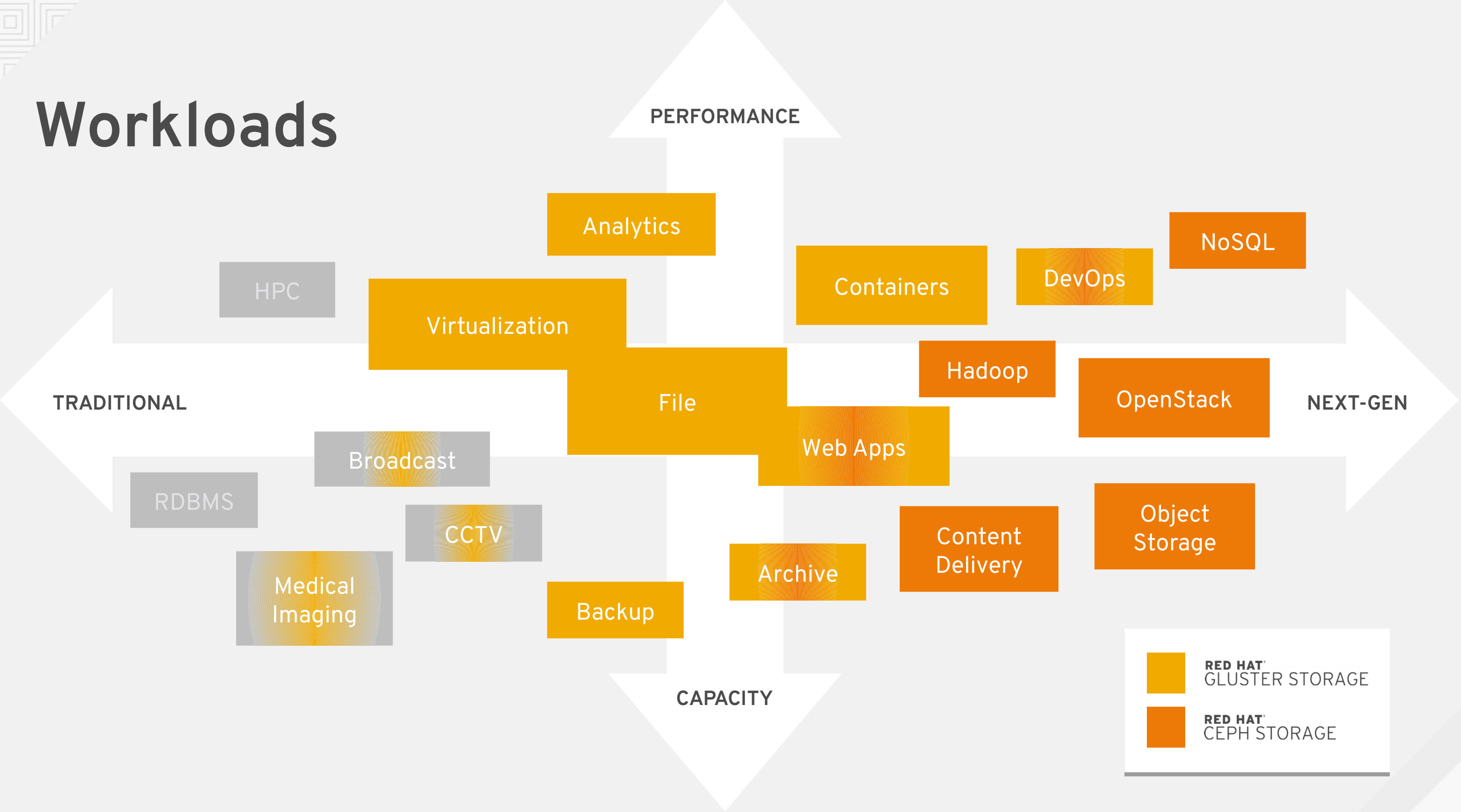
PUBLIC CLOUD

RED HAT® CEPH STORAGE

RED HAT® GLUSTER STORAGE

RED HAT® ENTERPRISE LINUX®

Workloads



PERFORMANCE

CAPACITY

TRADITIONAL

NEXT-GEN

Analytics

HPC

Virtualization

Containers

DevOps

NoSQL

Hadoop

File

OpenStack

Broadcast

Web Apps

RDBMS

CCTV

Object Storage

Content Delivery

Archive

Medical Imaging

Backup

🟨 RED HAT® GLUSTER STORAGE

🟧 RED HAT® CEPH STORAGE

Red Hat Gluster Storage

Red Hat Gluster Storage

Half the price for comparable features & greater flexibility

RED HAT® GLUSTER STORAGE

KEY STRENGTHS

- Straightforward, adaptable, embeddable architecture
- Competitive TCO
- Experience of large-scale production customers
- Thriving community

Open source, distributed, scalable, software-defined storage with enterprise-grade capabilities

Security
In-flight encryption
At-rest encryption
SELinux enforcing

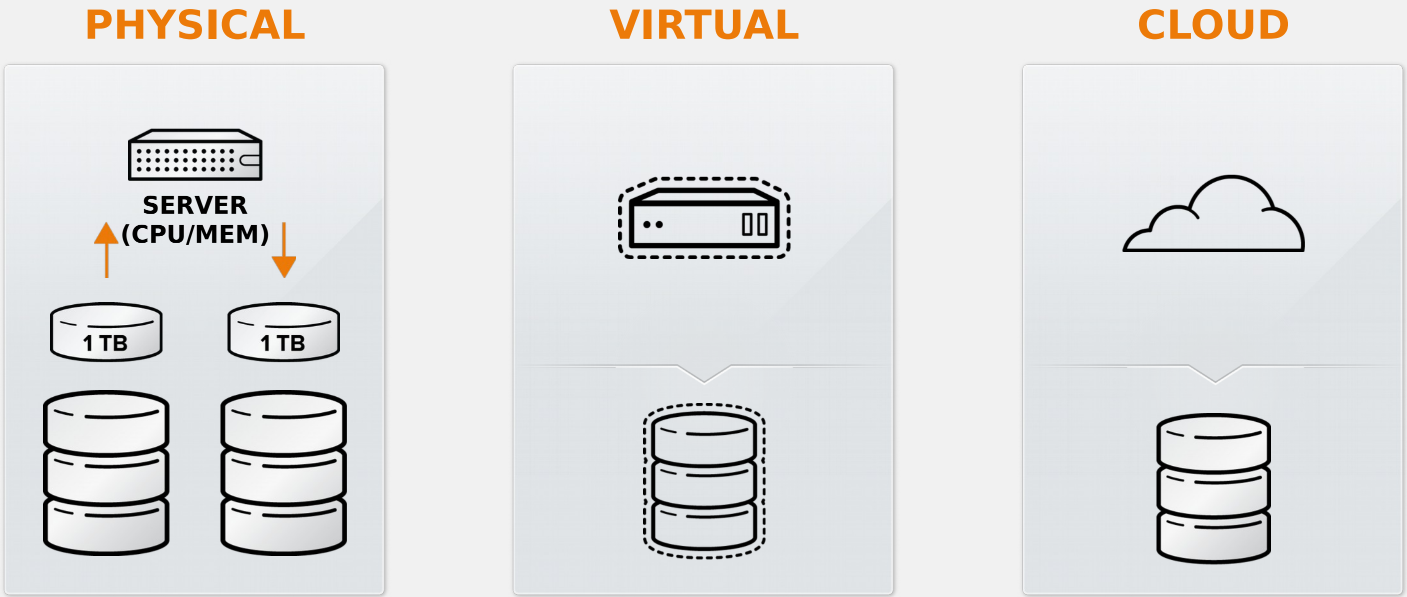
Data Services
NFS/SMB access
Snapshots
Clones
Quotas
Mirroring
Tiering

Data Integrity
Erasure coding
Replication
Geo-replication
Self-healing
Bit-rot detection

Architecture & Terms

What is a system?

Can be physical, virtual or cloud



PHYSICAL

SERVER
(CPU/MEM)

1 TB

1 TB

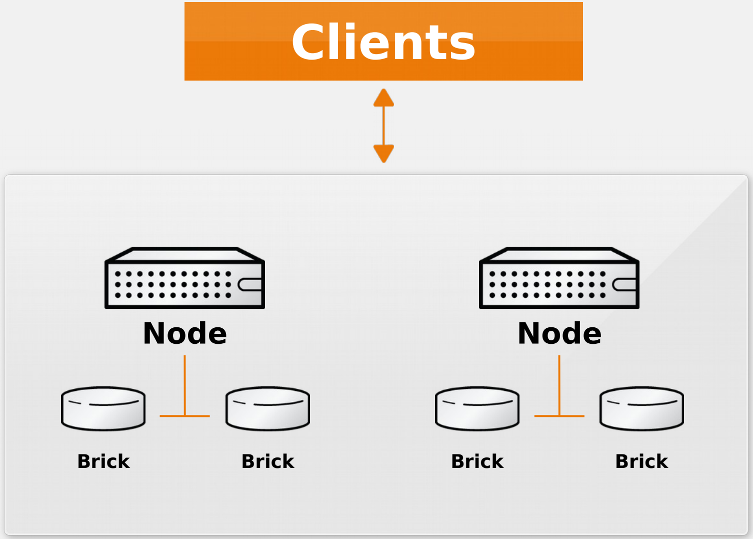
VIRTUAL

CLOUD

Volumes – Nodes – Bricks

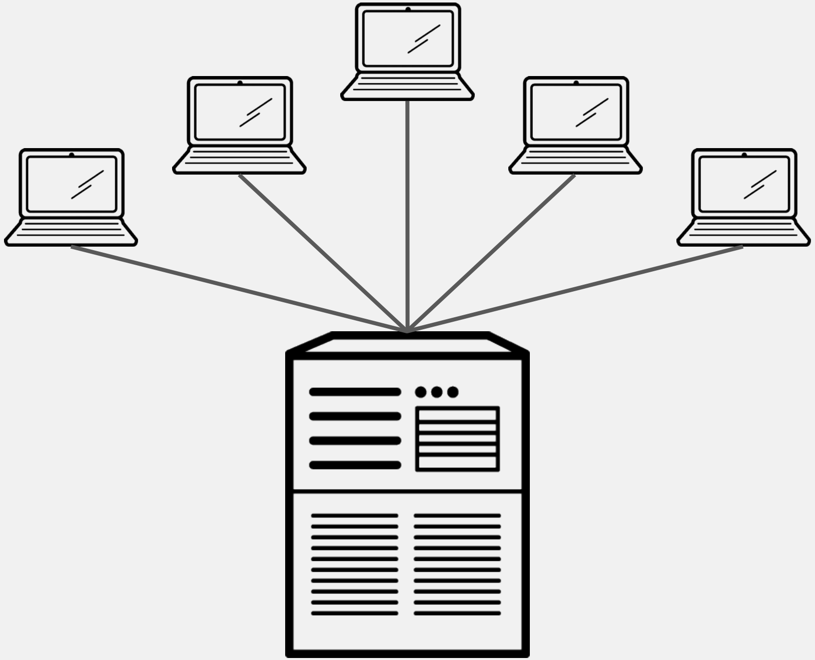
Bricks taken from multiple hosts become one addressable unit

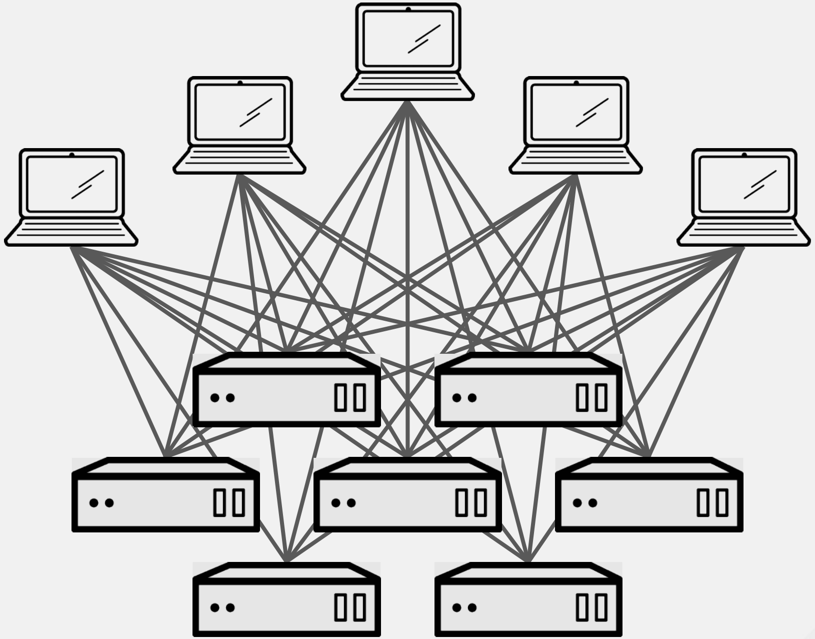
- High availability as needed
- Load balanced data
- Managed by Gluster



How Does Gluster Do It?

The Data Placement Challenge





The Data Placement Challenge

Imagine a storage pool of thousands of data volumes

- How can we store data reproducibly?
- What happens if we add disks?
- What happens if a disk fails?
- How can we ensure data is written evenly across all volumes?

Elastic Hashing Algorithm

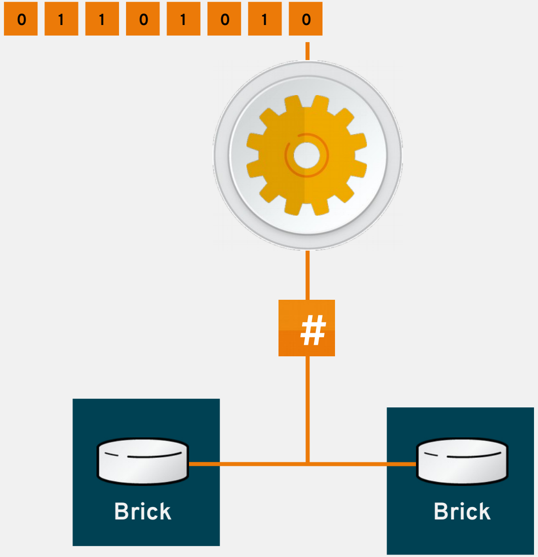
No metadata servers = No single point of failure

Elastic Hashing

- Enables petabyte scale
- Files assigned to virtual volumes
- Virtual volumes assigned to multiple bricks
- Volumes easily reassigned on-the-fly

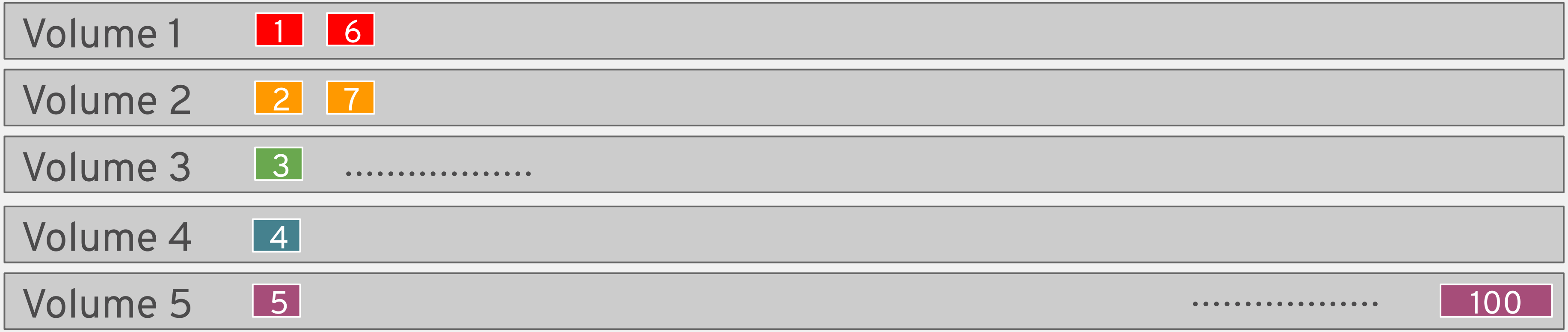
Location Hashed on Filename

- No performance bottleneck
- Eliminates risk scenarios



Simple Approach – Round Robin

How to store 100 objects on 5 disks



Volume 1 1 6

Volume 2 2 7

Volume 3 3 ....

Volume 4 4

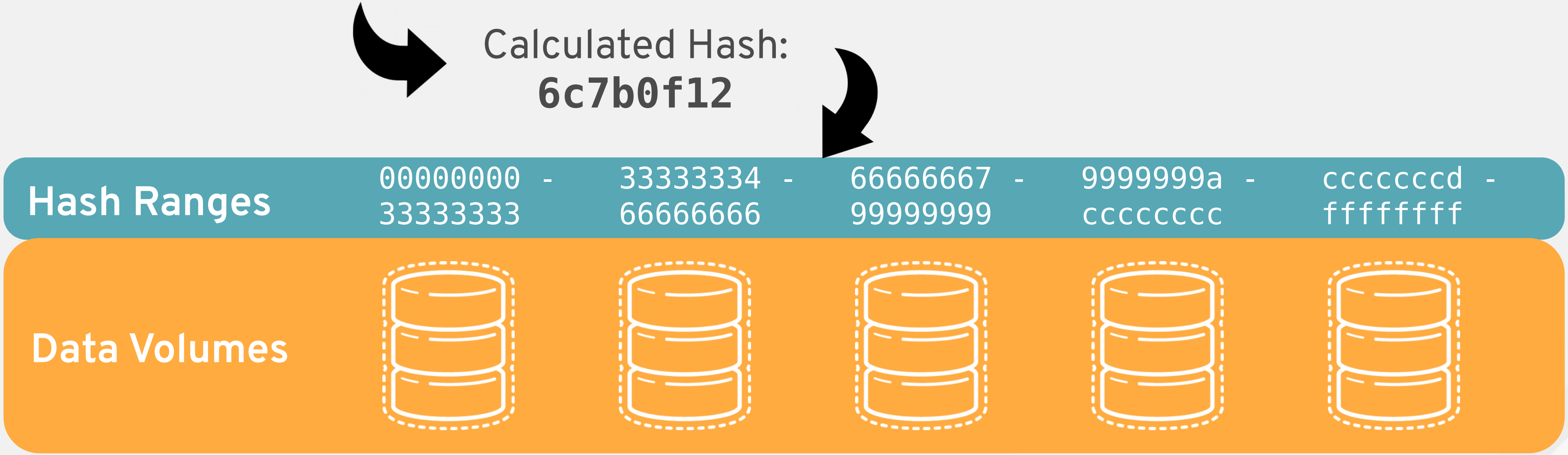
Volume 5 5 100

Round Robin works efficiently, but has a crucial bottleneck,

central metdata

Hash-based Data Placement

Identifying Key (i.e., file name)



Calculated Hash:
6c7b0f12

Hash Ranges	00000000 - 33333333	33333334 - 66666666	66666667 - 99999999	9999999a - cccccccc	cccccccd - ffffffff
Data Volumes					

Hash-based Data Placement

- Clients and daemons both use the hash algorithm to compute the object location (reading and writing)
- There is no centralized lookup table
- Enables massive scaling by cleanly distributing the work to all the clients and daemons
- Replication logic ensures data resilience

How Do We Maintain the Hash Tables?

Hash Ranges	00000000 - 33333333	33333334 - 66666666	66666667 - 99999999	9999999a - cccccccc	cccccccd - ffffffff
Data Volumes					

Hash Ranges	00000000 - 55555555	55555556 - aaaaaaaa	aaaaaaab - ffffffff
Data Volumes			

Modulo Division

Let’s calculate where we store objects

- Dividing our hash by 5 data volumes will always yield a remainder between 0 and 4 (range equal to number of data volumes)
- Pseudo-random hash values will result in statistically even distribution of remainders
- Calculating hashes and modules are lightweight computational tasks

Some examples (decimal numbers for simplicity):

Object #36:

36 mod 5 = 1

so we put object #36 on data volume #1

Object #7:

7 mod 5 = 2

so we put object #7 on data volume #2

Object #133:

133 mod 5 = 3

so we put object #133 on data volume #3

Distributing Data by Modulo

The actual distribution

Object ID	1	2	3	4	5	6	7	8	9	10	11	12	13
modulo(5)	1	2	3	4	0	1	2	3	4	0	1	2	3

So it seems we’ve found a solution to evenly distribute data and to easily retrieve it, BUT…

Distributing Data by Modulo

What happens if we add a disk?

With 5 data volumes we get this distribution:

Object ID	1	2	3	4	5	6	7	8	9	10	11	12	13
modulo(5)	1	2	3	4	0	1	2	3	4	0	1	2	3

With 6 data volumes we get this distribution:

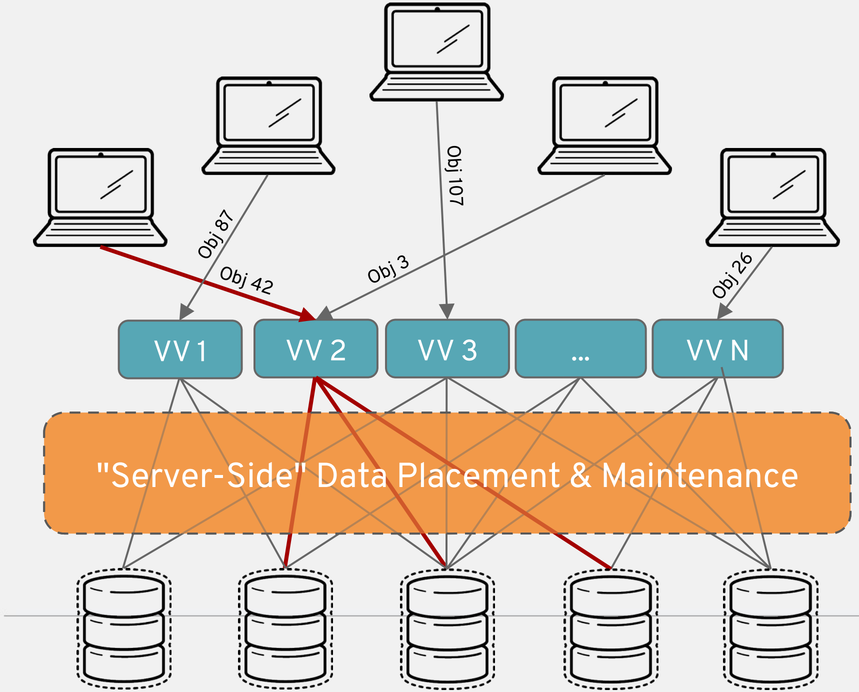
Object ID	1	2	3	4	5	6	7	8	9	10	11	12	13
modulo(6)	1	2	3	4	5	0	1	2	3	4	5	0	1

Every object with an ID > 4 needs to be relocated!

Layered Features

Virtual Data Volumes

Assigning data to virtual volumes allows us to scale data volumes independently of data placement

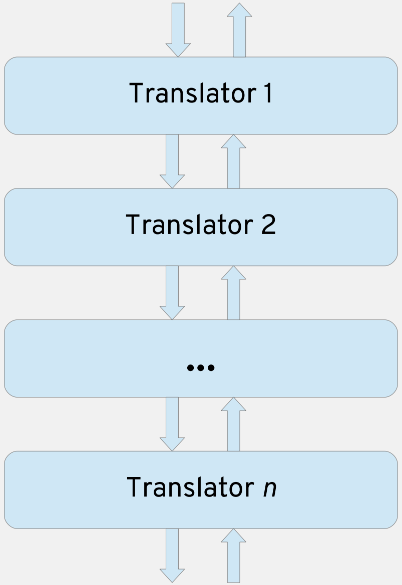


- Virtual volumes are an abstract concept. They work as a layer between objects and data volumes.
- Since they are, unlike data volumes, constant in their number, we can rely on them as the divider for the modulo() operation.
- Allows flexibility to handle replication or other data protection

"Server-Side" Data Placement & Maintenance

Translation layers handle:

- Data resilience scheme is maintained (replication, erasure coding)
- Metadata is stored and tracked with the object
- Dynamic mapping from virtual volumes to data volumes
- Heal, Rebalance, Bitrot Detection, Geo-Replication, …
- Data translation hierarchy (protocols, encryption, performance, …)
- Health monitoring, alerting, and response



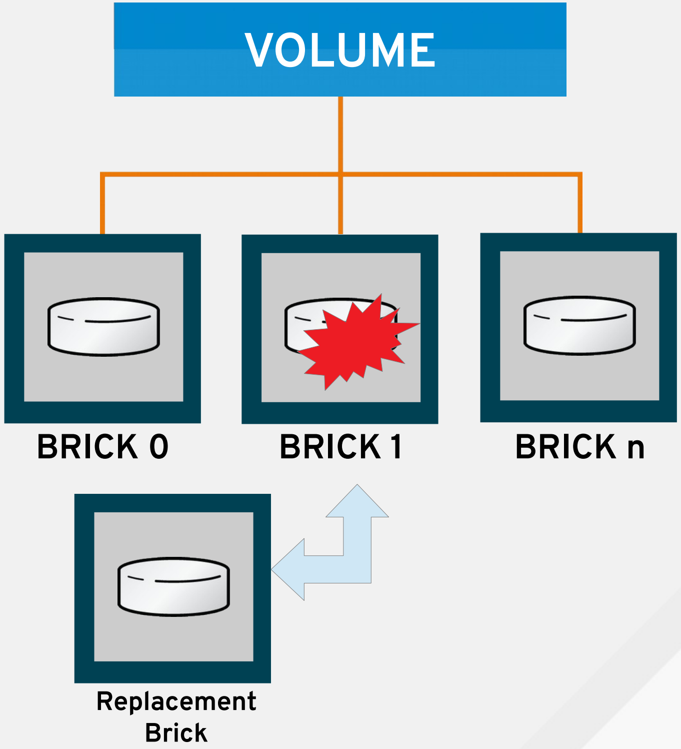
Self Healing

Automatic Repair

- Automatic Repair of Files
  - As they are accessed
  - Periodic via Daemon

Scenarios:

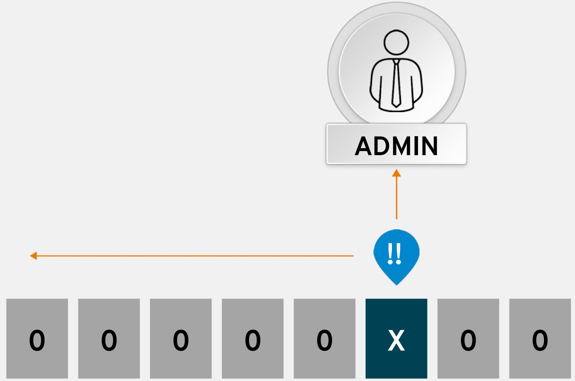
- Node offline
  - Bricks on node need to be caught up to current
- Node or brick loss
  - New brick needs to be completely rebuilt



Bit Rot Detection

Detecting silent data corruption

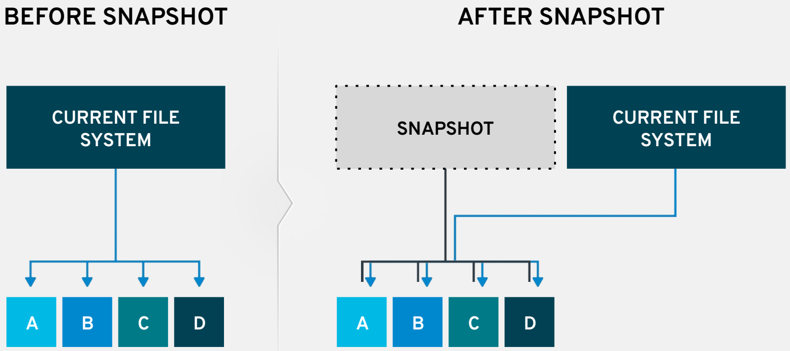
- Scans data periodically for bit rot
- Check sums are computed when files are accessed and compared against previously stored values
- On mismatch, an error is logged for the storage admin



Snapshots

Storing point in time state of the cluster

- Volume level, ability to create, list, restore, and delete
- LVM2 based, operates only on thin-provisioned volumes
- User serviceable snapshots
- Crash consistent image

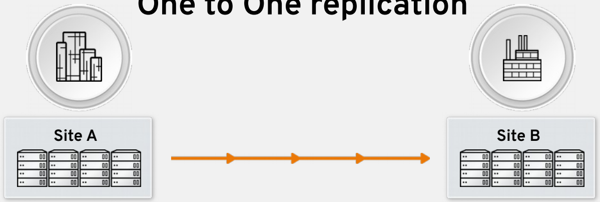


Geo Replication

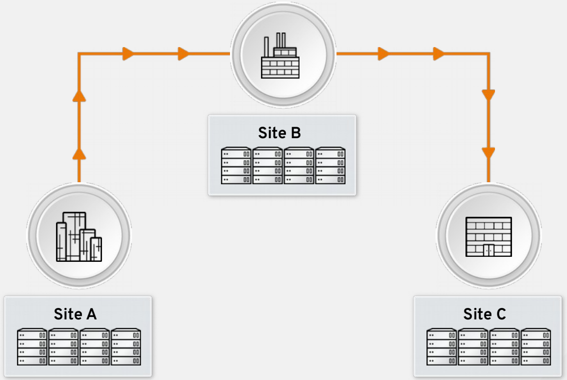
Multi-site content distribution

- Asynchronous across LAN, WAN, or Internet
- Performance considerations:
  - Parallel transfers
  - Efficient source scanning
  - Pipelined and batched
  - File type/layout agnostic
- Continuous and incremental
- Failover and Fallback
- Configurations:
  - One-to-one or one-to-many
  - Cascading

One to One replication

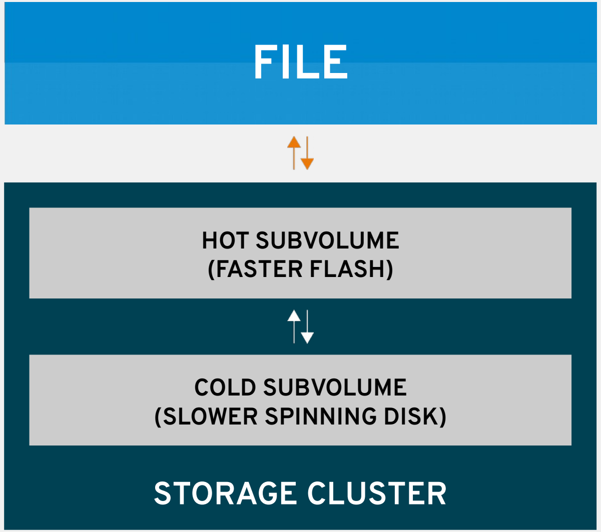


Cascading replication



Tiering

- Automated promotion and demotion of data between “hot” and “cold” sub volumes
- Based on frequency of access
- Cost-effective flash acceleration



Quotas

Volume and Directory Level Support

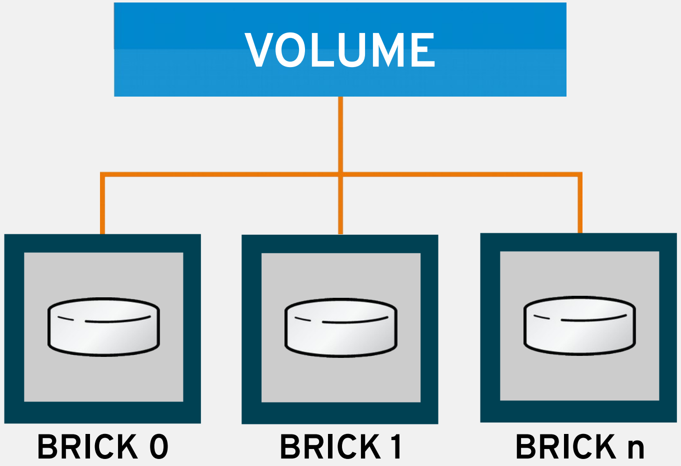
- Control disk utilization at both directory and volume level

Quota Limits

- Two levels of quota limits: Soft (default) and hard
- Warning messages issued on reaching soft quota limit
- Write failures with EDQUAT message after hard limit is reached

Global vs. Local Limits

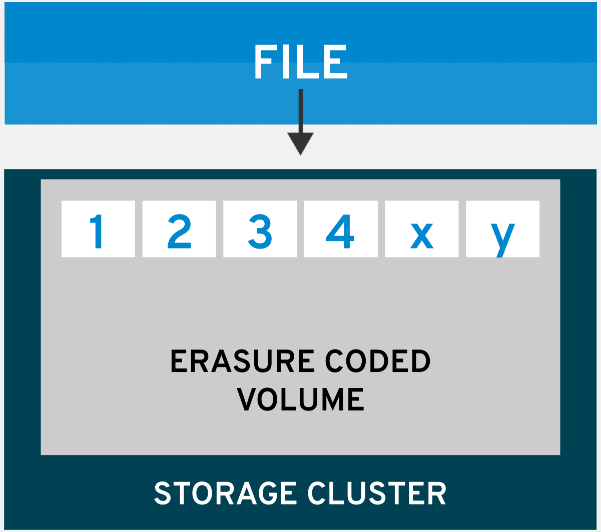
- Quota is global (per volume)
- Files are psuedo-randomly distributed across bricks



Erasure Coding

Storing data with less hardware

- Reconstruct corrupted or lost data
- Eliminates the need for RAID
- Consumes far less space than replication
- Appropriate for capacity-optimized use cases



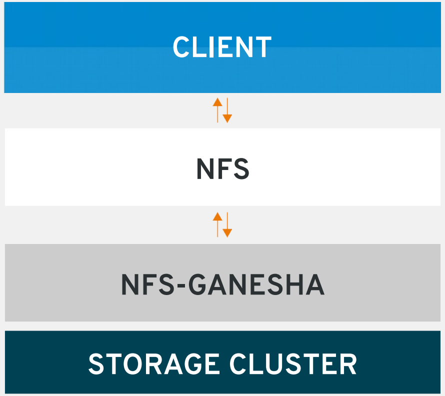
Security

Scalable NFSvs Client

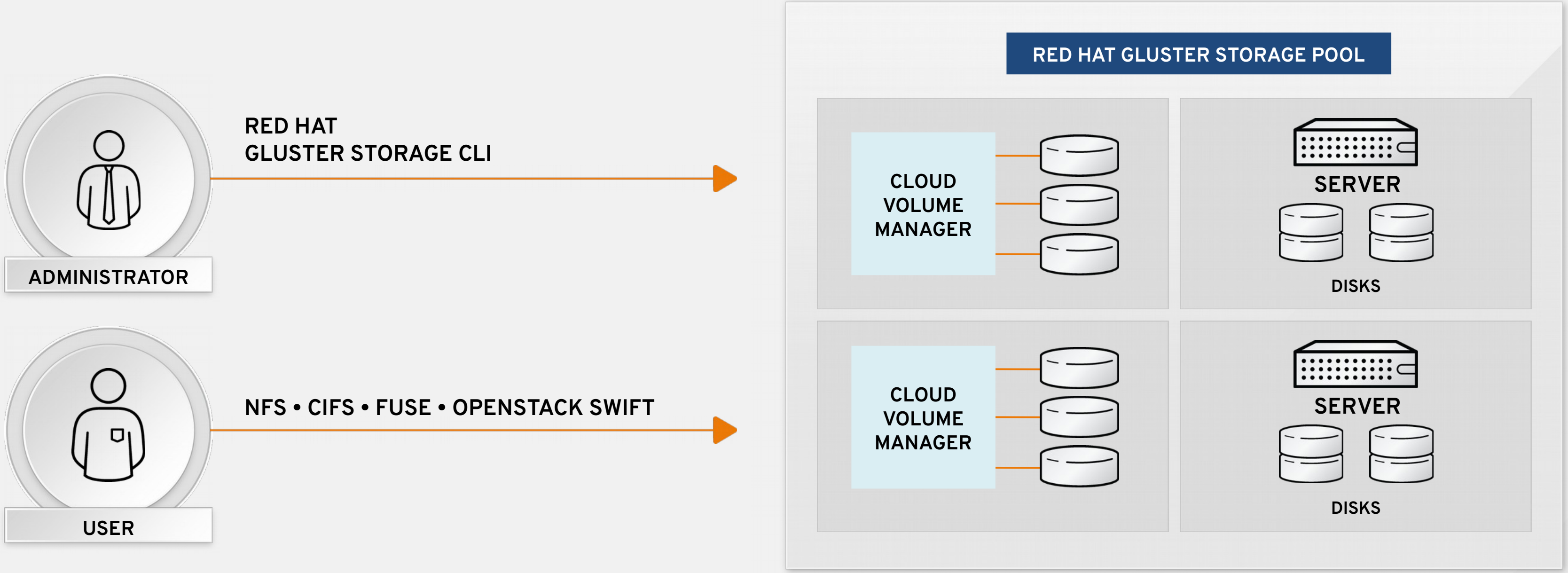
- Client access with simplified failover and failback in the case of a node or network failure
- ACLs for additional security
- Kerberos authentication
- Dynamic export management

Network Encyption

- TLS/SSL for authentication and authorization
- Encryption in transit and transparent encryption (at rest)
- I/O encryption and management encryption



Multi Protocol Support



ADMINISTRATOR

RED HAT
GLUSTER STORAGE CLI

USER

NFS • CIFS • FUSE • OPENSTACK SWIFT

RED HAT GLUSTER STORAGE POOL

Applications

A SIX-NODE CLUSTER CAN PROCESS...

JPEG Web Image Files (32KB)

72x 7.2K HDD

1700 JPEGs per second

or

Optimized 72x 7.2K HDD

12,000 JPEGs per second

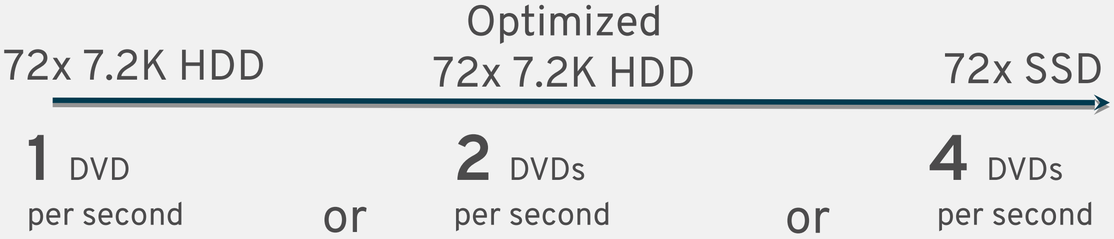
or

72x SSD

23,000 JPEGs per second

OR...

DVD
Movie Files
(4GB)



72x 7.2K HDD

1 DVD
per second

or

Optimized
72x 7.2K HDD

2 DVDs
per second

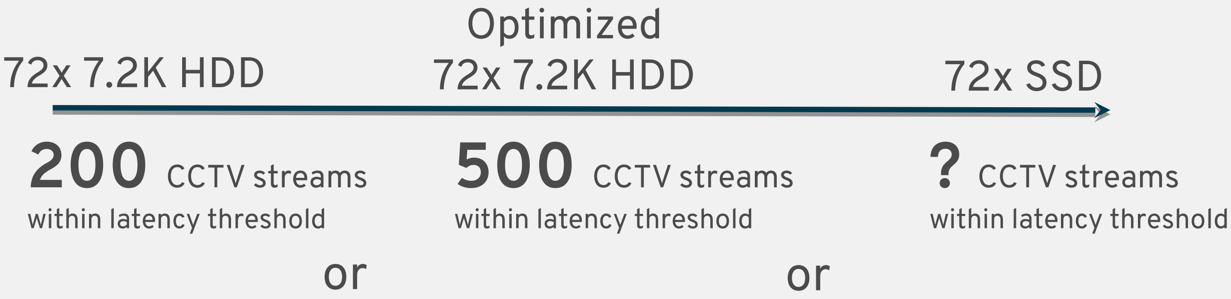
or

72x SSD

4 DVDs
per second

OR...

High-Def
CCTV Camera
Recording Streams



72x 7.2K HDD

200 CCTV streams
within latency threshold

or

Optimized
72x 7.2K HDD

500 CCTV streams
within latency threshold

or

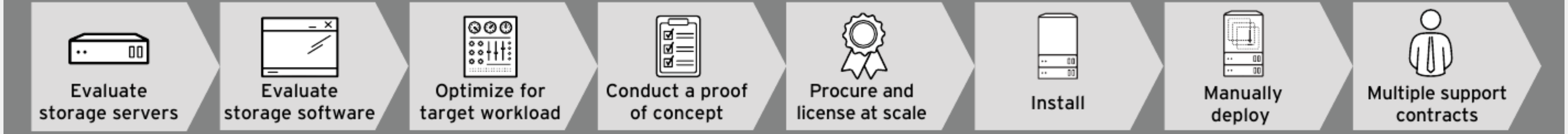
72x SSD

? CCTV streams
within latency threshold

Red Hat Storage One

Pre-configured Storage Hardware and Software

TRADITIONAL DIY SOFTWARE-DEFINED STORAGE DEPLOYMENT

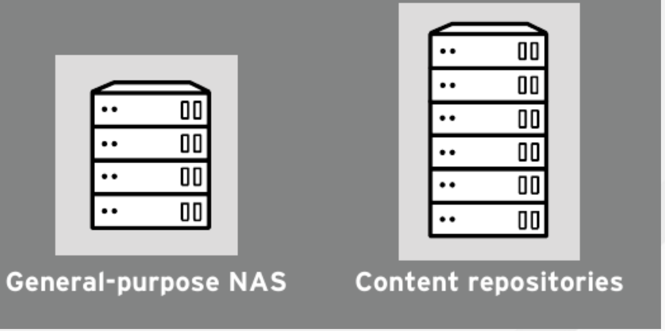


Evaluate storage servers → Evaluate storage software → Optimize for target workload → Conduct a proof of concept → Procure and license at scale → Install → Manually deploy → Multiple support contracts

OR...

RED HAT STORAGE ONE BY SUPERMICRO

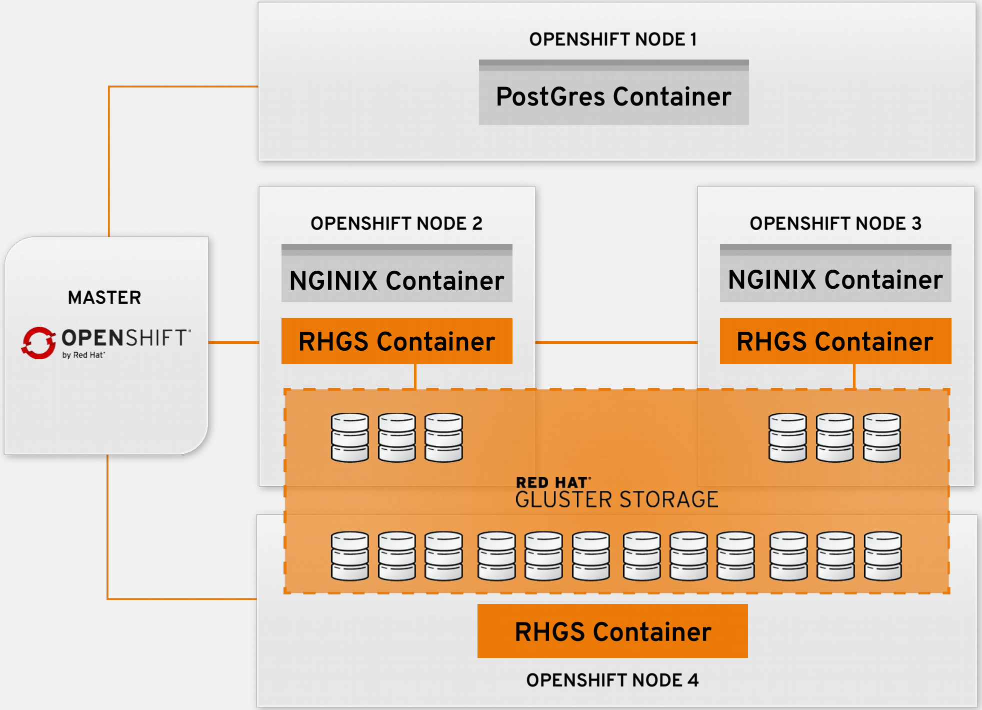
- Workload-optimized, tested, self-configuring, and ready in minutes
- Hundreds of terabytes to petaybtes of useable resilient Red Hat Gluster Storage
- Hardware, software, and support in a single Supermicro part number



General-purpose NAS

Content repositories

CONTAINER-NATIVE STORAGE



Lower TCO

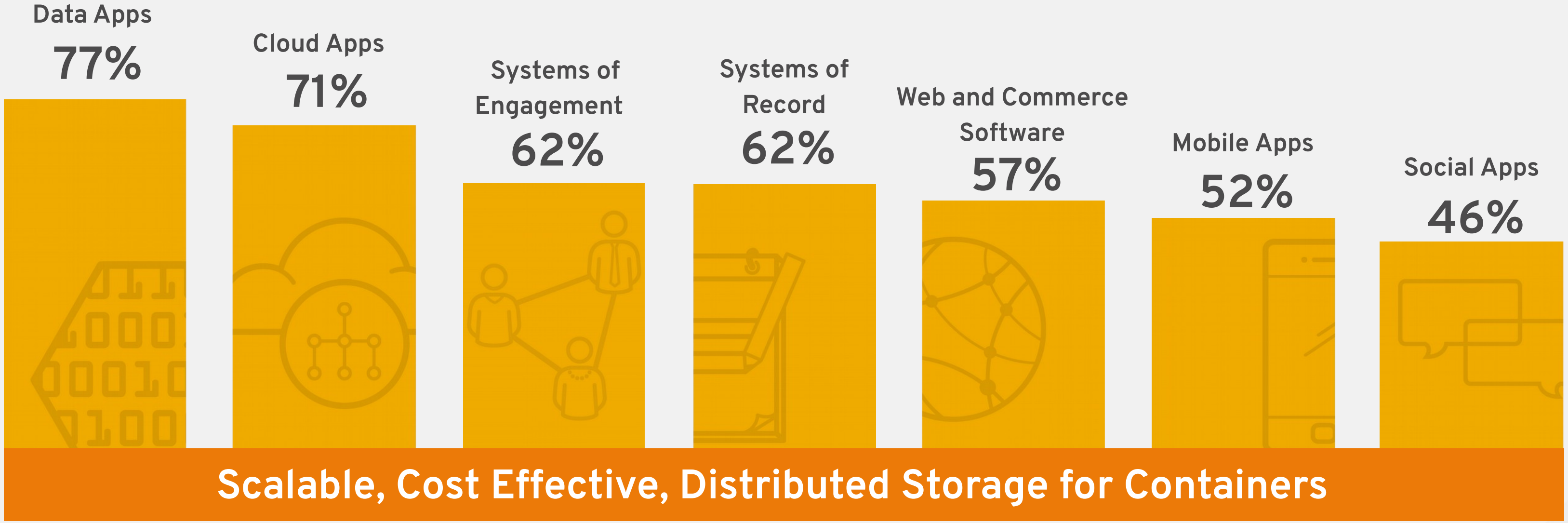
Unified Orchestration

Ease of Use

Greater control

WHY PERSISTENT STORAGE FOR CONTAINERS?

“For which workloads or application use cases have you used/do you anticipate to use containers?”



Base: 194 IT operations and development decision-makers at enterprise in APAC, EMEA, and North America
Source: A commissioned study conducted by Forrester Consulting on behalf of Red Hat, January 2015

Use Cases

IOPS OPTIMIZED

Use Case: MySQL

THROUGHPUT OPTIMIZED

Use Case: Rich Media

COST/ CAPACITY OPTIMIZED

Use Case: Active Archives

CISCO SanDisk SUPERMICRO QCT SAMSUNG intel

General SDS Hardware Performance & Sizing

Container Storage Center of Excellence

Joint Innovation With Partners

Performance and Sizing Guides

http://red.ht/2mg9kQ5

RED HAT STORAGE

GAIN A COMPETITIVE ADVANTAGE WITH FINANCIAL ANALYTICS

PENGUIN COMPUTING

INTRODUCTION

THE PROBLEM

http://red.ht/2k6uYcg

RED HAT STORAGE

PERFORMANCE AND SIZING GUIDE

Red Hat Gluster Storage on QCT servers

ABSTRACT

TABLE OF CONTENTS

1 INTRODUCTION 3

MANAGING UNSTRUCTURED FINANCIAL DATA AT WEB SCALE

> “Red Hat worked with us the entire way as we designed and built our architectures, helping with best practices, design considerations and layout, performance testing, and migration.”
>
> Mohit Anchlia
> Architect, Intuit TurboTax

BUSINESS CHALLENGE

Needed a fast, reliable and cost-effective storage solution to meet growing SaaS line of business

Tax returns and other data were being stored as BLOBs in an expensive Oracle Database

Replication required database hacks, disaster recovery was challenging

SOLUTION

Red Hat Gluster Storage

HP ProLiant DL2000 Multi Node Server

BENEFITS

Provides scalable on-demand storage for unstructured data

Cost-effective solution that leverages commodity hardware

Helps meet growing capacity and peak performance needs

Lets you achieve multisite DR strategy

SaskTel

> "By standardizing on Red Hat Storage Server on commodity hardware, we were able to quickly scale our infrastructure to manage massive amounts of data while significantly decreasing our costs."
>
> David Yaffe
> Technical Analyst, SaskTel

GATHERING TELCO BUSINESS INSIGHTS FROM MACHINE DATA

BUSINESS CHALLENGE

Storage and analysis of massive amounts of server and device logging information

Data analysis involved many separate tools and steps

Logical and physical silos led to high incident response times

Proprietary storage too expensive

SOLUTION

Red Hat Gluster Storage

Splunk Enterprise, HP servers

BENEFITS

CASIO®

> “Our costs, including various procurement costs and operating fees, fell to less than half of what we had been before implementing Red Hat Storage Server. The solution’s flexibility enabled us to build a storage environment using commodity servers and its ease of operational control was also a major advantage.”

Kazuyasu Yamazaki

SCALABLE, COST-EFFECTIVE STORAGE FOR RED HAT VIRTUALIZATION

BUSINESS CHALLENGE

Virtualized server infrastructure, but storage costs negated server virtualization cost benefits

Traditional and proprietary systems also limited flexibility which resulted in further cost escalation

Eliminate vendor lock in

SOLUTION

Red Hat Gluster Storage & Red Hat Enterprise Virtualization

IBM System x servers

BENEFITS

Reduced storage costs by 50%

Standardizing on RHEV and RHS provided flexibility

Able to use commodity servers and centrally manage server and storage infrastructure

Demo

Install

Peer Systems

Format & Mount Bricks

Create & Start Volume

Install Client

Mount Volume on Client

Self Healing

redhat

THANK YOU

G+ plus.google.com/+RedHat

f facebook.com/redhatinc

in linkedin.com/company/red-hat

twitter.com/RedHat

youtube.com/user/RedHatVideos