

# THE CLOUD STORAGE ARGUMENT

---

The argument over the right type of storage for data center applications is an ongoing battle. This argument gets amplified when discussing cloud architectures both private and public. Part of the reason for this disparity in thinking is that there is no ‘one size fits all solution.’ The other part of the problem is that there may not be a current right solution at all.

When we discuss modern enterprise data center storage options there are typically five major choices:

- Fibre Channel (FC)
- Fibre Channel over Ethernet (FCoE)
- Internet Small Computer System Interface (iSCSI)
- Network File System (NFS)
- Direct Attached Storage (DAS)

In a Windows server environment these will typically be coupled with Common internet File Service (CIFS) for file sharing. Behind these protocols there are a series of storage arrays and disk types that be used to meet the applications I/O requirements.

As people move from traditional server architectures to virtualized servers, and from static physical silos to cloud based architectures they will typically move

away from DAS into one of the other protocols listed above to gain the advantages, features and savings associated with shared storage. For the purpose of this discussion we will focus on these four: FC, FCoE, iSCSI, NFS.

The issue then becomes which storage protocol to use for transport of your data from the server to the disk. Depending on who you're talking to it's not uncommon to find extremely passionate opinions. There are quite a few consultants and engineers that are hard coded to one protocol or another. That being said most end-users just want something that works, performs adequately and isn't a headache to manage.

Most environments currently work on a combination of these protocols, plenty of FC data centers rely on DAS to boot the operating system and NFS/CIFS for file sharing. The same can be said for iSCSI. With current options a combination of these protocols is probably always going to be best, iSCSI, FCoE, and NFS/CIFS can be used side by side to provide the right performance at the right price on an application by application basis.

The one definite fact in all of the opinions is that running separate parallel networks as we do today with FC and Ethernet is not the way to move forward, it adds cost, complexity, management, power, cooling and infrastructure that isn't needed. Combining protocols down to one wire is key to the flexibility and cost savings promised by end-to-end virtualization and cloud architectures. If that's the

case which wire do we choose, and which protocol rides directly on top to transport the rest?

10 Gigabit Ethernet is currently the industries push for a single wire and with good reason:

- It's currently got enough bandwidth/throughput to do it (10gigabits using 64b/66b encoding as opposed to FC/Infiniband which currently use 8b/10b with 20% overhead)
- It's scaling fast 40GE and 100GE are well on their way to standardization (As opposed to 16G and 32G FC)
- Everyone already knows and uses it, yes that includes you.

For the sake of argument let's assume we all agree on 10GE as the right wire/protocol to carry all of our traffic, what do we layer on top? FCoE, iSCSI, NFS, something else? Well that is a tough question. the first part of the answer is **you don't have to decide**, this is very important because none of these protocols is mutually exclusive. The second part of the answer is, maybe none of these is the end-all-be-all long-term solution. Each current protocol has benefits and draw backs so let's take a quick look:

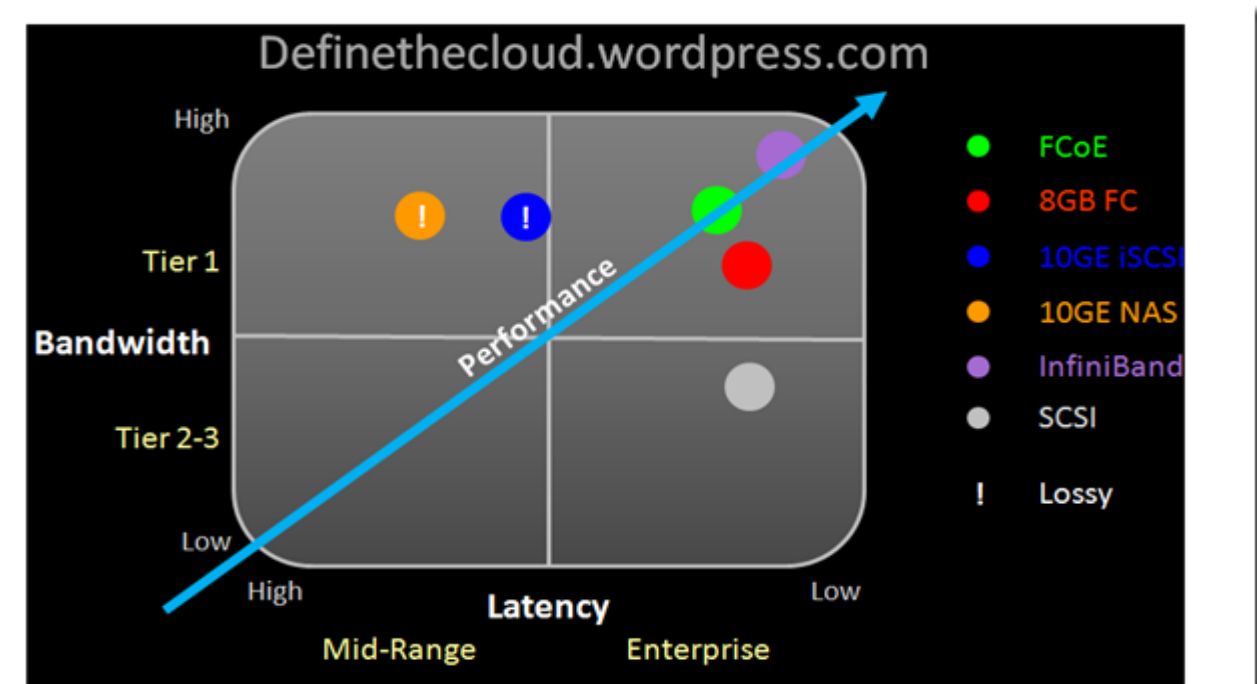
- iSCSI: Block level protocol carrying SCSI over IP. Works with standard Ethernet but can have performance issues on congested networks, also incurs IP protocol overhead. iSCSI is great on standard Ethernet networks until

congestion occurs, once the network becomes fully utilized iSCSI performance will tend to drop.

- FCoE: Block level protocol which maintains Fibre Channel reliability and security while using underlying Ethernet. Requires 10GE or above and DCB (<http://www.definethecloud.net/?p=31>) capable switches. FCoE is currently well proven and reliable at the access layer and a fantastic option there, but no current solutions exist to move it up further into the network. Products are on the road map to push FCoE further into the network but that may not necessarily be the best way forward.
- NFS: File level protocol which runs on top of UDP or TCP and IP.

And a quick look at comparative performance:

### Protocol Performance



While the above performance model is subjective and network tuning and specific equipment will play a big role the general idea holds sound.

One of the biggest factors that needs to be considered when choosing these protocols is block vs. file. Some applications require direct block access to disk, many databases fall into this category. As importantly if you want to boot an operating system from disk block level protocol (iSCSI, FCoE) are required. This means that for most diskless configurations you'll need to make a choice between FCoE and iSCSI (still within the assumption of consolidating on 10GE.) Diskless configurations have major benefits in large scale deployments including power, cooling, administration, and flexibility so you should at least be considering them. If you chosen a diskless configuration and settled on iSCSI or FCoE for your boot disks now you still need to figure out what to do about file shares? CIFS or NFS are your next decision, CIFS is typically the choice for Windows, and NFS for Linux/UNIX environments. Now you've wound up with 2-3 protocols running to get your storage settled and your stacking those alongside the rest of your typical LAN data.

Now to look at management step back and take a look at block data as a whole. If you're using enterprise class storage you've got several steps of management to configure the disk in that array.

It varies with vendor but typically something to the effect of:

1. Configure the RAID for groups of disks
2. Pool multiple RAID groups
3. Logically sub divide the pool
4. Assign the logical disks to the initiators/servers
5. Configure required network security (FC zoning/ IP security/ACL, etc)

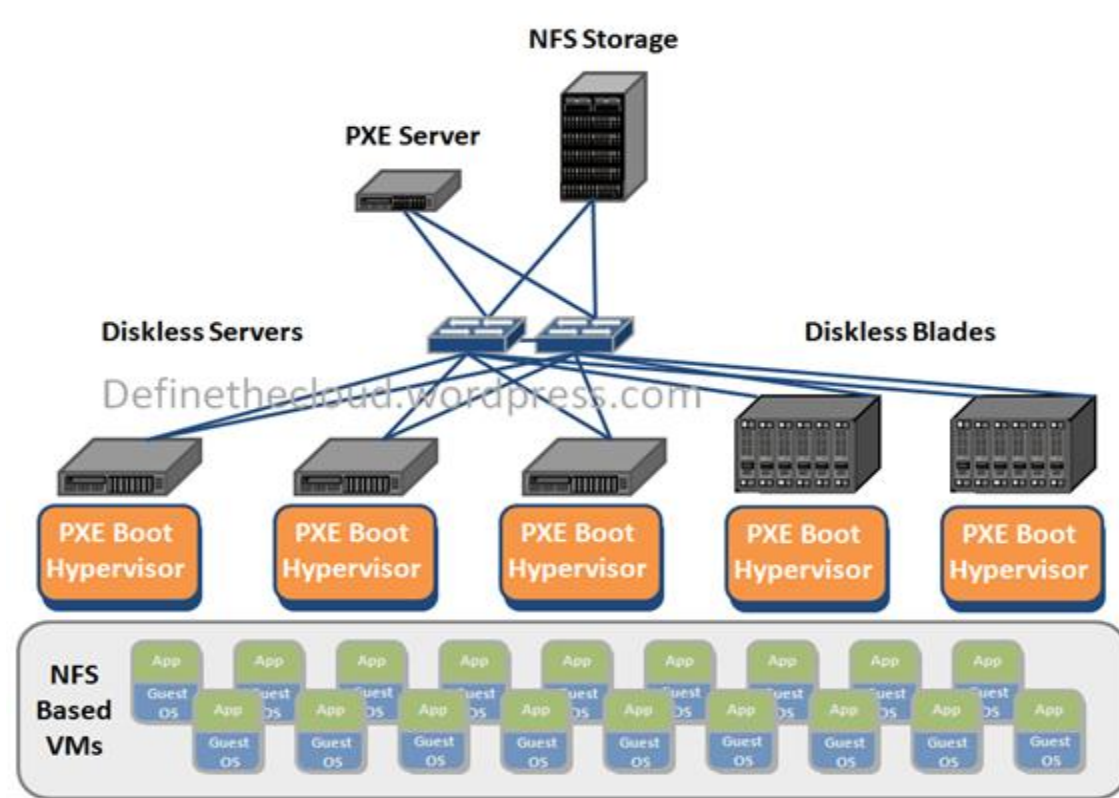
While this is easy stuff for storage and SAN administrators it's time consuming, especially when you start talking about cloud infrastructures with lots and lots of moves adds and changes. It becomes way to cumbersome to scale into petabytes with hundreds or thousands of customers. NFS has more streamlined management but it can't be used to boot an OS. This makes for extremely tough decisions when looking to scale into large virtualized data center architectures or cloud infrastructure.

There is a current option that allows you to consolidate on 10GE, reduce storage protocols and still get diskless servers. It's definitely not the solution for every use case (there isn't one), and it's only a great option because there aren't a whole lot of other great options.

In a fully virtualized environment NFS is a great low management overhead protocol for Virtual Machine disks. Because it can't boot we need another way to get the operating system to server memory. That's where PXE Boot comes in

Pre eXecutionEnvironment (PXE) is a network OS boot that works well for small operating systems, typically terminal clients or Linux images. It allows for a single instance of the operating system to be stored on a PXE server attached to the network, and a diskless server to retrieve that OS at boot time. Because some virtualization operating systems (Hypervisors) are light weight, they are great candidates for PXE boot. This allows the architecture below.

### PXE/NFS 100% Virtualized Environment



Source: <http://www.definethecloud.net/the-cloud-storage-argument/>