WarpDrive: A Composable Storage Substrate for Disaggregated Data Systems

Tejus Chandrashekar
Independent Researcher
tchandrashekar@acm.org

ABSTRACT

Object stores have become the de facto storage layer for disaggregated systems, yet they expose only a narrow interface (PUT, GET, DELETE) to workloads with fundamentally different requirements. Recent work has proposed increasingly sophisticated mechanisms for mitigating latency, cost, and consistency overheads above the object API [1, 2]; we argue these concerns belong inside the storage layer. WarpDrive is a modular, S3-compatible object store that exposes them as tunable primitives. Our contributions:

(1) Surveying the client-side mechanisms built over the years to reduce object-storage cost and latency, we observed that a small set of storage requirements emerged across otherwise unrelated systems. To explore whether this pattern held more broadly, we examined three representative disaggregated workloads, each representing a distinct class of disaggregated infrastructure: distributed AI training, databases, and agentic execution frameworks (IBM Vela [3], Neon [4], and LangGraph [5]). The four needs are I/O batching, multi-object consistency, liveness-aware reclamation, and co-residency control. In practice each system rebuilds its own solution independently. Vela's checkpoints need atomic cross-shard visibility for FSDP[1] barriers. LangGraph's superstep handoffs need *blocking versioned reads* (reads that block until a target version is visible) to avoid polling.
(2) We introduce WarpDrive [6], a modular, S3-compatible object store built around the principle that the storage layer should understand workload access patterns, and applications should understand storage primitives; this mutual visibility is what allows both sides to reduce cost simultaneously rather than compensating for each other's opacity.
(3) We develop a codebase-grounded analytical model of Neon's open-source pageserver as a representative instance of a recurring pattern: delta-chain reconstruction above flat object storage. Recent work measures this trap empirically and responds by relocating replay compute to idle instances, leaving the object layout untouched [7]. We instead explore the object layout itself as a complementary optimization lever. This pattern extends beyond Neon to storage-disaggregated database engines more broadly, with Aurora and other disaggregated PostgreSQL systems exhibiting structurally similar page-version reconstruction. Our model reveals a structural tradeoff: existing layouts force a tradeoff between fragmented reconstruction (k independent GETs per cold read, compounding across tenants) and bundled reconstruction (an $O(k)$ read-modify-write on every append, because S3 objects are immutable). WarpDrive avoids this tradeoff by storing independently addressable deltas in mutable co-located slabs and retrieving them through a single batched I/O.
(4) We are currently evaluating the analytical predictions using WarpDrive and a flat object-store baseline across increasing delta-chain lengths. Preliminary results will be presented at the workshop, assessing the impact of storage-workload co-design on operational cost and application complexity.

Across databases, AI training systems, and agentic runtimes, we repeatedly observe the same storage requirements emerging above a flat object interface. WarpDrive packages these requirements as reusable storage primitives, reducing the need for custom application-layer workarounds. Its four primitives are a first proof point for storage-workload co-design below the query layer, and we bring them to the community as a concrete starting point for composability.

VLDB Workshop Reference Format:
Tejus Chandrashekar. WarpDrive: A Composable Storage Substrate for Disaggregated Data Systems. VLDB 2026 Workshop: Fourth International Workshop on Composable Data Management Systems.

VLDB Workshop Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/vitality-ai/warpdrive.

REFERENCES

[1] Or Ozeri, Effi Ofer, and Ronen Kat. Object storage for deep learning frameworks. In *Second Workshop on Distributed Infrastructures for Deep Learning (DIDL '18)*. ACM, 2018.
[2] Dominik Durner and Viktor Leis. Exploiting cloud object storage for high-performance analytics. In *Proceedings of the VLDB Endowment*, volume 16, 2023.
[3] Brian Belgodere, Pierre Dognin, et al. The infrastructure powering IBM's gen AI model development. https://arxiv.org/abs/2407.05467, 2024.
[4] Jack Vanlightly. Neon serverless PostgreSQL — ASDS chapter 3. https://jack-vanlightly.com/analyses/2023/11/15, 2023.
[5] LangChain. LangGraph persistence documentation. https://langchain-ai.github.io/langgraph/concepts/persistence/, 2024.
[6] Tejus Chandrashekar. WarpDrive: Workload-aware object store. https://github.com/vitality-ai/warpdrive, 2025.
[7] Xi Pang and Jianguo Wang. Reducing tail latency in storage-disaggregated database systems. *Proc. ACM Manag. Data*, 4(1), 2026.

[1] Fully Sharded Data Parallel.

Proceedings of the VLDB Endowment. ISSN 2150-8097.