Using Hammerspace with RDMA

TECHNICAL BRIEF

Summary

Modern applications are fueled by data. Nearly every industry is impacted by data-driven applications. Workflows requiring Petabytes of data are no longer unusual, especially in fields such as artificial intelligence (AI). But data today is everywhere – at the edge, in multiple data centers, and in multiple clouds. The speed at which compute nodes can access data is sometimes the bottleneck in a data processing pipeline. Improved data transport efficiency is required to solve this problem, not just an increase in wire speed.

RDMA technology can provide an answer, using hardware offloading and a streamlined protocol stack to greatly reduce latency between storage systems and the servers that need to access them. But RDMA comes at a cost, and it's not always the right tool for the job.

Hammerspace is a software-defined data storage and orchestration system that provides unified access to files via a high performance Parallel Global File System that can span any storage type from any vendor, in any location. Hammerspace supports RDMA to provide clients with ultra-low latency access to storage when it makes sense.

This Tech Brief begins with a high-level explanation of how RDMA works and the circumstances in which it's beneficial, followed by a discussion of the key requirements for implementing RDMA with the Hammerspace Data Platform.

More Efficiency = Lower Latency

When two networked computers talk to each other, it is through a layered stack of protocols. A client connecting to a file server over Ethernet, for example, may use the NFS, TCP and IP protocols. As data moves up and down through the layers, associated copy operations must take place between memory buffers, as shown on the left side of Figure 1. The copies take time, and are executed by the system CPU, taking cycles that may be better spent elsewhere.

Table of Contents

- Requirements for Data Hungry Applications
- Remote Direct Memory Access (RDMA) Reduces Storage Latency
- Using RDMA with the Hammerspace Data Platform

HAMMERSPACE

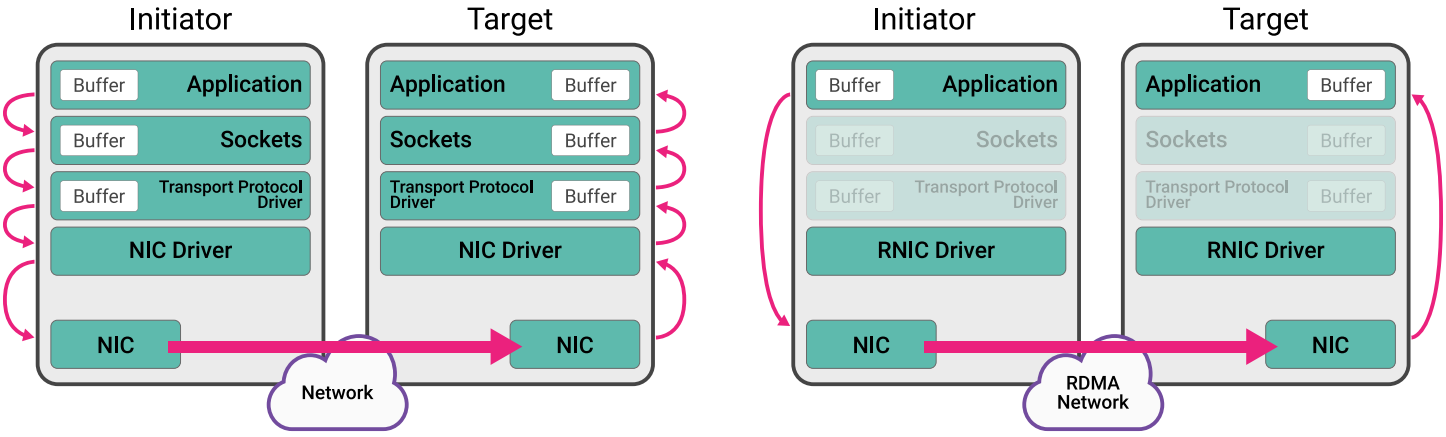


Figure 1 - Non-RDMA vs. RDMA

This all happens quickly, but if there is a lot of data, the extra time adds up and can slow down applications. Faster network infrastructure might help some, but just because the pipe gets larger doesn't mean that servers can fill it, and CPU utilization will spike as they try, leaving fewer cycles for applications.

What if there were a way for the CPU to set up a connection between the initiator and target applications and then 'get out of the way', letting them talk to each other directly? That would be more efficient, and it's the essential idea behind RDMA, as illustrated on the right side of Figure 1. This is sometimes referred to as a 'zero copy' approach, since the intermediate CPU-driven copies are eliminated. Instead of the main system CPUs, the work of moving the bits over the wire is delegated to small processors on RDMA-capable NICs, sometimes called RNICs. In the case of Hammerspace, the "target application" is an NFS server on a storage system.

The benefits? Applications get a lower-latency data path for more rapid roundtrips to storage, and there are more CPU cycles for them to use. But to experience these benefits you need the proper infrastructure.

Applications for RDMA

What is this new, more efficient method of networking good for? Many things. NVIDIA has GPUDirect Storage, (Figure 2) which uses RDMA to connect GPUs to local or networked storage. NVMe-oF enables NVMe flash storage devices to be shared across an RDMA-enabled network. And NFS over RDMA (NFSoRDMA) can speed up file sharing over NFS.

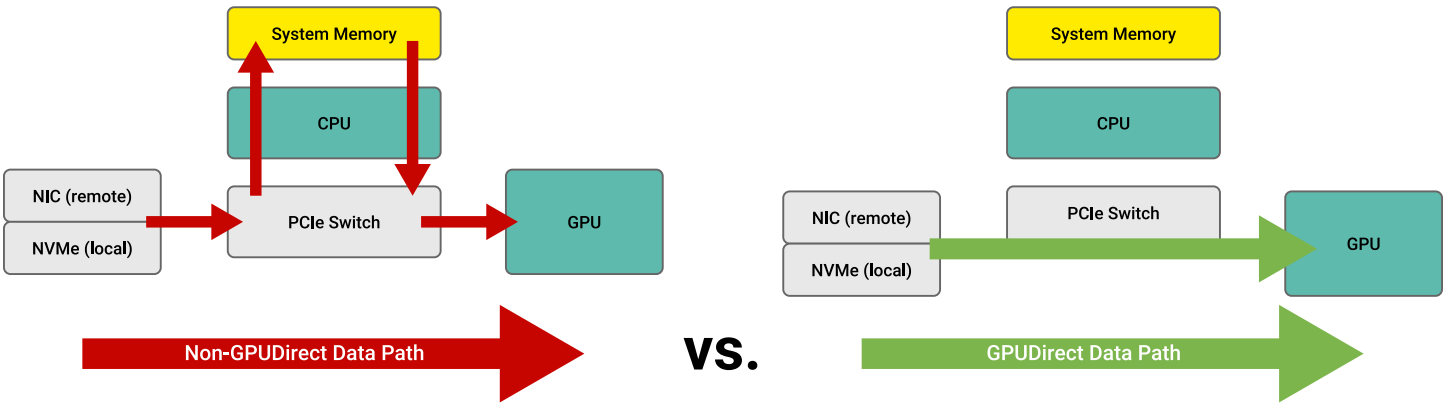


Figure 2 - GPUDirect Using RDMA

To determine if using RDMA is the right tool for a particular job, one must first examine whether the application suffers from a problem that RDMA can solve. If the application is compute bound, spending most of its time processing data vs. moving it, RDMA is not likely to increase performance. At the other extreme, if the application is I/O bound, with CPUs and GPUs suffering from low utilization because they are starved of data, RDMA could provide a big boost.

Another factor to consider is whether there are alternative, potentially easier ways to achieve the desired performance. With parallel applications, adding more compute nodes and storage to the cluster might be the most straightforward approach. In other words, scaling up, vs. spending lots of effort wringing the last bit of performance out of a smaller cluster.

Even if the application will benefit from RDMA, one must determine if the benefit is worth the additional complexity. If the network infrastructure supports it and the network team has experience with lossless Ethernet, or if there is an IB network already in place, the path may be smooth. If Ethernet is used and the network involved is large, complicated, or aged, or the network team is unfamiliar with the technology and resistant to change, implementing RDMA will be far more difficult.

NVIDIA GPUDirect has additional requirements, and applications must be written to specific APIs to take advantage of it. Again, organizations must weigh the costs vs. the benefits. More on using Hammerspace with GPU Direct may be found in a separate Tech Brief.

The good news is that the standards based Hammerspace Data Platform supports RDMA, but does not require it, and provides great performance either way. Without RDMA, it's possible to achieve 90% of theoretical network performance. With RDMA, 95% or even more is possible.

RDMA Transports

RDMA was born in the world of high-performance parallel computing (HPC) and is inherent to the InfiniBand (IB) networks which are common in that field. IB networks are still considered somewhat exotic outside HPC, though. What brings RDMA more into the mainstream is the ability to use it over Ethernet.

There are three commonly available transports for RDMA. InfiniBand of course, and two options on Ethernet; iWARP and RoCE. RoCEv1 is not routable, and has been supplanted by RoCEv2, which is.

RDMA requires a lossless network, meaning packets must not be dropped or the connection will be broken. IB networks are lossless by design, but one of the fundamental characteristics of Ethernet is that it is designed for "best effort" communication. On Ethernet, upper-level protocols are expected to deal with the fact that sometimes packets will be dropped.

Packet drops occur when a network device runs out of buffer space because packets are arriving faster than they can depart. Several technologies have been devised to avoid drops so that lossless channels may be created over Ethernet.

Some, like Priority-based Flow Control (PFC) and Enhanced Transmission Selection (ETS) provide the ability to prioritize traffic, giving lossless traffic a higher priority. Others such as Congestion Notification (CN) and Explicit Congestion Notification (ECN) enable receivers to communicate to senders that they are running out of buffer space, so that the senders may pause or throttle their rate of transmission. Data Center Quantized Congestion Notification (DCQCN) uses a combination of PFC and ECN to provide an end-to-end congestion

control protocol for RoCEv2. Data Center Bridging (DCB) is an umbrella term under which fall PFC, ETS, CN, and Data Center Bridging Capabilities Exchange (DCBCX), a method by which network neighbors may communicate to each other which of these features they support.

The previous paragraph may leave the reader with the impression that constructing a lossless Ethernet is complex. It can be. There are multiple ways to accomplish the goal, and both the endpoints and all switches and routers in between must be configured correctly for it to work. Configuration details are manufacturer specific. The good news is that the technology has been in the market for quite a few years, and more network teams are becoming familiar with it.

IB may seem simpler in some ways, and it probably is if there is IB infrastructure already in place and a team that is familiar with it. New deployments of IB mean installing and managing a separate parallel network, with the associated expenses. This may be practical for small, localized clusters, but will be less attractive when a pre-existing high-speed Ethernet network is available, especially at large scale.

RDMA In a Hammerspace Data Platform Environment

Hammerspace Architecture Overview

Hammerspace is a software-defined data platform that provides unified file access via a high-performance Parallel Global File System that can incorporate any storage type, from any vendor, across multiple geographic locations, clouds, and cloud regions.

Designed to make data a global resource across distributed or otherwise incompatible storage platforms, Hammerspace presents a cross-platform global namespace where users and applications have direct multi-protocol access to all files, regardless of storage type or location.

A Hammerspace environment contains a few different components, illustrated in Figure 3. Note that metadata is handled separately from data, as is common in proprietary parallel file systems. Hammerspace, however, leverages parallel NFS to provide scalable performance without requiring proprietary client software.

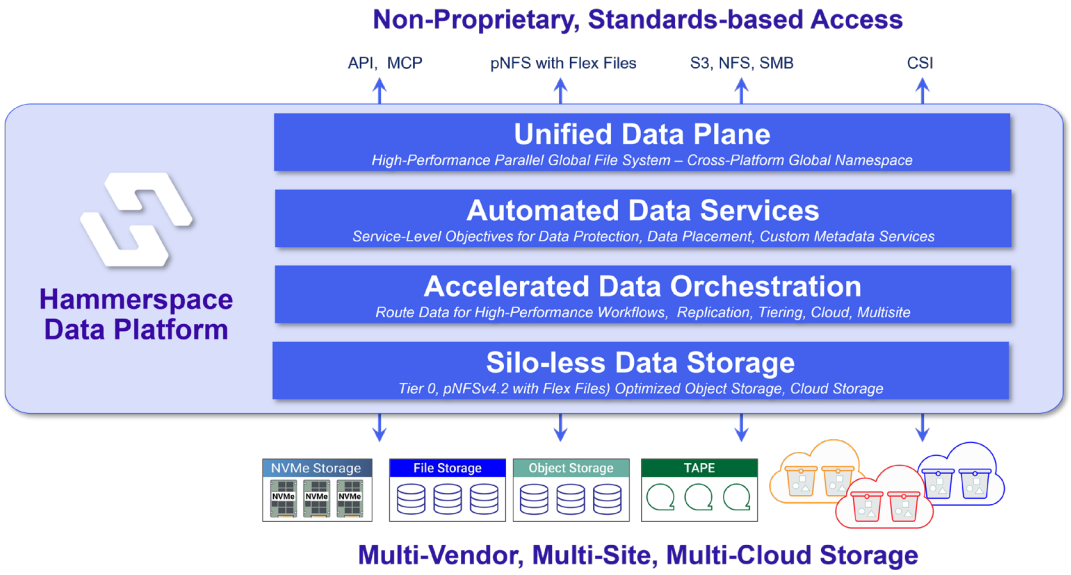


Figure 3 - Hammerspace Architecture Overview

Anvil nodes manage all the metadata within the system and supervise all the actions of the DSX nodes, and are typically deployed in an HA pair at each site to provide resilience.

DSX (Data Services eXtension) nodes do the heavy lifting. They are the scale-out engines that connect data and storage to users and applications, performing I/O operations on any local storage, as well as replication and data orchestration.

Storage used with Hammerspace may be any combination of direct attached storage (e.g. NVMe drives attached to DSX nodes), NAS, object/cloud, and even tape.

For a much more detailed discussion of the Hammerspace solution architecture and capabilities, refer to the Hammerspace Technology White Paper.

RDMA with Hammerspace

Hammerspace supports RDMA on RoCEv2 or InfiniBand between clients and DSX nodes, and between DSX nodes. RDMA is not supported (or needed) between clients and Anvil nodes.

When Hammerspace software is installed on DSX nodes that contain NVIDIA ConnectX-4 or later generation RNICs, RDMA will be automatically enabled, using standard Linux drivers. The Hammerspace installation package includes the appropriate Linux distribution and drivers, so as long as approved RNICs are installed, compatibility is assured.

The Hammerspace Parallel Global File System uses NFSv4.2 and parallel NFS (pNFS). This enables clients to directly access storage devices once provided a 'layout' by the metadata server/Anvil. The layout defines the storage the client is being granted access to, and a path (or paths) to reach that storage.

When RDMA is enabled, layouts provided by the Anvil to clients will include both RDMA and non-RDMA paths, with RDMA being preferred. Clients that support RDMA will use the RDMA paths, and non-RDMA clients will access storage using the non-RDMA paths. Clients mount shares on the Anvil using NFSv4.2 but use NFSv3 to directly access storage devices. This is standard for pNFS using the FlexFiles layout type.

Hammerspace Tier 0 configurations also support RDMA, since Tier 0 uses pNFSv4.2. With Tier 0, Hammerspace turns local storage on GPU or CPU compute servers into an ultra-fast tier of shared persistent storage. Tier 0 nodes will use RDMA (provided it is enabled) whenever they need to access non-local data. To learn more about Tier 0 see the Solution Brief.

RDMA communication between DSX nodes is used for data mobility/data orchestration operations within a single site or Hammerspace installation.

For more details regarding NFSv4.2, pNFS, and how Hammerspace uses these protocols to provide a standards based, high-performance parallel file system, refer to the NFSv4.2 Tech Brief.

Requirements Summary

The following requirements must be fulfilled to use RDMA with Hammerspace:

Network

- Lossless Ethernet using RoCEv2 or InfiniBand configured on all network devices (switches, etc.) connecting clients to RDMA-capable storage. Configuration will be manufacturer specific and is beyond the scope of this document. Contact your network equipment vendor or VAR for assistance

DSX Nodes

- Whether DSX nodes are presenting attached storage devices directly, re-exporting NAS storage via the DSX Portal function, or simply used for data orchestration, RoCEv2 or InfiniBand networking is required to use RDMA
- Only NVIDIA/Mellanox ConnectX-4 or later RNICs are supported

NAS Storage Systems

- RDMA support via RoCEv2 or InfiniBand configured and enabled
- RNICs connected to the RDMA network
- NFSv3 configured and enabled

Clients

- OS and RNIC drivers that support RDMA
- RNICs configured for RDMA and connected to the RDMA network

Tier 0 nodes act as both NAS storage systems and clients, and must meet the requirements for both.

Conclusion

RDMA is supported in a Hammerspace Global Data Environment and can be a useful tool for accelerating storage performance for applications bottlenecked by high-latency storage paths.

The networking required to support RDMA may be unfamiliar to some organizations and may be complex, which can complicate deployments. Customers contemplating RDMA should:

- Carefully evaluate the ability of their applications to take advantage of RDMA and the scale of the expected benefits
- Be realistic about the level of effort required to deploy RDMA in their environment
- Compare alternatives to achieving the required performance, such as scaling out compute and storage clusters