



Introduction to Gluster

Versions 3.0.x

Table of Contents

Table of Contents	2
Overview	3
Gluster File System.....	3
Gluster Storage Platform	3
No metadata with the Elastic Hash Algorithm	4
A Gluster Cluster	4
Gluster server.....	4
Gluster client	5
NFS and CIFS considerations	5
Choosing hardware	5
Disk storage	5
Hardware RAID	6
Volume Managers	6
File systems	6
Operating systems	6
A Gluster storage server	7
Network connections.....	8
Common cluster file distribution methods	9
Distribute-only (RAID 0)	9
Distribute over mirrors (RAID 10).....	10
Stripe.....	10
Mixed environments	10
Data Flow in a Gluster Environment.....	11
Gluster File System client only.....	11
Accessing the Gluster Cluster via other protocols	12
Conclusion.....	13
Glossary	14
References and Further Reading	15

Introduction to Gluster

Gluster software simplifies storing and managing large quantities of unstructured data using commodity hardware. This guide explains fundamental concepts required to understand Gluster. The intended audience includes:

- Storage architects
- Systems administrators
- IT management

It may be helpful to refer to the glossary on page 14 while reading this guide.

Overview

Gluster provides open sourceⁱ storage software that runs on commodity hardware. The Gluster File System [distributed file system](#) aggregates disk and memory resources into a pool of storage in a single [namespace](#) accessed via multiple file-level protocols. Gluster uses a scale-out architecture where storage resources are added in building block fashion to meet performance and capacity requirements.

Gluster can be deployed in two ways; as [userspace](#) software installable on most major 64-bit Linux distributions and as a software appliance that integrates the Gluster File System file system with an operating system and includes a Web GUI for management and installation.

Note that regardless of the packaging, the underlying Gluster File System code is the same.

Gluster File System

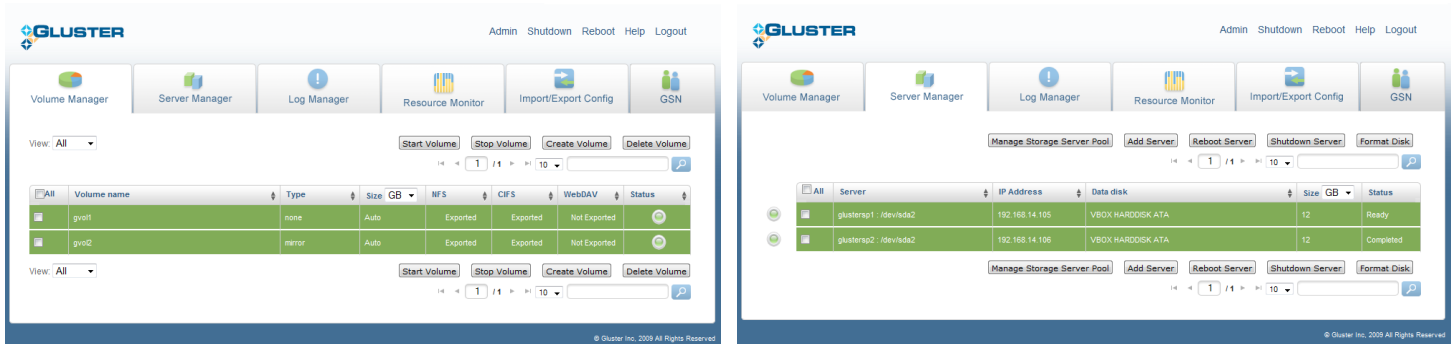
Gluster File Systemⁱⁱ is the core of the Gluster solution. Distributed as RPM and Debian packages in addition to source codeⁱⁱⁱ, it is supported on most major Linux distributions (64-bit). As a user space application it can be installed alongside applications and be re-exported as NFS, CIFS, WebDAV, FTP, and HTTP(S). Gluster File System supports locally attached, iSCSI, and Fibre Channel storage. Managed with a single command, Gluster File System is a flexible distributed file system.

Gluster Storage Platform

Gluster Storage Platform^{iv} is a comprehensive storage management offering. It includes its own Linux-based operating system layer as well as a GUI installer and web interface for administration. The software is installed via USB onto commodity hardware (64-bit) and uses all of the locally attached storage. Storage servers can be added easily and

quickly. The Platform exports the Gluster File System protocol, NFS and CIFS simultaneously. Gluster supports distributing, mirroring and striping data and can be configured to be completely redundant. To simplify administration, the Platform version has most of the configurable features turned on, limiting configuration access to a subset of the features available in Gluster File System.

Figure 1, 1a – Screenshots of the Gluster web interface.



No metadata with the Elastic Hash Algorithm

Unlike other distributed file systems Gluster does not create, store, or use a separate index of [metadata](#) in any way. Gluster distributes and locates data in the cluster on-the-fly using the [Elastic Hash Algorithm](#). The results of those calculations are dynamic values acquired as needed by any one of the storage servers in the cluster without the need to look up the information in a metadata index or communicate with other storage servers. The performance, availability, and stability advantages of not using metadata are significant.

A Gluster Cluster

A Gluster cluster is a collection of individual commodity servers and associated back-end disk resources exported as a [POSIX](#) compliant file-level protocol. All of the storage servers will run Gluster File System or Gluster Storage Platform. File locking is managed automatically and in a predictable way. **Every storage server in the cluster is active, and any file in the entire namespace can be accessed from any server using any protocol simultaneously.**

Gluster implements the distributed file system with two pieces of software; the Gluster server and the Gluster client.

Gluster server

The Gluster server clusters all of the physical storage servers in an all-active cluster and exports the combined disk space of all of the servers as the Gluster file system, NFS, CIFS, DAV, FTP, or HTTP(S).

Gluster client

The optional Gluster client implements highly-available, massively-parallel access to every storage node in the cluster simultaneously. With the Gluster client installed on existing applications servers, failure of any single node is completely transparent. The Gluster client exports a completely POSIX compliant file system. The Gluster client is available for 64-bit Linux systems only and is strongly recommended over other protocols when possible.

NFS and CIFS considerations

If using the Gluster File System client is not feasible, Gluster fully supports both NFS and CIFS. However there are some challenges with these protocols.

- Each of these protocols require additional software for a load balancing layer, usually [RRDNS](#) and a layer to provide high availability, usually [UCARP](#) or [CTDB](#), which add additional layers of complexity to the storage environment.
- Unlike when using the Gluster File System client, under certain configurations failure of a node can result in an application error. In a mirrored environment using UCARP or CTDB application servers will automatically connect to another storage server.
- Because neither NFS nor CIFS enable parallel access to clustered storage, they are generally slower than using the Gluster File System client.

Choosing hardware

Gluster runs on commodity, user supplied hardware and in the case of the Gluster File System a customer installed operating system layer and standard file systems. Each storage server in the Gluster cluster can have different hardware, in the case of mirror pairs we suggest the same amount of disk space per server, Gluster will not expose the additional space on a mirrored storage server, much like a standard [RAID 1](#).

Disk storage

Gluster File System supports the following storage attachment options:

- Locally attached (SAS, SATA, JBOD, etc.)
- Fibre Channel
- iSCSI
- Infiniband

Gluster Storage Platform only supports locally attached storage and does not support software RAID.

With any type of storage, Gluster recommends hardware RAID. Software RAID will work, but there are performance issues with this approach. Gluster only provides redundancy at the server level, not at the individual disk level.

Hardware RAID

For data availability and integrity reasons Gluster recommends RAID 6 or RAID 5 for general use cases. For high-performance computing applications, RAID 10 is recommended.

Volume Managers

Volume managers are fully supported but are not required. LVM2, ZFS, and Symantec Storage Foundation volume managers are known to work well with Gluster. Gluster fully supports snapshots created by a volume manager.

File systems

All POSIX-compliant file systems that support extended attributes will work with Gluster, including:

- ext3
- ZFS
- btrfs (experimental)
- ext4
- XFS (can be slow)

Gluster tests with and recommends ext3, ext4, and ZFS. There are known challenges with other file systems. For kernel versions 2.6.30 or below, Gluster recommends ext3. For kernel versions 2.6.31 and above, ext4 is the suggested option. Customers planning on using a file system other than ext3 or ext4 should contact Gluster.

Operating systems

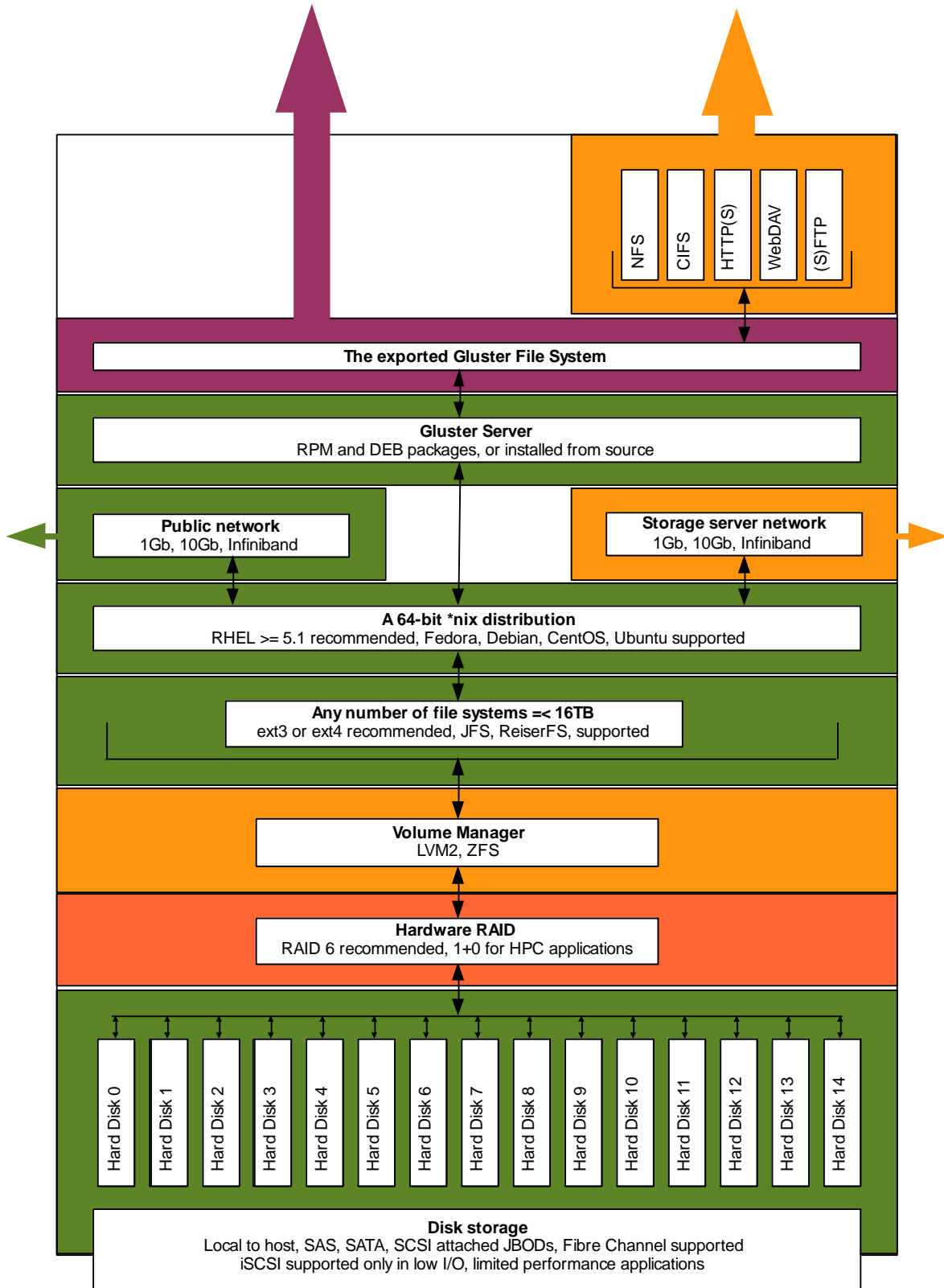
Gluster File System supports most 64-bit Linux environments, including:

- RHEL (5.1 or newer)
- CentOS
- Fedora
- Ubuntu
- Debian
- Solaris (server only)

A Gluster storage server

Figure 2 - An illustration of all of the components, both required and optional, of a Gluster storage server.

required
strongly recommended
optional
a Gluster service



Network connections

Gluster supports:

- 1GbE and 1GbE bonded interfaces
- 10GbE and 10GbE bonded interfaces
- InfiniBand SDR, QDR, and DDR using IB-Verbs (recommended) or IPoIB

Performance is typically constrained by network connectivity. If 1GbE interconnect is not sufficient Gluster also supports 10GbE and InfiniBand connections. If clients are not using the Gluster File System client server-to-server network communication can consume network resources and impact performance. In these situations, Gluster recommends a second, dedicated “back-end” network for Gluster server communications.

Common cluster file distribution methods

These common use cases help illustrate how Gluster might be implemented. These examples include:

- Distribute-only (RAID 0)
- Distribute over mirrors (RAID 10)
- Stripe
- Mixed environments.

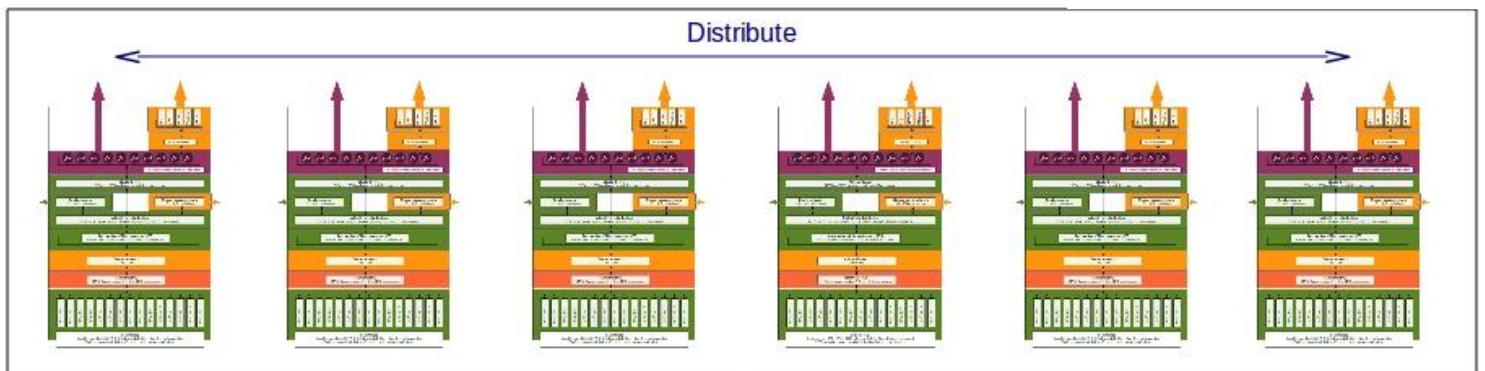
Distribute-only (RAID 0)^v

In this example (shown in figure 3), Gluster distributes files evenly among the storage servers using the Elastic hash algorithm described above. Each file is stored only once. The advantage of a distribute-only Gluster cluster is lower storage costs and the fastest possible write speeds.

There is no fault tolerance in this approach. Should a storage server fail;

- Reads from the failed storage server will fail.
- Writes destined to the failed storage server will fail.
- Reads and writes on all other storage servers will continue without interruption.

Figure 3 – Distribute-only

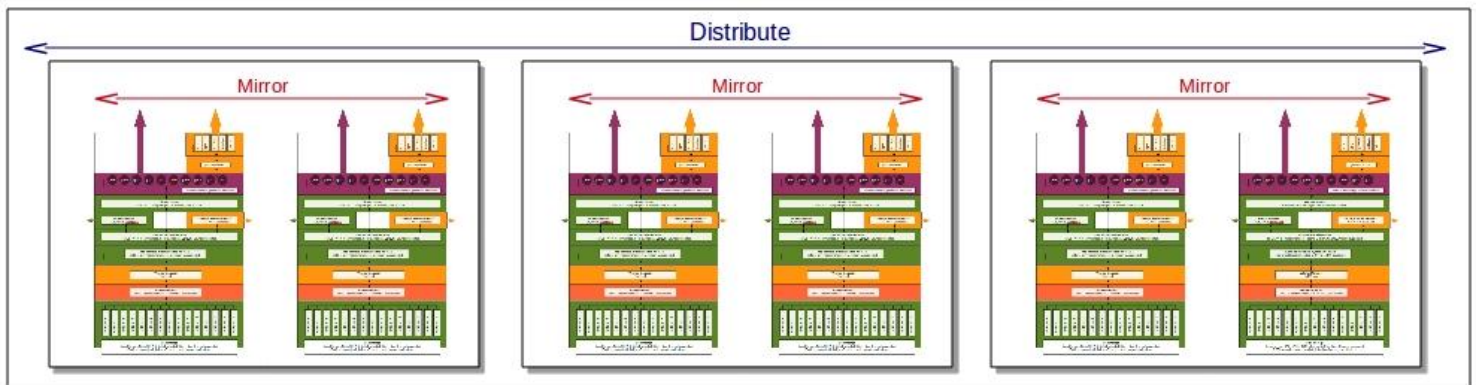


In a 6 storage server Gluster Cluster running only distribute the Elastic Hash Algorithm will distribute $(100/6) \sim 17\%$ of the files in the file system onto each server. Any single file will be stored on a single storage server.

Distribute over mirrors (RAID 10)

In this scenario (illustrated in figure 4), each storage server is replicated to another storage server using synchronous writes. The benefits of this strategy are full fault-tolerance; failure of a single storage server is completely transparent to Gluster File System clients. In addition, reads are spread across all members of the mirror. Using Gluster File System there can be an unlimited number of members in a mirror, using Gluster Storage Platform there can only be two members. The redundancy and performance gained by distribute over mirrors can increase storage costs. Mirroring without distributing is supported on Gluster clusters with only two storage servers.

Figure 4 – Distribute over mirrors



In a 6 storage server Gluster Cluster running distribute over mirror (RAID 10) the Elastic Hash Algorithm will distribute $(100/3) \sim 33\%$ of the files in the file system onto each mirror pair. Any single file will be stored on two storage servers.

Stripe

Gluster supports striping individual files across multiple storage servers^{vi}, something like a more traditional RAID 0. Stripe is appropriate for use cases with very large files (a minimum of 50GB) with very limited writes and simultaneous access from many clients. The default stripe size is 128KB; this means that all files smaller than that will always be on the first storage server, rather than spread across multiple servers. Distribute or mirror (over | under) stripe is not supported.

Mixed environments

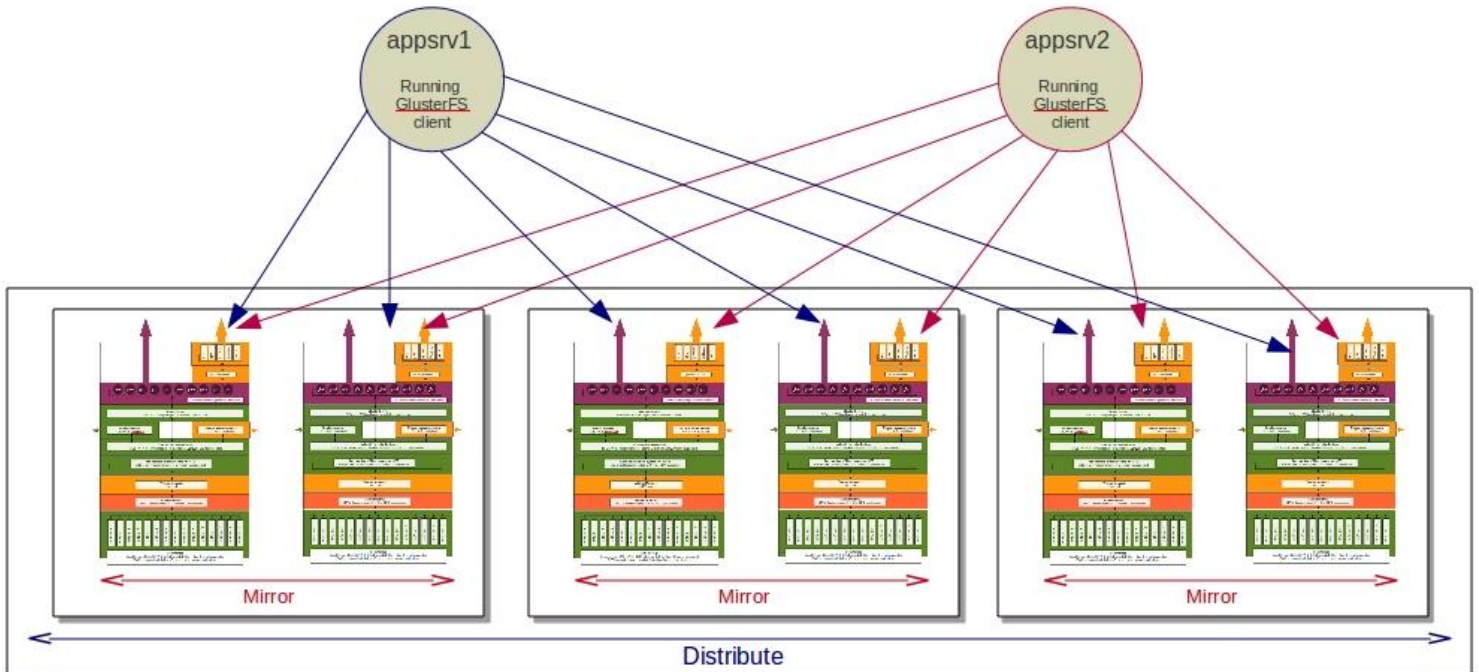
Any single, logical Gluster cluster is limited to a single distribution method; however it is possible to run multiple logical Gluster clusters on one set of hardware. By creating multiple volume specification files, each using a different port number it's possible to run clusters with different distribution types without additional hardware^{vii}. The Gluster client mounts the multiple file systems, making the multiple instances and various port numbers transparent to any application servers. This approach also takes advantage of improved processor parallelism on the storage servers.

Data Flow in a Gluster Environment

Depending on whether or not the Gluster File System client is used, there are two different ways that data flows within a Gluster environment.

Gluster File System client only

Figure 5 – Data flow using the GlusterFS client only



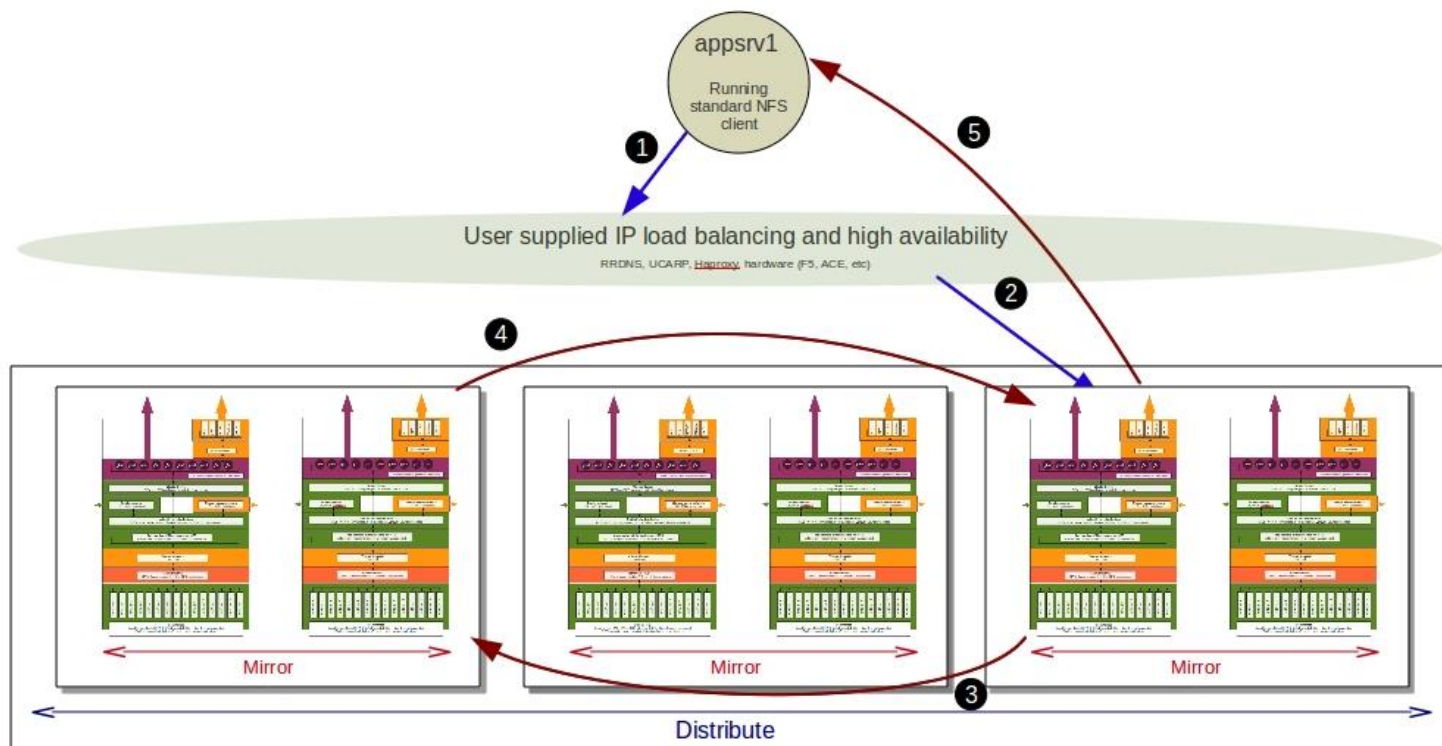
Using the Gluster File System client, every application server maintains parallel connections to every Gluster storage server. Applications access the file system via the standard POSIX interface while the Gluster client transparently enables massively parallel access to the file system.

This architecture is extremely scalable yet easy to deploy. In addition to its performance advantages, it also offers robust fault-tolerance, even if a storage server fails during I/O, application servers aren't aware of this event as long as mirroring has been setup.

Gluster supports data access via the Gluster File System and any other protocol simultaneously.

Accessing the Gluster Cluster via other protocols

Figure 6 – Data flow using NFS, CIFS, CIFS, DAV, FTP, or HTTP(S).



The following is the process for accessing files using protocols other than native Gluster File System:

1. The application server (client) connects to a customer-supplied load balancing layer (usually RRDNS and/or UCARP, CTDB).
2. The load balancer directs the request to one of the storage servers in the cluster.
3. If the file is not on the selected storage server, the Gluster server requests the file from the correct storage server.
4. The file is then sent to the selected server.
5. And then sent to the application server (client); note the application server immediately starts receiving data and does not wait for the entire file to be delivered.

The back-end file access process is completely transparent to the application server regardless of the protocol. Any client can connect to any Gluster storage server for any file. If the storage server fails during I/O, the client will receive connection reset. In addition, this approach can create significant server-server communication and typically a dedicated, private communication network for storage server to storage server is used.

Gluster supports accessing the same file via any protocol simultaneously. File locking is managed at the file and block-range level in a POSIX compliant manner.

Conclusion

By delivering increased performance, scalability, and ease-of-use in concert with reduced cost of acquisition and maintenance, the Gluster Storage Platform is a revolutionary step forward in data management. The complete elimination of metadata is at the heart of many of its fundamental advantages, including its remarkable resilience, which dramatically reduces the risk of data loss, data corruption, or data becoming unavailable.

To download a copy of Gluster File System and Gluster Storage Platform, visit us at <http://gluster.com/products/download.php>.

If you would like to start a 60 day proof concept, including direct access to Gluster support engineers for architecture and design through testing and tuning please visit Gluster at <http://www.gluster.com/products/poc.php>.

To speak with a Gluster representative about how to solve your particular storage challenges, phone us at +1 (800) 805-5215.

Glossary

Block storage: Block special files or block devices correspond to devices through which the system moves data in the form of blocks. These device nodes often represent addressable devices such as hard disks, CD-ROM drives, or memory-regions.^{viii} Gluster supports most POSIX compliant block level file systems with extended attributes. Examples include ext3, ext4, ZFS, etc.

Distributed file system: is any file system that allows access to files from multiple hosts sharing via a computer network.^{ix}

Metadata: is defined as data providing information about one or more other pieces of data.^x

Namespace: is an abstract container or environment created to hold a logical grouping of unique identifiers or symbols.^{xi} Each Gluster cluster exposes a single namespace as a POSIX mount point that contains every file in the cluster.

POSIX: or "Portable Operating System Interface [for Unix]" is the name of a family of related standards specified by the IEEE to define the application programming interface (API), along with shell and utilities interfaces for software compatible with variants of the Unix operating system.^{xii} Gluster exports a fully POSIX compliant file system.

RAID: or "Redundant Array of Inexpensive Disks", is a technology that provides increased storage reliability through redundancy, combining multiple low-cost, less-reliable disk drives components into a logical unit where all drives in the array are interdependent.^{xiii}

RRDNS: or "Round Robin Domain Name Service". RRDNS is a method to distribute load across application servers. It is implemented by creating multiple A records with the same name and different IP addresses in the zone file of a DNS server.

Userspace: Applications running in user space don't directly interact with hardware, instead using the kernel to moderate access. Userspace applications are generally more portable than applications in kernel space. Gluster is a user space application.

References and Further Reading

CTDB

CTDB is primarily developed around the concept of having a shared cluster file system across all the nodes in the cluster to provide the features required for building a NAS cluster.

<http://ctdb.samba.org/>

Elastic Hash Algorithm

Get a more detailed explanation of the Elastic Hash Algorithm here -

http://ftp.gluster.com/pub/gluster/documentation/Gluster_Architecture.pdf

UCARP

Find more information and configuration guides for UCARP here -

<http://www.ucarp.org/project/ucarp>

ⁱ GlusterFS is released under GNU General Public License v3 or later. Documentation is released under GNU Free Documentation License v1.2 or later.

ⁱⁱ GlusterFS is available at <http://ftp.gluster.com/pub/gluster/glusterfs/>

ⁱⁱⁱ The Gluster source code repository is at <http://git.gluster.com/>, it is strongly suggested that users only install Gluster from the source and packages at <http://ftp.gluster.com>, the code in the Git repository has not been through the QA process.

^{iv} GlusterSP is available at <http://ftp.gluster.com/pub/gluster/gluster-platform/>

^v To create a distributed file system using the GlusterSP web interface choose "None" as the volume type.

^{vi} Striping requires at least four storage servers.

^{vii} Check the [glusterfs-volgen man](#) page, -p option for more information.

^{viii} http://en.wikipedia.org/wiki/Device_file_system#Block_devices

^{ix} http://en.wikipedia.org/wiki/Distributed_file_system

^x http://en.wikipedia.org/wiki/Metadata#Metadata_definition

^{xi} http://en.wikipedia.org/wiki/Namespace_%28computer_science%29

^{xii} <http://en.wikipedia.org/wiki/POSIX>

^{xiii} <http://en.wikipedia.org/wiki/RAID>