

Project Gismo

Global I/O-free Shared Memory Objects

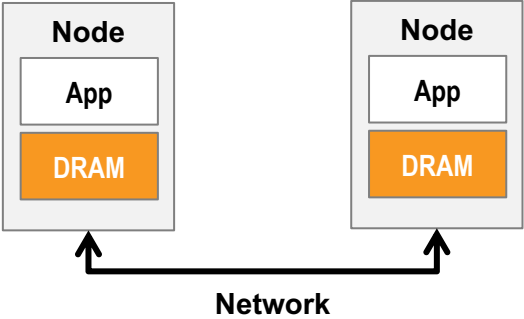


Yong Tian

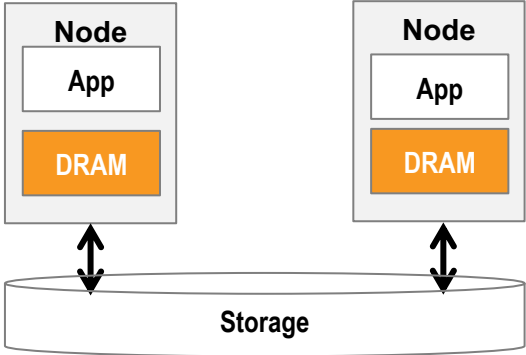
Field CTO, MemVerge



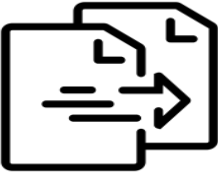
The I/O Wall in Distributed Applications



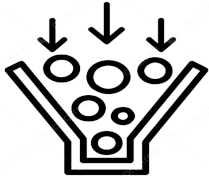
Serialization & Deserialization



Data Copies

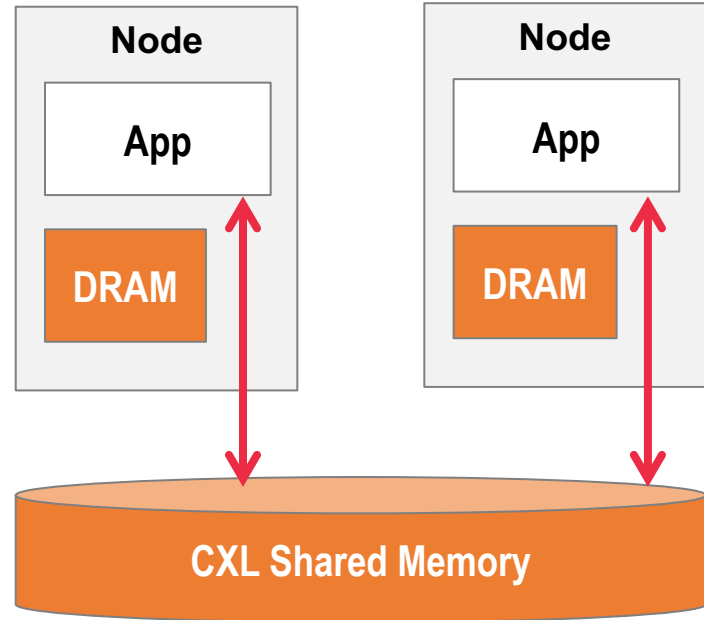


Media Speed



CXL Makes Cross-node Shared Memory Possible!

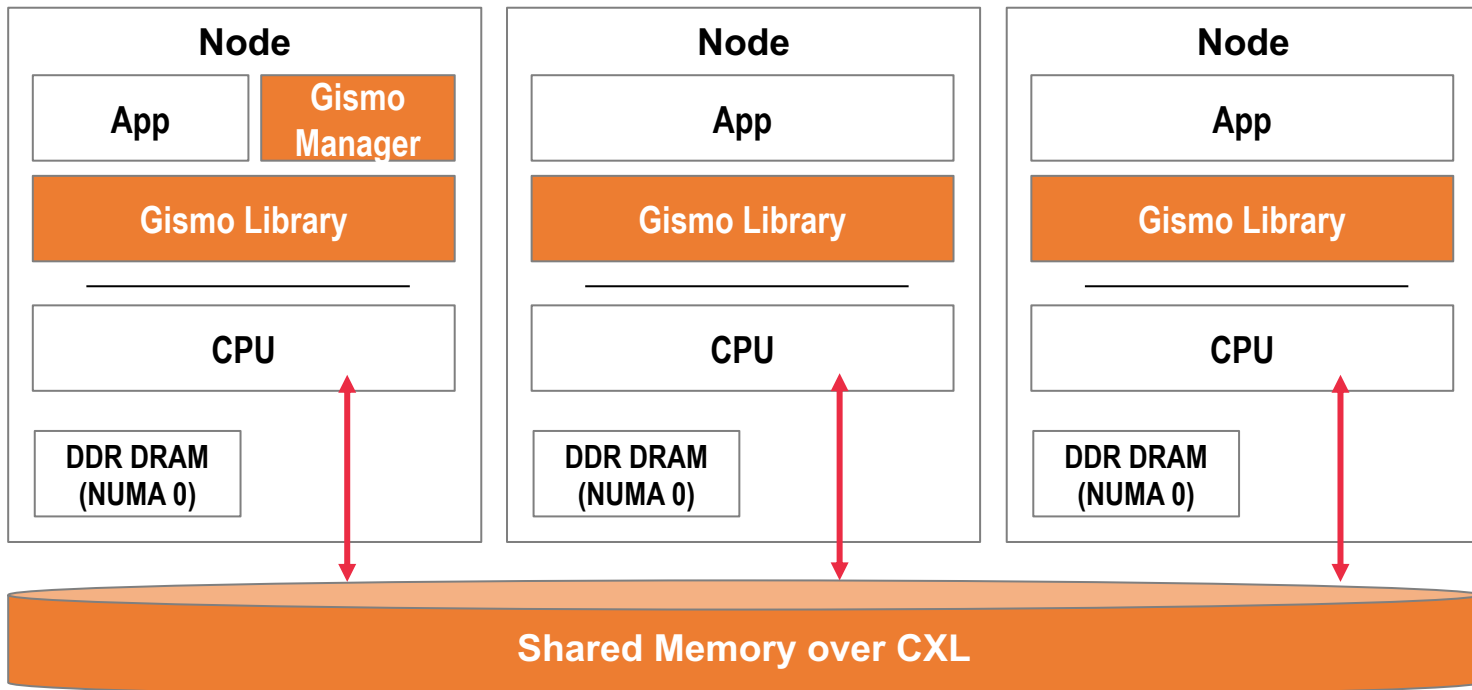
- Multiple nodes have direct memory access to the same CXL memory region
- Cache Coherence is part of CXL 3.0 hardware specification
- Software cache coherence can be implemented on top of CXL 2.0 hardware





Introducing Project Gismo

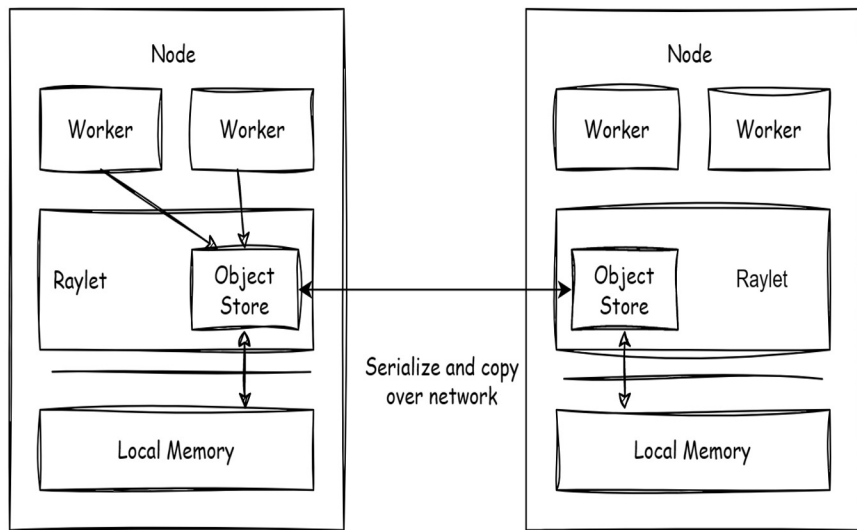
Global I/O-free Shared Memory Objects



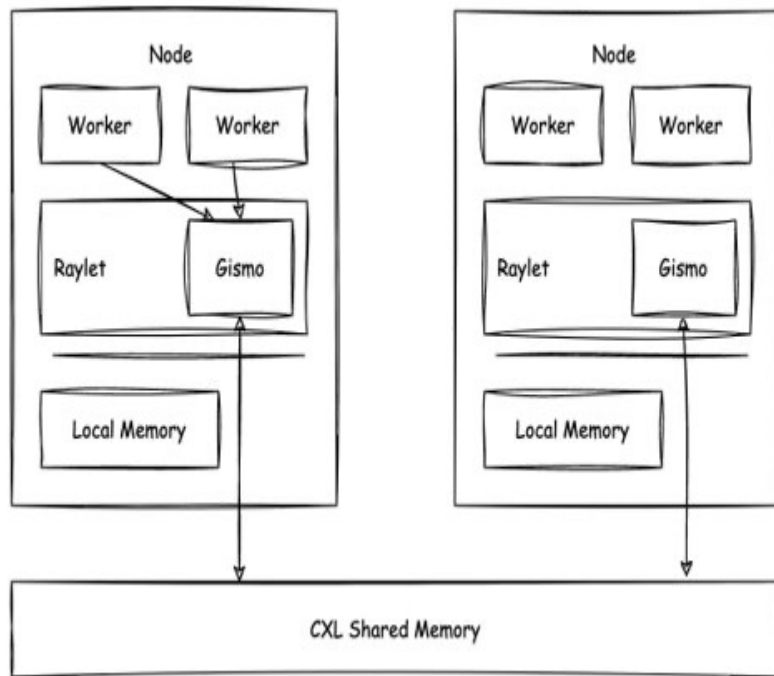
Use Case 1: AI/ML



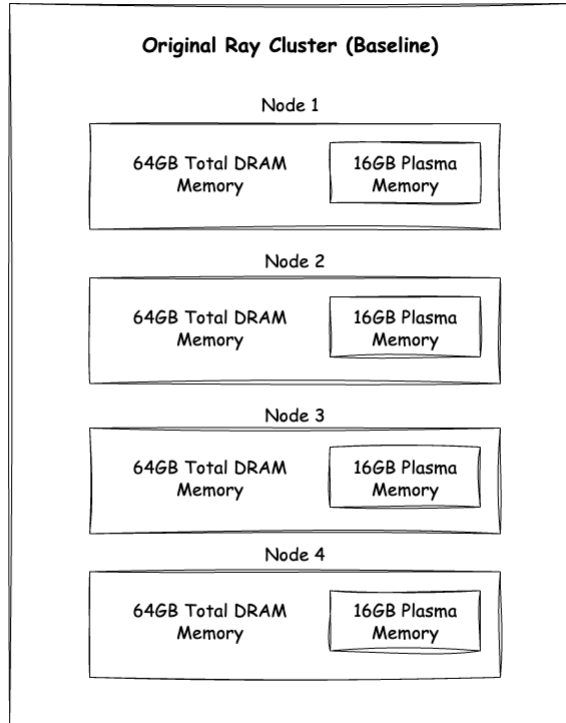
Baseline Ray



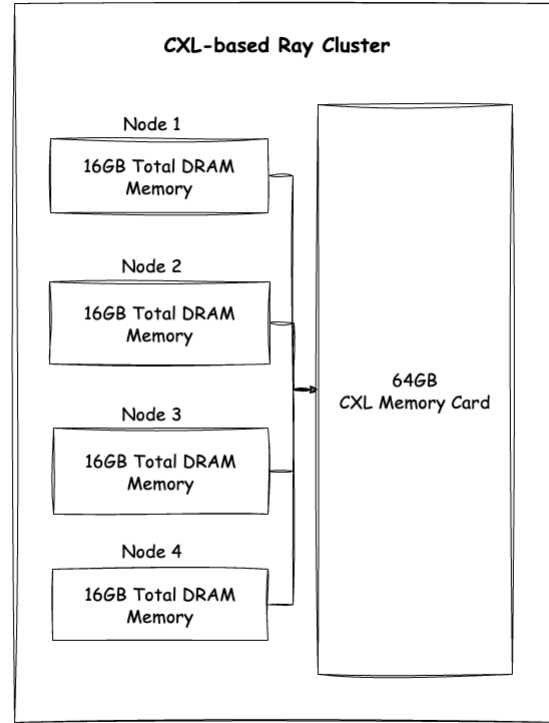
Ray+Gismo



Demo



**Benchmark
setup**



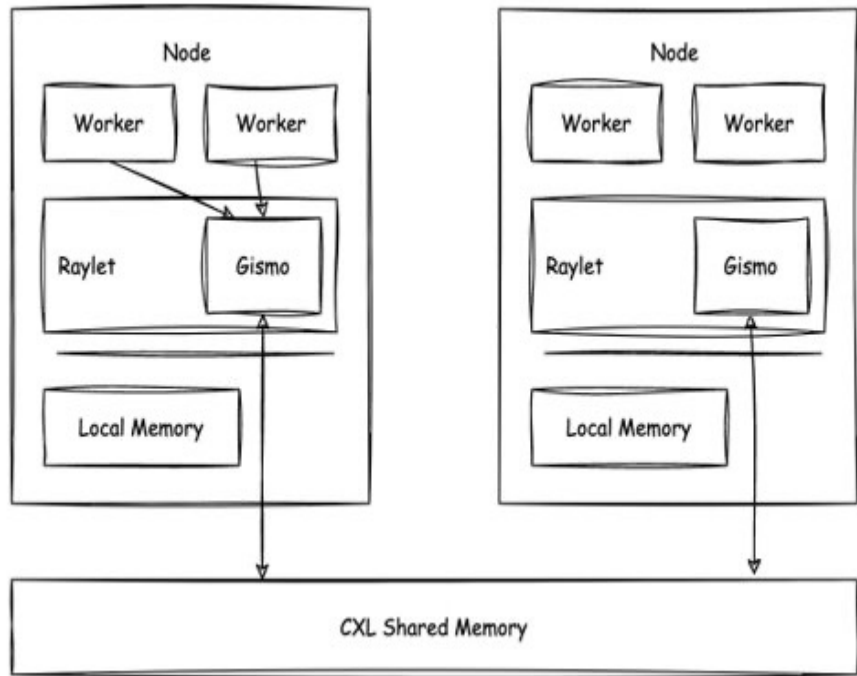
Shuffle Benchmark Results

	Baseline Ray	Ray + Gismo
Local Get 1GB object	0.4 sec	0.4 sec CXL shared memory as fast as local memory
Remote Get 1GB object	2.7 sec	0.4 sec 675% faster
Shuffle 50GB 4 nodes, each 4 cores, 128 GB object store	515 sec	185 sec 280% faster

* Running in emulation environment

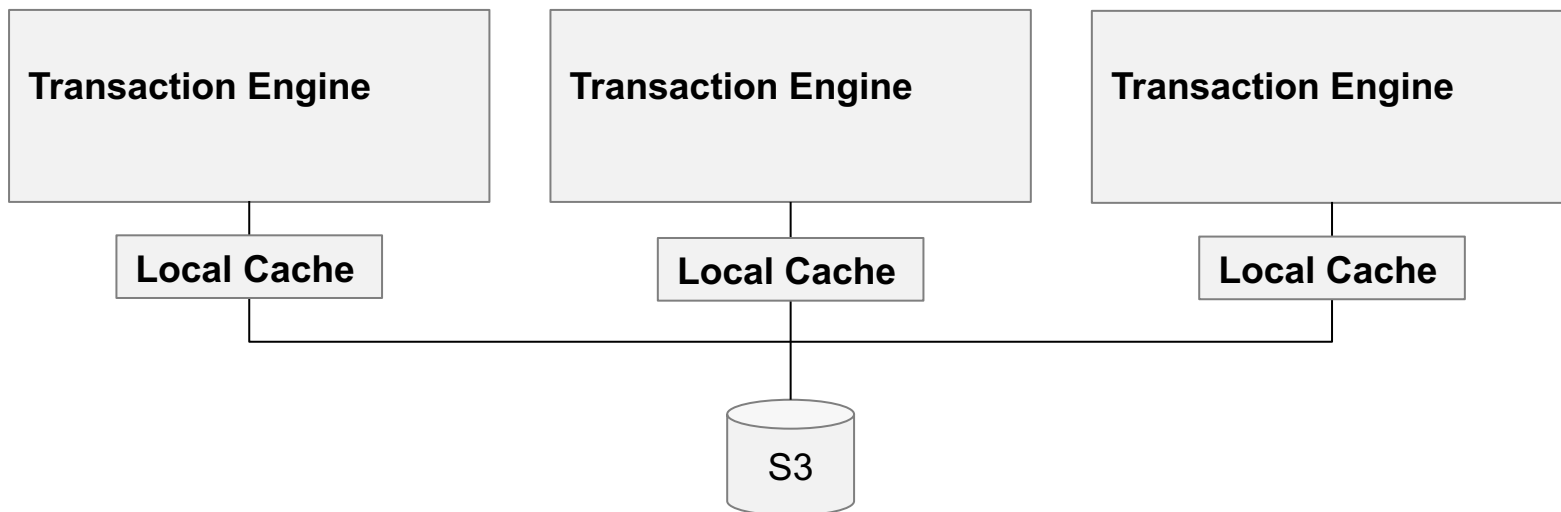
Benefits of Ray+Gismo

- **IO-free:** No more object serialization and transfer over network for remote object access
- **Zero Copy:** No more duplicate object copies on different nodes
- **No Spilling:** Reduce object spilling and data skewing because each node has access to the whole memory pool



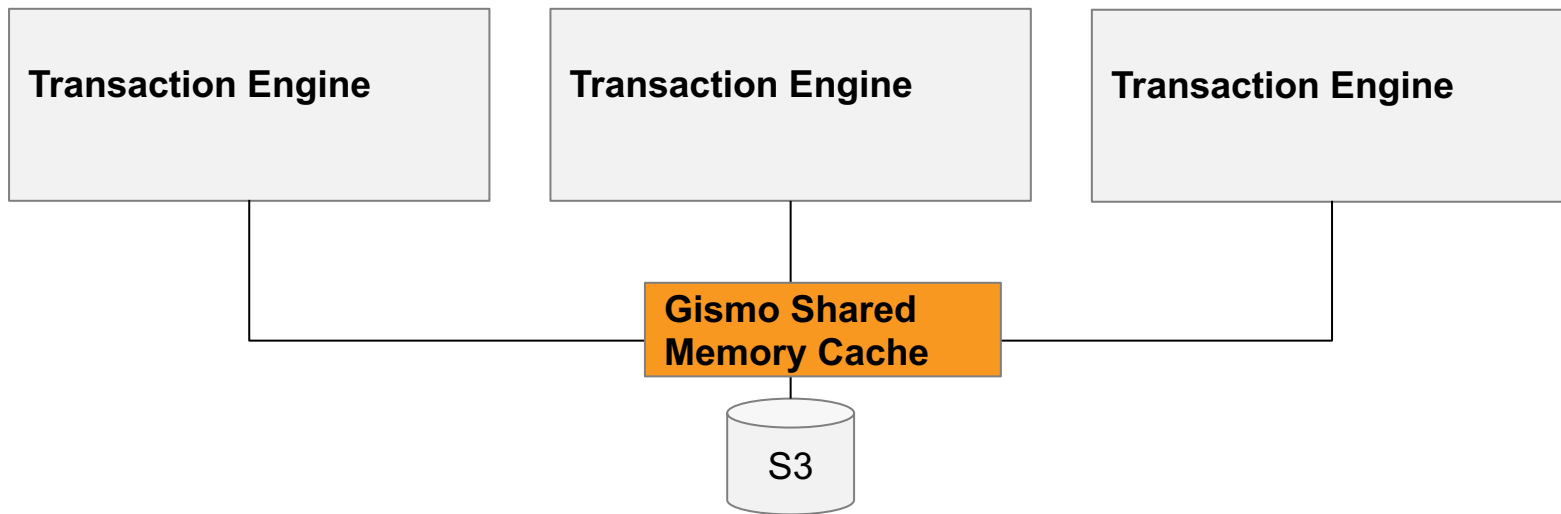
Use Case 2: Memory Cache for HTAP Database

- MatrixOrigin HTAP database features a scale-out architecture
- Local cache can have high cache miss rate
- Cache coherence is complicated.



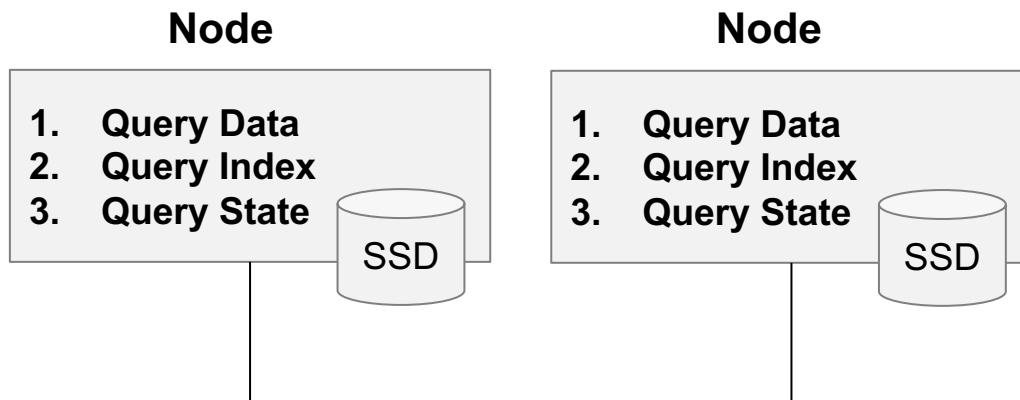
Gismo Enables Shared Memory Cache

- Higher cache hit rate
- Less cache copies
- No cache coherency handling



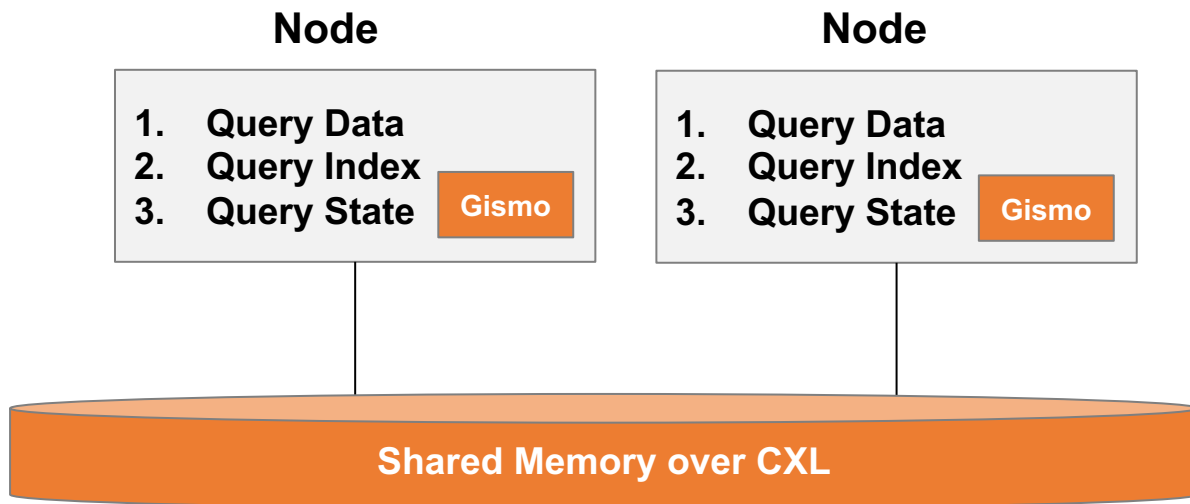
Use Case 3: Database Fail-over

- Timeplus distributed streaming analytics used for real-time trade monitoring
- Synchronization of data for fail-over recovery is time consuming



Faster Database Failover with Gismo

- Using Gismo-based checkpoint and restore
- Synchronization of data and fail-over recovery will be significantly faster



Gismo API

Gismo_connect(): Connect to Gismo Manager

Gismo_disconnect(): Disconnect from Gismo Manager

Gismo_create(): Create an object

Gismo_seal(): Seal an object. Object becomes immutable after this call

Gismo_get(): get/read an object

Gismo_release(): Tell the Gismo manager that this host is done reading the object

Gismo_delete(): Delete an object

Gismo_subscribe(): Subscribe to notifications when objects are sealed in Gismo

CXL+MemVerge
Memory is the Network
Memory is the Storage
Welcome to Memory-Centric Computing





Contact gismo@memverge.com

- If you are a developer of distributed applications
- If you are in the field of AI/ML or distributed databases
- If you are looking for differentiated performance advantages for your applications

