

EMC Celerra MPFS Performance Benefits with Celerra SnapSure and Celerra Replicator V2

Applied Technology

Abstract

This white paper explains how EMC[®] Celerra[®] MPFS performance is better when compared to NFS even when accessing a Celerra file system protected by EMC Celerra SnapSure[™] and/or EMC Celerra Replicator[™] V2.

April 2010

Copyright © 2010 EMC Corporation. All rights reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS IS.” EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com.

All other trademarks used herein are the property of their respective owners. Part number h7081

Table of Contents

Executive summary.....	4
Introduction	7
The storage demands of protected environments.....	8
MPFS performance measurement for Linux clients in protected environments	14
Conclusion.....	20
References	20

Executive summary

Business case Medium to large-size enterprises require best-in-class storage to meet their demand for high-performance shared file storage systems that can serve across multiple platforms and distributed environments. Meeting the needs of applications that require high throughput, near-linear scalability of both the client and storage system, and manageability are some of the biggest challenges for any organization.

Present-day applications inevitably require data protection; such data protection is applied using snapshot and/or replication technologies. However, these features may add overhead to the storage subsystem, thus affecting performance.

The major challenge for storage in these environments is meeting the combined need to provide continual, shared access to data, to ensure high levels of data protection and integrity, and achieve maximum performance while minimizing the idle compute time.

Product solution EMC® Celerra® Multi-Path File System (MPFS) accelerates NFS file sharing when in use with EMC Celerra SnapSure™ and/or EMC Celerra Replicator™ V2 to protect file system data.

The MPFS parallel data access architecture allows hundreds to thousands of computers to share files protected by snaps, at a speed limited only by the underlying storage subsystem configuration and topology.

Key results EMC Celerra MPFS allows customers to experience optimal performance compared to traditional NFS systems, even when protected using EMC Celerra SnapSure and/or EMC Celerra Replicator V2. As with NFS and CIFS, MPFS experiences an initial blip in write performance, followed by speeds equivalent to before the snap was made. Even in the worst case, MPFS performance is still better than NFS or CIFS.

[Figure 1](#) on page 5 illustrates results derived from extensive testing with a 100 percent sequential write I/O load, and shows MPFS performing better than NFS.

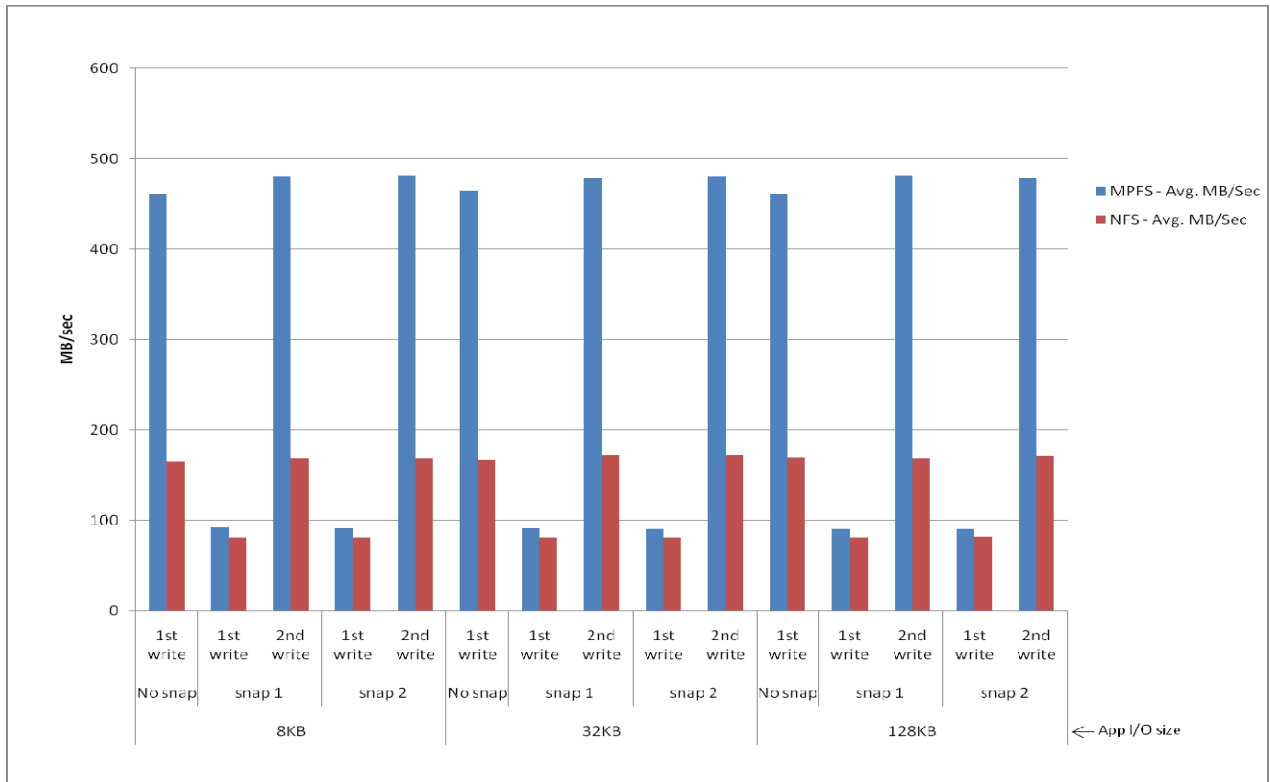


Figure 1 MPFS vs. NFS performance with 100% sequential write I/O

About MPFS

EMC Celerra Multi-Path File System (MPFS) is a combination of patented technology and the NFS protocol that enables file sharing by hundreds to thousands of client nodes, while realizing up to four times the aggregate bandwidth as compared to conventional NFS file serving. MPFS accomplishes this without requiring any application changes. A client-resident MPFS agent interacts with the Celerra file server through a special File Mapping Protocol (FMP) to split the file content data flow from the NFS metadata flow, permitting file content data to move directly between MPFS clients and EMC storage arrays by using an FC or iSCSI link. This process is reflected below in [Figure 2](#).

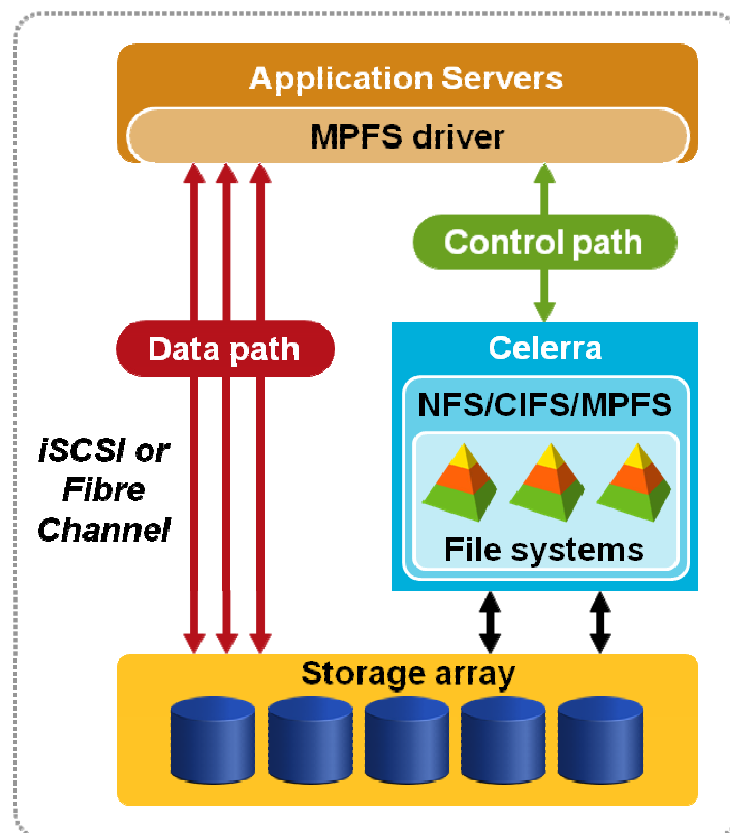


Figure 2 EMC Celerra MPFS data and control path flow

All of the file system operations, including block allocations, file locking, and metadata operation and logging, are performed by the Celerra server. But actual data movement of the file content occurs directly between the MPFS client and the storage arrays without the involvement of the Celerra file server. By directing data movement to the low latency storage channel, a client realizes line-speed data delivery, enabling higher bandwidth than conventional NFS would support. The MPFS SAN data delivery also dramatically reduces the Celerra file server's workload as it is not moving data to and from the clients, permitting 10 times more clients to share the namespace of a single file server blade.

Benefits

By separating file system operations from data delivery, and by supporting direct parallel access to the storage devices, MPFS offers the following:

- Use of NFS standards, making the MPFS benefits available without changing client applications
- Ability to serve data to large numbers of client limited only by the storage array bandwidth
- Ability to leverage high-performance block access caching built into the client operating environment
- Sophisticated distributing locking, enabling efficient concurrent access by multiple clients
- The extensive, proven capabilities and performance optimization of Celerra file servers
- Ability to exploit the prefetching and caching features of CLARiiON® or Symmetrix®.
- Use of all security features of today's NAS environments
- Sharing of files for all NFS clients, with or without the MPFS agent

Introduction

Overview

The information in this paper demonstrates the storage demands of protected environments, and the performance benefits of EMC Celerra MPFS when compared to NFS, even when advanced protection of EMC Celerra SnapSure and/or EMC Celerra Replicator V2 is required. Test result sections illustrate performance measurements for Linux clients under various load scenarios.

Audience

This white paper is intended for midsize or enterprise architects, and system or storage administrators who have developed or are contemplating deployment for a large-scale environment using NFS, and who seek the highest possible data I/O bandwidth to maximize computer throughput.

The storage demands of protected environments

Shared file system protection

In Celerra, shared file systems are protected using EMC Celerra SnapSure technology. It enables the creation of point-in-time logical images of this Production File System (PFS). These point-in-time views of the PFS are called checkpoints.

SnapSure uses a “copy on first modify” principle. A PFS consists of blocks of data. After a checkpoint is created, when a block within the PFS is modified, a copy containing the block’s original contents is saved to a separate volume called a SavVol. To see the point-in-time version, the preserved blocks in the SavVol and the unchanged PFS blocks are read by SnapSure according to the bitmap and blockmap data-tracking structure. These blocks combine to provide a complete point-in-time image called a checkpoint. Figure 3 illustrates how PFS data blocks are copied into SavVol.

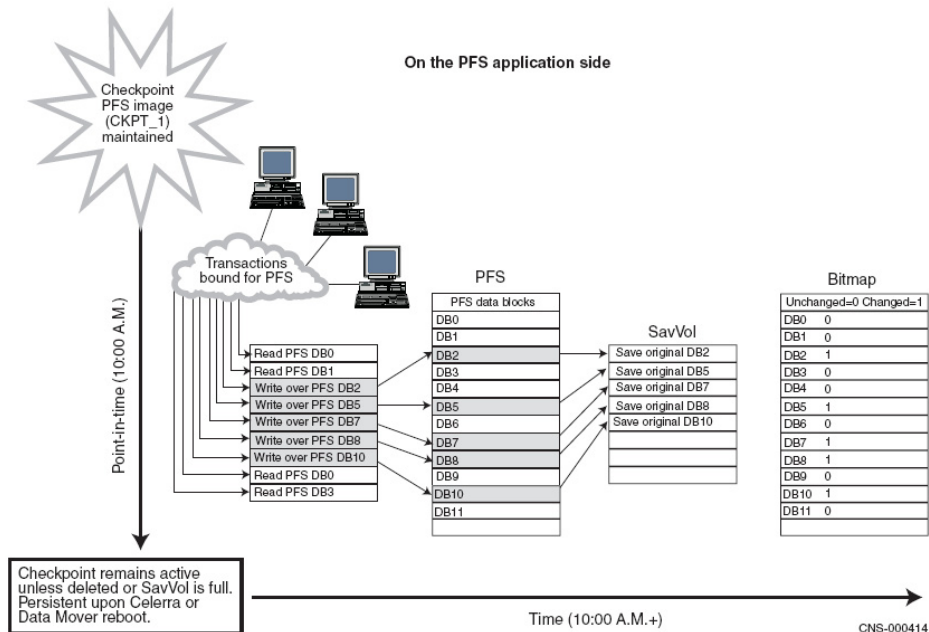


Figure 3 Celerra SnapSure uses the SavVol to save PFS blocks

The major challenge for BC/DR protected storage in MPFS environments is providing access to PFS and snaps with guaranteed data integrity and high performance levels.

For I/O-intensive environments, more NFS servers must be used, making the management of PFS and snaps difficult, and at times impossible.

Celerra Replicator V2

Celerra Replicator V2 provides efficient, asynchronous data replication over Internet Protocol (IP) networks. With Celerra Replicator, file system copies and consistent iSCSI LUN copies can be made available on local or remote sites.

The file system replication creates a read-only, point-in-time copy of a source file system at a destination and periodically updates this copy, making it consistent with the source file system.

Celerra Replicator V2 also uses internal checkpoints to ensure availability of the most recent point-in-time copy. These internal checkpoints are based on SnapSure technology.

So, the performance implications in this paper about SnapSure apply equally to Celerra Replicator V2.

Client I/O performance scaling

The total aggregate throughput of a shared file system is a function of the demands on individual nodes, multiplied by the number of concurrent nodes.

The ability to support large numbers of clients is not the only dimension of scalability. Per-client performance is also important. In practice, there are a number of factors, including caching and low latency access, that combine to improve per-client performance.

MPFS dramatically enhances file-sharing capabilities and accelerates performance of high-bandwidth, collaborative applications. The shared file system can accommodate applications of any size across tens of thousands of clients.

MPFS file sharing

MPFS with Celerra SnapSure or Celerra Replicator V2 incorporates the conventional advantage of accessing the shared file system, which permits file content data to move directly between MPFS clients and EMC storage arrays by using an FC or iSCSI link.

Along with this advantage, MPFS allows file sharing among multiple users. The MPFS server manages access to the file by granting read or write permissions to the clients. File data is stored in “extents,” each of which is composed of file system data blocks. Locks govern access permission to the blocks. Each block may be either unlocked or locked for write by a single client or locked for read by one or multiple clients.

When a client needs to read data, and if the requested blocks are available (that is, they are either not locked or locked for read), the server will mark them as locked for read by the requesting client.

When a client needs to write data, and if the requested blocks are available (that is, they are not locked), then the server will mark them as locked for write by the requesting client.

When a server cannot grant the client request, the server will try to obtain the lock for the requesting client by notifying the client(s) that hold the lock(s) to release the lock(s). This process may take some time, and it is not desired that the requesting client just wait.

With MPFS, when a client request cannot be granted immediately, the server will queue the request and update the client. After the server completes its processing, it will notify the client and complete the client’s original request.

A client may queue multiple requests for one file to the server. If the server cannot queue the block lock request, it will reject the request. The client may subsequently retry the request.

Protected file sharing – snap access

When an MPFS client attempts to mount a versioned file system, the MPFS client asks the MPFS server if it is mounting the primary file system view, a point-in-time view, or a “snap” of the file system. If it is mounting a snap, the MPFS client then treats the file system as if it were an NFS or CIFS file system. All subsequent read or write data packets are transmitted over NFS or CIFS instead of MPFS. The MPFS protocol is used only for the primary file systems. [Figure 4](#) illustrates a flowchart snippet for this operation.

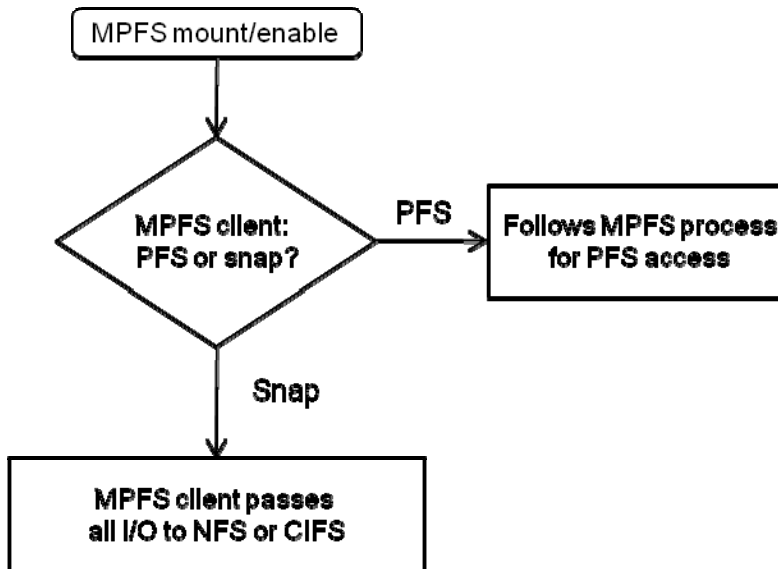


Figure 4 Snap access

Protected file sharing – PFS read access

When an MPFS client needs to read data from the PFS, and if the requested blocks are available (that is, they are either not locked or locked for read), the server will mark them as locked for read by the requesting client and the data is read by the client over the MPFS high-speed block I/O path. If the blocks are unavailable, then the request is queued and the request is completed later when the server completes its processing. MPFS read performance is equivalent to read performance of a file system without any snaps. The flowchart for this operation is illustrated below in [Figure 5](#).

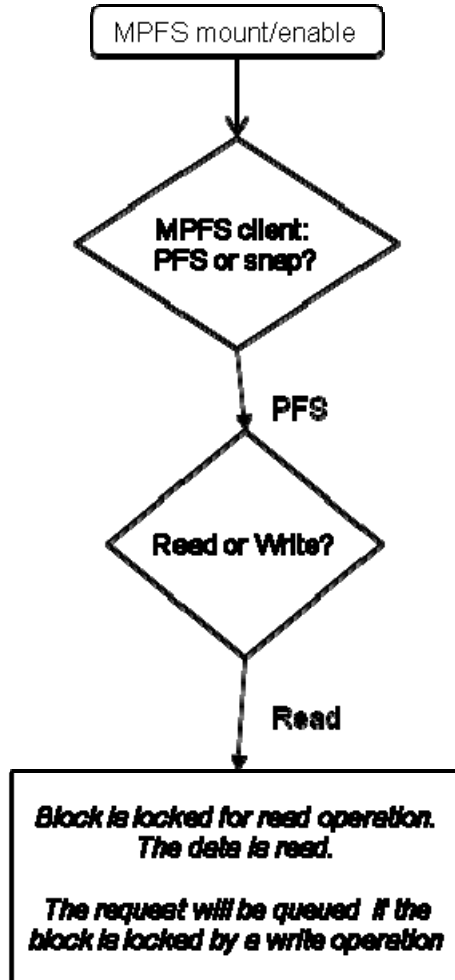


Figure 5 PFS read access

Protected file sharing – PFS write access

When an MPFS client needs to write data from the PFS, the MPFS client sends an MPFS “AllocSpace” request to the server. If the file system is versioned, and the block to be written is not snapped (that is, has not been copied to the SavVol), then the MPFS server reads the existing block into memory and writes it into the SavVol. The server then grants the AllocSpace request to the MPFS client. This request is also granted immediately if the file system is not versioned or if the block has already been snapped. The MPFS client then performs the write operation directly over the MPFS high-speed path. So, as one might imagine, there is a moderate performance impact for the first write operation after a snap. But, subsequent write operations happen at about the same speed as a file system without any snaps. The flowchart for this operation is illustrated below in [Figure 6](#).

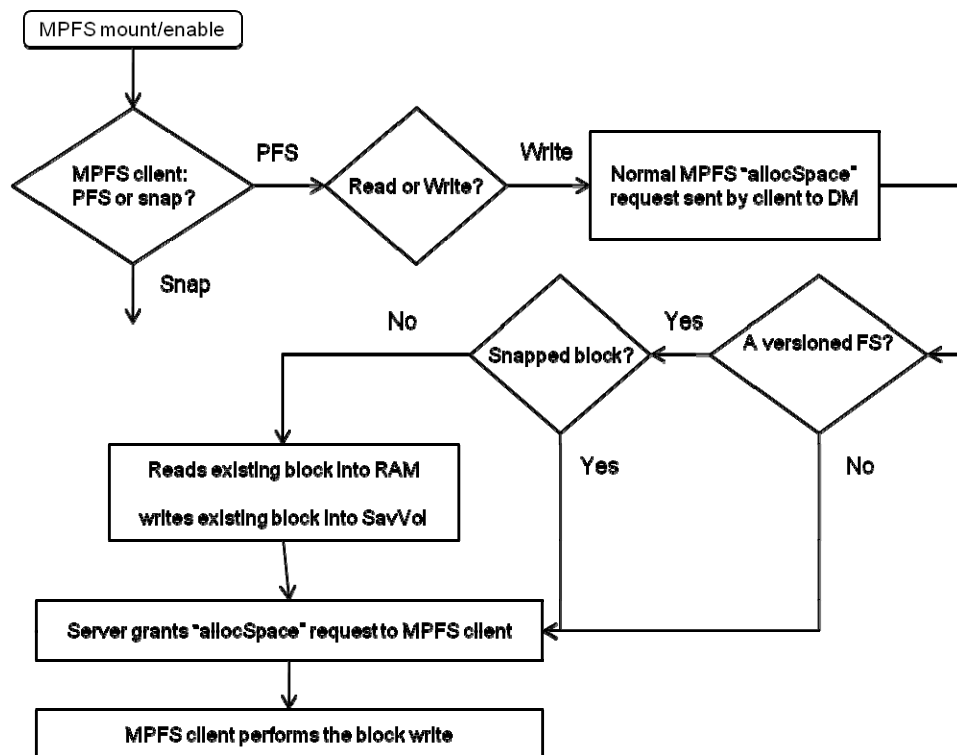


Figure 6 PFS write access

MPFS performance measurement for Linux clients in protected environments

Introduction

MPFS has been extensively characterized in Celerra SnapSure-protected environments. The following characteristics were observed:

- With 100 percent sequential write I/O load, MPFS performs three times better than NFS with snaps, except for the first write after each snap.
 - With 100 percent sequential write I/O load on first write on the PFS with snaps, MPFS performs marginally better than NFS.
-

Test outline

The following test result sections show performance measurements for Linux clients. Synthetic I/O load generators (for example, IOZone) were used to generate client load on a Celerra NS-480FC file server to assess the performance advantage afforded by MPFS as compared to NFS. EMC Celerra with DART 5.6.46-418 and the MPFS client 5.0.31.7.1 on x86-based Linux (RHEL 5 Update 5) machines were used for the testing.

Following are the major components of the MPFS architecture tested:

- MPFS file system clients (Linux clients with an MPFS agent and iSCSI initiator/HBA)
 - EMC Connectrix[®] MDS with either FC and IPS blade modules or a CLARiiON CX4 FC/iSCSI array that provides protocol conversion between FC and iSCSI
 - EMC Celerra blades running MPFS server processes and FC modules
 - CLARiiON with Navisphere[®] Analyzer and Access Logix[™]
-

Testbed layout Figure 7 below shows the high-level layout of the test environment.

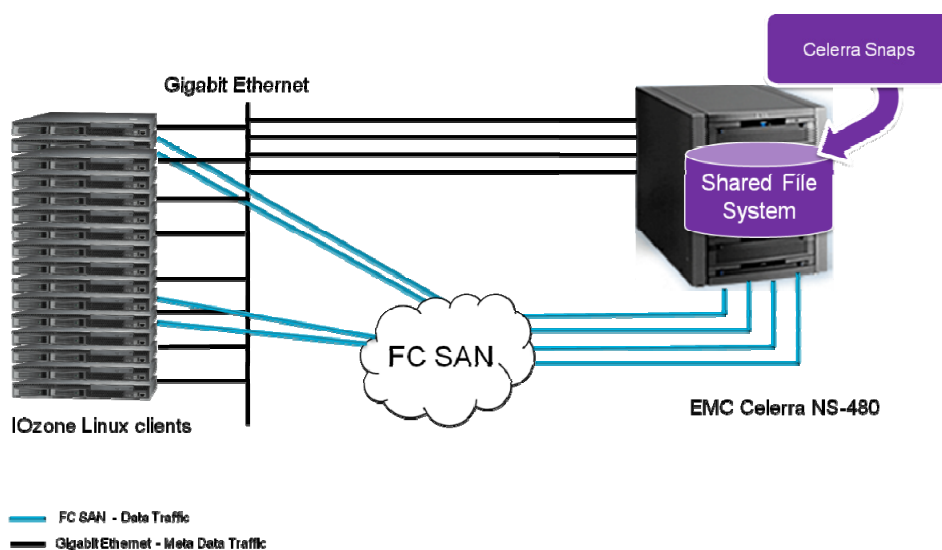


Figure 7 Testbed layout

Test methodology

The performance testing was completed using IOzone-based tests.

All tests carried out used the following test parameters:

- 1 MPFS thread running for 100 percent sequential I/O write or read. Similar tests were carried with random mixed I/O
- 16 Linux (RHEL version 4 update 4) clients
- 16 TB file system size
- 8/32/128K application I/O size

Test result – 8K 100% sequential write

Figure 8 reflects the test results of 8K 100 percent sequential write.

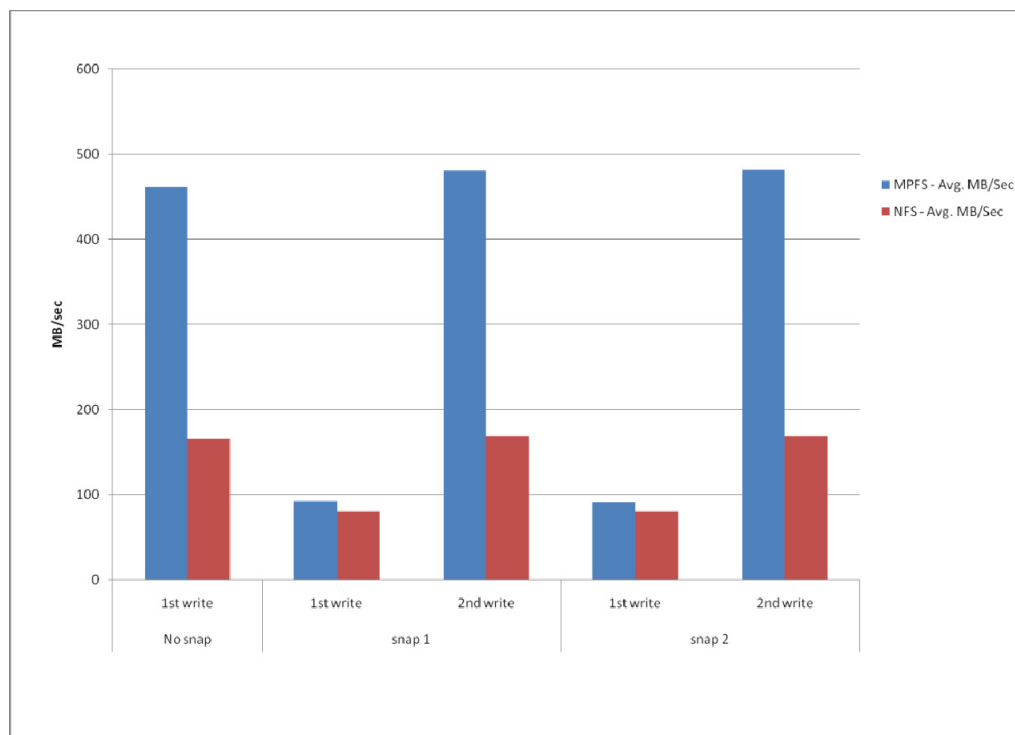


Figure 8 8K 100% sequential write MPFS vs. NFS with Celerra snaps

Observation: In this configured test environment, MPFS performs almost three times better than NFS in write operations on a shared file system without snap, the second write operation on the first snap, and the second write operation on the subsequent snaps.

In scenarios of shared file systems with first write on first snap and first write on subsequent snaps, MPFS yields 8 percent better performance than NFS. This is due to “copy on first modify.”

**Snap writes –
copy on first
modify**

Write operations on a snapped (or replicated) Celerra file system uses a “copy on first modify” principle (CoFM). CoFM occurs the first time a file system’s blocks are modified after a snap. Even with MPFS, before a block within the PFS can be modified after a snap, the Celerra Data Mover must first preserve the existing contents of that block. The Data Mover reads the existing block from the PFS, and then writes it to a separate volume called the SavVol. These two I/Os must occur prior to any PFS modifications because this preserves the view of the file system as it existed at the time the snap was taken. These extra synchronous I/Os add overhead to the first write operations after each snap, resulting in slower performance compared to previous and subsequent write operations, regardless of the access protocol. Subsequent changes made to the same block in the PFS are not copied into the SavVol. This is illustrated below in [Figure 9](#).

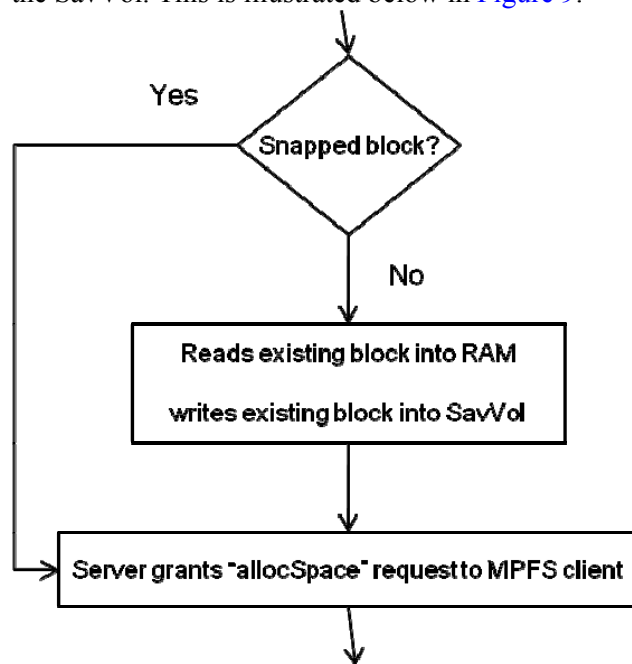


Figure 9 Copy on first modify flowchart

**Test result –
32K 100%
sequential write**

Figure 10 shows the test results of 32K 100 percent sequential write.

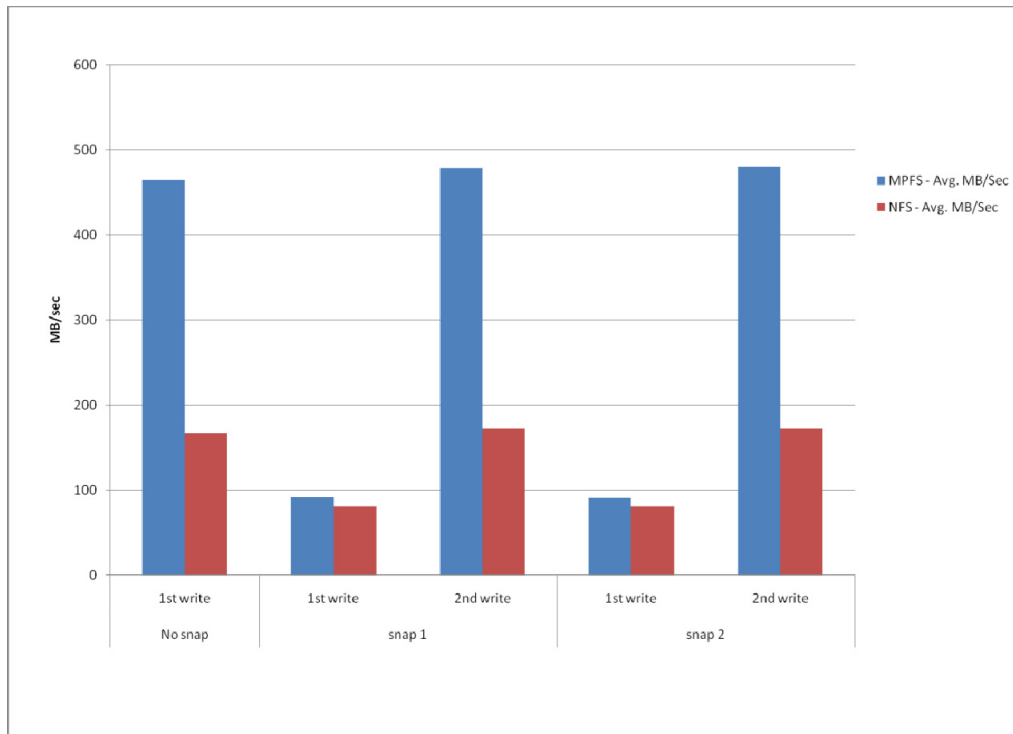


Figure 10 32K 100% sequential write MPFS vs. NFS with Celerra snaps

Observation: In the configured test environment, MPFS performs almost three times better than NFS in write operations on shared file system without snap, the second write operation on the first snap, and the second write operation on the subsequent snaps.

In scenarios of shared file system with first write on first snap and first write on subsequent snaps, MPFS performs yields 8 percent better performance than NFS. This is due to CoFM.

These characteristics are similar to that of an 8K I/O load.

**Test results –
128K 100%
sequential write**

Figure 11 shows the test results of a 128K 100 percent sequential write.

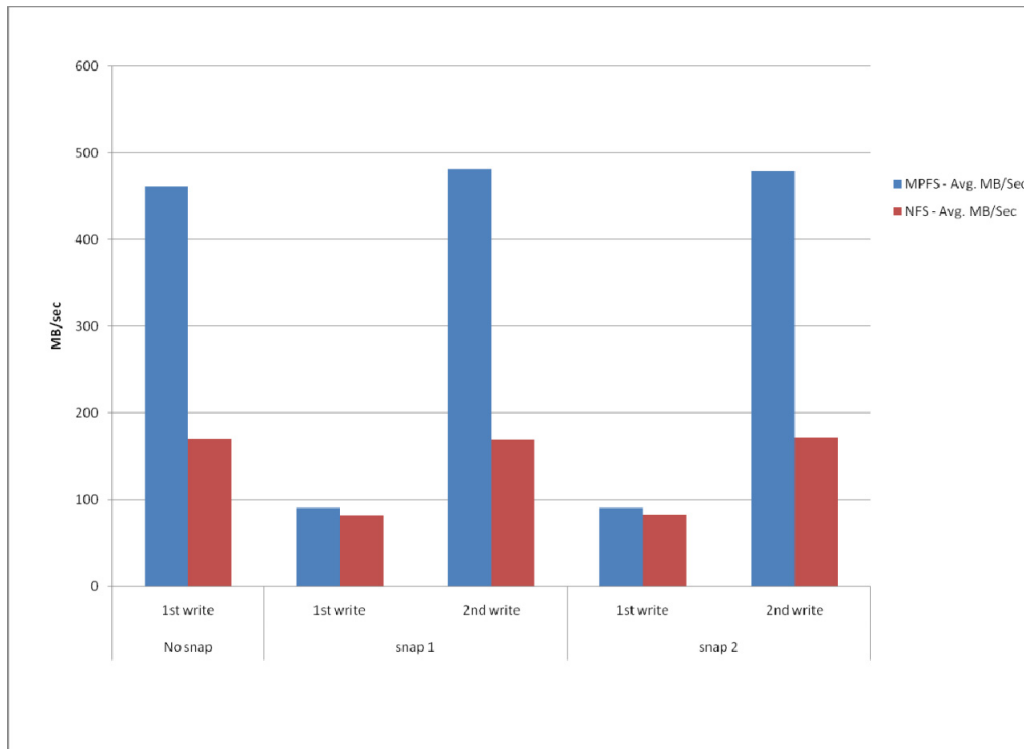


Figure 11 128K 100% sequential write MPFS vs. NFS with Celerra snaps

Observation: In the configured test environment, MPFS performs almost three times better than NFS in write operations on shared file system without snap, the second write operation on the first snap, and the second write operation on the subsequent snaps.

In scenarios of shared file systems with first write on first snap and first write on subsequent snaps, MPFS performs yields 8 percent better performance than NFS. This is due to due to CoFM.

These characteristics are similar to those of 8K and 32K I/O loads.

**Test result -
100%
sequential read**

Figure 12 illustrates the test results of 8K, 32K, and 128K 100 percent sequential read.

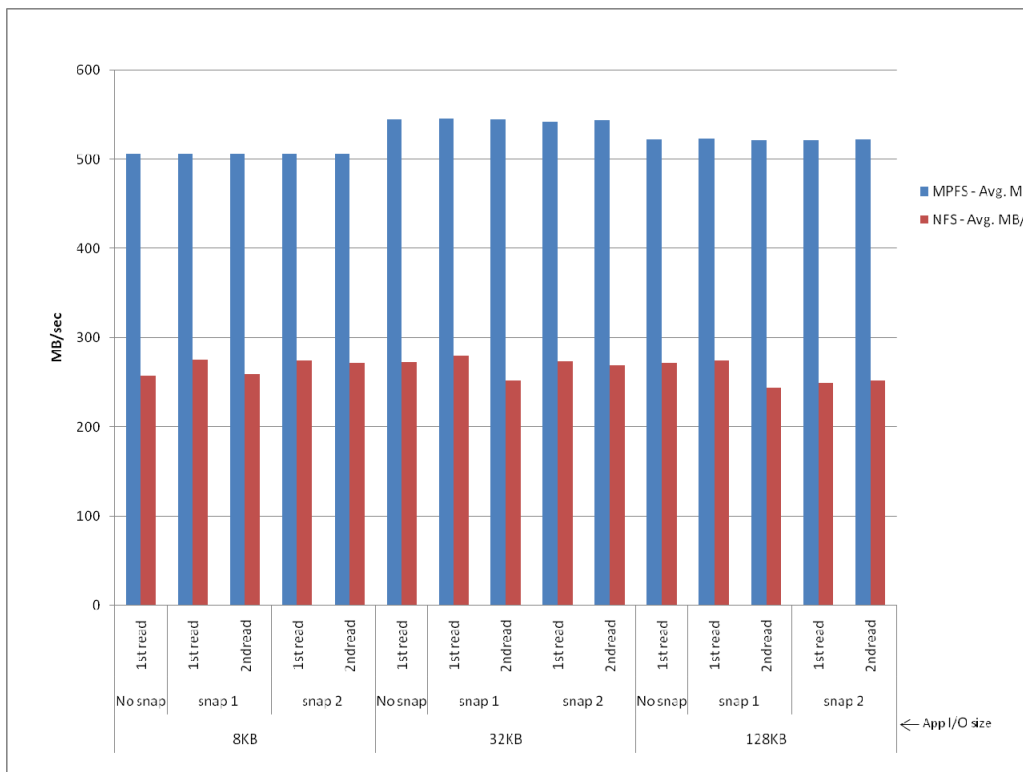


Figure 12 MPFS vs. NFS performance with 100% sequential read I/O

Observation: In the configured test environment, MPFS performs almost two times better than NFS in read operations on shared file system without snap and with snaps (first read or second read operations).

The MPFS and NFS read I/O pattern is almost the same for 8K, 32K, and 128K I/O loads.

**Test result -
random mixed
I/O workload**

In the configured test environment, MPFS performs almost the same as NFS with a random mixed I/O workload. This is a result of the time required to determine if random blocks have been modified. Ultimately, performance is bottlenecked by this process for both transport protocols.

**Testing with
new file or
existing file**

In the configured test environment, all new file creations and existing file appendages (where blocks were not allocated prior to the snap) exhibited the write performance equivalent to CoFM performance. Note that CoFM does not occur when blocks written to were previously unallocated within the file system. This helps save space in the SavVol, but does not improve performance above CoFM.

Conclusion

Summary discussion

MPFS offers standards-based simplicity of NFS data sharing, with block access performance attributes, in environments protected by EMC Celerra SnapSure and EMC Celerra Replicator V2.

The tests show that MPFS performance on shared file systems protected by Celerra SnapSure or Celerra Replicator V2 can be equivalent to MPFS performance without protection. While CoFM introduces a performance penalty for the MPFS protocol, the performance is still better than what can be achieved via the traditional NFS protocol.

Consider MPFS for high-performance shared file system needs, even when advanced protection of Celerra SnapSure and/or Celerra Replicator V2 is required.

References

Product documentation

For additional information, see the product documents listed below.

- *EMC Celerra MPFS for Unified Storage Configurations Quick Start Guide*
 - *EMC Celerra MPFS over FC and iSCSI Linux Clients Product Guide*
 - *EMC Celerra MPFS for Linux Clients Release Notes*
 - *Using MPFS on Celerra*
 - Various Celerra QuickStart: Celerra MPFS module documents
-