

Delivering HPC Performance with Standards-Based Parallel NFS

Demonstrating How Hammerspace Delivers Performance for HPC and AI Workloads Using a Standards-Based Architecture

TECHNICAL BRIEF

Introduction

HPC performance depends on both the data path and the availability of the datasets a job needs. Hammerspace combines direct parallel access through pNFS with data-in-place assimilation, automated placement, and shared server-local NVMe through Tier 0. This technical brief explains how these capabilities support research computing and examines the IO500 and MLPerf Storage results that demonstrate performance in tested configurations.

Parallel File System, Built on Open Standards

Mixed HPC and AI environments need to share datasets across simulation, analysis, and GPU workloads while connecting to existing storage. Maintaining a separate file system client stack across these resources adds deployment and upgrade work. A parallel architecture accessed through the standard Linux NFS client offers a familiar interface for this broader environment. The key technical requirement is to distribute I/O across storage resources so that file data does not funnel through the metadata service.

With NFSv4.2 and pNFS, metadata and data paths are separate. Clients request a FlexFiles layout from the metadata server describing where file data resides and how to access it. They then read and write directly to the designated storage servers, allowing I/O to proceed across multiple servers in parallel without passing file data through the metadata server. FlexFiles also enables clients to report I/O errors and layout usage statistics back to the metadata server. Hammerspace builds on these capabilities to deliver parallel access across supported storage resources.

This follows the same proven architecture used by traditional HPC parallel file systems. The difference is that with pNFS, that parallel access is available through open standards and the NFS client included with modern Linux distributions, without requiring an additional proprietary file system client.



Architecture Highlights

Separate metadata and data paths

Metadata services coordinate access while clients move data directly to and from storage.

Direct parallel I/O

Clients can access multiple storage resources concurrently rather than funneling all data through a single NAS head.

No proprietary client required

NFSv4.2 and pNFS capabilities are implemented in the standard Linux NFS client.

Integrate Existing NFS Storage

FlexFiles supports direct file access to NFS data servers from any vendor, including standard NFSv3 systems. This gives customers flexibility on which storage resources they deploy; attainable performance still depends on the storage, network, client software, and configuration.

Hammerspace also controls how datasets become available to these parallel clients. **Data-in-place assimilation** brings metadata from existing storage into a global namespace while the underlying data remains where it is. Administrators can then use Hammerspace's service-level Objectives to place selected files on the storage needed for a workload, including server-local NVMe (Tier 0). Live Data Mobility allows placement to change without changing the application-visible file path, even on active files that are in use. This connects existing datasets to high-performance resources without first requiring a wholesale migration.

Learn more about parallel NFS architecture by reading this white paper:

[Parallel NFS – A Modern Protocol for High-Performance Workloads](#)

Proven HPC Performance on IO500

IO500 was created to evaluate storage performance under HPC workloads commonly found in modeling, simulation, academic research, government laboratories, and other compute-intensive environments. The benchmark combines bandwidth and metadata tests across a range of workload patterns and summarizes them into composite bandwidth, IOPS, and overall scores.



At ISC 2025, Hammerspace submitted its first IO500 result in the 10-Node Research category.

Hammerspace IO500 10-Node Research Results									
Overall Score			Composite Bandwidth			Composite Metadata Performance			
63.77			37.25 GiB/s			109.16 kIO/s			

The result placed Hammerspace alongside established HPC file systems including Lustre, IBM Spectrum Scale, WekaFS, and BeeGFS on the 10-Node Research list.

79	SC 19	DDN	AI400	DDN	Lustre	10	240	63.88	19.65	207.63
80	ISC 25	Hammerspace	Hammerspace Performance Lab	Hammerspace	pNFS v4.2 via Hammerspace Anvil Metadata Service	10	640	63.77	37.25	109.16
81	SC 21	Raytheon	MCK-HPC	Penguin Computing	WekaFS	10	360	62.55	25.96	150.70
82	SC 23	Center for Research Informatics at University of Chicago	Randi	IBM	Spectrum Scale	10	160	60.88	31.05	119.36
83	SC 19	University of Cambridge	Data Accelerator	Dell EMC	Lustre	10	320	60.25	21.73	167.09

Hammerspace also delivered particularly strong performance on IOEasy, the IO500 test using file-per-process, large-block sequential I/O:

- 420.35 GiB/s IOEasy Write
- 446.79 GiB/s IOEasy Read

These results placed Hammerspace among the top 10 performers for both IOEasy Write and IOEasy Read on the ISC 2025 10-Node Research results.

- The IO500 configuration used 10 Linux clients, eight NVMe storage servers, and one Hammerspace metadata server connected over 400GbE using RoCE. File access used pNFSv4.2 with FlexFiles through the standard Linux NFS client, without requiring an additional proprietary file system client.

The result demonstrates that a standards-based parallel file system architecture can deliver competitive HPC performance without requiring a proprietary client stack or specialized storage infrastructure.

Learn more by reading this technical brief:
[Hammerspace Data Platform IO500 Benchmark Results](#)

HPC-class performance does not require a proprietary file system client.



Extending HPC Performance to AI and GPU Workloads

Traditional HPC and emerging AI workloads place different demands on data infrastructure. IO500 evaluates bandwidth, metadata, and other I/O characteristics associated with HPC environments. MLPerf Storage evaluates whether storage can deliver data fast enough to support AI/ML workloads at GPU scale.

Together, the benchmarks provide complementary views of performance for modern HPC + AI environments.

For MLPerf Storage v2.0, Hammerspace submitted a verified Closed Division result using the bandwidth-intensive 3D U-Net workload and simulated NVIDIA H100 GPUs. The implementation used pNFS and Hammerspace Tier 0.

MLPerf Storage v2.0 Results			
Tier 0 Nodes	Simulated H100 GPUs	Total Throughput	Mean Simulated GPU Utilization
1	28	85.6 GB/s	94.7%
3	84	253.1 GB/s	95.0%
5	140	420.8 GB/s	96.4%

As Tier 0 resources increased from one to three to five nodes, throughput increased from 85.6 GB/s to 253.1 GB/s to 420.8 GB/s while supporting 28, 84, and 140 simulated H100 GPUs respectively.

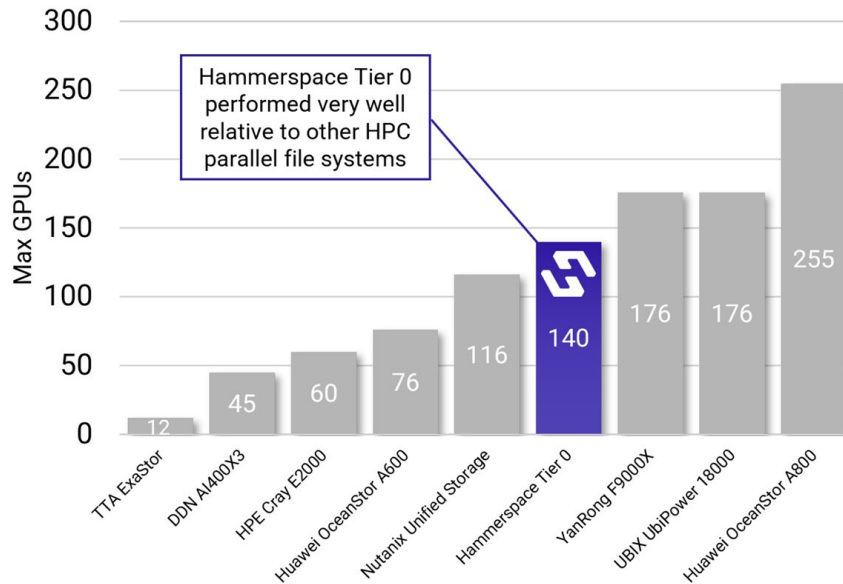
These results also benefited from LOCALIO, a Linux NFS optimization developed by Hammerspace that detects when the client and storage are on the same server. In that situation, LOCALIO bypasses the network stack and NFS server to access the local file system directly, reducing overhead while preserving access through NFS. This complements Tier 0 by accelerating access to data on the compute server’s own NVMe drives. The benchmark used a best-case configuration in which the primary dataset could reside entirely on the host. Access to data on other servers still uses the network, so performance depends on data placement and system configuration.

AI and GPU workloads demand high-throughput, low-latency data access - both for training and checkpointing, as well as for inferencing. Hammerspace orchestrates data to bring data close to compute, and a Tier 0 architecture extends the Hammerspace parallel file system architecture into the server-local NVMe storage already installed inside compute servers.



3D U-Net Max Simulated H100 GPUs Supported

(MLPerf Storage v2.0 Closed Division, Verified Results)



Instead of treating this NVMe solely as isolated node-local scratch, Tier 0 aggregates it into the Hammerspace shared high-performance tier of storage, part of the Hammerspace file system. Clients mount the file system using pNFSv4.2 and, after receiving a layout from Hammerspace metadata services, access storage directly. No application data flows through the metadata server.

For an HPC workflow, Tier 0 can hold active datasets or checkpoint files on compute-server NVMe while retaining shared file-system access. Hammerspace Objectives can govern placement and protection, including creating additional file instances on other storage. When protection is asynchronous, completing a local write and completing the downstream protection are separate milestones. This makes the placement and durability policy part of the workflow design.

The MLPerf results demonstrate the performance potential of extending the parallel data path into the compute layer: as additional Tier 0 resources were added, aggregate throughput scaled to support a larger simulated GPU environment.

One standards-based architecture can support traditional HPC and emerging AI workloads.

Learn more about the MLPerf results by reading this technical brief:

[Hammerspace MLPerf Storage v2.0 Benchmark Results](#)



A Standards-Based Architecture for Modern HPC + AI

In mixed HPC and AI environments, simulation output may become input for GPU analysis or model training, with the same datasets needed across different compute and storage resources. Hammerspace combines standard Linux client access with a global namespace and policy-driven data placement, allowing data to move between workflow stages while retaining consistent file paths. Standards matter because they reduce the need to maintain a separate client stack as the environment expands; orchestration makes that flexibility useful by placing data where the next stage needs it.

The IO500 and MLPerf Storage results provide measured examples of Hammerspace's parallel I/O and Tier 0 performance. Evaluating a deployment should begin with its own I/O patterns, dataset locality, network capacity, and protection requirements.

**Discuss your HPC workload with Hammerspace
to identify a configuration and validation approach
suited to your environment.**

Contact Us

