Paper SAS6760-2016

Architecting Your SAS Grid®: Networking for Performance

Tony Brown and Margaret Crevar, SAS Institute Inc.

ABSTRACT

With the popularity of network-attached storage for shared file systems, IT shops are increasingly turning to them for SAS Grid® implementations. Given the high I/O demand of SAS large block processing, how do you ensure good performance between SAS Grid nodes and network-attached storage? This paper discusses different types of network-attached implementations, what works, and what does not. It provides advice for bandwidth planning, types of network technology to use, and what has been practically successful in the field.

.

INTRODUCTION

Implementing a SAS Grid system has many parts and layers. From the overlying application architecture, to the underlying physical hardware and file system architecture, there are many layers to integrate. When network-attached storage is used for SAS Grid, it introduces another layer of performance that must be assured. There are tried and true best practices that should be followed when using network-attached storage for the best performance of your SAS Grid.

This paper will introduce the basic building blocks of network storage systems, and give a definition of network-attached storage, delineating it from direct-attached and SAN storage. It will also present some common network file system variants that are used. In practicality it will give tips for maximizing SAS Grid performance via a network including:

- Network fabric choices, and their performance implications
    - Recommended bandwidth for each SAS Grid architecture layer
- Network file systems and their known issues
- Typical grid architectures – network usage pros and cons
- Complicating factors:
    - Management for expanding and for large systems
    - Software defined storage (SDS)
    - Virtualization

WHAT ARE THE BUILDING BLOCKS OF NETWORK STORAGE?

There are fundamental building blocks in the chain of network-attached storage. The components list begins with the interface cards in your host server used to attach network resources, feeds out to the network fabric itself (network, routers, bridges, switches, and so on), and ends with the final connection to some type of storage. When using storage over networks, the performance you achieve in your SAS Grid system will depend on how well you provision each of the components in this chain. Each CPU core you use in a SAS Grid node will require a minimum of 100 megabytes per second (MB/s) of throughput to ensure adequate performance for your SAS applications. Planning bandwidth for throughput in the network chain is crucial to ensure satisfactory SAS Grid operations and performance.

WHAT IS A NETWORK-ATTACHED STORAGE DEVICE?

Network-attached storage (NAS) is a type of dedicated file system storage that provides local-area network (LAN) nodes with file-based shared storage through a standard Ethernet connection. “Uh, what?” you might ask? Perhaps the easiest way to define network-attached storage is to contrast it with traditional direct-attached storage.

Storage comes in a couple of basic dimensions:

- Direct-attached storage (DAS), typically storage area network (SAN) – traditionally block-based architectures
- Network-attached storage – traditionally file based architectures

DAS/SAN - BLOCK BASED STORAGE

In traditional block-based storage, either the compute server, or the SAS Grid node (in the case of a SAS Grid), has its own map of the data (volumes=> logical unit numbers (LUNs) => file systems => files => individual storage blocks comprising a file) in physical blocks, on the storage device (such as disk or flash drive). SAS Grid requires a clustered (or shared) file system to share data across nodes. Other shared file system types such as distributed file systems, or network file systems (NFS), are often used. The results are often sub-optimal as compared to using a clustered file system. In the case of a clustered file system, the clustered file system itself is defined as the server-based file stack.

The server file stack translates the file address into the disk blocks that it consists of on the storage device. This type of storage is called block-based storage. It has evolved from blocks stored on internal server disks, to blocks stored on external direct-attached storage arrays, to SANs. The block-based storage format is fronted by a server-based file stack that in turn maps these blocks to an individual file. Block I/O is the underlying I/O mechanism, and data is read from storage in individual blocks. The key takeaway word here is "server".

Originally the blocks comprising your files were defined on a single server, so only that server could access that file. The block-based storage device was **DAS, connected directly to the server,** without going through a network. The most common protocols used for the direct-attached connections are Fibre Channel protocols. Eventually SANs – (storage arrays with powerful private networks across the storage frame) came along and allowed those blocks to be shared by multiple servers that were **direct-attached** to the SAN, as well as to the files that they comprise.

To further confuse things, some modern SAN offerings now support Internet Small Computer Systems Interface (iSCSI), which is a protocol for transmission over an Ethernet network. So now you can network attach a SAN! We will discuss more on the performance of this later.

Direct-attached storage typically represents the highest performing storage option to attach to a SAS Grid. Attaching a SAN with iSCSI will function well, but does not typically perform as well as direct-attached storage.

NAS - FILE BASED STORAGE

File-based storage moves the file stack definition to a NAS. To access a file the server sends a request to the NAS. The NAS runs **its own** file stack to translate the addresses needed into underlying physical device blocks on the NAS storage devices (for example, a disk) to store or retrieve a file. For file-based storage, this is typically called a filer. It is essentially a file sharing system. Since the NAS owns the underlying physical data and that data's definition, instead of the host server, it can share these across multiple servers via a network. The underlying reason for NAS is to allow file sharing across multiple servers or compute nodes via a network (hence its name). The protocols that NAS uses are the Common Internet File System (CIFS) for Microsoft Windows, and NFS for UNIX and LINUX.

In general, filers do not perform as well as block-based storage for the large block, sequential I/O that SAS uses. The best successes in the field with filers have involved direct-attaching them to SAS Grid hosts via Fibre Channel. Lesser performance, but acceptable in many instances, can be achieved with network-attached filers. In this case, careful attention has to be paid to both fabric and bandwidth throughput.

COMBINING BLOCK AND FILE BASED STORAGE – UNIFIED STORAGE

Modern storage vendors are offering block-based and file-based storage in one system – for those who want it all. These hybrid SAN/NAS systems are often referred to as unified storage. Unified storage is the "wild West" of new offerings. Some solutions implement a NAS head (the computer brain of the NAS that defines and controls the file stack and operations) that controls a SAN. Other SAN vendors are offering file-based I/O functionality that is integrated into their block-based SANs (without the use of NAS heads), under the same SAN switches that also service block devices. There are several variants of the architecture and operation of these units.

In the approach that uses a NAS head fronted to the SAN, users connect to the NAS via an Ethernet network, and the NAS head is cabled to the SAN device with Fibre Channel connections. The overlying protocol of this approach is still a network protocol –NFS or CIFS. In a true "converged beneath the covers" storage approach, the unified storage device allows both NFS, CIFS, and iSCSI attachments via a LAN, and direct-attached connections via Fibre Channel. These connections allow both file- and block-based I/O to be serviced from the same storage unit.

Most storage vendors have unified storage offerings. They all differ in their approach and technology, so you must work with their engineers to understand their offerings carefully if you wish to combine file and block I/O operations. Please understand that SAS Foundation performs very heavy, large-block, sequential I/O. Our bias is toward block-based storage for SAS Foundation, whether using traditional block storage or file-based storage that manipulates block operations. Network-attached storage can be adequate in many situations, but it requires careful bandwidth planning, and avoidance or mitigation of known network file system issues. This is discussed later in the paper.

The above is a simplified view of network storage devices, and how they differ from traditional SAN or block-based storage. Implementations of network storage devices have grown quite complex, and some are illustrated below.

NETWORK FABRIC CHOICES AND PERFORMANCE IMPLICATIONS

The following are rough approximations of some common network fabric choices and their approximate throughput bandwidth:

- 1-Gigabit Ethernet, 1GbE - ~ 120 megabytes per second
- 10-Gigabit Ethernet, 10 GbE – ~ 1.2 gigabytes per second
- InfiniBand:
  - Mellanox 40 – 100-Gigabits per second fabrics – ~ 3.2 – 12.5 gigabytes per second
  - Intel 40-80 gigabits per second for QDR1B cards – ~ 3.2 – 6.4 gigabytes per second

Some of these rates are published via vendor testing, so they are only approximations for your workload. Actual rates are influenced by network switching, routing, and inter-connect choices. The above fabric types can use bundle components to achieve bandwidth goals. The important thing to note here is the graduation between the fabric element throughput ranges.

When it comes to network fabrics, you must pay careful attention to the throughput that SAS Grid application layers demand. The underlying network fabric, ranges from network interface cards (NICs) in your SAS Grid servers (SAS Metadata Server, SAS Web Application Server, Platform Process Manager, and grid compute nodes), to the underlying Ethernet cables, switches and routers. High bandwidth should be the driving factor in the decision on what technology to use in the network fabric for the grid compute node layer. The other layers easily suffice with the fabric types listed below.

The requirements for network throughput vary in the SAS Grid architecture layers. For simplification of discussion, let's split a SAS Grid architecture into a separate layers, and deal with each layer independently. The first layer consists of the SAS Grid client hosts that are ostensibly on a corporate public network. This network might be on a LAN, or distributed on a wide-area network (WAN) using Citrix servers to attach to SAS Grid systems. At the entry portion of the SAS Grid architecture is the SAS Web Application or middle tier Servers, and the SAS Metadata Servers. There might be multiple instances of these servers if high

availability has been architected. The third layer consists of SAS Grid Manager and the grid host compute nodes. The fourth and final layer consists of the underlying storage that SAS Grid Manager and the grid host compute nodes use. The basic performance profile of each layer follows:

The first client layer of the Grid architecture:

- SAS Grid client nodes, linked into the SAS Grid system
- Traditional 1GbE public corporate network bandwidth – this generally suffices for SAS clients.
- WAN implementations – you will need to consider distance, attenuation and latency across WAN switches, and interconnect into the SAS Grid application network (for example, via CITRIX® Servers).

The second (SAS Grid® services) layer:

- SAS® Grid web-tier and middle tier server(s)
- SAS Grid Metadata Servers – multiple servers if they are clustered for failover
- We have experienced acceptable performance for very small (4-node/16 CPU core total) SAS Grid implementations using 1 GbE network bandwidth in this area. We often recommend placing the SAS Grid Metadata Servers on a private LAN segment, separating traffic from the busy corporate LAN. This implementation typically uses a single instance of a SAS Metadata server. For mid-size to larger grids (larger than 4 node/16 cores, systems with large metadata implementations, or web-services components) , we highly suggest deploying a 10 GbE fabric for this layer. See your SAS Grid Technical Team for guidance.

The third (SAS Grid processing and compute) layer:

- SASGrid control server and compute nodes
- SAS Grid compute nodes interconnect with the SAS Grid control server node and the services layer nodes via 10 GbE fabric. The performance of 10GbE fabric is quite adequate for performance. The size of both job requests shipped via the services layer through the SAS Grid control server and of the resulting output are generally small enough that this bandwidth is adequate.
- An issue in the SAS Grid processing layer is the heavy lifting that the processing nodes must perform on the storage. The connections of the processing nodes to the storage must meet a core minimum throughput requirement of sustained 100 megabytes per second per host.
- The 10 GbE fabric that is servicing the SAS Grid processing and compute layer should be placed on a private network segment, or a separated LAN segment, dividing its local traffic from the primary corporate network.

The fourth (Storage) layer of the SAS Grid architecture:

- Underlying physical and/or virtual storage
- The storage layer attachments must be able to produce 100 MB/s per core to each SAS Grid node that it services. Because of the myriad of storage architectures that can be deployed, a discussion of the fabric choices in this layer are given in the remainder of this section, along with examples.

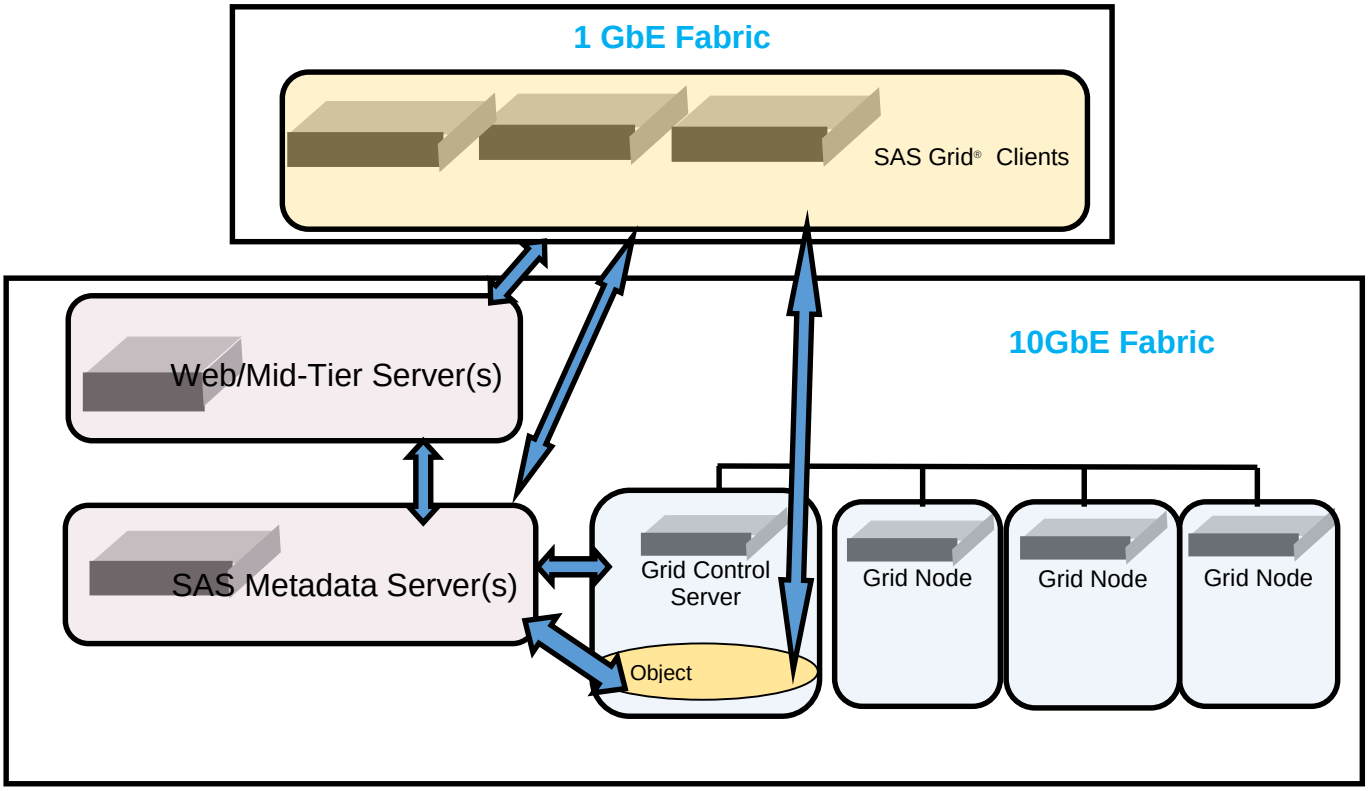

Figure 1. Appropriate Network Fabric Choices for the SAS Grid Client, Services, and Grid Node Tiers.

NETWORK FABRIC CHOICES FOR THE STORAGE LAYER

Given the above discussion, let's examine three fundamental types of SAS Grid storage architecture and the architecture's ramifications for bandwidth delivery.

Direct-Attached Storage

Direct-attached storage has been shown to be the best option for providing and sustaining the best throughput bandwidth. In short, it cuts out the shared network. By using Fibre Channel host bus adapters that are directly connected to the storage array, bandwidth is limited only by the specifications of adapters and the storage configuration chosen. We currently have customers successfully attaching SAN and NAS in this fashion. Some implementations have involved using iSCSI connections from host to storage (still a network option), but its performance is not as good as Fibre Channel attachment. While direct attached storage has generally proven to be the best option for performance and throughput, storage arrays must be scaled up to increase capacity, and can be more difficult to manage for a growing number of SAS Grid nodes. Figure 2 below illustrates the direct-attached option using Fibre Channel.

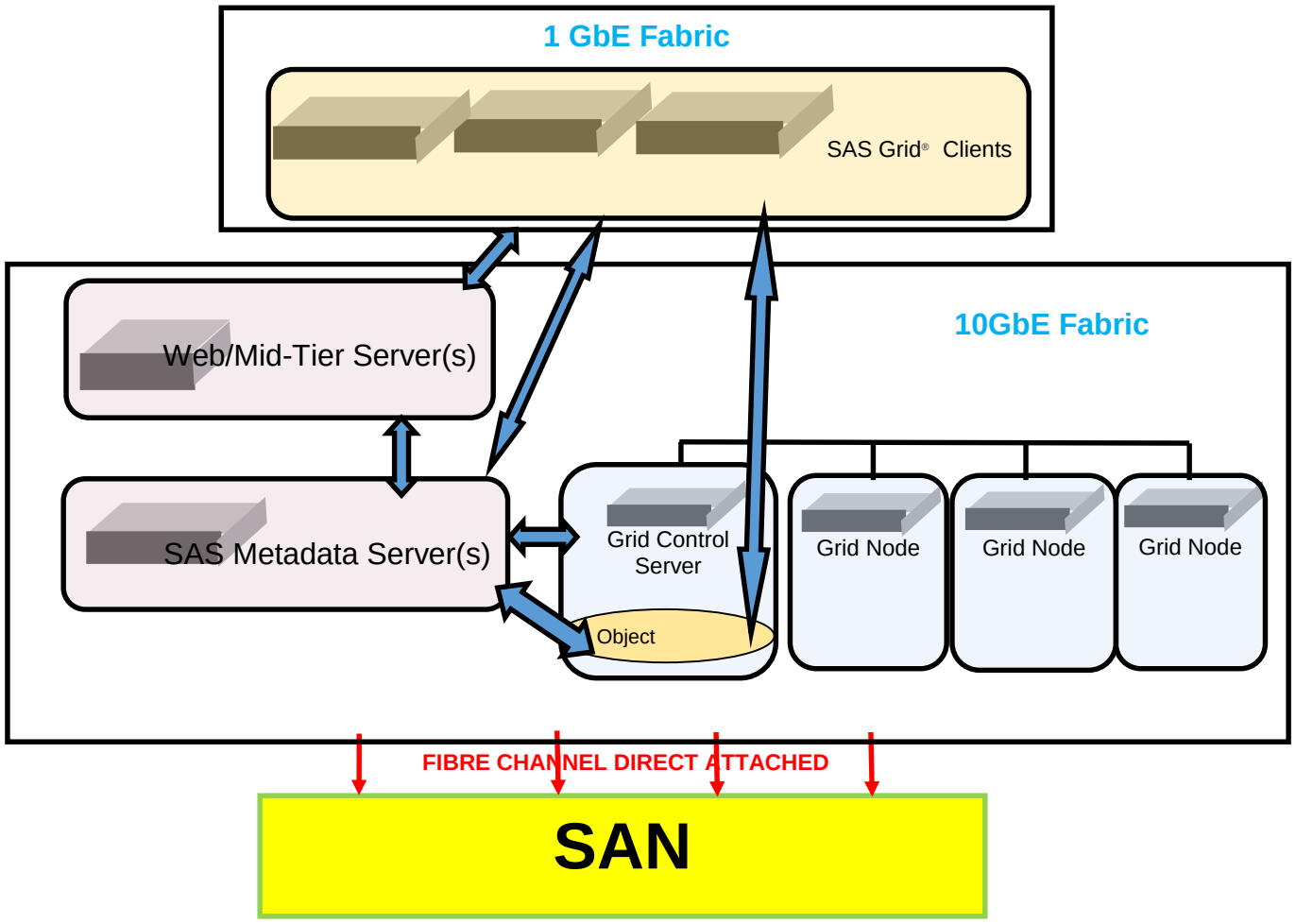

Figure 2. Direct Attached Storage (Non-Network Option) for SAS Grid.

Network-attached Storage

For network-attached storage, there are several options. Two common methods are depicted in Figure 3. One method is to use Network Shared Disk (NSD) protocol servers. This is a protocol supported by IBM Spectrum Scale storage with File Placement Optimizer. The NSD servers are in turn direct-attached, to a SAN or NAS. NSD servers allow significant flexibility in storage placement and sharing, but its real strengths can lie in redundancy in the face of failure (asynchronous file recovery and failover), and in providing load balancing to storage. Typically, at least two NSD servers are used to front each storage array for bandwidth and redundancy in the event of failure, providing more safety. This option is depicted on the left side of Figure 3.

The right side of Figure 3 depicts a traditional NAS or clustered NAS (NAS devices can be clustered together for management via storage cluster management software). This configuration comes with all the underlying pros and cons discussed previously in this paper for filer-based and network-based storage.

Network-attached storage allows significant flexibility for both scaling up and scaling out. It can take advantage of software-defined storage (SDS) management, discussed later in this paper, with underlying virtual creation and management of storage resources. It can be used successfully for SAS Grid utilization of storage. It typically will not perform quite as well as direct-attached storage for all the reasons discussed previously.

However, it is flexible, and it can be more cost effective. So there are trade-offs. Pay close attention to the section in this paper that discusses complicating factors.

The overlying factor here is that network bandwidth supporting an NSD or traditional NAS approach must supply the 100 MB/sec per core requirement for all of your SAS Grid compute nodes. In very small implementations, 10 GbE connections can be bundled and might suffice, but typically we have had better experiences with InfiniBand fabrics for this layer.

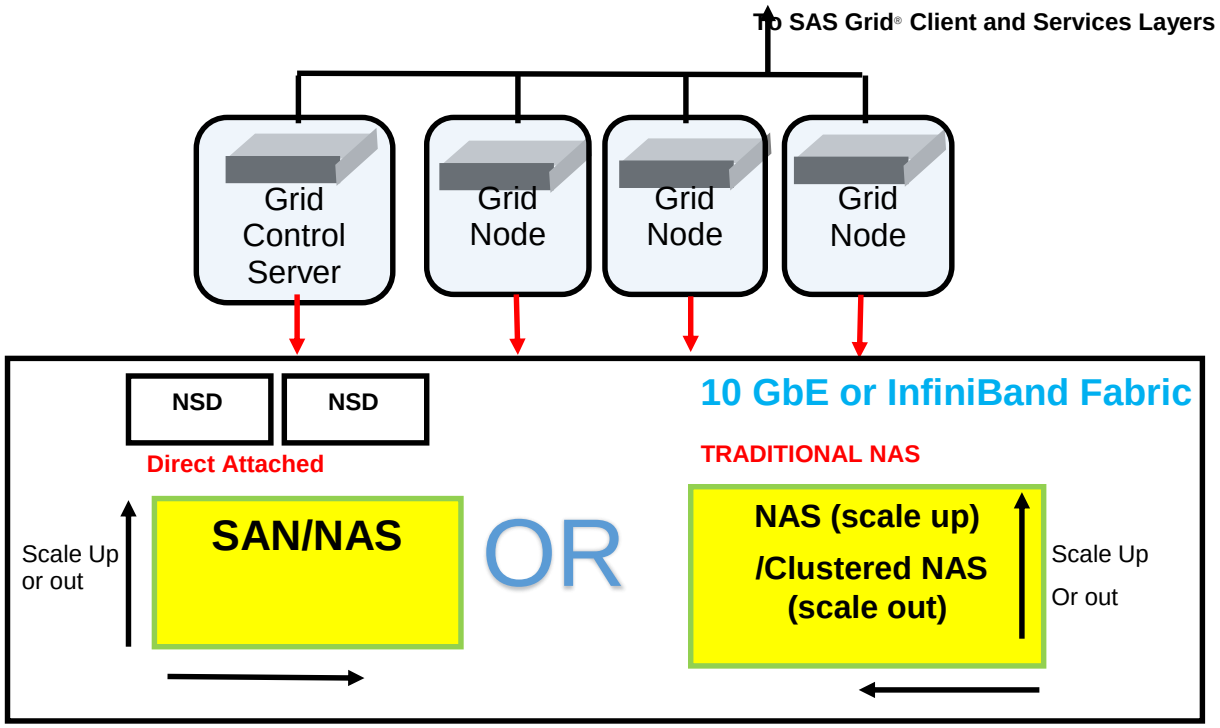

Figure 3. Network Shared Disk and Traditional NAS Arrangements for SAS Grid

Scale-Out Storage

A new type of networked storage is becoming popular, and it consists of scaled-out appliances. The storage is integrated into the actual individual SAS Grid node. There are a couple of vendors who have this available. The example shown below uses IBM Spectrum Scale storage with File Placement Optimizer. It uses the Spectrum Scale clustered file system with the File Placement Optimizer functionality that manages file placement across the nodes. Data is physically stored on one node, to be shared across all the nodes. The File Placement Optimizer collocates the stored data with the node where the data is processed the most. This configuration is meant to be an easily expandable, scale-up arrangement for SAS Grid growth. You add a node and its associate storage as a building block unit.

This type of appliance scale-out is software managed, and it requires a very robust network bandwidth for appliance attachment. Our testing has achieved excellent results using a 40 Gb InfiniBand fabric. Using a lesser fabric with this type of data-sharing arrangement across nodes is not recommended. The data flow from node to node can be heavy at times, despite intelligent file placement for collocated node processing. Figure 4shows the basic concept of scale-out appliance nodes.

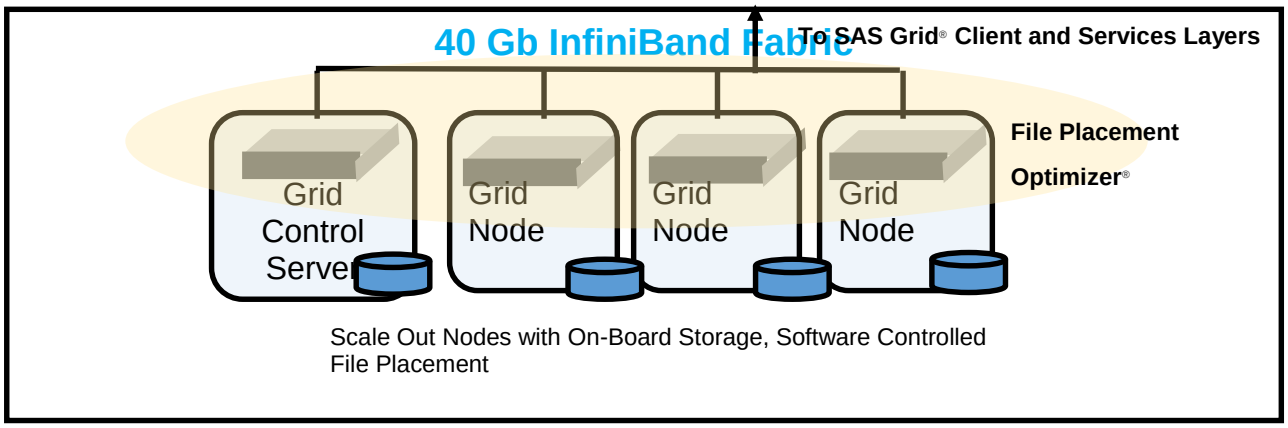

Figure 4. Scale-Out Node Appliances, with Software Managed File Placement, Best on High-bandwidth InfiniBand Networks

There are several important additional factors to consider when you select the network fabrics for the various layers of your SAS Grid. These include:

- Physical construction of the network, such as LAN, regional, or WAN
- Private versus public LAN segments
- Number of bridges, routers, and switches that data must traverse (each hop can bring some added latency)
- Utilization of sub-nets and segmented networks to localize and manage heavy traffic
- Dedicated versus virtualized servers and storage (addressed later in this document)
- Network load for layers that involve heavy data movement

These considerations are part of normal network performance architecture. Please work closely with LAN and WAN administrators to ensure that the physical construction and throughput of your entire network is optimized to achieve and sustain the suggested bandwidth for each of the SAS Grid architectural layers covered above.

NETWORK-BASED FILE SYSTEMS AND KNOWN ISSUES

NETWORK FILE SYSTEM (NFS) CACHE COHERENCY ISSUES

In order to ensure file data consistency, any metadata in the local file cache is invalidated (or flushed) when an NFS client detects a change in a file system attribute. The next time the file is accessed, its metadata will be retrieved from the NFS server. This means that retention in the file cache might have very different behavior with an NFS file system than when compared to other file systems. The file system storage devices and network must be provisioned to handle a larger demand when compared to either a local file system or a shared file system that uses a different strategy for cache coherency.

This dropping of the cached attributes causes metadata re-reads from the NFS server in order to re-obtain them. This in turn results in high process and I/O latency. In a SAS environment where there are many processes performing Write activities, this latency often introduces very noticeable application I/O slowness. This issue is persistent in both NFS3 and NFS4. It is exacerbated by frequent file metadata access in the heavily used SASWORK file system. This behavior is also more prevalent in Write operations than in Read operations. For these reasons, we typically recommend that NFS and file systems based on NFS (like EMC Isilon OneFS) be used for file systems that have heavy read usage (such as permanent SAS data), and not

used for file systems that have heavy write usage (such as SASWORK or UTILLOC).

NFS FILE AVAILABILITY ON UPDATE

The NFS client maintains a cache of file and directory attributes. The default NFS settings, which are associated with file closings and file metadata updates, will not ensure that files created or modified on one system will be visible on another system within a minute of file creation or modification. The default settings might cause software to malfunction if multiple computer systems are accessing data that is created or modified on other computer systems. For example, if a workspace server on system A creates a new SAS data set, that file might not be visible on System B within one minute of its creation.

In order to assure a consistent view of the file system, the file system mount option ACTIMEO= (attribute cache timeout) should be set to 0. This setting will increase the number of requests to the NFS server for directory and file attribute information, but it will also ensure that the NFS client systems have a consistent view of the file system. File data modifications might not be visible on any NFS client system (other than the system where the modifications are being made) until an NFS commit is executed. Most NFS clients will issue a commit as part of closing a file. If multiple systems are reading files that can be modified, file system locking should be used. This is controlled by the SAS system option FILELOCKS=, the SAS library option FILELOCKS=, or the SAS LOCK statement.

Additional NFS mount options to reduce messaging traffic include:

- Noatime – disables updates to metadata timestamps on the file's last access
- Nomtime – disables updates to metadata timestamps on the file's last modification.

These options are frequently used in SAS systems and can reduce some file system messaging traffic when used.

EMC ISILON KNOWN ISSUES

EMC Isilon storage uses the OneFS file system. The minimum recommended release for use with SAS Grid is 7.2.0.3 or higher. The OneFS file system is based on NFS and its underlying protocols. Because it is based on NFS, it experiences the same types of issues as those noted above for other NFS file systems. These issues include cache coherency, file locking, and general network performance . These issues require tuning to mitigate some of the behavior. There has been significant testing of EMC Isilon with OneFS, and this testing has resulted in the creation of a usage and advisory guide, "Advisory Regarding SAS Grid Manager with Isilon" (listed in the Recommended Reading section). It contains file mount options, network interface tuning options, and ancillary tuning options.

SUMMARY – NFS AND ONEFS

In summary, given the above stated issues with NFS and OneFS, we have proposed a general advisory to not use these network-based file systems for SASWORK or UTILLOC activity. We recommend relegating this type of network access to permanent SAS data storage utilization. Please review the Recommended Reading section below for detailed suggested file system usage and tuning for SAS Grid configurations that use NFS and OneFS file systems.

COMPLICATING FACTORS

We often run into conflicting priorities customers face when architecting a storage system. Performance is typically the goal that is a stated priority, along with reliability, security, and stability. These attributes rae typically followed in priority with ease of use and management. While these priorities always tend to begin in this general order, we have found that they can be quickly rearranged to suit budget and administration constraints.

While performance is given as the most important attribute for a storage infrastructure, in practice we often see it relegated far below the decisions made concerning commodity hardware, ease of management, and scalability. This section will touch on issues that arise with balancing this primary goal of performance against ease of management, maximum utilization of shared resources, and scalability.

MANAGEMENT, SCALABILITY, AND EASE OF USE

Management While Scaling

One of the biggest challenges IT faces is scaling storage and processing systems to meet both rising demand and data and process growth. Building silos of scale-up systems becomes risky, isolates data, and requires significant human management for setup and ongoing storage provisioning. There are a myriad of approaches toward scaling systems, including virtualizing systems, greater pool sharing of existing resources for maximization, and implementing all of the above via software applications to abstract logical and physical layers. Part of the solution involves software defined storage and host virtualization.

Software Defined Storage

There are a myriad of definitions for SDS. SDS can mean different things to different vendors, depending on their offerings. Essentially, it is the abstraction of storage management away from physical storage. It is a similar concept to host virtualization, but for storage. With a friendly user interface you can create storage resources, assign them, grow them, move them around, manage them, and so on. You can manage a single storage unit or many networked units. One of the issues with this approach is the underlying virtual storage management that SDS relies on. It has many of the same concepts as server virtualization management – creating large pools of shared resources and allocating out of those pools to assign space across the virtual network. The key issues that have to be guarded against are typically thin provisioning leading to over-subscription, sharing storage pools with small-block applications, creating pools with insufficient physical device backing for throughput, and so on. The primary issues for the network-related performance aspects of these virtual resources are:

- Defining virtual storage that is physically collocated too far from the consuming host (in other words, too many IP jumps to travel)
- Virtual networked storage that must traverse insufficient bandwidth fabric to support SAS Grid throughput
- Storage that resides on network file system types that have previously discussed issues (such as cache coherency, and locking).

If you are using software defined storage or virtual storage systems, due diligence must be paid to the physical network layer, its bandwidth, physical co-location, and the type of file system protocol it uses. If this is not examined, insufficient throughput and bandwidth can result, degrading your SAS Grid performance.

CONCLUSION

SAS Grid performance relies on the underlying hardware delivery systems. Sufficiently resourced bandwidth and throughput are critical. The target system bandwidth is to provide100 MB/sec per core for each Grid compute node. Consider these minimum bandwidth recommendations for the underlying fabric and connections for the SAS Grid layers:

SAS client tier – 1 GbE fabric, Citrix servers if served via WAN

SAS services tier – SAS Web ApplicationServer, middle tier server - 10 GbE fabric

SAS host-to-storage tier – (if not directly attached) bundled 10 GbE, bundled InfiniBand fabric to propagate bandwidth (preferred).

Providing the appropriate network fabric, interface cards, routers, switches, and so on, carefully managing network file system issues via tuning, and implementing proper usage per file system type (DATA, WORK, UTILLOC) can

ensure your SAS Grid functions and performs well.

RECOMMENDED READING

SAS Institute Inc., 2015. SAS Institute white paper. “Advisory Regarding SAS Grid Manager With Isilon.” Available http://support.sas.com/resources/papers/Advisory-Regarding-SAS-Grid-Manager-with-Isilon.pdf

EMC Corporation. 2016. Support article “OneFS: Best Practices for NFS client settings” (Registration required for access.) Available https://support.emc.com/kb/90041

EMC Corporation. 2012. EMC Corporation white paper. “EMC Isilon Storage and VMware vSphere 5: Optimize VMware vSphere 5 Using Isilon Scale-Out Storage from EMC.” (Registration required for access.) Available https://support.emc.com/docu45332_White-Paper:-EMC-Isilon-Storage-and-VMware-vSphere-5.pdf?language=en_US

VMware, Inc. 2009. VMware Inc. white paper. “Best Practices for Running VMware vSphere on Network Attached Storage.” Available http://www.vmware.com/files/pdf/VMware_NFS_BestPractices_WP_EN.pdf

VMware, Inc. 2013. VMware technical documentation. “Best Practices for Runing VMware vSphere on Network-Attached Storage (NAS).” Available http://www.vmware.com/files/pdf/techpaper/VMware-NFS-Best-Practices-WP-EN-New.pdf

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Tony Brown
100 SAS Campus Drive
SAS Institute Inc.
Cary, NC 27513
+1 (469) 807-7455
Tony.brown@sas.com

Margaret Crevar
100 SAS Campus Drive
SAS Institute Inc.
Cary, NC 27513
+1 (919) 531-7095
Margaret.crevar@sas.com