The S3 Object Store That Never Makes GPUs Wait

The only S3 object store that's also your fast file system. No gateway. No copy. No second system to run.

Challenges

- GPU idle time from storage bottlenecks costs providers and enterprises revenue.
- Separate file and object tiers double infrastructure spend
- Slow object storage under GPU cluster concurrency raises cost per token.

Solution

NeuralMesh™ delivers 7.5M ListObject ops/sec, 5,000 concurrent connections per node, and 10.2 TB/s per rack — with S3 and POSIX running natively on the same data.

Benefits

- GPU cycles bill instead of wait; the storage layer stops being the constraint.
- Training, fine-tuning, and inference run from one data store with no copies between stages.
- Infrastructure gets more resilient and more cost-efficient as it grows, not more fragile.

Object storage now sits in the middle of the AI pipeline, not at the edge. Training reads millions of small files in unpredictable, metadata-heavy lookups. Checkpointing writes multi-terabyte sequential dumps every few minutes, often from thousands of GPUs at once. Inference reads under strict latency budgets, with thousands of concurrent sessions hitting the same objects at once. Most S3 object stores were built for one of those patterns, not all three. NeuralMesh was built for this traffic profile. Not adapted for it.

NeuralMesh Fast S3 Object Storage runs on the same distributed architecture as every other NeuralMesh protocol, with no centralized metadata and no single point of failure to break under load. S3 and POSIX operate on the same physical data, so training, fine-tuning, and inference share one deployment with no copies between stages. On top of that foundation, a full set of S3-specific engineering covers performance, data management, security, and multitenancy. On its own, this architecture already removes the ceilings that stall conventional S3 object stores. Paired with the new, next-generation WEKApod systems announced today, it reaches rack-scale numbers no general-purpose hardware platform can match.

Why Conventional Object Stores Can't Keep Up

Most S3 object stores concentrate metadata in a small number of controller nodes and manage client connections through a shared pool, both sized for sequential access by a modest

number of clients. GPU clusters running hundreds of simultaneous training and inference jobs blow past that design point. Once the connection ceiling hits, typically around 1,000 connections per node, request queuing cascades through every job on the cluster, not just the one that caused it.

Performance that holds at moderate load doesn't mean it holds at production scale. Conventional object stores degrade unpredictably once GPU cluster concurrency exceeds what the architecture was designed for, making SLA commitments to tenants unreliable exactly when they matter most. Cost and density follow the same pattern: capacity and performance per rack unit stop scaling once concurrency exceeds the architecture's design point.

NeuralMesh's Architecture Advantage for S3

NeuralMesh removes each of those ceilings at the architecture level, not by adding hardware on top of a design that still breaks under load.

Capability	Why it matters for S3
:--	:--
Distributed metadata and data across every node, no centralized controller.	No node becomes a bottleneck. Performance scales linearly from six nodes to hundreds.
Independent failure domains, no single point of failure in software.	A single node event stays a single node event. Resilience compounds as the cluster grows.
AlloyFlash mixes TLC and QLC NVMe, steering writes by I/O type.	QLC density without the QLC performance penalty.
Always-on data reduction.	Up to 6x capacity savings on AI training data, write overhead under 5%.

One Platform, No Copies Between Stages

A conventional AI pipeline, ingest, ETL, train, validate, infer, archive, copies the full dataset at multiple stage boundaries before a model ships, sometimes two or three, sometimes five or more, depending on the pipeline. Every copy is a sequencing dependency the next stage has to wait out. It's what happens when an S3 object store and a file system have no shared foundation.

NeuralMesh Fast S3 Object Store runs S3 and POSIX on the same physical data blocks, so that requirement doesn't exist. New files are immediately readable through S3, with no gateway and no translation layer between them. Training, fine-tuning, and inference all read and write against the same S3 deployment. However many copies a pipeline needed before, it needs one now.

Governance follows the same path. S3 lifecycle rules, IAM policies, encryption, and audit logging set once apply automatically across POSIX, NFS, and SMB. One compliance posture covers every protocol instead of one per protocol, which for organizations with data sovereignty requirements means every pipeline stage can run on-premises without giving anything up to get there.

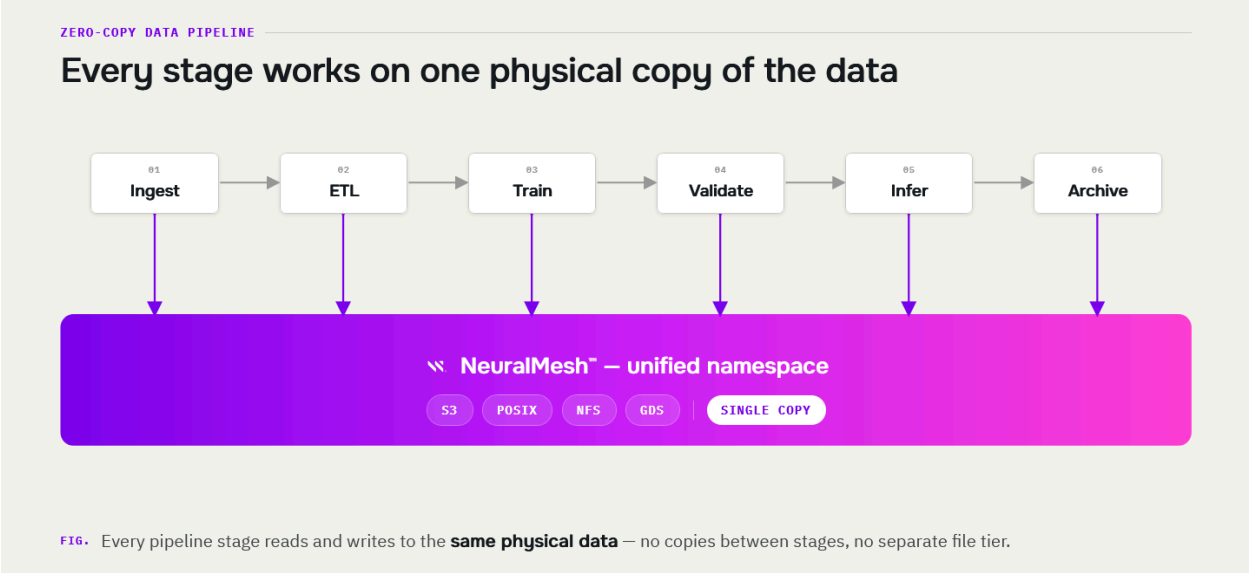

FIG. Every pipeline stage reads and writes to the **same physical data** — no copies between stages, no separate file tier.

What's Built Specifically for S3

S3 Performance Bucket. S3 over RDMA moves S3 data directly into GPU memory over RDMA instead of through TCP and a shared connection pool, cutting CPU overhead and latency out of every S3 read and write. For inference operators serving weight loads under concurrent sessions, that shows up as time-to-first-token. For training and checkpointing, it's the difference in how fast a multi-terabyte dataset or checkpoint moves without stealing CPU cycles from the job. The same engineering also speeds up catalog scans and high-concurrency writes, the two other operations that saturate first under AI load. Enumeration alone runs about 4.5x faster than the nearest published alternative on large catalogs, and the full set runs 5 to 6x faster than standard S3 configuration on AI workload patterns.

S3 Data Management. Conventional S3 object stores make pipelines poll for changes, and enforce lifecycle rules that don't survive a protocol switch. Event notifications push to Kafka the moment an

object changes, so preprocessing workers and indexers react in real time instead of waiting on a poll cycle. Lifecycle rules, versioning, and delete protection apply the same way no matter which protocol touched the object, so a training dataset or checkpoint stays governed and recoverable across the whole pipeline.

S3 Security and Identity. This closes the permission gap conventional S3 gateways can't: access set once, at the POSIX layer or in S3 IAM policies, holds no matter which protocol touches the data next. Temporary, zero-trust credentials and existing LDAP identities work directly against S3, so security teams don't need a separate identity system just for the object tier. Every S3 API call streams as a structured event to SIEM platforms for audit.

S3 Multitenancy and Operations. Slow S3 tenant provisioning delays revenue directly for AI cloud providers scaling their tenant base. New tenants come online in minutes through standard Kubernetes manifests, the same workflow already used for compute, and run alongside each other under Multitenancy 2.0, whether that's one large anchor tenant on dedicated infrastructure or dozens of smaller ones sharing it. Observe puts request rates, latency, and per-tenant consumption in one dashboard, so managing hundreds of tenants doesn't require a team that grows with them.

See NeuralMesh Fast S3 Object Storage in action. Request a benchmark walkthrough with a NeuralMesh specialist at weka.io/contact.