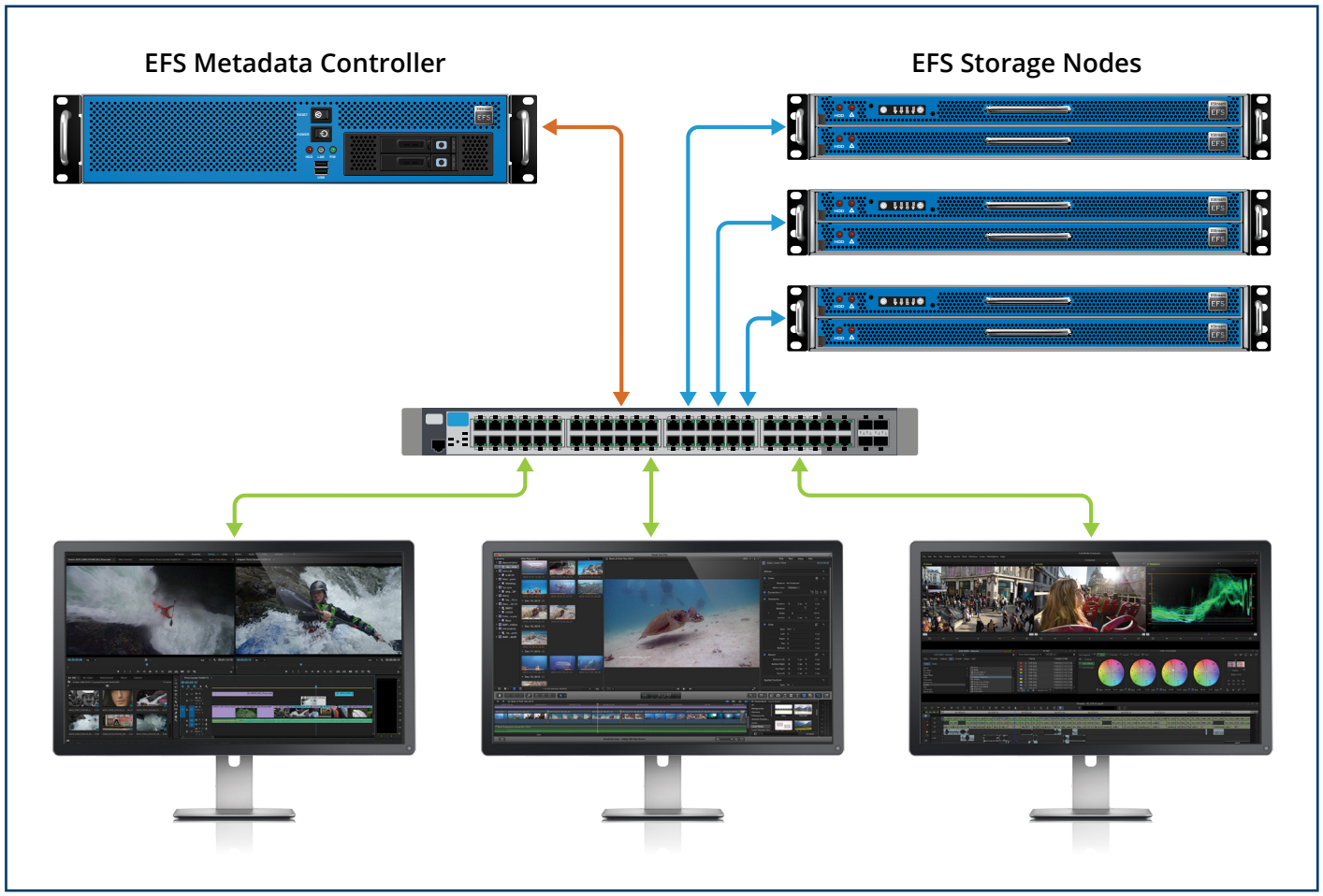
WHITE PAPER

EditShare EFS Shared Storage

Advantages of EFS Native Client network protocol in media intensive storage workflows



EditShare®

Introduction

The EditShare EFS is an enterprise-class scale-out storage system designed for the most demanding "mission critical" collaborative media production environments. It combines an advanced distributed file system, intelligent load-balancing, a scalable, fault-tolerant architecture with cost-effective 10/40 GbE interconnectivity. The result is a powerful shared storage system that is free of single points of failure, that can reliably deliver massive payloads to dozens of concurrent client devices, that presents an easily managed single global namespace and sets new standards in storage cost and efficiency. Key to ensuring this ultra-fast and reliable performance is the EFS Native Client network protocol.

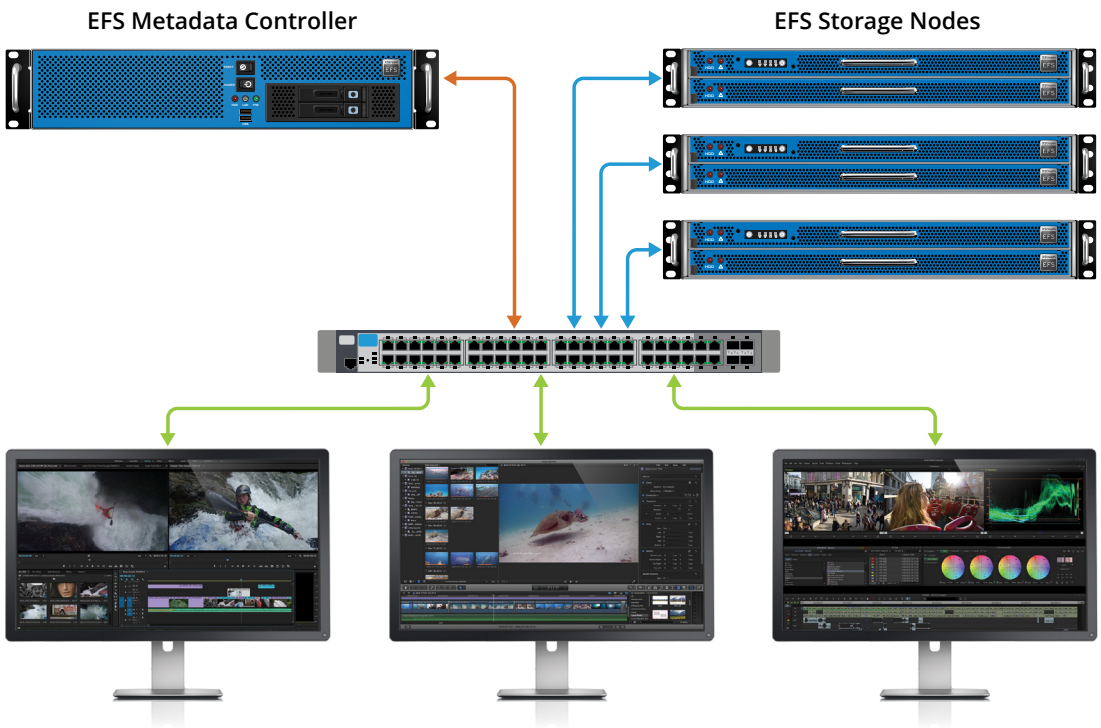


Figure 1 - Basic EFS storage cluster

Network Protocols

Network protocols specify the structure and content of messages passed between computers and define the mechanisms used to address, secure and respond to messages. There is no universal network protocol that meets the need of all computer networks or applications. In Media & Entertainment applications Apple Filing Protocol (AFP), Server Message Block (SMB), Common Internet File System (CIFS - a variant of SMB), Network File System (NFS) and File Transfer Protocol (FTP) are examples of popular network protocols used to connect a storage solution to NLE or creative clients. However, network protocols like these have evolved to support the many types of devices and network applications. As a result, this evolution in capabilities comes at the cost of a great deal of overhead to recognize and implement specific connections. While these protocols have become "Jacks of all trades" they are unfortunately "Masters of none."

EFS Native Client

While EFS storage supports each of these protocols, EditShare believes the best way to improve shared media performance is to use an application-specific network protocol instead of accepting the limitations of the legacy AFP, SMB and NFS protocols.

EFS Native Client implementations have been developed for the OS X, Windows and Linux platform, providing a powerful multi-threaded application that tightly manages read and write streams while avoiding the latency and overhead of legacy protocols.

Unlike shared storage solutions that rely on legacy network protocols, the EFS Native Client has enabled powerful new features to improve storage performance and resilience.

Reducing Latency and Overhead

When clients transact with NAS storage via protocols like SMB and AFP, the client mounts just one storage node (IP address) and that node transacts with remaining nodes to collect associated data chunks on behalf of the client. This produces several problems. First, read requests are slowed by the multiple hops the requests take from client to primary node to secondary nodes and back again. Second, since the primary node processes its own transactions as well as those of all secondary nodes, it becomes a hot spot for transactions and can become a bottleneck.

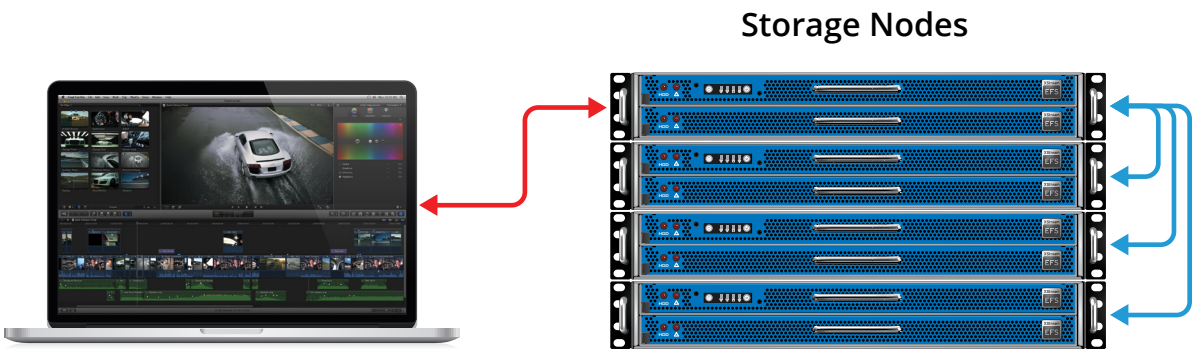


Figure 2 - Traditional NAS scale-out storage communication

To make matters worse, when clients randomly mount a primary node it is possible for multiple clients to choose the same primary node. That node can easily be overwhelmed by the need to support multiple clients. Some scale-out NAS systems prevent this situation by using an expensive dedicated DNS server to ensure that clients don't mount the same primary node. Not only does this increase the cost of ownership, it also adds a new single point of failure.

When clients transact via the (multi-threaded) EFS Native Client protocol, the client mounts and transacts directly with all of the nodes containing the relevant data and parity chunks. This distributes transactions evenly and eliminates the hot spot associated with the primary server. It also eliminates the multi-hop round trips between primary and secondary nodes. In comparison, the transaction load using the EFS Native Client in a 5 node configuration is only 10% per server versus an equivalent 'traditional' SMB/AFP scale-out system.

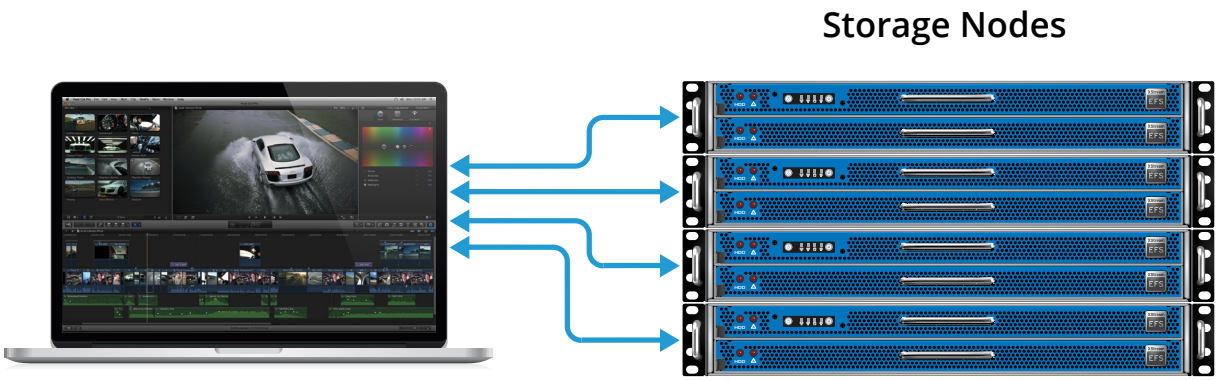


Figure 3 - EFS Native Client communication

Performance Features and Benefits

Eliminates contention issues that lead to dropped video frames

In heavily loaded shared NAS or SAN storage systems, node contention is often a source of performance degradation. It occurs when two or more clients make read requests that involve the same storage node. The EFS SwiftRead feature is an example of the unique capability and benefit that is unmatched with other storage-specific file systems. Recall that an EFS storage cluster writes data and parity across 3 or more storage nodes. The EFS SwiftRead feature monitors the response times to node read requests. If a node fails to respond within a defined time window, the client simply requests the associated parity data and uses it instead of the delayed data chunk to construct the media data. Thanks to the native SwiftRead feature, an EFS storage cluster can complete more client read transactions than could normally be completed by other storage solutions.

Lower overhead and latency delivers 20% more payload than legacy network protocols

As previously explained, legacy network protocols must cope with an incredibly wide range of client-server connection types. In contrast, the EFS Native Client needs only support a specific set of Windows, OS X and LINUX client applications and the EFS files system. As a result, the EFS Native Client has far less overhead and presents a much lower latency path than do legacy network protocols. In tests comparing the AFP protocol with the EFS OSX Native Client, when a MAC-based Avid Media Composer mounted the EFS storage system via AFP protocol, it delivered 4 streams of DNxHD 145 on a standard 1GbE connection. However, when the EFS OS X Native Client replaced the AFP protocol, that same system delivered 5 streams of DNxHD 145 on the standard 1GbE connection.

Bonded Ethernet ports deliver up to twice the payload of a single port

Bonded Client Ethernet ports is another powerful feature enabled by the EFS Native Client. This allows two or more client machine ports to be bonded for the purpose of delivering greater payloads than are possible with one port. In tests comparing the performance of a bonded pair of 1GbE ports to a single 1GbE port (both cases using the EFS OS X Native Client) the bonded pair delivered 9 streams of DNxHD 145 while the single port topped out at 5 streams.

Fault Tolerance Features

Managing storage node hardware failure

In addition to detecting slow storage nodes, SwiftRead also detects nodes that have ceased responding. Therefore, in cases where a storage node has failed, SwiftRead uses parity data stored on other storage nodes instead of the data on the failed node. Thus, even in the presence of a failed storage node, an EFS storage system is able to operate without any performance degradation.

Managing storage node connection failures without interruption of streaming services

SMB or AFP-based client connections always mount a primary NAS storage node and then rely upon that node to transact with other storage nodes. When connection to the primary storage node is lost due to a system failure, the client will need to establish a new connection with a primary node to resume operations. During the interruption no information can be transmitted between the storage system and the affected client device.

Because an EFS Native Client – base client connection always mounts all storage nodes, the loss of connection to one storage node is detected and managed via SwiftRead and services continue without

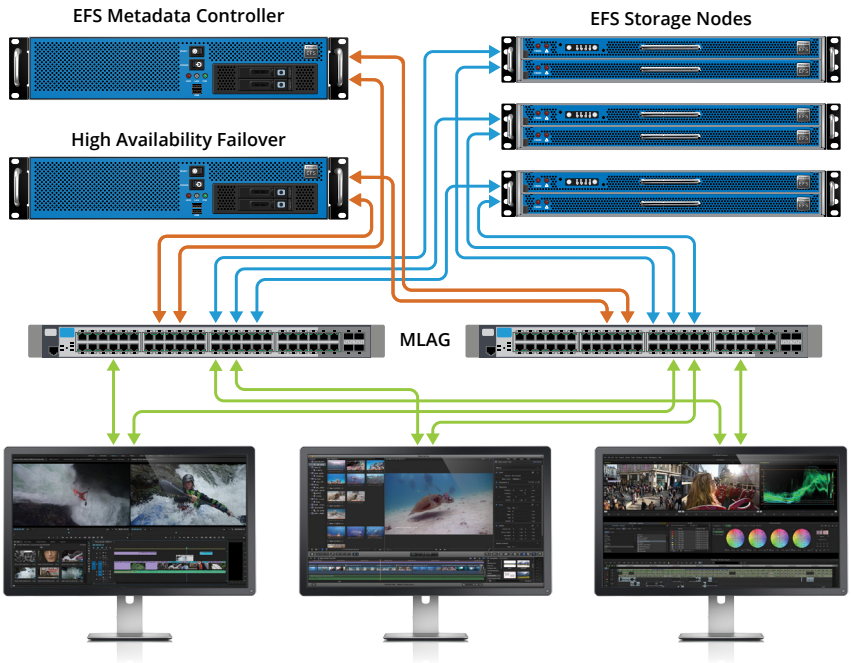


Figure 4 - EFS redundant network configuration

Summary

By employing the application-specific EFS Native Client instead of legacy AFP, SMB or NFS network protocols, the EditShare EFS shared storage system delivers a higher level of performance than similar NAS scale-out storage solutions. In addition, the EFS Native Client enables powerful new features that enhance performance and provide increased levels of fault tolerance. Armed with these capabilities, EditShare EFS storage systems are an excellent choice for today's demanding 24 x 7 media production environments.