SOLUTION GUIDE

Hammerspace Tier 0 in the Cloud

HAMMERSPACE

Boost Cloud GPU Utilization with a Hammerspace Tier 0 Architecture

For enterprises running AI/ML workloads in the cloud, a new challenge is emerging: despite the abundance of GPU-accelerated compute, I/O bottlenecks continue to erode utilization, inflate costs, and extend time-to-insight. Many organizations rely on cloud-native object stores or managed file services, which are constrained by network bottlenecks that introducing latency between the storage and the compute, resulting in idle cloud GPUs.

By enabling the use of instance-local NVMe storage as a tier of high-performance shared storage in the cloud, a Hammerspace Tier 0 architecture delivers **2–2.5x higher throughput and 50% lower latency** vs. cloud file systems, boosting GPU utilization to accelerate AI pipelines.

This Solution Guide provides architecture, configuration, and operational guidance for deploying Hammerspace in the cloud to maximize GPU ROI, streamline hybrid cloud workflows, and minimize infrastructure overhead.

About Hammerspace

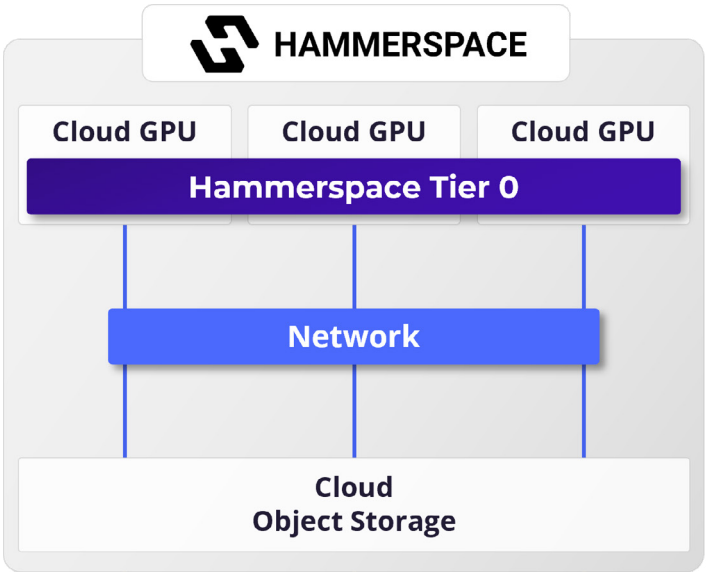
Hammerspace is a data platform that helps customers unlock the full value of their data by breaking down silos across on-premises, cloud providers, and cloud regions. Together with our cloud partners, we make it easy to move data where it's needed, speed up AI and HPC projects, and simplify hybrid-cloud operations. The result is faster time to value, lower infrastructure complexity, and greater consumption of cloud compute and storage.

Hammerspace is available on all four of the major US cloud platforms.

What is Hammerspace Tier 0?

A Hammerspace Tier 0 architecture turns the local NVMe storage inside GPU – and CPU - compute nodes into high-performance, shared, orchestrated storage.

On-premises, this means the local NVMe drives that are included in physical GPU servers can be added to the Hammerspace file system, providing low-latency data access to the



Unify instance-Local NVMe as high-performance shared storage

Automate data movement between Tier 0 and object, while maintaining visibility

Accelerate AI Pipelines by removing I/O bottlenecks to boost GPU utilization

GPUs while eliminating the operational complexity of manually moving data between local storage and external networked storage.

In the cloud, using Hammerspace Tier 0 means using the local NVMe that is available in most cloud GPU and CPU instances as a 'tier 0' of high-performance shared storage. This "tier 0" behaves like a shared file system while leveraging local storage that would otherwise remain underutilized.

Common cloud GPU instance types and their options for local NVMe storage are shown in the table in Appendix 1.

Highlights of a Hammerspace Tier 0 architecture in the cloud include:

- **No client or agent required:** Any cloud compute instance running Linux can take advantage of this architecture.
- **Deploy on standard cloud infrastructure:** Hammerspace software runs on standard cloud instance types, and uses standard cloud compute and storage resources.
- **Provides low-latency, high-throughput data access:** By using local NVMe, data can be served to GPUs at local NVMe speed, maximizing cloud GPU utilization and reducing GPU idle time.
- **Avoids bottlenecks of cloud north/south networks:** Tier 0 enables the use of the high bandwidth, "east/west" or internode networks used to communicate between compute nodes in the cloud.
- **Makes it easy to scale compute, performance, and capacity:** Tier 0 makes capacity scaling easy and predictable – scale capacity linearly as you add compute nodes to your cluster.
- **Supports multi-protocol data access** using industry standard file and object protocols – NFSv4.2, NFSv3, SMB, and S3. Hammerspace also supports the Kubernetes Container Storage Interface (CSI) for containerized workloads.
- **Automates data tiering to cloud object storage while maintaining visibility:** Tier 0 is a 'tier' of storage within a Hammerspace cloud file system. A Hammerspace cluster can extend to external flash storage, and cloud object storage – within a unified global namespace. This means older data can be moved to cost effective cloud object storage without disrupting user access.

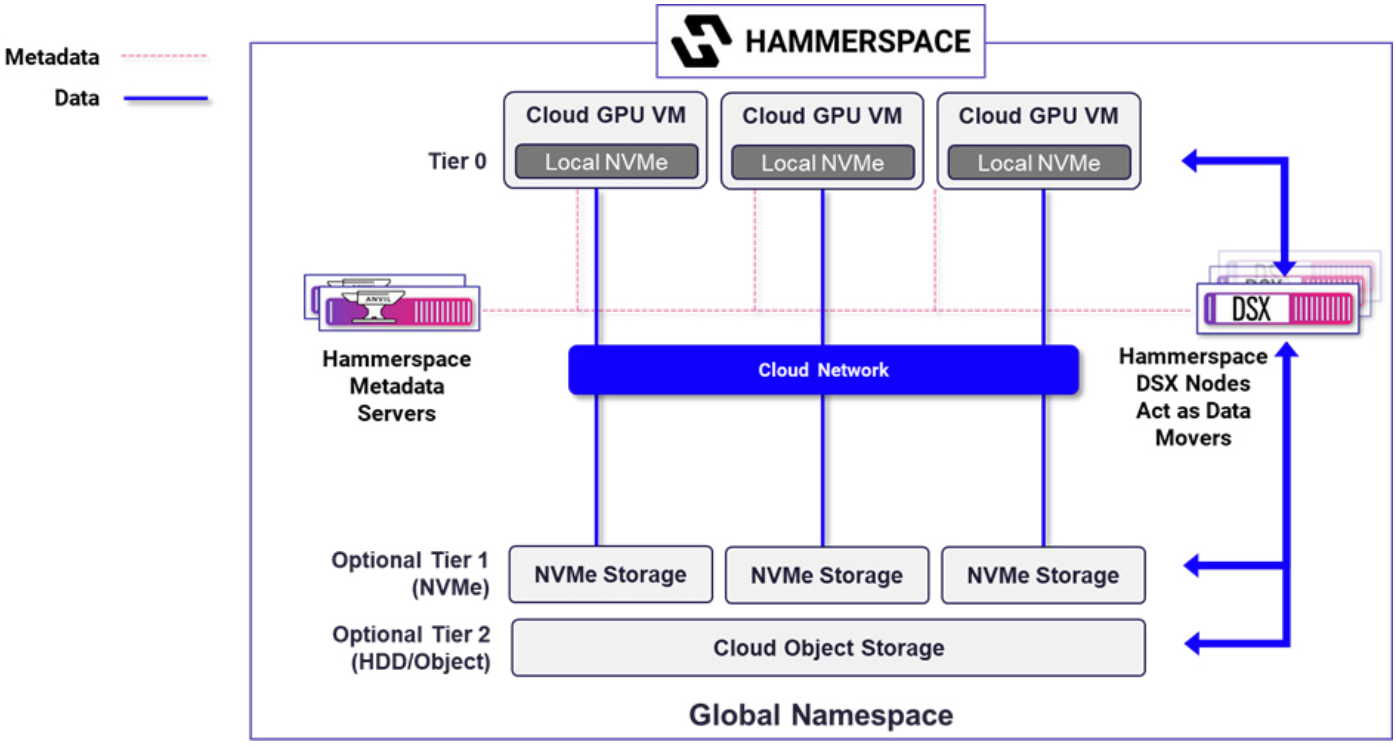
Hammerspace makes it easy to migrate on-prem data to the cloud, making hybrid-cloud and multi-cloud architectures a reality.

Hammerspace → Unifies on-prem and cloud environments into a single global namespace, automates the movement of data within that environment to bring your data to the compute that needs it, and delivers it at high speed.

Hammerspace Tier 0 Architecture → Maximizes GPU/CPU utilization in the cloud by uenabling the use of instance-local NVMe storage as high-performance shared storage.

Cloud Architecture for Hammerspace Tier 0

An example cloud architecture for Hammerspace is shown in the figure below:



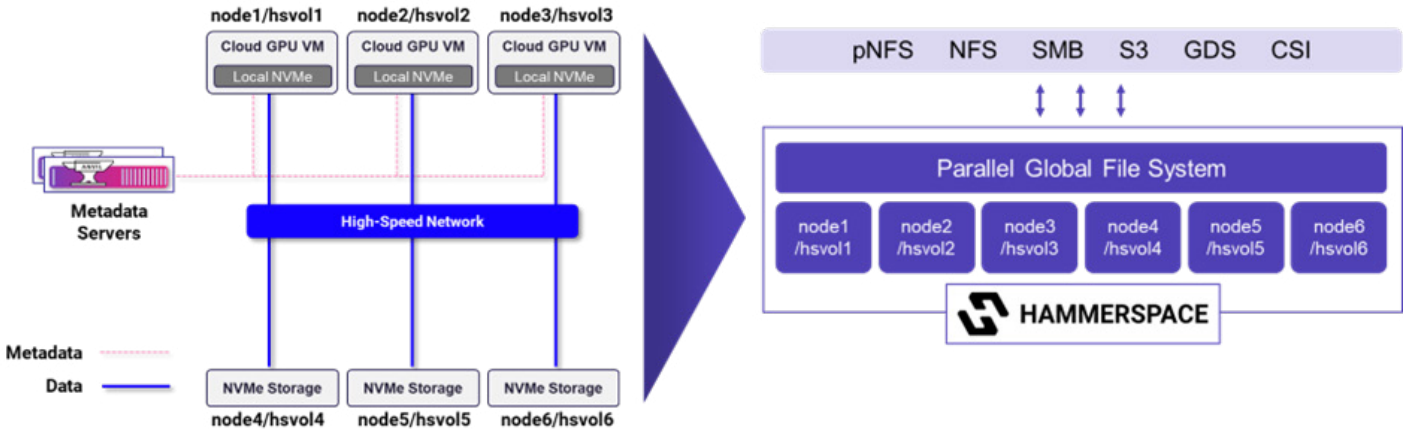
A cloud-based Tier 0 deployment includes:

- **GPU/CPU compute nodes (Tier 0 nodes)** with internal NVMe used as shared storage. Tier 0 nodes participate as both compute and storage, making full use of instance-local NVMe.
- **Hammerspace Metadata Servers ("Anvils")** for metadata management, orchestration, and share exports
- **Hammerspace Data Services ("DSX")** Nodes for data movement, both file-to-file, and file-to-object.

Hammerspace also provides for optional use of external 'tier 1' and 'tier 2' cloud storage:

- **Cloud object storage (S3, Blob, GCS)** for persistent capacity tiers
- **Optional Linux Storage Servers (LSS)** for additional storage or mover services

Storage systems and volumes added to Hammerspace, for example the local NVMe storage volumes in the GPU instances, are added to the Hammerspace file system and presented using standard file and object protocols. In this way, data across the Tier 0 nodes can be protected, orchestrated, and tiered to external storage based on administrator-defined policies.



Cloud Deployment Playbooks & Automation

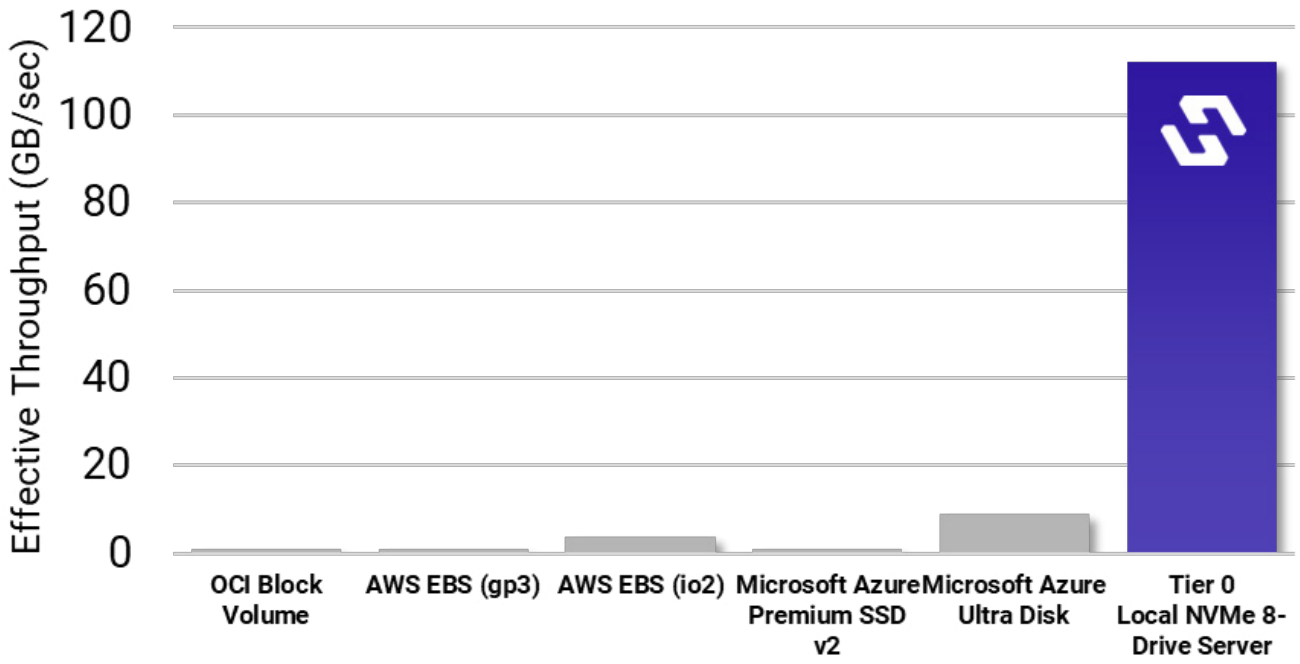
Hammerspace cloud deployment is fast and repeatable:

- Use **Ansible playbooks or REST APIs** for automation
- Integrate with **cloud-native deployment templates** (Terraform, CloudFormation)
- **Configure Tier 0 exports, RAID, objectives, and shares** via GUI, CLI, or script
- Hammerspace supports validation tools (HSTK), Grafana dashboards, and robust observability through Prometheus.

Performance Benefits of a Tier 0 Architecture in the Cloud

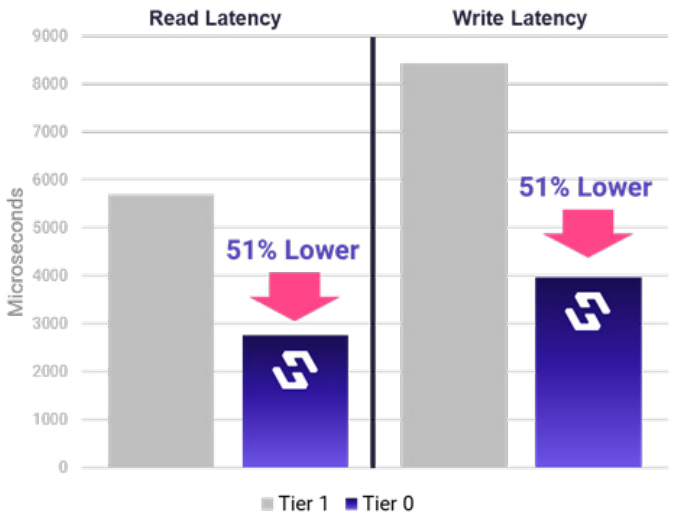
As discussed above, a Hammerspace Tier 0 architecture maximizes cloud GPU utilization by enabling the use of local NVMe storage as high-performance shared storage, and bypassing the north/south networks used by external storage.

The chart below (based on data from this white paper) compares the effective throughput of a GPU cloud VM with 8 local NVMe drives to the effective throughput of common NVMe storage VMs from some of the leading clouds.



Instance-Local NVMe Storage delivers 10x-20x the throughput of external cloud flash storage.

In order to further characterize the performance benefits of using a Hammerspace Tier 0 architecture in the cloud, Hammerspace has also conducted a series of tests in OCI comparing read and write bandwidth and read and write latencies between Hammerspace Tier 0 storage (local NVMe storage in the OCI GPU shapes) and Tier 1 storage (external networked NVMe storage in OCI).



The results of the testing are summarized in the charts above and show that Hammerspace Tier 0 delivered:

- **2.5×** faster read bandwidth
- **2×** higher write throughput
- **51%** lower latency

compared to external networked storage running on OCI.

Getting Data into Hammerspace in the Cloud

There are essentially three ways to get data into Hammerspace:

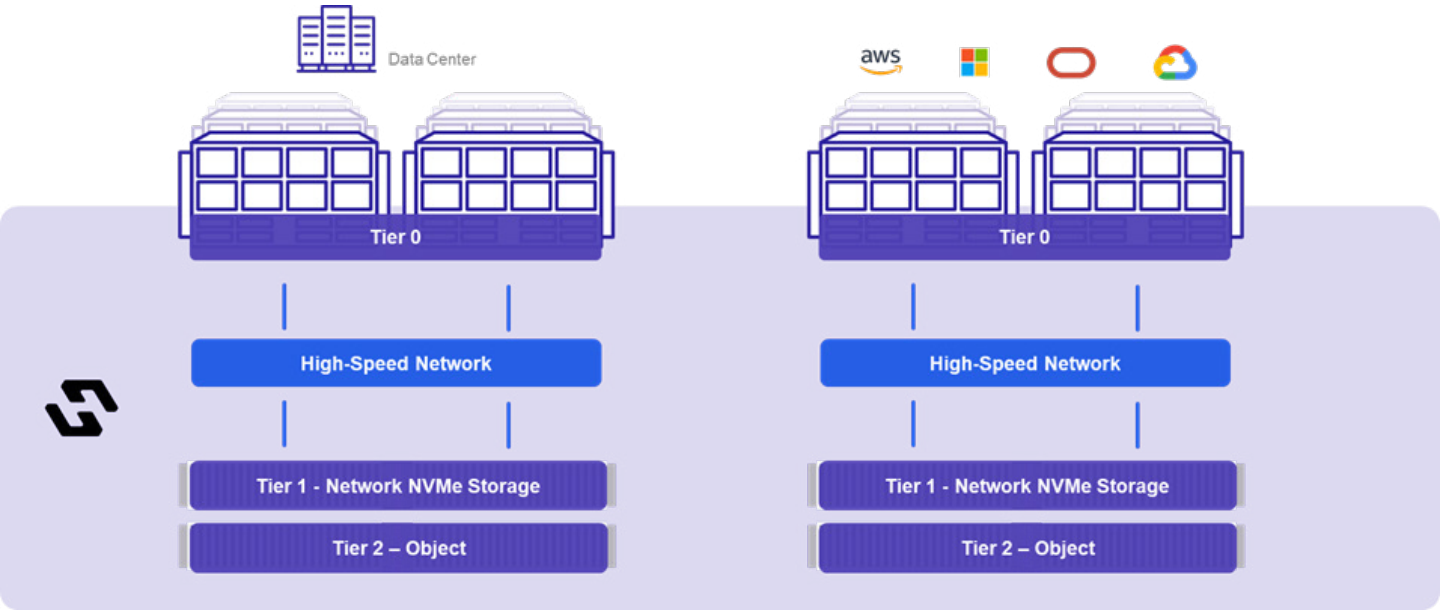
- **Copy in** - Any data copy tool that can write to an NFS mount could be used to bring existing data into Hammerspace.
- **Create new** - Workloads produce some kind of output or result. The Tier 0 architecture is designed to store that output.
- **Assimilation** - For existing data currently on other storage platforms, Hammerspace provides a way to “import” the metadata and hook a filesystem into a Hammerspace share, initially without moving the actual data. Objectives can be applied to the share into which the data is assimilated to move instances of the data onto Tier 0 based on any metadata criteria such as time stamps, filename or extension, or path.

In addition, when deployed as a Global File System, Hammerspace data orchestration policies can be used to automate the movement of data between on-prem environments and cloud environments.

Hammerspace Simplifies Hybrid-Cloud Architectures

Hammerspace Tier 0 is all about maximizing GPU utilization in the cloud. But Tier 0 is only a small part of what Hammerspace can do -

Organizations can join multiple Hammerspace clusters together as a hybrid-cloud global file system. This creates a unified Global Namespace that spans multiple sites, multiple cloud regions, and even multiple cloud providers, and allows all users, applications, and GPU clusters to operate on a consistent, global dataset via standard POSIX file and S3 object protocols.



By providing a single source of truth for all unstructured data, Hammerspace enables a new class of global applications. It eliminates the AI data bottleneck and maximizes GPU utilization, while making hybrid-cloud and multi-cloud architectures a reality. The result is accelerated innovation, improved operational efficiency, and a unified data strategy fit for the modern global enterprise.

Appendix 1 – Common Cloud GPU Instance Types and Local NVMe Options

Cloud Provider	GPU Instance Type	Local NVMe Storage?
AWS (EC2)	P3 (Tesla V100)	Selectable
	P4d (A100)	Included by default
	G4 (T4), G5 (A10G)	Included by default
Azure	CPU - L family GPU – ND-H100-v5, ND-H200-v5, ND-MI300X-V5	Varies
OCI	**CPU Bare-metal shapes** BM.DenseIO.E4.128 BM.DenseIO.E5.128 **GPU Bare Metal Shapes** BM.GPU4.8 BM.GPU.A10.4 BM.GPU.A100-v2.8 BM.GPU.MI300X.8 BM.GPU.H100.8 BM.GPU.H200.8 BM.GPU.B200.8 BM.GPU.GB200.4 **CPU VM Shapes** VM.DenseIO.E4.Flex VM.DenseIO.E5.Flex	Included by default
GCP	CPU – N2 family, C2 Family, A2 Family, Z3 family GPU – A2 family, A4 family	Varies