SOLUTION BRIEF

PeerGFS with Dell PowerScale

USING PEERGFS FOR
HYBRID CLOUD MEDIA WORKFLOWS

Overview

Media and Entertainment companies are facing increasingly difficult data challenges, accelerated by two primary drivers. On one hand, data continues to grow at an alarming scale, driven by increasing resolutions and the sheer volume of content being produced to meet viewers' insatiable demand for content. On the other hand, the need for global workflows is increasing across the industry as companies look to access new talent pools of creatives and break down geographic barriers to collaboration between them.

MovieLabs 2030 Vision sets out 10 principles as to where we expect the entertainment industry to be by the year 2030. Much of the emphasis is on 'cloud'. The reality is that most organizations have been running their own private clouds for many years without even realizing it. For example, running Dell VxRail Hyper-Converged Infrastructure with native vSAN can be classed as a "private cloud" because it contains all the necessary automation, compute, storage, and network infrastructure to serve out any workload.
Workflow and process transformation are at the heart of the MovieLabs paper. By defining where the industry is aiming, and the path that must be taken to get there, will help organizations to begin future proofing and shaping the future of the industry.

The usual challenge presents itself, however: Will traditional IT technology advance fast enough to keep up with the ever-growing needs of media & production IT / Tech and make the 2030 vision a reality?

Global file systems or global storage silos?

Global file systems have been in use in the entertainment industry for years under various guises. However, they are often just a form of replication between two or more discreet silos of storage that replicate changes in data at a given point in time, or they use a simple watch folder workflow to synchronize known data sets between locations.

When the goal is to unify file storage globally, there are a few things that come into play to enable a truly collaborative global file system.

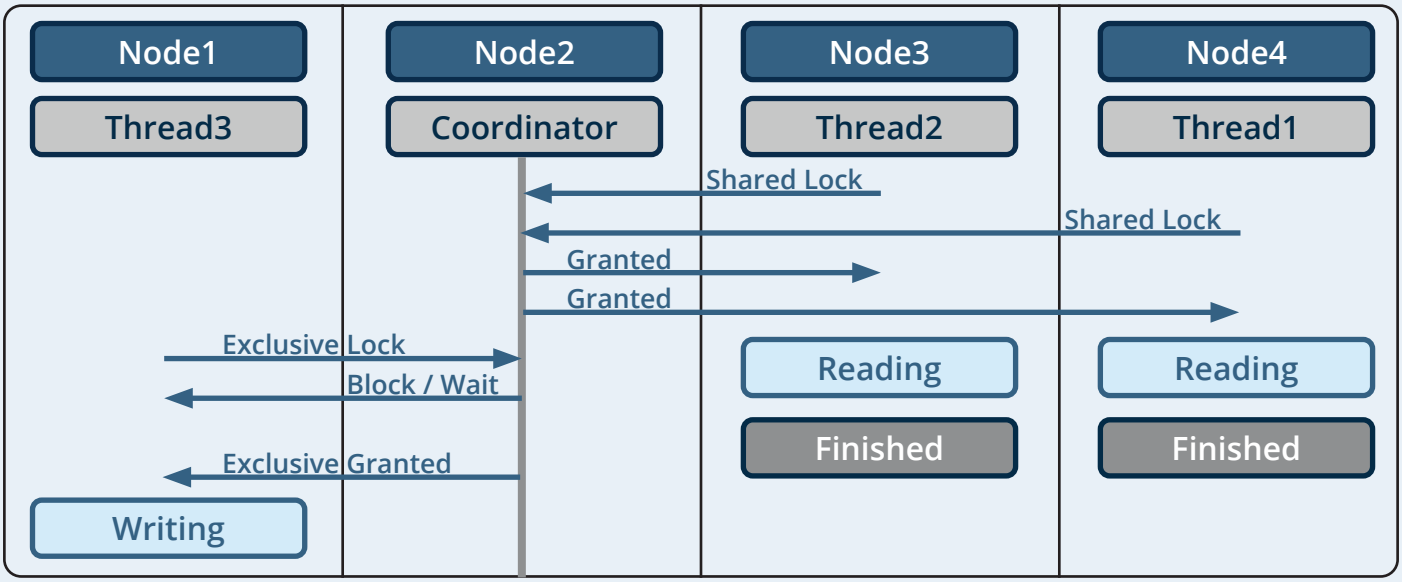
- File Locking
- Network Latency
- Data Consistency

FILE LOCKING

File locking is in place to avoid potential corruption and write conflicts when users work with files on a filesystem. Users work on the assumption that when they open a file for modifying, other users will not be able to modify the same file at the same time.

On a single site, PowerScale OneFS enables this with a cluster wide lock manager. When a given user opens a file for writing, OneFS actively tells all other nodes in a cluster that the file is in use by UserX at WorkstationY and puts a 'lock' on the file. When another user tries to open the file for writing they receive an error message stating that the file is already in use.

Global file systems or global storage silos?



PowerScale OneFS cluster-wide file locking

Some applications grant permission to open a file in a 'Read-Only' state and subsequently inform the user when it becomes available for writing. Some also inform the user who is trying to access the file whom it is who currently has the file open for writing.

When we expand out past a single storage volume, site, or silo to multiples in different geographies, we are now just working on siloed copies of the same data at each location. If the data is not properly managed or version controlled, the lack of geographically distributed file locking can, as mentioned above, cause replication / save conflicts and file inconsistency. In essence, this is not a Global file system at all, and more resembles a semi-synchronous global file silo.

LATENCY

Typically, when talking about latency in storage, we are referring to the "round trip time" taken for a request to be acknowledged by any given client. That is usually as deep as we go as a piece of storage, because we are then at the mercy of the client side network.

The problem deepens however, when it comes to global file systems. The reality of network latency over a WAN causes issues that are not seen in LAN environments. That is because we must also try to mitigate the time it takes data to move across the world. You end up in a catch 22 of either having fast access to siloed data in multiple locations, or all of your data visible everywhere which suffers with increased latency back to the origin storage.

DATA CONSISTENCY

The final piece of the puzzle is: How do we keep data up to date everywhere, accessible by everyone, and protected? PowerScale OneFS and PeerGFS: Solution and approach

Overview

There are multiple ways of addressing a global filesystem challenge - Object Core with a Performance File Edge, Object Core with a Client Side File Cache, and so on.

PeerGFS addresses this in a slightly different way. Using File storage everywhere and keeping the data in sync by integrating with native auditing APIs, PeerGFS ensures that modifications are taken care of in real-time and changes are distributed to the other storage targets as quickly as possible.

PeerGFS is a software-only solution developed to solve file management and file replication challenges in multi-site, multi-platform, and hybrid multi-cloud environments. It directly integrates with PowerScale OneFS, using out of band agent servers to monitor the real time events on a given cluster or clusters and then transfers the delta changes from one storage target to one or more others.

PeerGFS provides the data orchestration and management tool that keeps all the managed file systems in sync. It also preserves all the extended metadata attributes of a file on PowerScale OneFS during synchronization (time stamps, permissions ACEs, and so on).

The platform supports multiple job types for different storage targets and workflows. In this paper, we focus on the File Collaboration, Cloud Backup, and Cloud Replication jobs because they directly integrate with PowerScale OneFS to provide a real-time global file system experience.

How PeerGFS addresses the problems

In opposition to many global filesystems, PeerGFS sits outside of the data path between user and storage, using real-time messaging brokers to learn about what is happening on a storage target.

This means that you are not bound to a proprietary gateway or access method to get to your data. Additionally, you are not performance restricted by the hardware that is serving it. It also maintains locks on all files across every storage target under its management.

FILE LOCKING

PeerGFS is constantly monitoring the audit event logs of the storage, keeping track of changes to enable synchronization. When a file is opened for write, a real-time metadata replication occurs and places a lock on the file on all of the other storage targets managed by the policy. This means that even though a file is being modified in Site 1, all other locations will have a lock to prevent overwrites or corruption from occurring.

NETWORK LATENCY

By synchronizing the data or metadata changes at an incremental delta level, you decrease the amount of data being sent over the WAN between your synchronization target(s). This minimizes the amount of time spent transferring data over the wire. Note that some types of files will need to be synced in their entirety if a modification causes an entirely new file to be created.

Maintaining the required data set in real time at a storage target that is close to the user effectively eliminates the WAN network latency problem by ensuring that your data is always current on each site.

DATA CONSISTENCY

When PeerGFS is managing storage targets, it is keeping them all in check with each other, maintaining lock status and updating the incremental changes of data in the background to keep the filesystems up to date. If something were to happen that caused a site or piece of storage to go offline, the transactional logs of the PeerGFS system are stored as a changelist within the system until the storage target returns or is removed. When the storage target re-connects, the changelist is re-played and the storage target is updated with the required changes.

If the changelist is small, it parses using the PMC server and replays to any outdated targets. If however the changelist is large, the PMC performs a rescan of the volume that is out of date and brings it back up to date that way instead. While the filesystem scan is in progress, the real-time collaboration is already back up and running so that multi-site collaboration can continue as usual.

For jobs with more than two participants, collaboration will continue to work with the sites that are alive. If quorum is lost, the impacted job pauses.

There is also an option to require all storage targets to be online for a given job to run (this is not enabled by default). The default behavior is that the offline site is bought up to date when it is back online.

What makes PeerGFS different?

Peer Software addresses this problem differently, compared to other Global Filesystem vendors, because PeerGFS does not sit in the data path to function. Unlike other global file system vendors where your data must go through a proprietary gateway with files locked into another ecosystem, or an object-based core/edge tiering approach where you have effectively an entire copy of your dataset sitting in multiple geographies being shuttled about by a data mover, PeerGFS is an overlayed data management solution that ensures data freedom and future agility while providing a globally consistent file system.

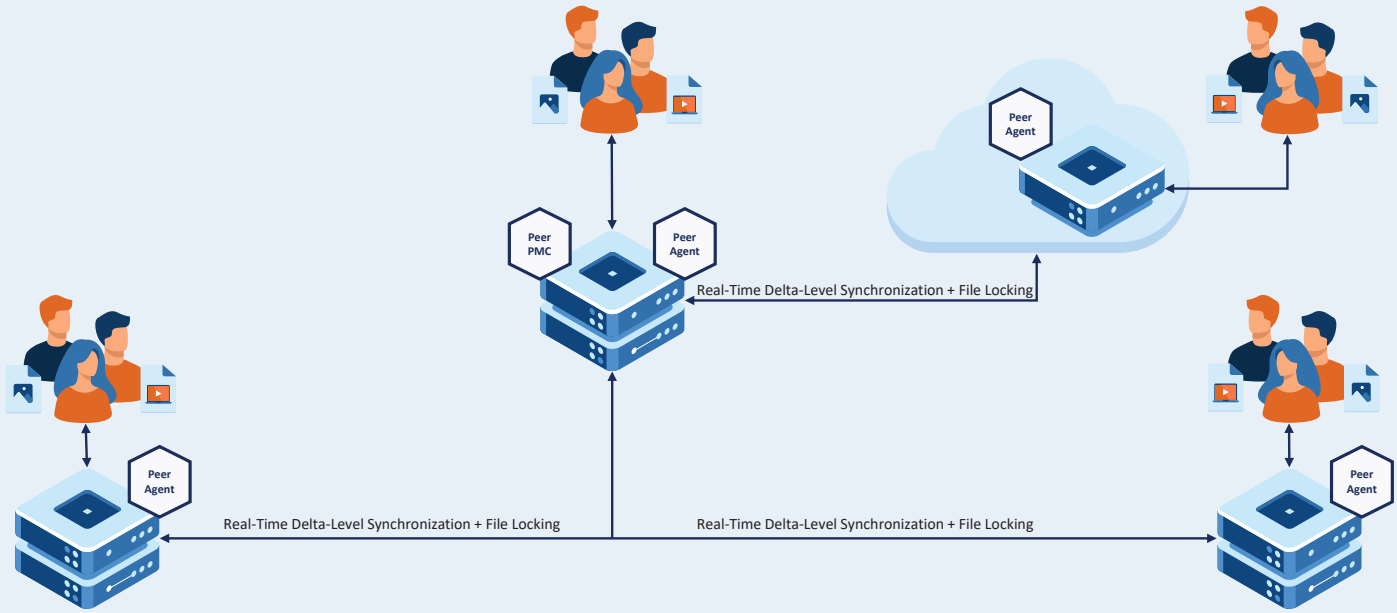
PeerGFS Achitecture

HIGH LEVEL ARCHITECTURE

Because PeerGFS is an overlayed system, it requires ancillary infrastructure to be deployed to monitor, move, and manage your data.

At a minimum, you must deploy:

- A PMC (Peer Management Console) instance.
  A single PMC instance manages all of the agents.
- Two or more PeerGFS Agent instances.
  One agent instance is required for each storage target / edge location.



Example PeerGFS high level architecture

 |

PeerGFS Test Scenarios

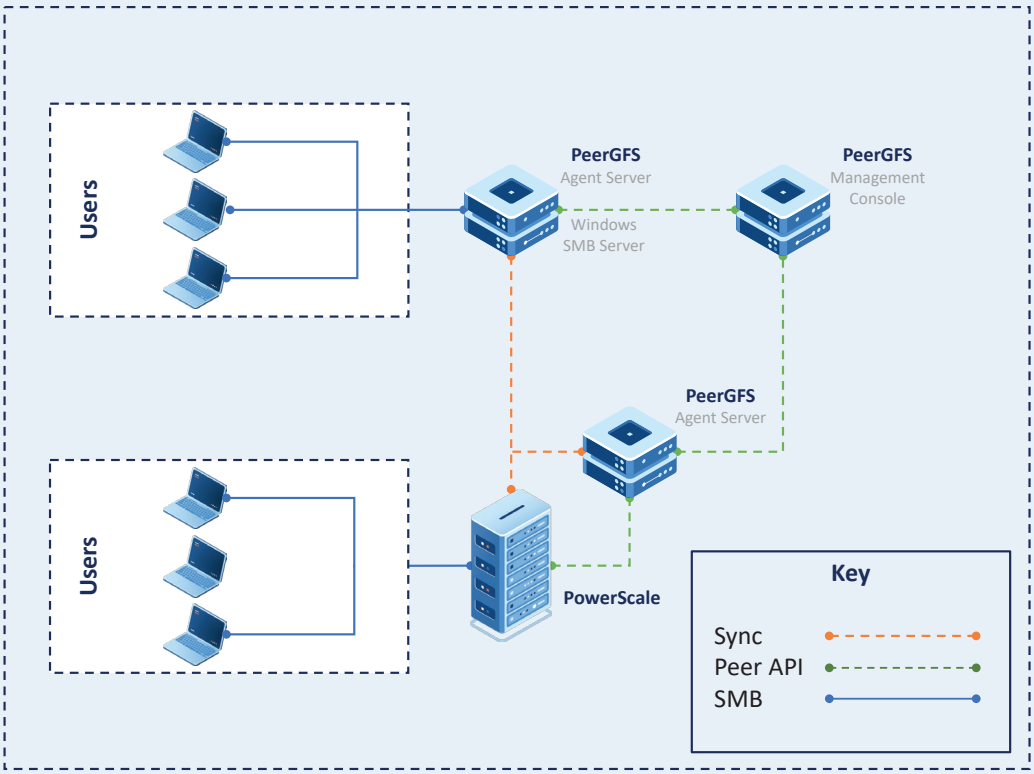
LAB INFRASTRUCTURE DEPLOYMENT

The test environment included a virtual estate containing MS Windows servers, a PowerScale OneFS cluster, and ECS Test Drive for functional testing of all of the PeerGFS capabilities.

Because PeerGFS does not sit in the data path, performance testing is not necessary because the storage platform that serves the data determines the performance of the locally accessed data.

TEST ENVIRONMENT INFRASTRUCTURE

- 1 x Windows 2016 Server VM (Management Console)
  - PeerGFS Management Console
- 1 x Windows 2016 Server VM (SMB Server PeerGFS Agent)
  - SMB Server Features
  - PeerGFS Agent
- 1 x Windows 2016 Server VM (OneFS PeerGFS Agent)
  - PeerGFS Agent
  - Dell CEE Framework
  - RabbitMQ
  - Erlang
- 1 x PowerScale OneFS 9.4 Simulator VM



Test Environment - Real-Time Collaboration

REAL-TIME COLLABORATION

The first phase of testing involved setting up both a PowerScale PeerGFS Agent and a native Windows SMB server environment to prove the real-time synchronization and locking between disparate storage targets.

You must then complete some pre-configuration steps on the Windows Peer Agent server and PowerScale OneFS to ensure that the API is able to communicate with the PMC servers. (For complete details, see the PowerScale OneFS PeerGFS Pre-requisites and Configuration Guide).

The next step is to configure a job. When configuring the collaboration job, we created two storage targets; the Windows SMB server, and the Windows Server running the Dell CEE framework. (This framework acts as the broker for the PowerScale system.)

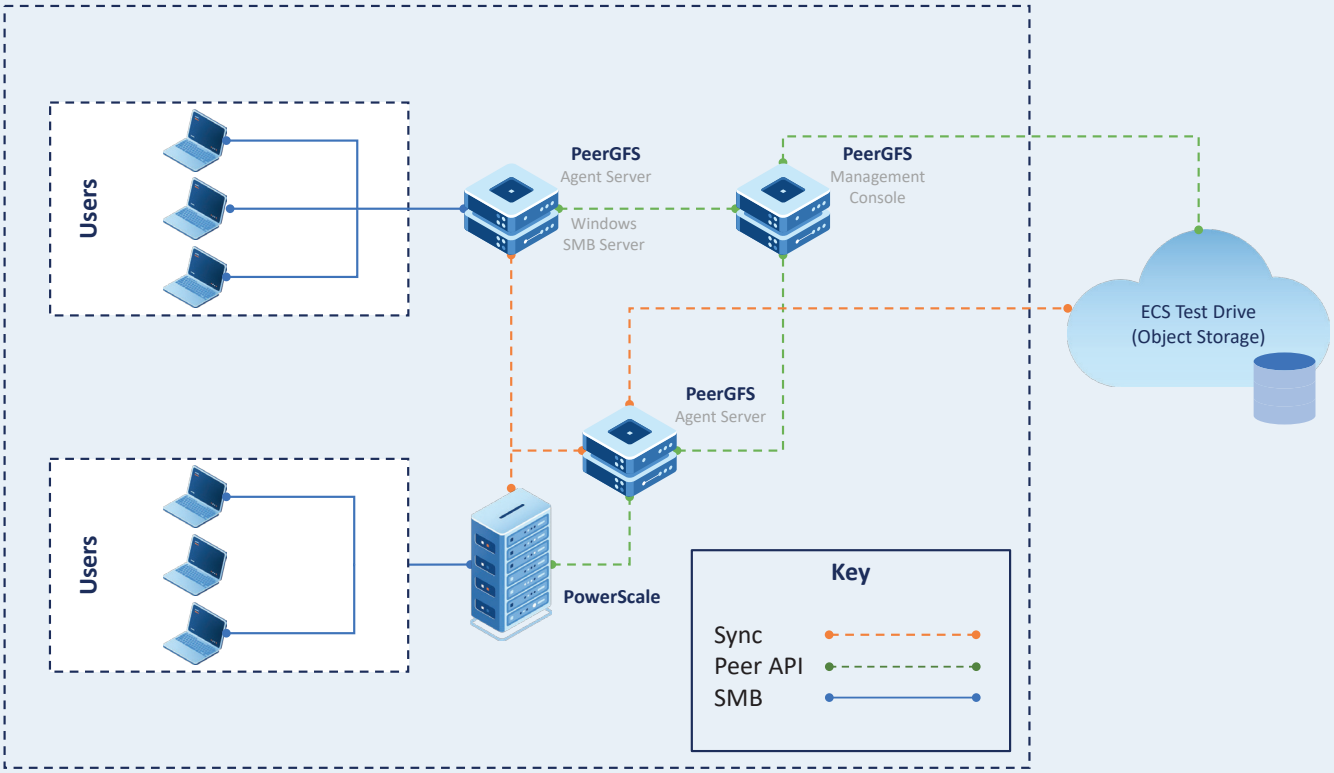
When the policy was configured there was instantaneous feedback from the PMC that files were being indexed and shared between the storage targets. The data began to appear almost instantaneously in the second target and changes were propagated in real-time.

PeerGFS Test Scenarios

The most obviously visible test of the PeerGFS file locking ability was to open an MS Word document and begin editing it, then open the same document on another client using the other storage target and receive a dialog message stating that "the file is currently being edited by another user, would you like to open it Read-Only?"

The same was true on a Windows and Mac when editing files. The dialog messages were different depending on the operating system and applications that were in use, but in general you received a 'cannot open this file, file is in use' type of message.

All of the Owner, DACL (Discretionary Access Control List), and SACL (System Access Control List) metadata was maintained on all storage targets. When permissions were changed on a file or folder, they were then updated on every storage target.



Test Environment - Collaboration with S3 Backup

COLLABORATION WITH S3 BACKUP

The second phase of this testing added a "Cloud tier" to the environment to offload backups to an S3 compatible target using a Cloud Backup & Replication job.

Configuring the backup job is a straightforward process. This requires just the source that is to be synchronized, the S3 target to which to sync, and a Retention Policy for the job.

This job type is intended to replace the traditional backup to tape methodology and to enable fast, reliable, and scalable backup that uses modern object storage targets such as Dell ECS / ObjectScale.

ABOUT US

Since 1993, Peer Software has developed data management solutions that address unique challenges related to data synchronization, backup, and file collaboration in multi-site (WAN) environments.

EMEA OFFICES

P +44 20 77098 22 (UK)
P +49 89 2620763 22 (DE)
P +971 4 424 5005 (ME)

sales.emea@peersoftware.com

US OFFICE (HQ)

P +1 703 763 7700

sales@peersoftware.com

 |