GlusterFS Storage Administration Deep Dive

Dustin L. Black, RHCA
Principal Cloud Success Architect
Red Hat Customer Experience & Engagement

Dustin L. Black

dustin@redhat.com

@dustinlblack

linkedin.com/in/dustinblack

people.redhat.com/dblack

RED HAT CERTIFIED

ARCHITECT

redhat

Agenda

- Hour 1 – GlusterFS Fundamentals
  - GlusterFS Overview
  - Use Cases
  - Technology Stack
  - Algorithmic Data Placement & Translators
  - Volumes and Layered Functionality
  - Asynchronous Replication
  - Data Access

Agenda

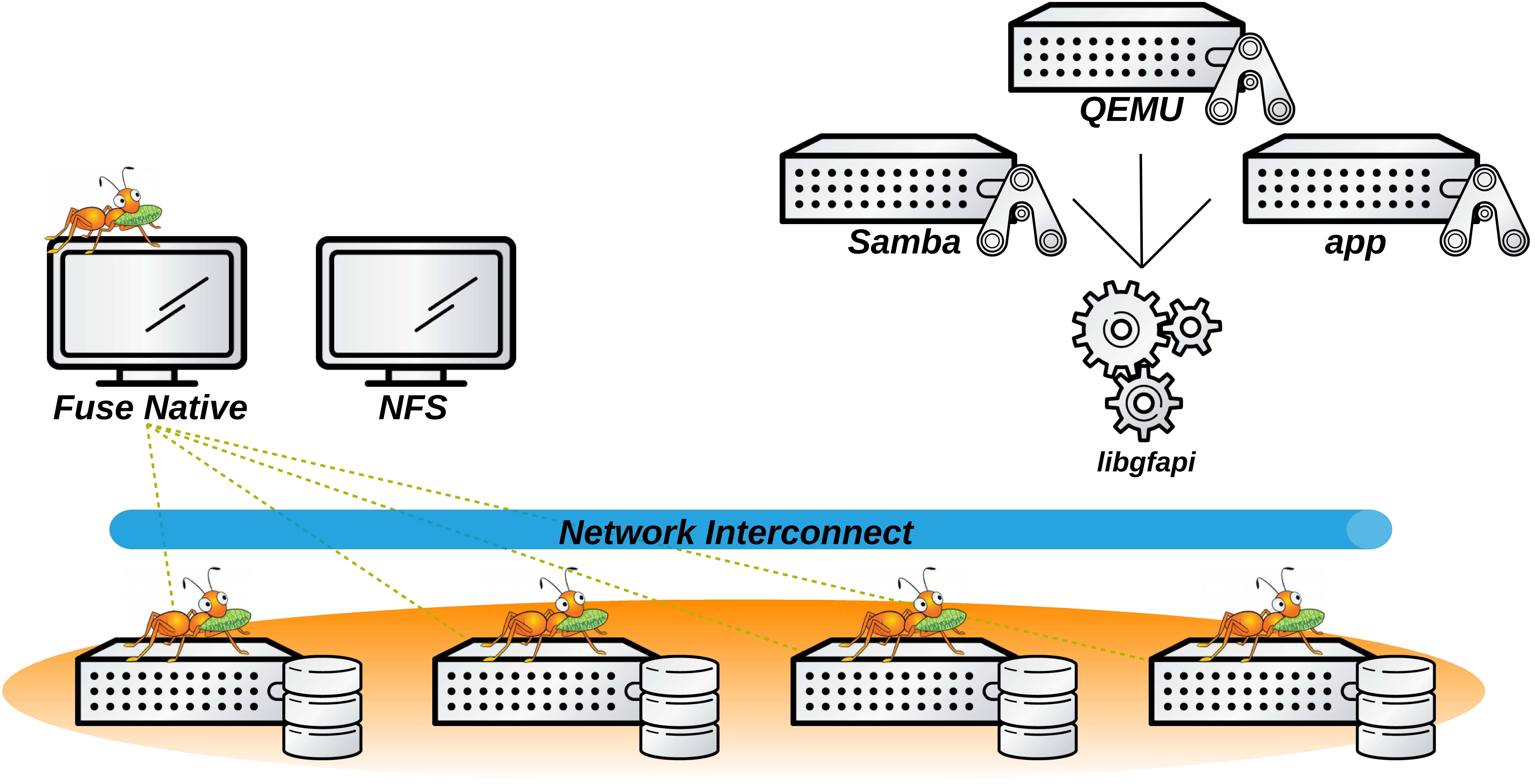
- Hour 2 – Advanced Features Demo
  - Metadata internals
  - Volfiles
  - Quotas
  - Split-Brain & Quorum Enforcement
  - Configuring Geo-Replication
  - Snapshots
  - Disperse Volumes (erasure coding)

Technology Overview

GlusterFS Storage Administration Deep Dive

What is GlusterFS?

- Clustered Scale-out **General Purpose** Storage Platform
  - POSIX-y Distributed File System
  - ...and so much more
- Built on Commodity systems
  - x86_64 Linux ++
  - POSIX filesystems underneath (XFS, EXT4)
- No Metadata Server
- Standards-Based – Clients, Applications, Networks
- Modular Architecture for Scale and Functionality


QEMU
Samba
app
libgfapi
Fuse Native
NFS
Network Interconnect

GlusterFS vs. Traditional Solutions

- A basic NAS has limited scalability and redundancy
- Other distributed filesystems are limited by metadata service
- SAN is costly & complicated, but high performance & scalable
- *GlusterFS is...*
  - *Linear Scaling*
  - *Minimal Overhead*
  - *High Redundancy*
  - *Simple and Inexpensive Deployment*

Use Cases

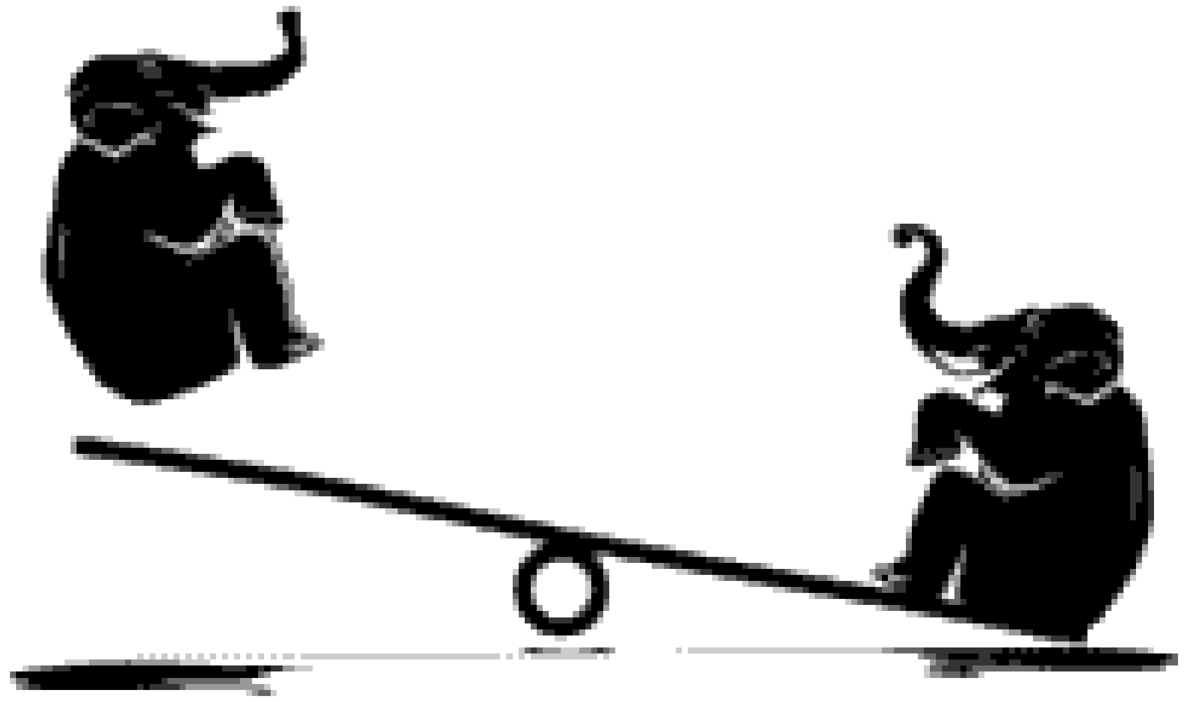
GlusterFS Storage Administration Deep Dive

Common Solutions

- Large Scale File Server
- Media / Content Distribution Network (CDN)
- Backup / Archive / Disaster Recovery (DR)
- High Performance Computing (HPC)
- Infrastructure as a Service (IaaS) storage layer
- Database offload (blobs)
- Unified Object Store + File Access

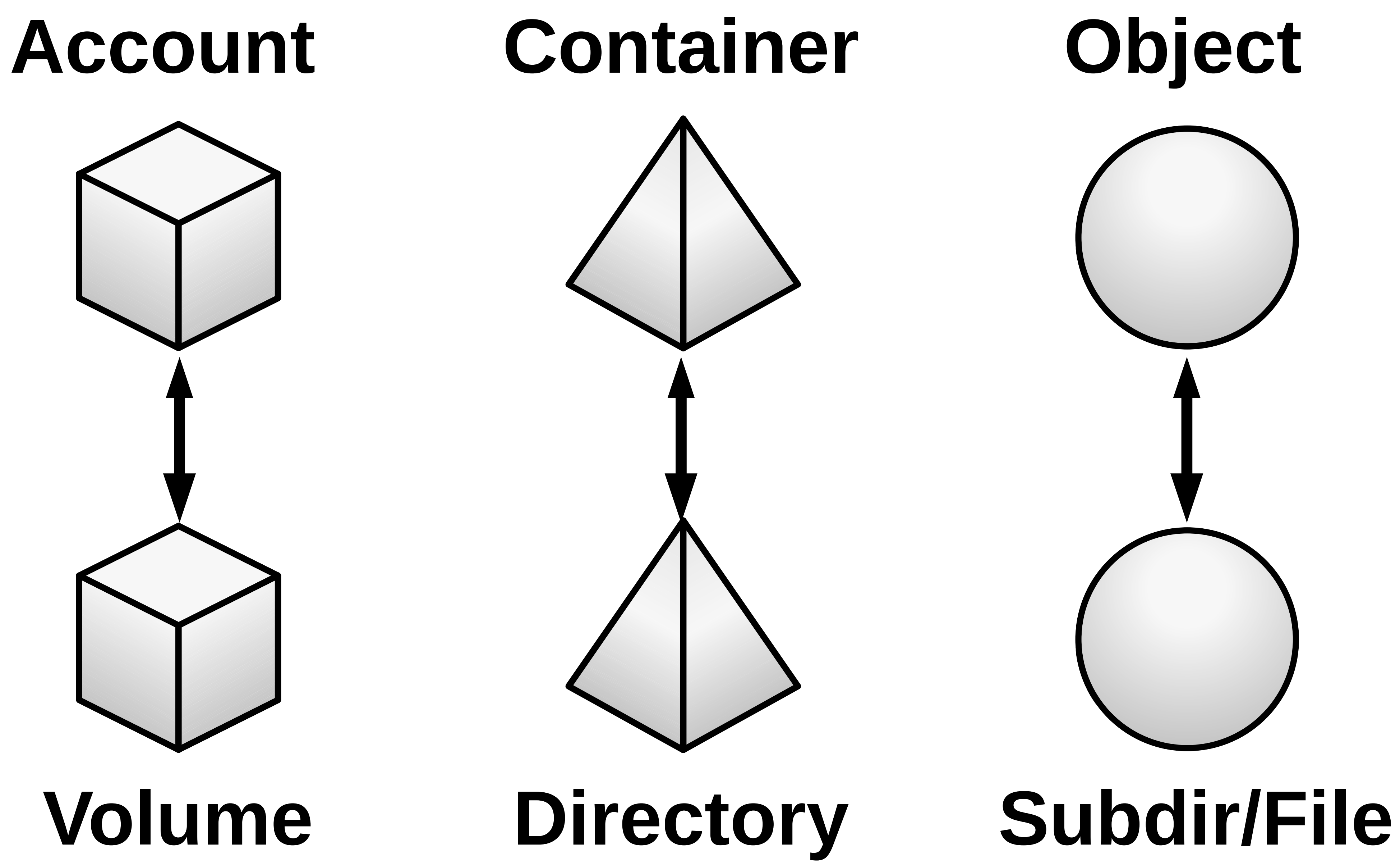
Hadoop – Map Reduce

- Access data within and outside of Hadoop
- No HDFS name node single point of failure / bottleneck
- Seamless replacement for HDFS
- Scales with the massive growth of big data



openstack

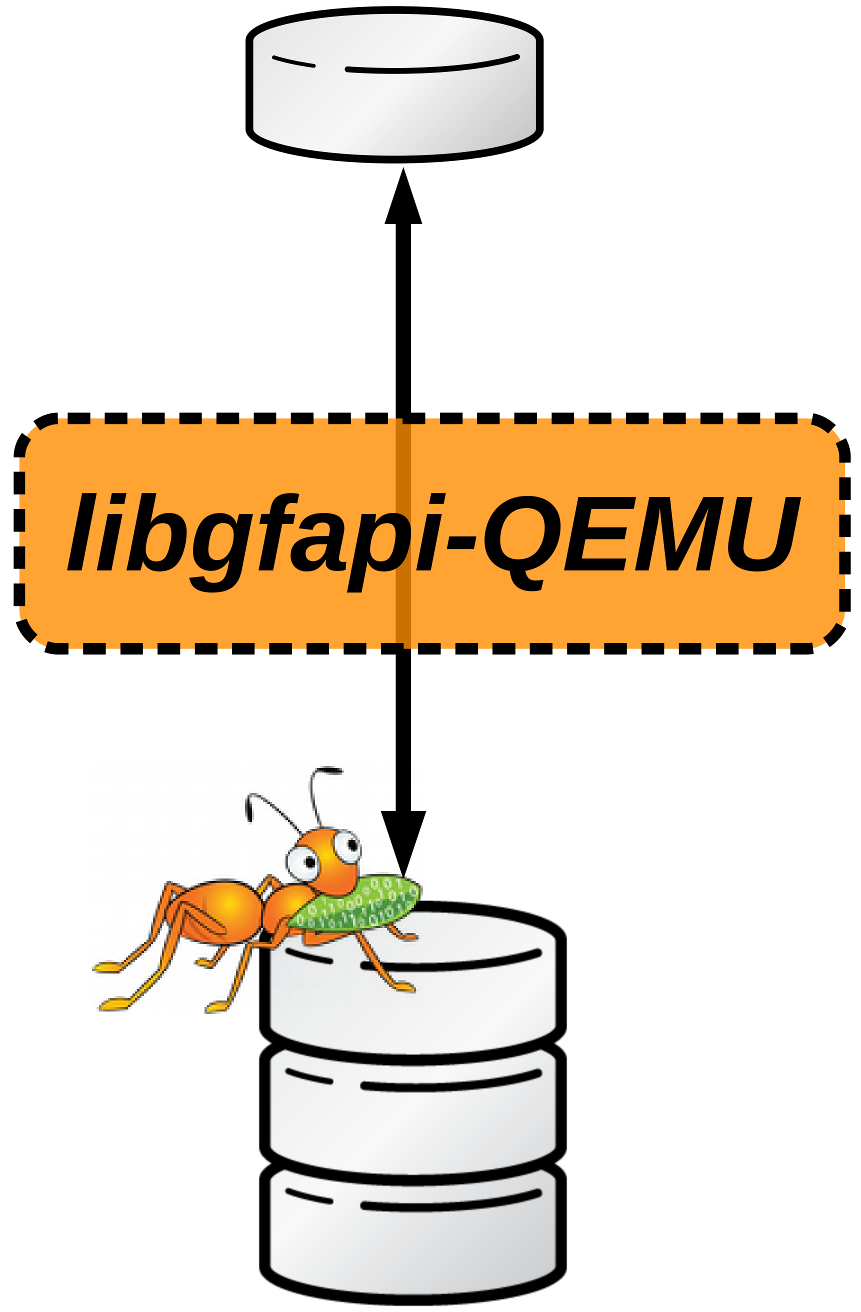
Swift



Account | Container | Object

Volume | Directory | Subdir/File

Cinder / Glance



libgfapi-QEMU

Technology Stack

GlusterFS Storage Administration Deep Dive

Terminology

- Brick
  - Fundamentally, a filesystem mountpoint
  - A unit of storage used as a ***capacity*** building block
- Translator
  - Logic between the file bits and the Global Namespace
  - Layered to provide GlusterFS ***functionality***

Everything is Modular

Terminology

- Volume
  - Bricks combined and passed through translators
  - Ultimately, what's presented to the end user
- Peer / Node
  - Server hosting the brick filesystems
  - Runs the gluster daemons and participates in volumes

Disk, LVM, and Filesystems

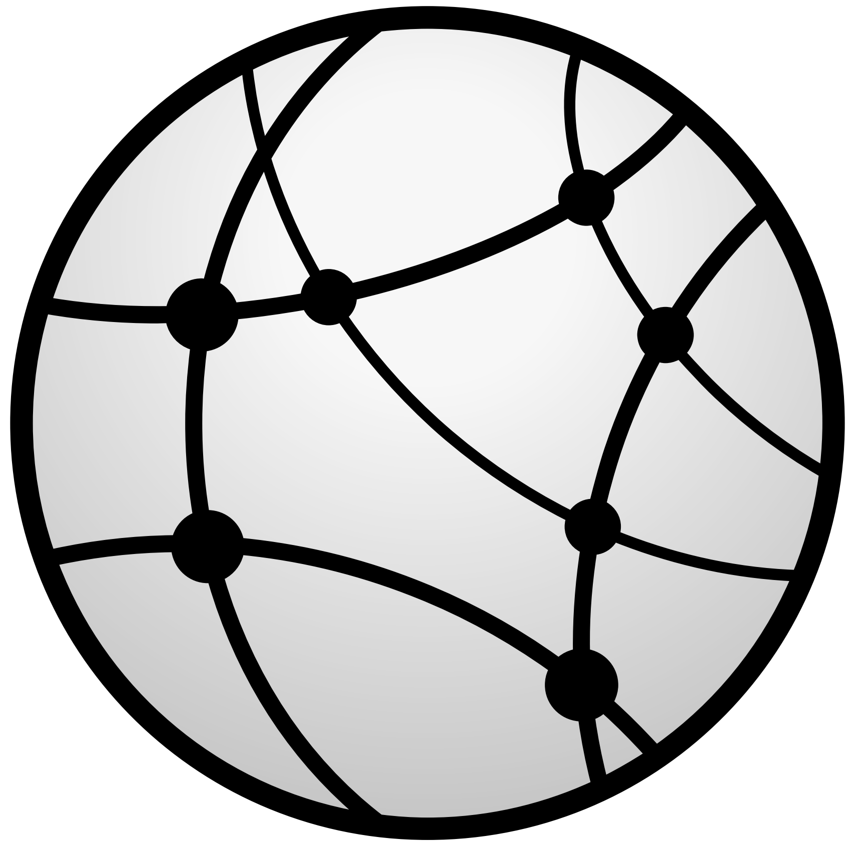
- Direct-Attached Storage (DAS)

  -or-

- Just a Bunch Of Disks (JBOD)
- Hardware RAID
  - RHGS: RAID 6 required
- Logical Volume Management (LVM)
- POSIX filesystem w/ Extended Attributes (EXT4, XFS, BTRFS, …)
  - RHGS: XFS required

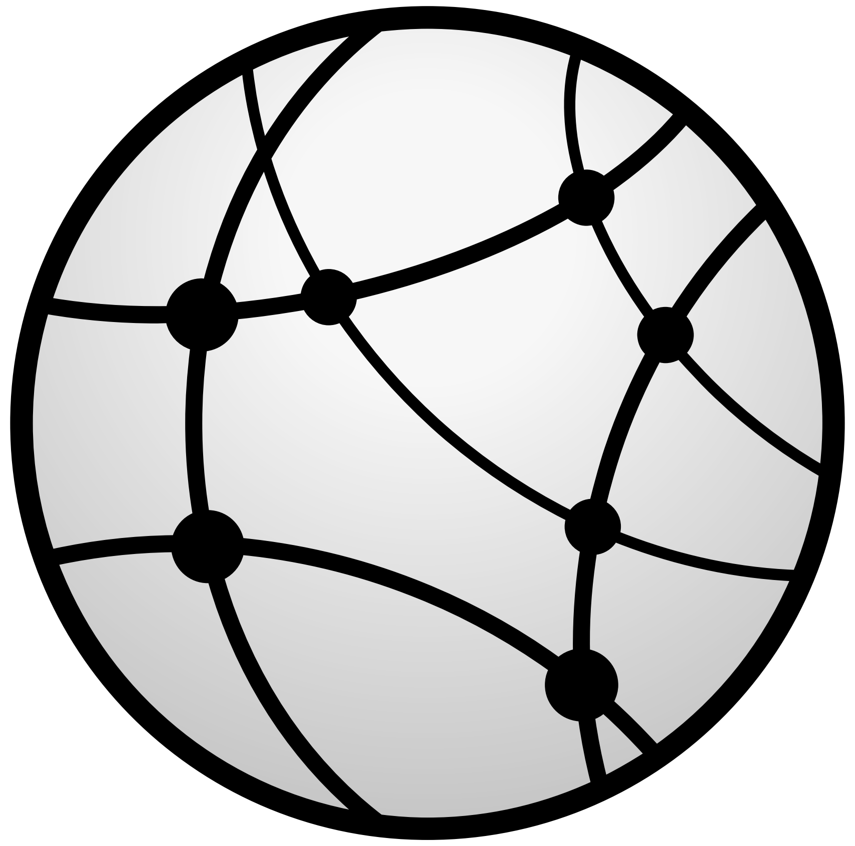
Data Access Overview

- GlusterFS Native Client
  - Filesystem in Userspace (FUSE)
- NFS
  - Built-in Service
- SMB/CIFS
  - Samba server required; NOW libgfapi-integrated!



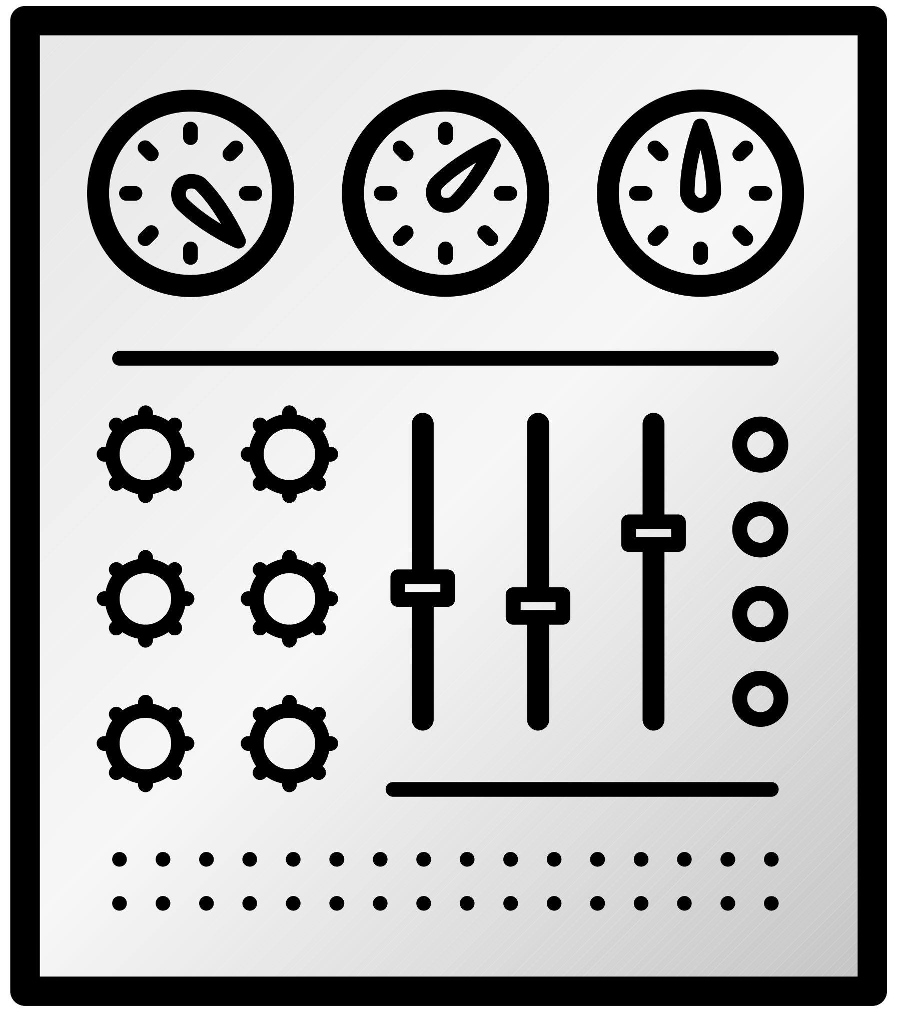
Data Access Overview

- Gluster For OpenStack (G4O; aka UFO)
  - Simultaneous object-based access via OpenStack Swift
- libgfapi flexible abstracted storage
  - Integrated with upstream Samba and NFS-Ganesha



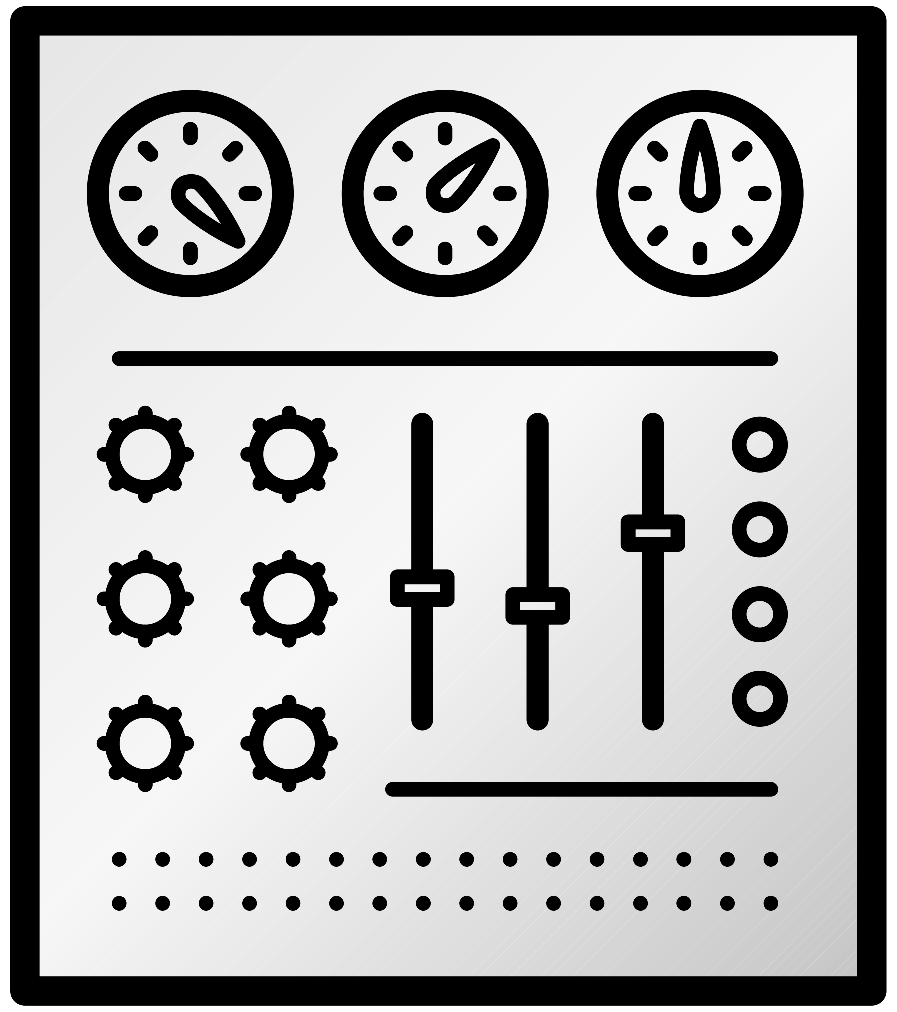
Gluster Components

- `glusterd`
  - Management daemon
  - One instance on each GlusterFS server
  - Interfaced through `gluster` CLI
- `glusterfsd`
  - GlusterFS brick daemon
  - One process for each brick on each server
  - Managed by `glusterd`

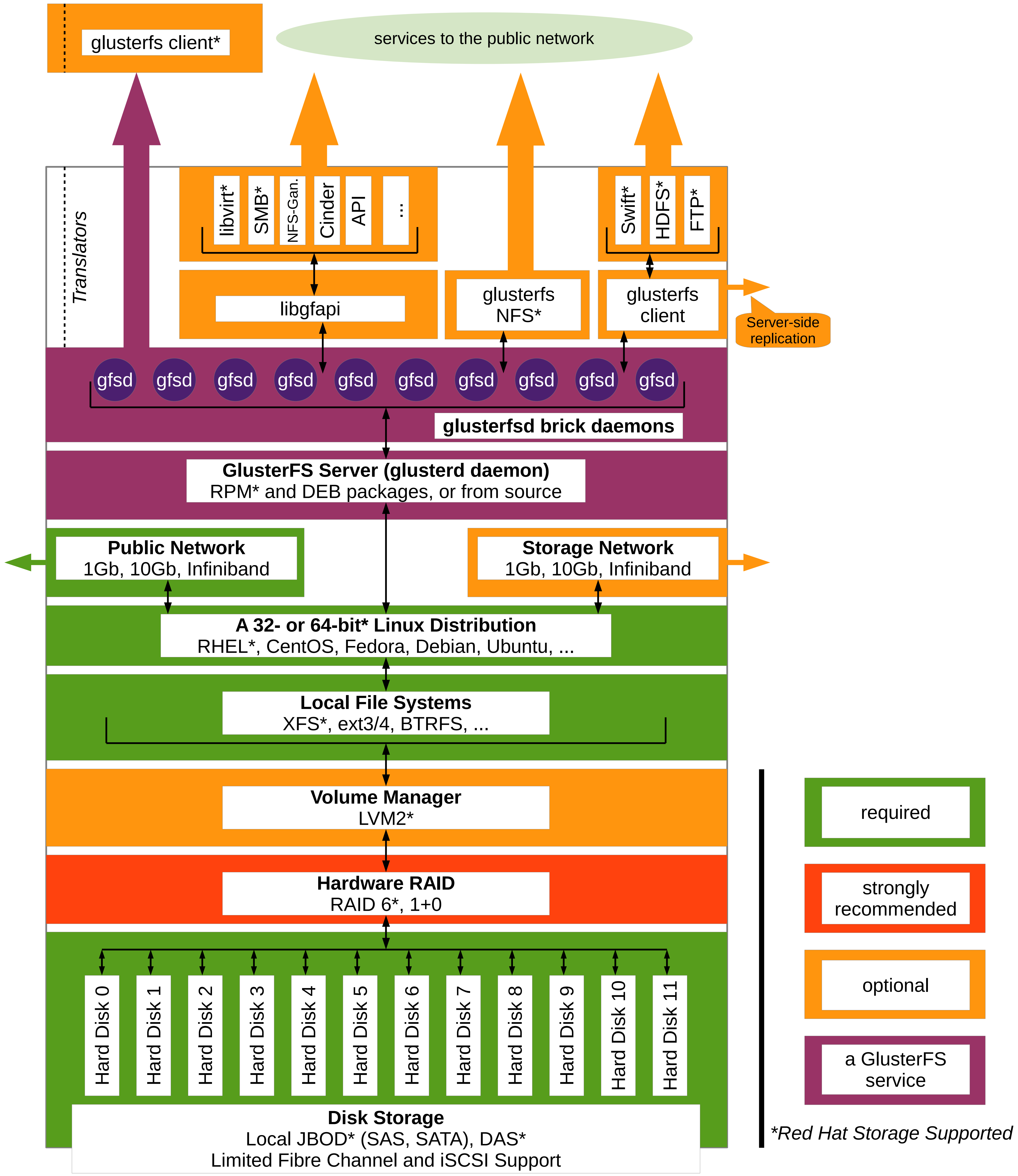


Gluster Components

- `glusterfs`
  - Volume service daemon
  - One process for each volume service
    - NFS server, FUSE client, Self-Heal, Quota, ...
- `mount.glusterfs`
  - FUSE native client mount extension
- `gluster`
  - Gluster Console Manager (CLI)



Putting it Together



glusterfs client*

services to the public network

Translators

libvirt* SMB* NFS-Gan. Cinder API ...

libgfapi

glusterfs NFS*

Swift* HDFS* FTP*

glusterfs client

Server-side replication

gfsd gfsd gfsd gfsd gfsd gfsd gfsd gfsd gfsd gfsd

glusterfsd brick daemons

GlusterFS Server (glusterd daemon)
RPM* and DEB packages, or from source

Public Network
1Gb, 10Gb, Infiniband

Storage Network
1Gb, 10Gb, Infiniband

A 32- or 64-bit* Linux Distribution
RHEL*, CentOS, Fedora, Debian, Ubuntu, ...

Local File Systems
XFS*, ext3/4, BTRFS, ...

Volume Manager
LVM2*

Hardware RAID
RAID 6*, 1+0

Hard Disk 0 Hard Disk 1 Hard Disk 2 Hard Disk 3 Hard Disk 4 Hard Disk 5 Hard Disk 6 Hard Disk 7 Hard Disk 8 Hard Disk 9 Hard Disk 10 Hard Disk 11

Disk Storage
Local JBOD* (SAS, SATA), DAS*
Limited Fibre Channel and iSCSI Support

🟩 required

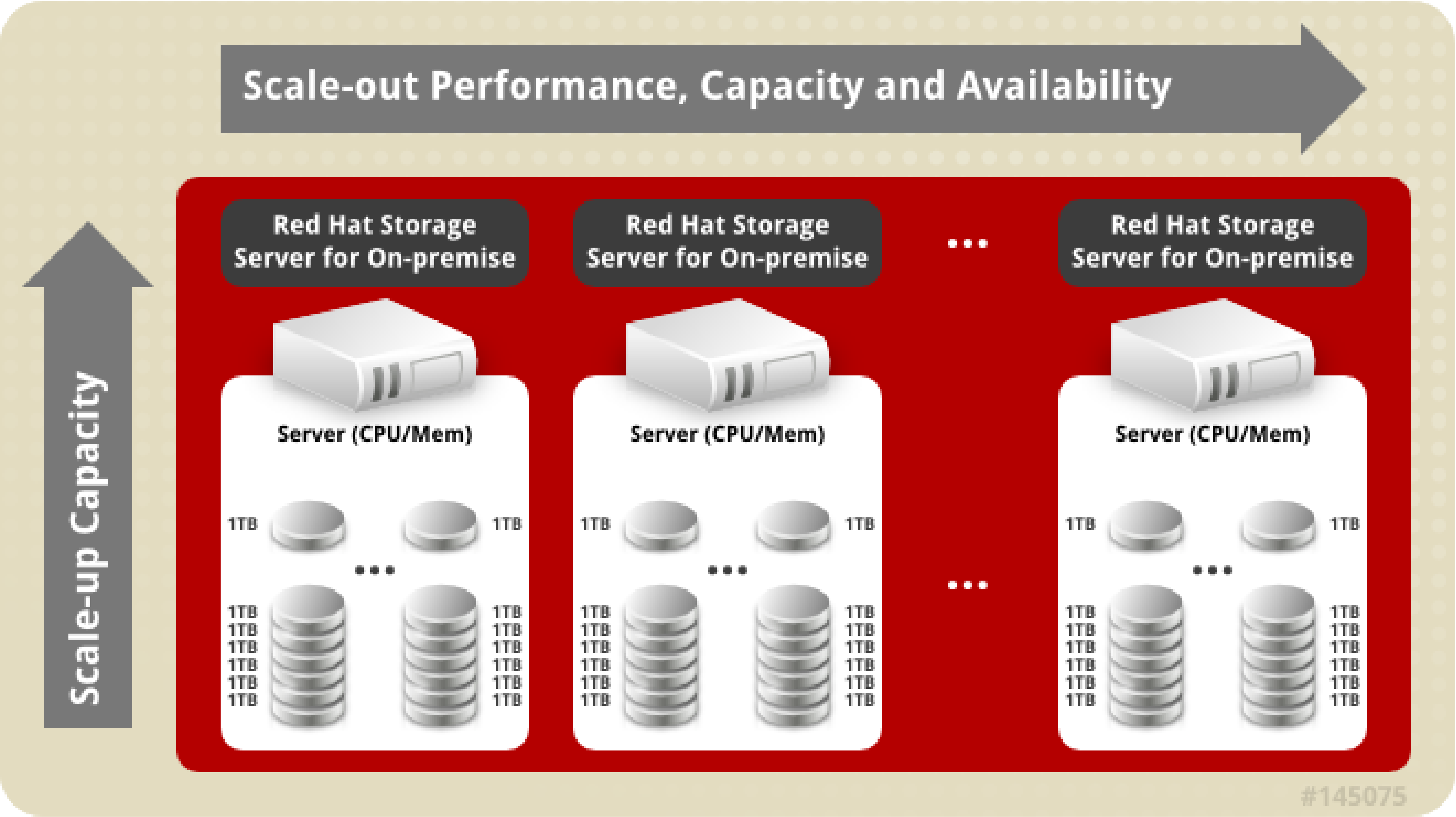
🟥 strongly recommended

🟧 optional

🟪 a GlusterFS service

**Red Hat Storage Supported*

Up and Out!

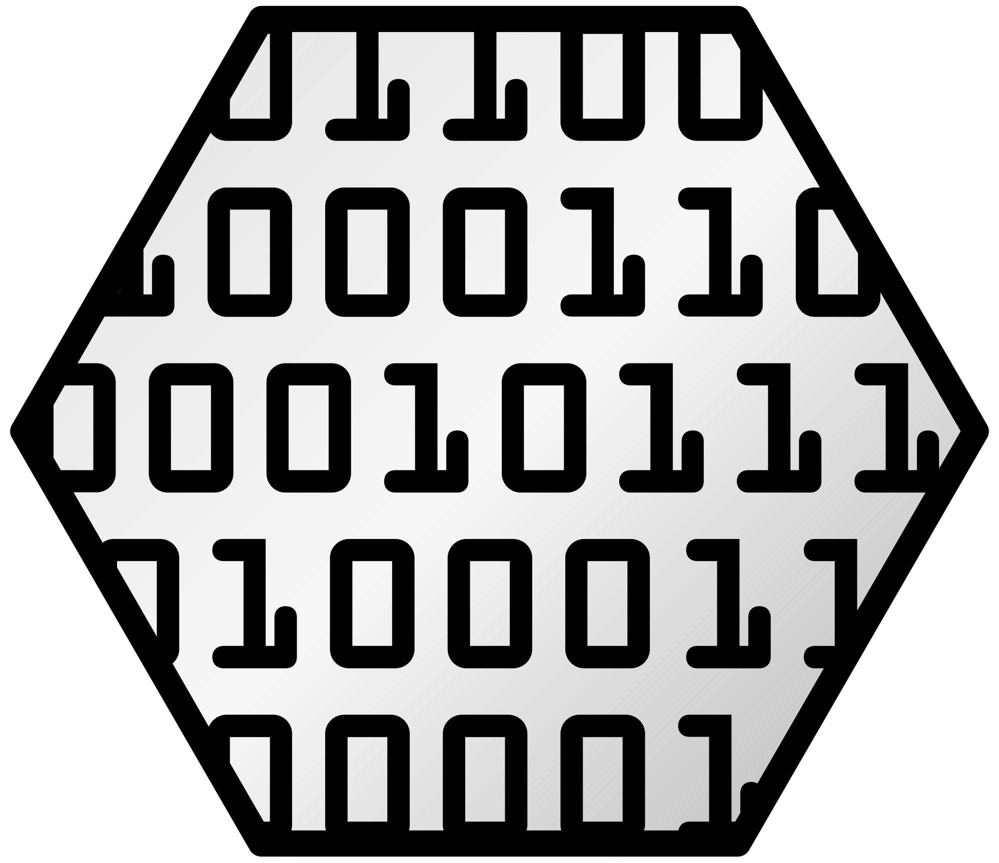

Scale-out Performance, Capacity and Availability
Scale-up Capacity
Red Hat Storage Server for On-premise
Red Hat Storage Server for On-premise
...
Red Hat Storage Server for On-premise
Server (CPU/Mem)
Server (CPU/Mem)
Server (CPU/Mem)
1TB
1TB
...
...
#145075

Under the Hood

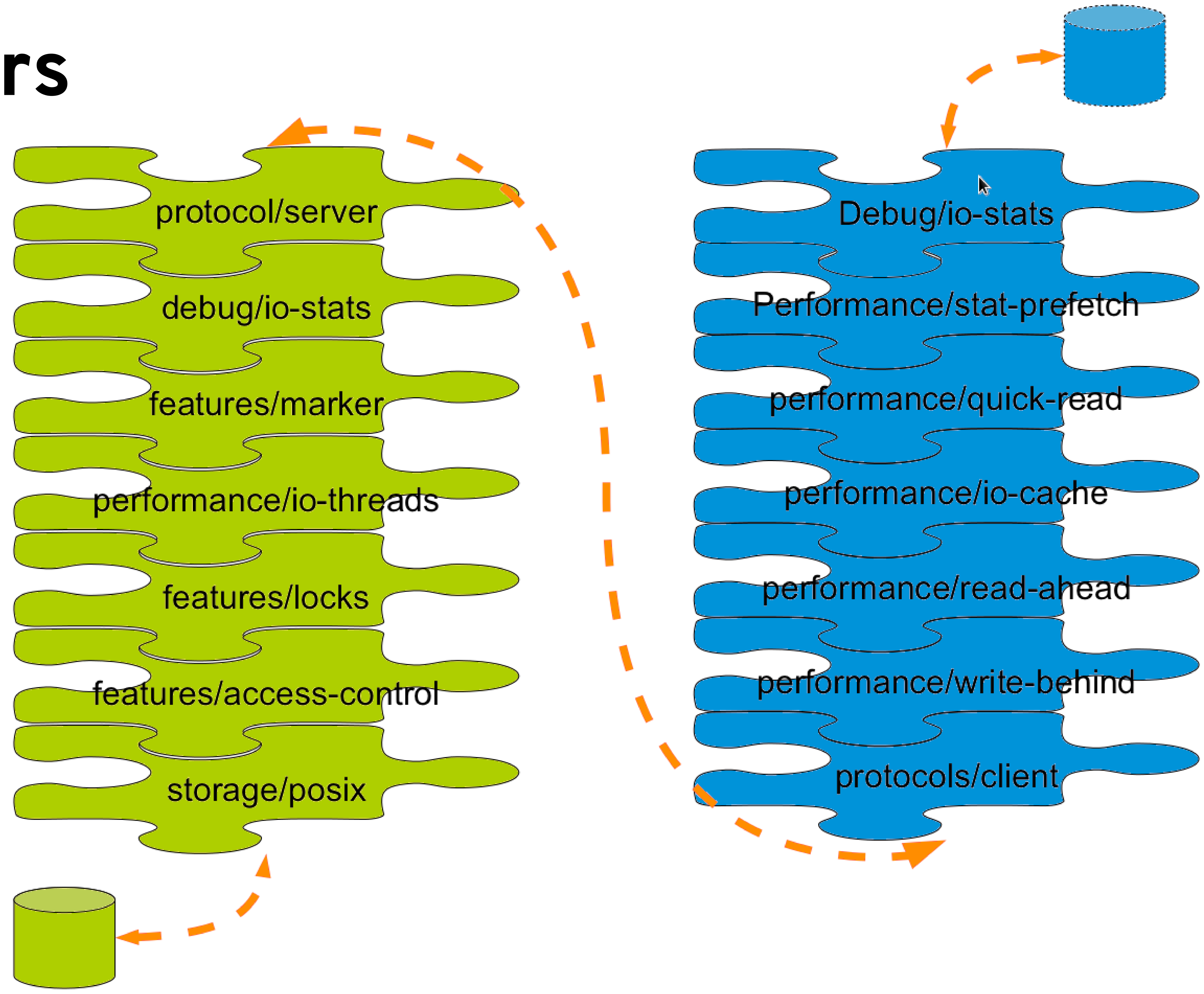
GlusterFS Storage Administration Deep Dive

Elastic Hash Algorithm

- No central metadata
  - No Performance Bottleneck
  - Eliminates risk scenarios
- Location hashed intelligently on filename
  - Unique identifiers, similar to md5sum
- The “Elastic” Part
  - Files assigned to virtual volumes
  - Virtual volumes assigned to multiple bricks
  - Volumes easily reassigned on the fly



Translators



Your Storage Servers are Sacred!

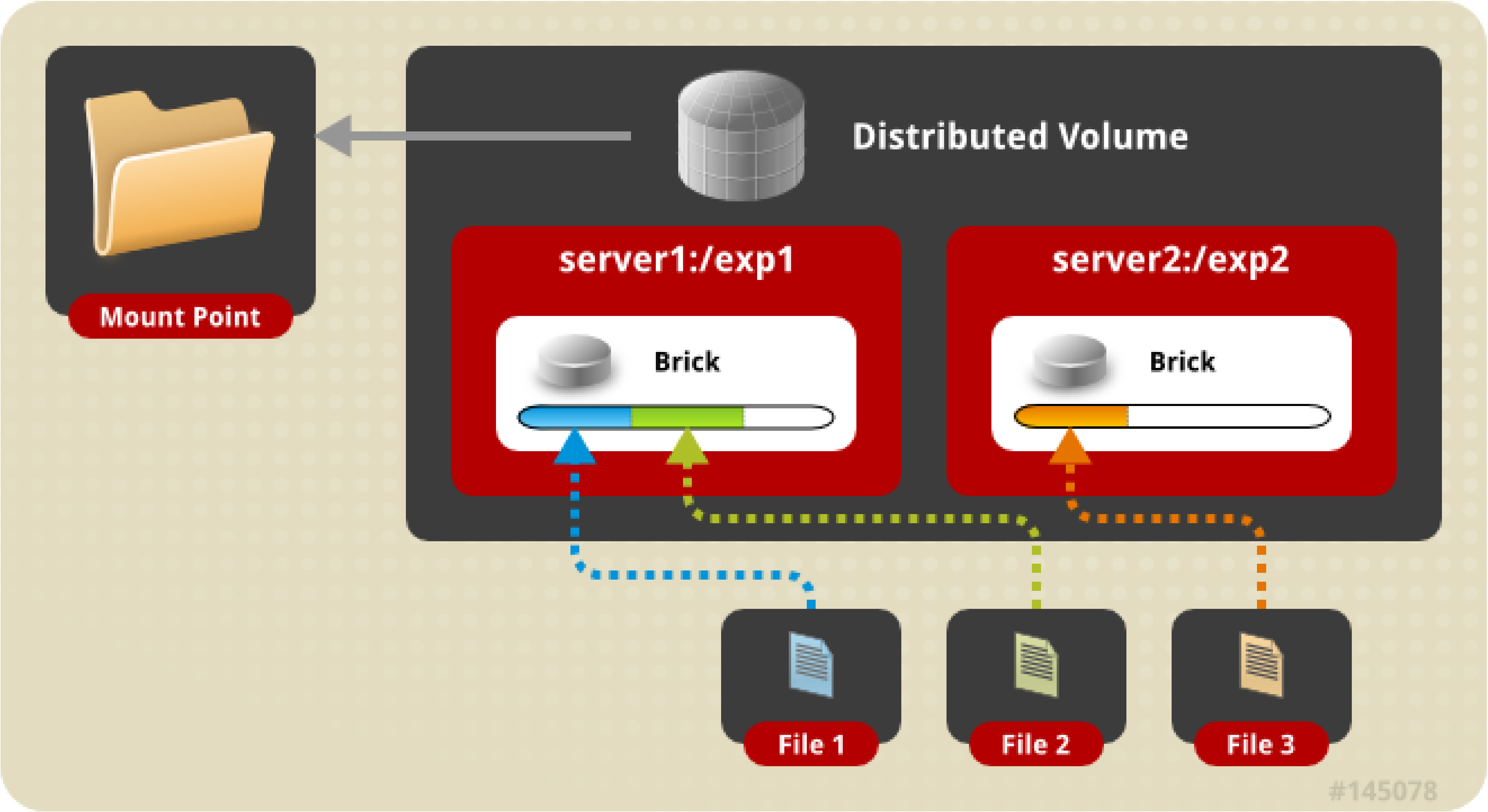
- Don't touch the brick filesystems directly!
- They're Linux servers, but treat them like storage appliances
  - Separate security protocols
  - Separate access standards
- Don't let your Jr. Linux admins in!
  - A well-meaning sysadmin can quickly break your system or destroy your data

Basic Volumes

GlusterFS Storage Administration Deep Dive

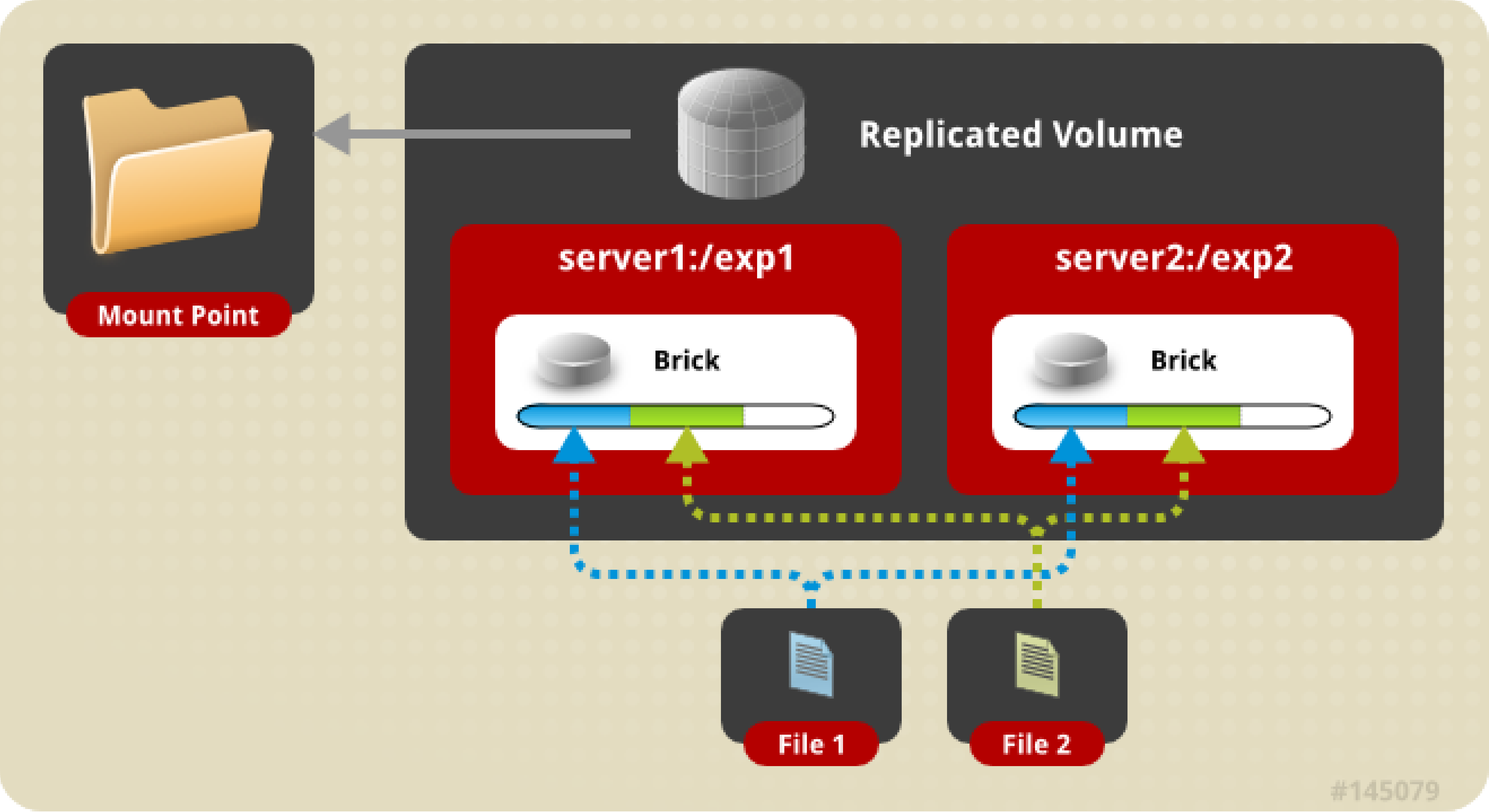
Distributed Volume

- The default configuration
- Files “evenly” spread across bricks
- Similar to file-level RAID 0
- Server/Disk failure could be catastrophic



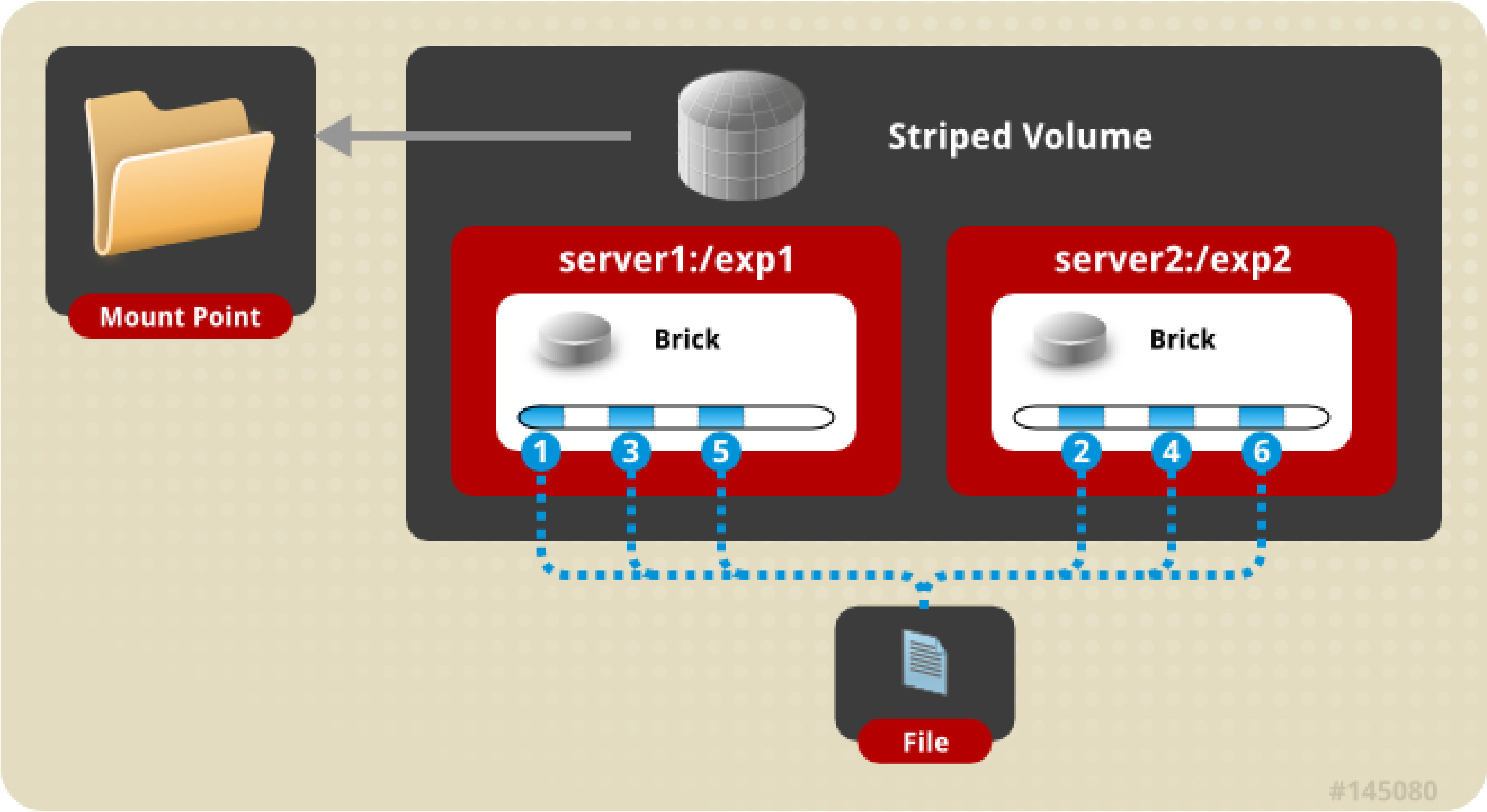
Replicated Volume

- Files written synchronously to replica peers
- Files read synchronously, but ultimately serviced by the first responder
- Similar to file-level RAID 1



Striped Volumes

- Individual files split among bricks (sparse files)
- Similar to block-level RAID 0
- Limited Use Cases
  - HPC Pre/Post Processing
  - File size exceeds brick size

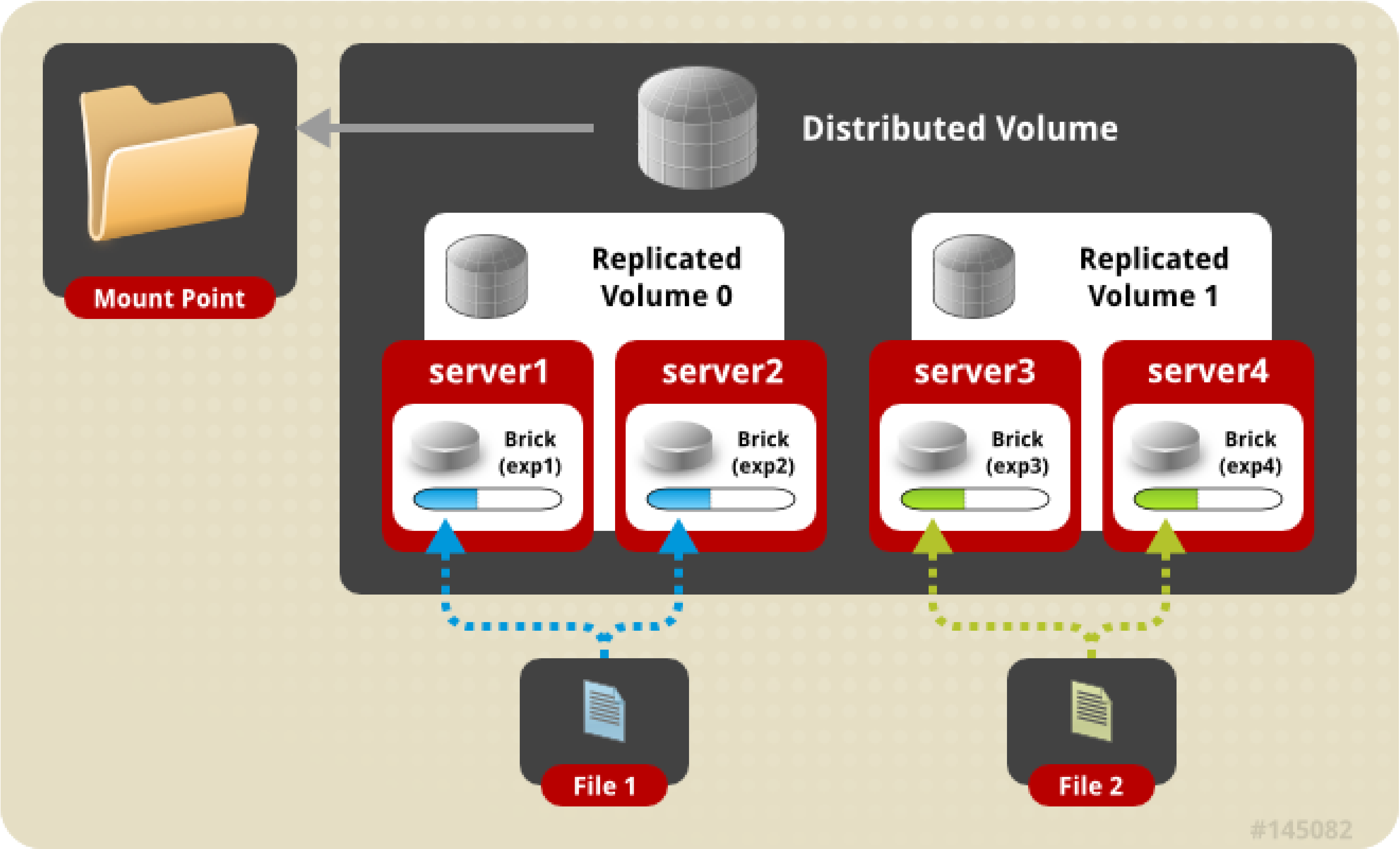


Layered Functionality

GlusterFS Storage Administration Deep Dive

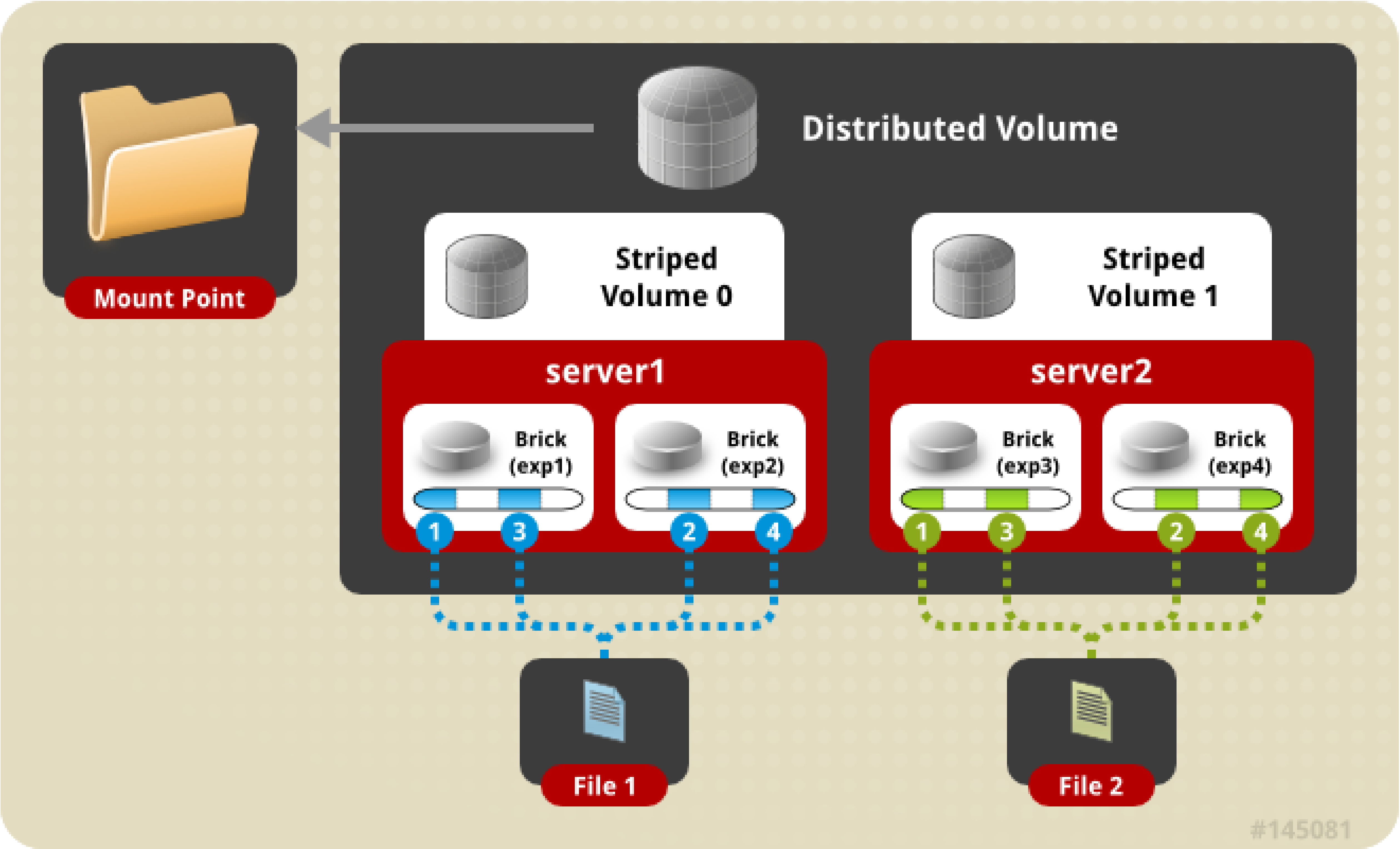
Distributed Replicated Volume

- Distributes files across multiple replica sets



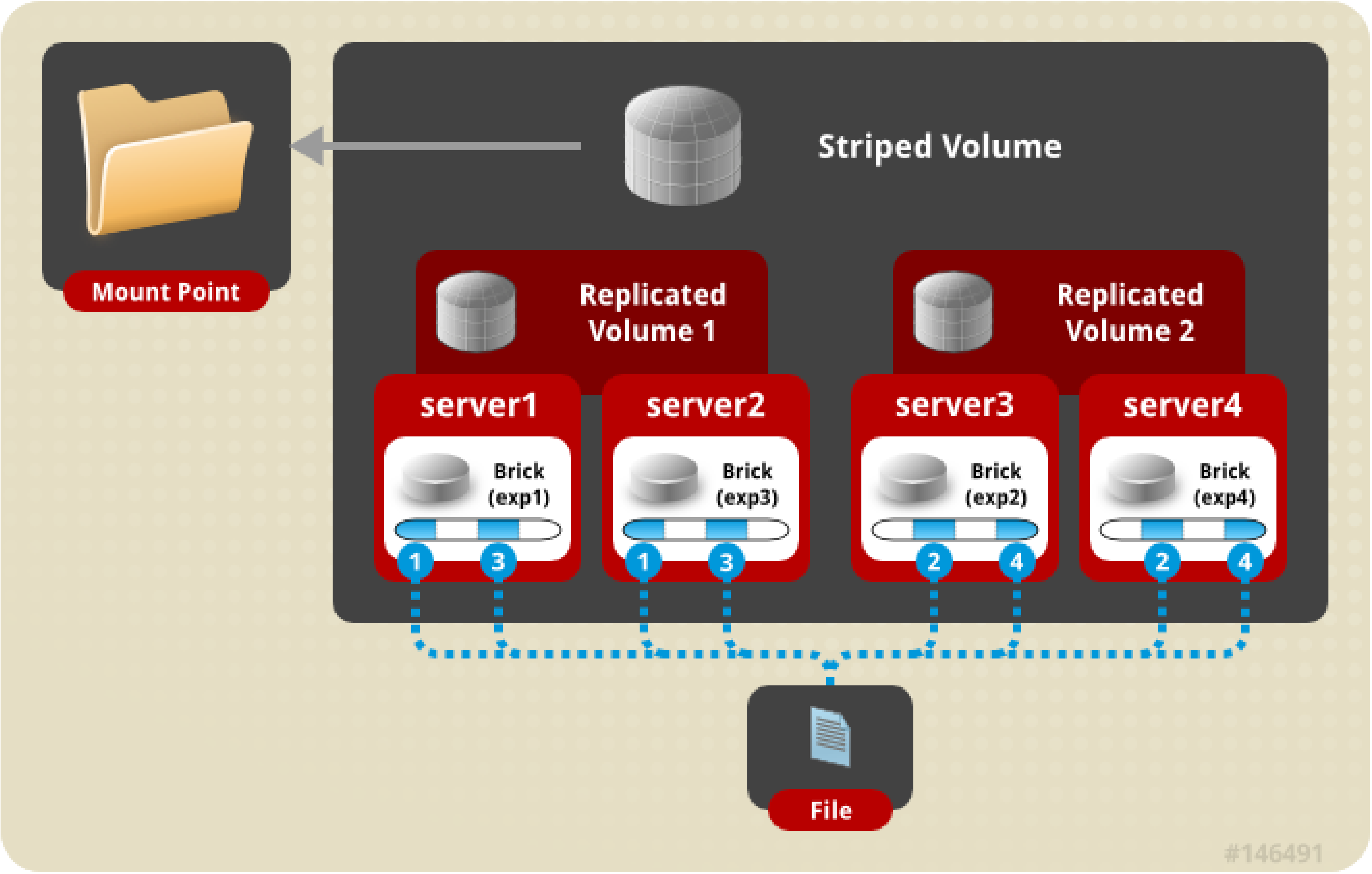
Distributed Striped Volume

- Distributes files across multiple stripe sets
- Striping plus scalability



Striped Replicated Volume

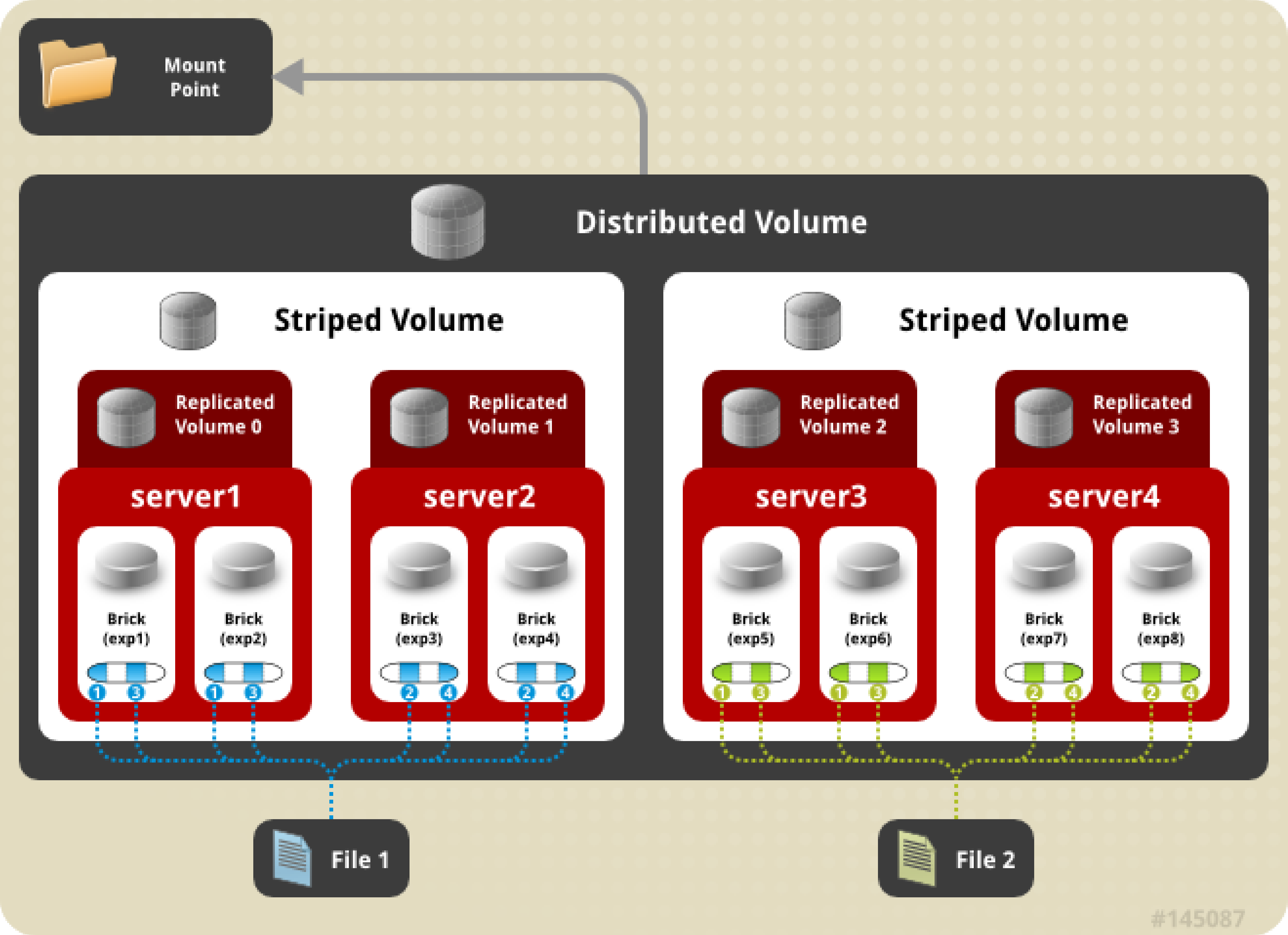
- Replicated sets of stripe sets
- Similar to RAID 10 (1+0)



Distributed Striped Replicated Volume

- Limited Use Cases – Map Reduce

Don't do it like this - ->

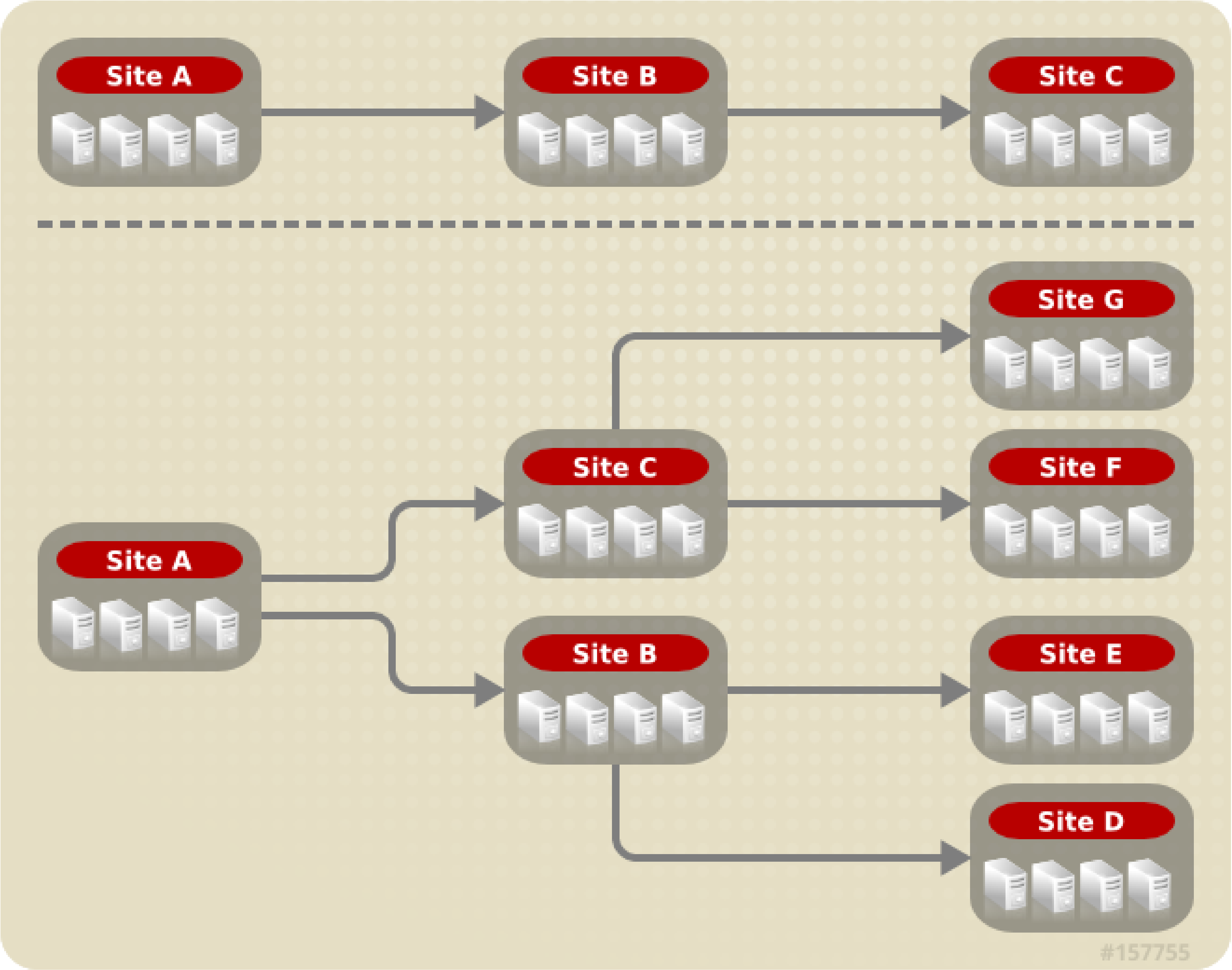


Asynchronous Replication

GlusterFS Storage Administration Deep Dive

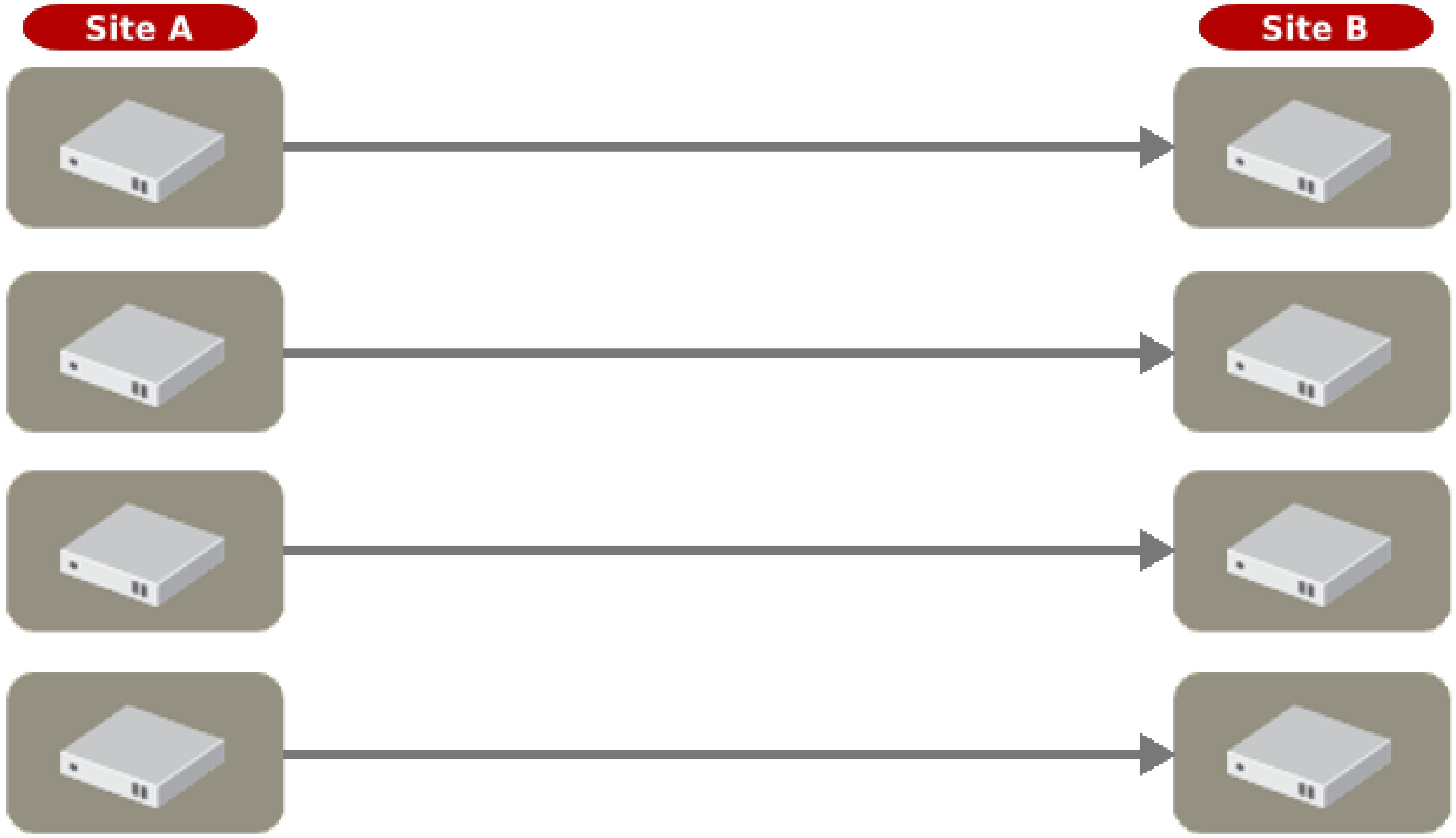
Geo Replication

- Asynchronous across LAN, WAN, or Internet
- Master-Slave model
  - Cascading possible
- Continuous and incremental
- One Way



Distributed Geo-Replication

- Drastic performance improvements
  - Parallel transfers
  - Efficient source scanning
  - Pipelined and batched
  - File type/layout agnostic



Data Access

GlusterFS Storage Administration Deep Dive

GlusterFS Native Client (FUSE)

- FUSE kernel module allows the filesystem to be built and operated entirely in userspace
- Specify mount to any GlusterFS server
- Native Client fetches volfile from mount server, then communicates directly with all nodes to access data
- Recommended for high concurrency and high write performance
- Load is inherently balanced across distributed volumes

NFS

- Standard NFS v3 clients
- Standard automounter is supported
- Mount to any server, or use a load balancer
- GlusterFS NFS server includes Network Lock Manager (NLM) to synchronize locks across clients
- Better performance for reading many small files from a single client
- HA with CTDB; Load balancing must be managed externally

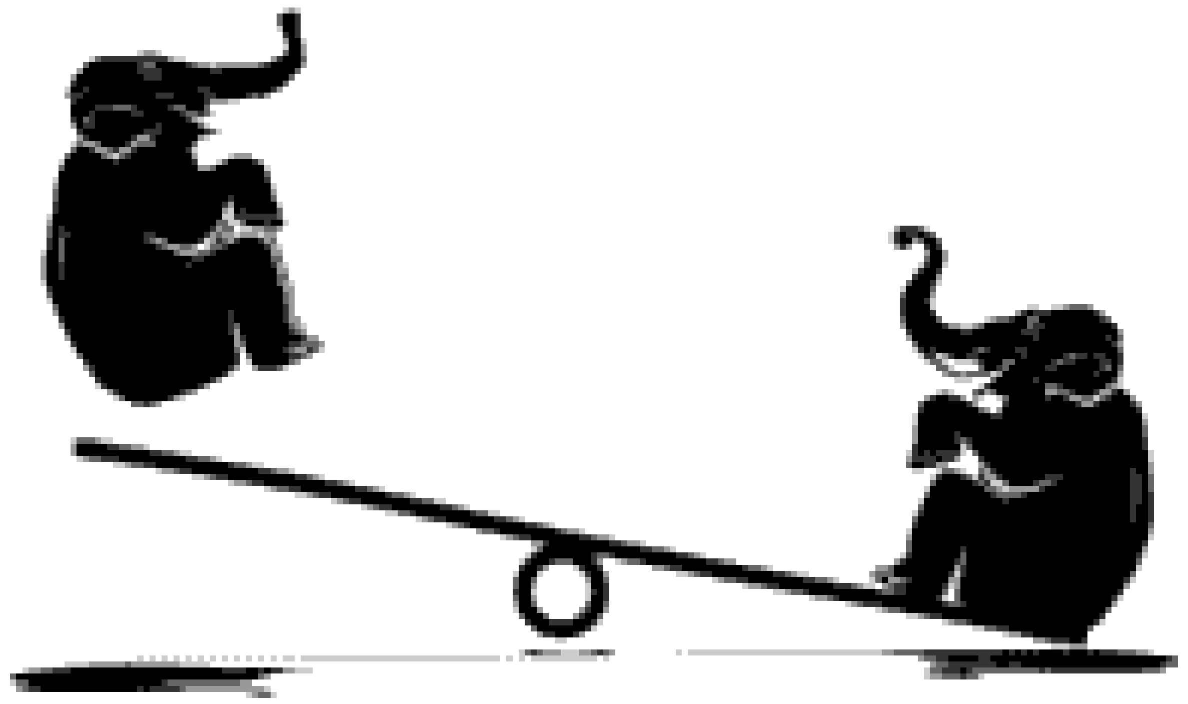
libgfapi

- Introduced with GlusterFS 3.4
- User-space library for accessing data in GlusterFS
- Filesystem-like API
- Runs in application process
- no FUSE, no copies, no context switches
- ...but same volfiles, translators, etc.

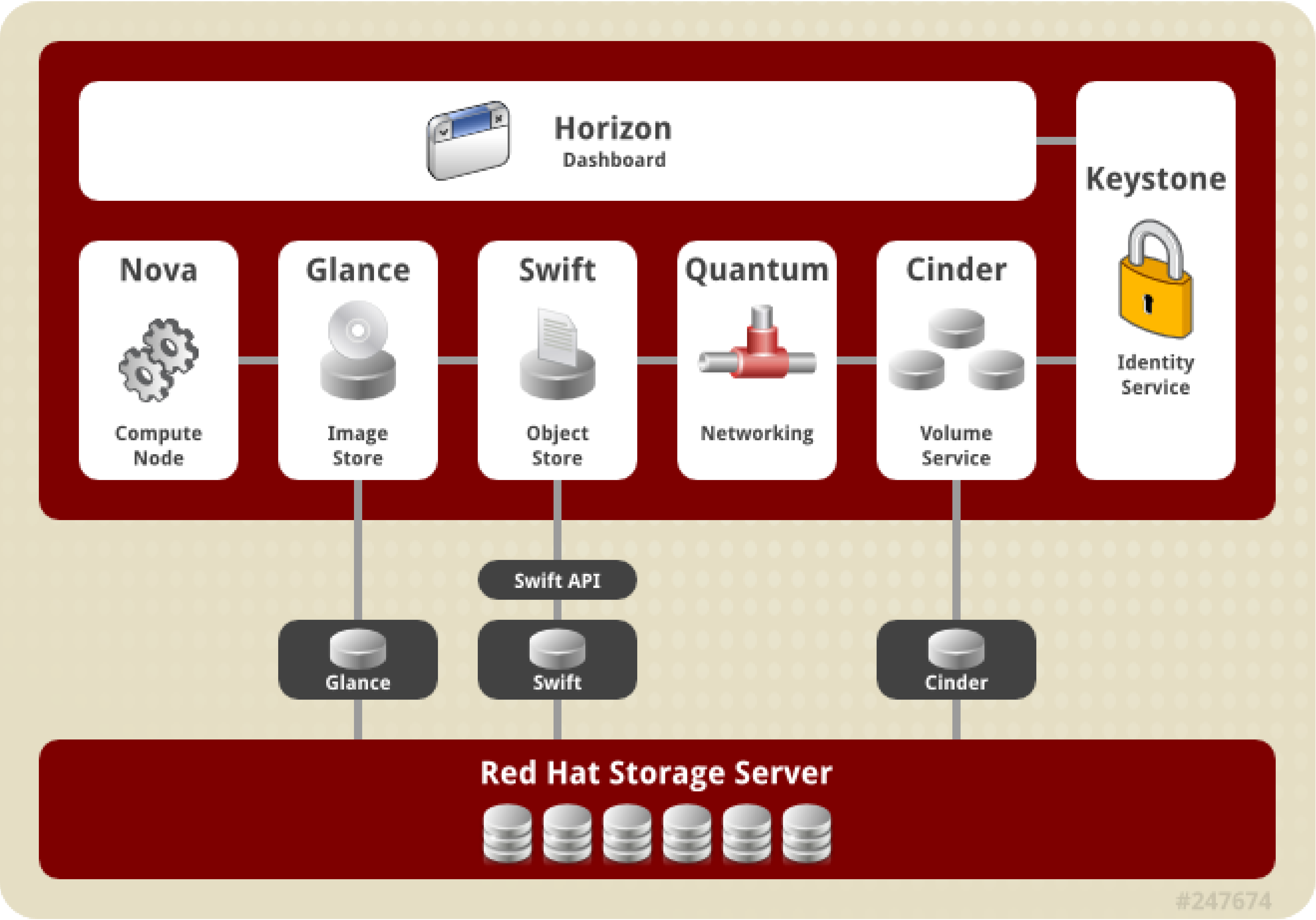
SMB/CIFS

- Samba + libgfapi
  - No need for local native client mount & re-export
  - Significant performance improvements with FUSE removed from the equation
- Must be setup on each server you wish to connect to via CIFS
- CTDB is required for Samba HA

HDFS Compatibility



Gluster 4 OpenStack (G4O)⚥



⚥The feature formerly known as UFO

http://people.redhat.com/dblack

Demo Time!

GlusterFS Storage Administration Deep Dive

Do it!

GlusterFS Storage Administration Deep Dive

Do it!

- Build a test environment in VMs in just minutes!
- Get the bits:
  - Fedora has GlusterFS packages natively: fedoraproject.org
  - RHGS ISO available on the Red Hat Portal: access.redhat.com
  - Go upstream: gluster.org
  - Amazon Web Services (AWS)
    - Amazon Linux AMI includes GlusterFS packages
    - RHGS AMI is available

Thank You!

- Contact
  - dustin@redhat.com
  - storage-sales@redhat.com
- Resources
  - www.gluster.org
  - www.redhat.com/storage/
  - access.redhat.com/support/offerings/tam/

- Twitter

  @dustinlblack

  @gluster

  @RedHatStorage

GlusterFS Storage Administration Deep Dive

Slides Available at: people.redhat.com/dblack

redhat.