GlusterFS Cloud Storage

John Mark Walker | Gluster Community Leader, RED HAT
Adventures in Cloud Storage

John Mark Walker
Gluster Community Leader
November 8, 2013
Topics

- Usage and Performance
- GlusterFS & Gluster Community
- GlusterFS – OpenStack Integration
- Related Projects
The big idea:
Storage should be simple
Core design principles:

- No data silos
- No single point of failure
- Global namespace
What is GlusterFS?

Gluster is a unified, distributed storage system

- User space, global namespace, stackable, scale-out arch, inspired by GNU Hurd
- Everything treated as a file
Ghosts of Gluster Past

• Nice idea, but kinda slow
• Ate my data
• Doesn't perform well for block storage
• No object storage
Ghosts of Gluster Past

• A lot has changed in 2 years:

<table>
<thead>
<tr>
<th>2011</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>~20 developers</td>
<td>~75 developers</td>
</tr>
<tr>
<td>NFS v3 and GlusterFS client</td>
<td>NFS v3, GlusterFS client, SAMBA, QEMU, Swift API, libgfapi</td>
</tr>
<tr>
<td>Long release cycle (~1 year)</td>
<td>Shorter release cycle (4 – 6 mos.)</td>
</tr>
<tr>
<td>No OpenStack integration</td>
<td>File, block and object OpenStack integration</td>
</tr>
</tbody>
</table>
GlusterFS Performance

Throughput with four compute nodes

<table>
<thead>
<tr>
<th>Total VMs</th>
<th>Read</th>
<th>Write</th>
<th>Read</th>
<th>Write</th>
<th>Read</th>
<th>Write</th>
<th>Read</th>
<th>Write</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 VMs</td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td>8 VMs</td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td>12 VMs</td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td>16 VMs</td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
</tbody>
</table>
GlusterFS Overview

Storage Servers

- Glusterd
  - Brick 1
  - Brick 2
  - ...

- Glusterd
  - Brick 3
  - Brick 4
  - ...

- Glusterd
  - Brick 1
  - Brick 2
  - ...

- Glusterd
  - Brick 3
  - Brick 4
  - ...

Swift API (Fuse)

NFSv3 Client

QEMU /KVM

Libgfapi

SAMBA

The Network
Client Access Overview

- Swift API
- GlusterFS Client (Fuse)
- Glusterd Brick 1
  - Brick 2
  - ...
- QEMU /KVM
- Libgfapi
  - Glusterd Brick 3
    - Brick 4
    - ...
- SAMBA
- NFSv3 Client
- Glusterd Brick 3
  - Brick 4
  - ...

Storage Servers
Some Features

• No metadata server
  • Elastic Hash (DHT xlator)
  • Distributed over multiple servers
• Multi-protocol access
• Replication (synchronous and async)
• Proactive self-healing
• No silos – files, objects and block devices all in the same namespace.
Making Hard Stuff Easier

- The easiest DFS to set up, manage
- Up and running in four CLI commands
  - probe peer, create volume, start volume, mount
- Cluster membership, process management, port mapping, dynamic config changes, etc.
  - add/remove nodes on the fly
  - add/remove features on the fly
  - rolling upgrade
Q: How Do We Do It?

A: Modularity!
What Can You Do With It?

- **Media** – Docs, Photos, Video
- **Shared storage** – multi-tenant environments
- **Big Data** – Log Files, RFID Data
- **Objects** – Long Tail Data
Storage for Any Environment

- Consistency across multiple platforms
- Single namespace

Private Cloud  ➔  Public Cloud

On-premises

Public Cloud
Multi-protocol Access

- HDFS, Swift API, NFS, GlusterFS client, SAMBA, QEMU

**Diagram:**
- **Object (Swift)**
  - HTTP Request
    - ID=/dir/sub/sub2/file
  - Swift Proxy
    - Account
    - Container
    - Object

- **File**
  - NFS, SMB or GlusterFS Mount

- **Block (QEMU)**
  - gfapi library

- **Global Namespace**
  - Volume
  - Directory
  - File

11/11/13
GlusterFS for Virtualization Use

- QEMU-GlusterFS integration
  - Native integration using libgfapi, no FUSE mount
  - GlusterFS as QEMU block back end
  - QEMU talks to gluster and gluster hides different image formats and storage types underneath
    -drive file=gluster://server[:port]/volname/image[?transport=...]

- Block device support in GlusterFS via Block Device translator
GlusterFS & QEMU
Libgfapi Client API

- Previously abandoned
- Brought back to life
  - In part because of QEMU Fuse bypass contributions
- Basis for Nova integration
- SAMBA integration, plus all non-NAS use cases
libgfapi v/s FUSE – FUSE access

Client Node

Server Node
libgfapi v/s FUSE – libgfapi access
OpenStack Integration

- Separate Compute and Storage Pools
- GlusterFS directly provides Swift object service
  - GeoReplication for multi-site support
- Swift data also available via other protocols
- Supports non-OpenStack use in addition to OpenStack use
OpenStack Integration Points

- **Grizzly**
  - Swift add-on
  - Cinder integration (FUSE mounted FS)
  - Glance integration (FUSE mounted)

- **Havana**
  - Swift implementation “baked in” upstream Swift
  - Nova integration (libgfapi)
  - Cinder integration via libgfapi (per Nova)
Glance Integration

- Shared storage capacity
- Geo-Replication
- Storage resizing

Nova Integration

- Live migration
- Protection using replicated storage
- Enhanced performance using libgfapi (Havana)
Swift Integration

- Collaborating with upstream Swift project
- Making Swift API pluggable for multiple storage backends (including GlusterFS)
  - Icehouse?
- Pushing Gluster implementation upstream

Cinder Integration

- Hypervisor-assisted snapshots
- Mostly complete capability matrix
OpenStack and GlusterFS

• Why would you do this?
  • No silos for Block, File, Object and HDFS
  • Modular extensible architecture
  • Choice of transport – RDMA, IP
  • Data locality
  • Compute/Virtualization transparent storage
  • Easy manageability and maintenance
Implementation Details
Cinder

- GlusterFS volume driver
  - Added in Grizzly
  - Improved in Havana
- Havana features
  - Snapshots
  - Cloning
  - Copy to/from image
  - Extend volume
Cinder Volume Snapshots

- *WARNING* Cinder volume != Gluster volume
- Snapshots are QCOW2 file chains
  - Detached: Cinder creates snapshot files
  - Attached: Cinder coordinates with Nova
    - libvirt and QEMU create snapshot live
Nova libgfapi

• Havana supports QEMU libgfapi integration
  • nova.conf
    - qemu_allowed_storage_drivers=gluster

• Limitations
  • Needs work to function with Cinder volume snapshots
Cinder + GlusterFS Roadmap

- Guest quiescing for snapshots
- Volume backup
Deployment

• Packstack
  • CONFIG_CINDER_BACKEND=gluster
  • CONFIG_CINDER_GLUSTERFS_MOUNTS=192.168.7.7:/volume1
  • For Libgfapi/Fuse bypass: GlusterFS 3.4+, QEMU 1.3+
OpenStack Future Integration

- Manila Project – Icehouse
  - Shared FS management project for OpenStack.
  - Provides tenant-specific file shares
  - Planned for incubation
  - More details: https://launchpad.net/manila
Savanna Project

- Mirantis, HortonWorks, Red Hat
- OpenStack becomes vehicle for distributing scale-out Hadoop MapReduce jobs
Organizational Structure

- **Gluster Community Board**
  - Votes on project graduation
  - Decides GCSD releases
  - Governs steering committees

- **GlusterFS Project**

- **HDFS Plugin**

- **Swift Integration**

- **Pmux + gflocator**

- **SAMBA Integration**

- + lots of other projects...

- Projects guided by independent technical committees
The Goal

• Create the world's largest and most dynamic community for open software-defined storage
More releases -

More aggressive backporting of fixes to point releases
More documentation -

Community projects to lead docs development
Heya johnmark!

Flagship Projects

**GlusterFS Core**
This is the core platform for GlusterFS, providing all the major feature functionality

Project Home | Developer Home | Documentation

Incubating Projects

**pmux**
Pmux is a lightweight file-based MapReduce system, written in Ruby. Applying the philosophy of Unix pipeline processing to distributed computing on a GlusterFS cluster, pmux provides a tool capable of handling large amounts of data stored in files.

Project Home | Developer Home | Documentation

From Planet Gluster

**GlusterFS is Ready for OpenStack**
Amidst the madness of the OpenStack Summit a couple of weeks ago, you could be forgiven for not seeing a Red Hat announcement about GlusterFS being "OpenStack-ready". You may wonder, what exactly do we mean by “OpenStack-ready”?

Tue, 07 May 2013 07:57:50 +0000

**PHP playing fast and loose with your data integrity**
Had a potential GlusterFS user state that the filesystem incorrectly reported that a write succeeded even though all the servers were powered off. Since this sounded rather impossible, I asked for details and duplicated the problem. This is the php cod...

Tue, 07 May 2013 00:21:29 +0000
On the Forge

PMUX

- lightweight file-based MapReduce system, written in Ruby.
- Includes HTTP gateway, log viewer
- Run distributed grep: `pmux --mapper="grep PATTERN" /glusterfs/xxx/\*\.log`
- forge.gluster.org/pmux
On the Forge

Gluster-Swift

• Swift implementation layer for GlusterFS
• Available at forge.gluster.org/gluster-swift
On the Forge

HDFS Plugin

• Drop-in compatibility w/ HDFS
  – Per HCFS guidelines
• Available at forge.gluster.org/hadoop
http://gluster.org/presos.php

johnmark@redhat.com

Twitter: @johnmark