Scaling CXL® to Millions of Servers Vistara for Hyperscale Efficiency

Neha Gholkar and Hasan Al Maruf (Meta, Inc.)

July 21, 2026

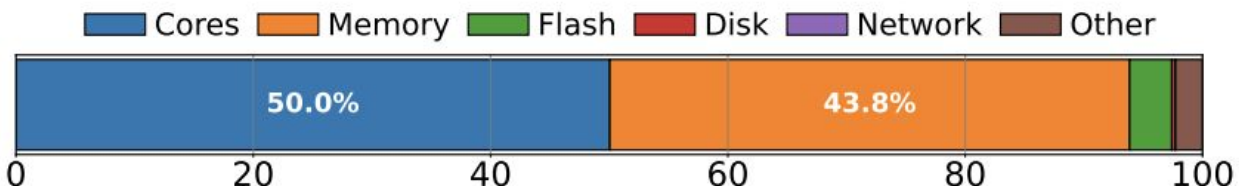
Memory is the Hyperscale Bottleneck

43%
of servers are memory-bound
not compute-bound

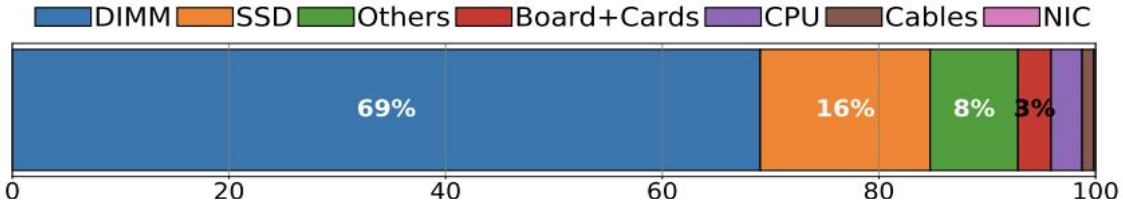
69%
of a server's embodied
carbon footprint is DRAM

10–14yr
DRAM lifetime
servers retire at 5–7 yr

Fraction of Servers bottlenecked by resource (%)



Emissions by Component (%)



CXL promised to address all three at once

Six Years In– Where is CXL at Scale?

6 years

since CXL was introduced

yet limited reports of at-scale deployment.

The CXL Promise

One standard to expand memory capacity and bandwidth across the datacenter.

The Reality

Off-the-shelf solutions didn't fit bill

Software readiness to make CXL useful

Why Broad Adoption Has Been Slow?

It's Architecture, Not the Protocol

CXL Memory Modules Bundled New DRAM

Required buying new DRAM chips with the controller– adds cost.

No DDR4 Support

Blocked reuse of decommissioned memory– killing the cost and carbon win.

Tiering is too Slow

A persistent myth: high tiering overheads and unstable tail latency rule out production.

Vistara, end-to-end CXL stack to turn CXL into production reality across millions of servers

What Fleet-Wide CXL Demands?

Cost-effective Expansion

Lower Carbon

Easy Adoption

Generic & Broad

Operational Efficiency & Low Maintenance Overheads

Vistara: one co-designed stack that meets all the demands

Vistara: End-to-End CXL Stack

Software Linux Memory Tiering

The complexity of CXL is invisible to applications. Keeping tooling and maintenance simple

Platform MemServer

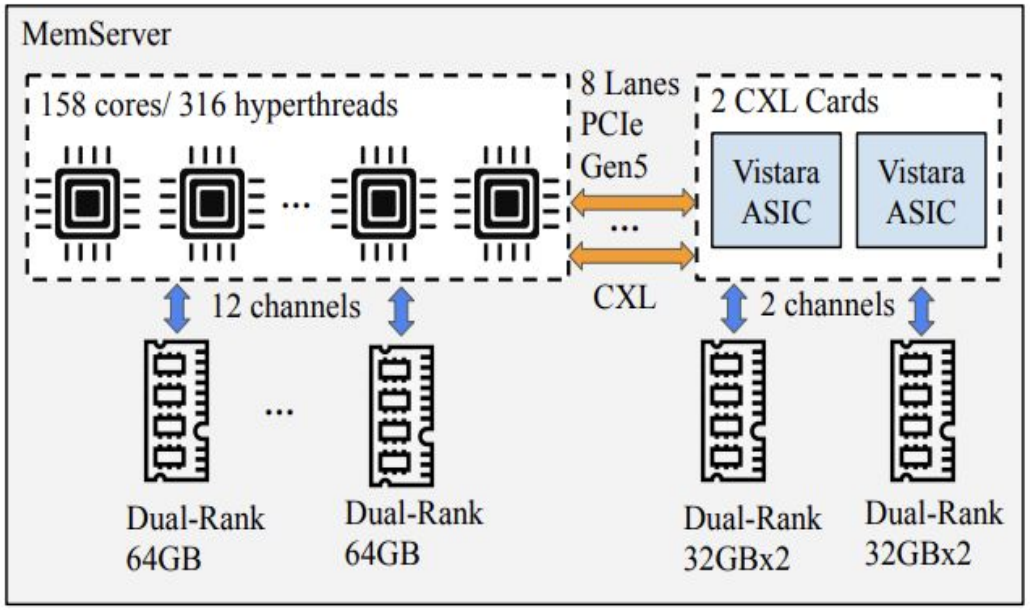
High memory-to-compute ratio for memory-bound services

ASIC Vistara

Power & cost efficient, performant expansion with DDR4 reuse

One stack, co-designed end-to-end, running in production today

MemServer: 1TB Server



Vistara ASIC (2x ASICs)

Host Connectivity	x16 PCIe Gen5 (x8 used in prod)
Memory Interface	2x DDR4 2DPC channels
Configuration	4x 32GB DDR4 DIMMs 2400MT/s

Production Operation Conditions	Local memory [60% BW util]	CXL Memory [20% BW util]
	234 ns	280 ns (+50ns)

1 TB
memory / server

3 : 1
local : CXL

250 ns
CXL Idle Latency

48 GB/s
Peak Effective CXL BW

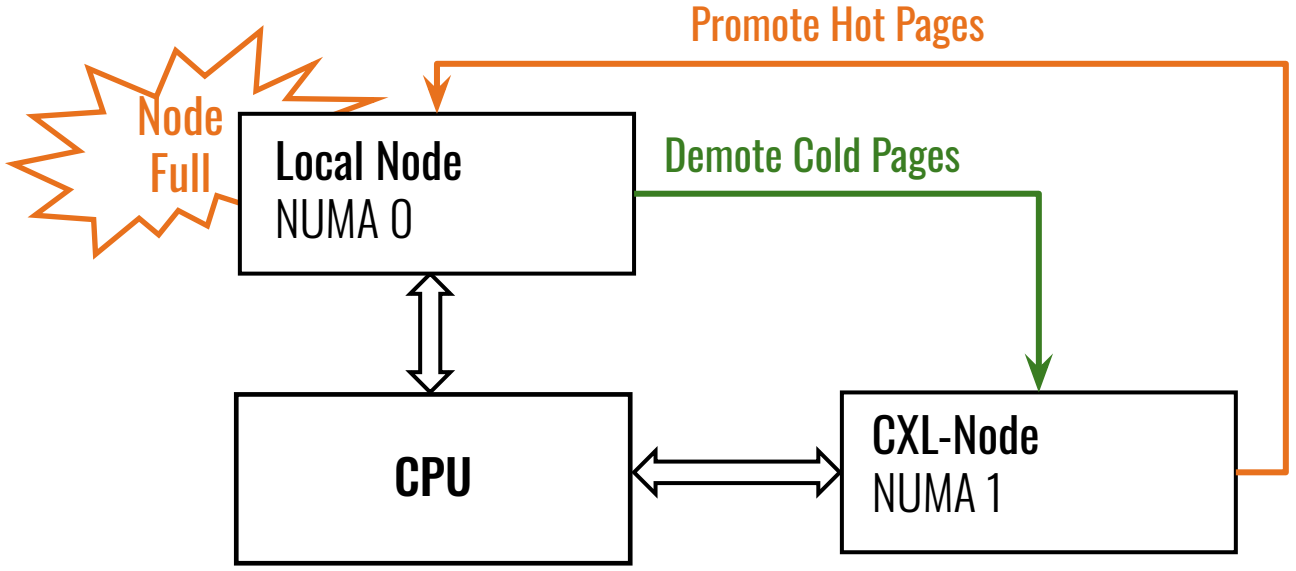
+50 W
for both ASICs+DIMMs

MemServer platform designed for memory constrained services with 3:1 local:CXL ratio

One Uniform Fleet– Opt Out As Needed

Transparent Tiering

CXL appears as CPU-less NUMA node
Onlined as ZONE_MOVABLE by Linux Driver
Secondary tier via HMAT / CEDT



TPP for Memory Tiering

OS allocates memory to local-first
When local is full TPP demotes cold pages to CXL
Trapped hot pages in CXL gets promoted back
Hot working set mostly in local; cold pages in CXL
Special tiering-aware memory allocation policies benefit specific workloads

Transparent memory tiering. Minimal to no application changes. One fungible fleet, Simple to run

One Uniform Fleet– Opt Out As Needed

TMO for Proactive Demotion

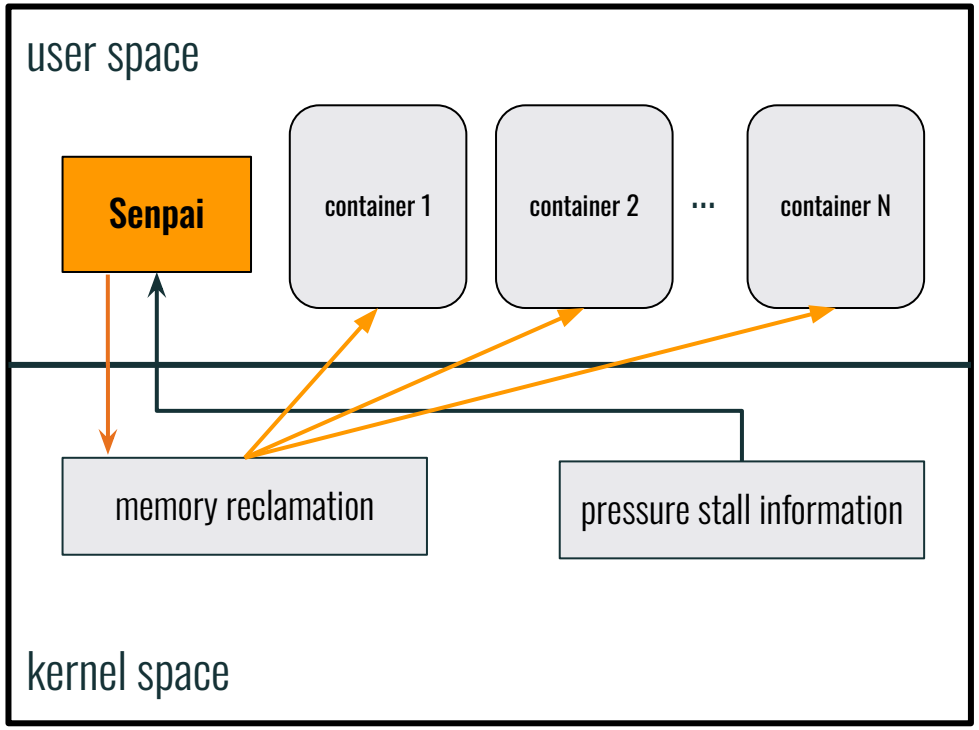
User-space tool senpai monitors PSI and triggers reclamation

Stalls due to lack of memory monitored in the kernel space

Impact on a container adjusts the reclamation rate dynamically

Reclamation Pipeline

Local memory gets demoted to CXL, CXL gets offloaded to storage

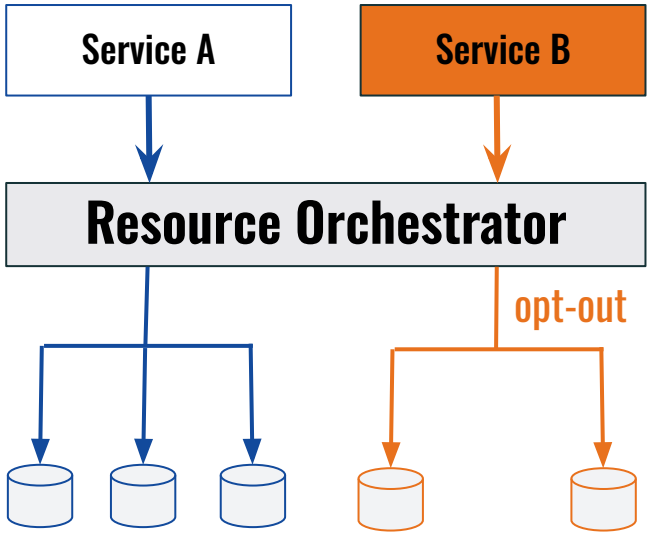


Transparent memory tiering. Minimal to no application changes. One fungible fleet, Simple to run

One Uniform Fleet– Opt Out As Needed

Uniform Fleet

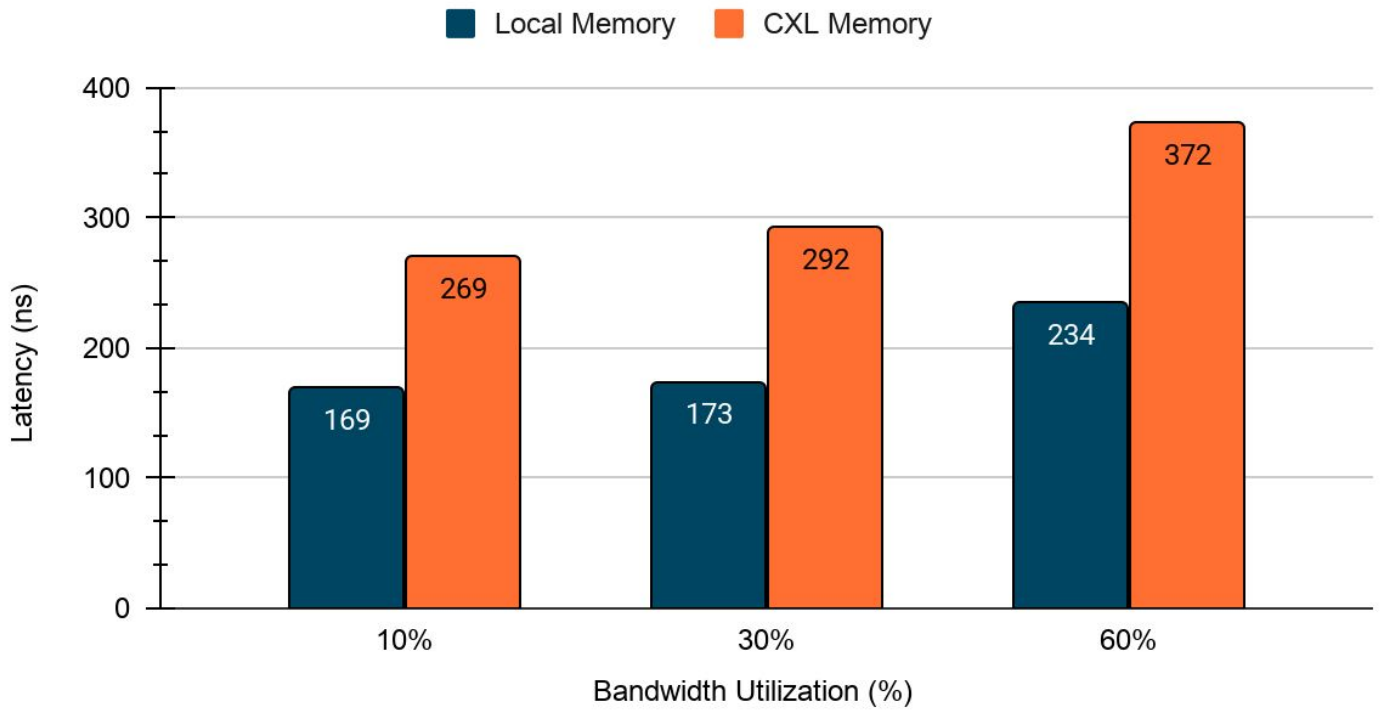
Uniform 1TB everywhere, 768GB local + 256GB CXL
Per-service opt-out is implemented at cgroup-level (cpuset.mems)
Service-level policies are specified at launch time
No BIOS changes or reboots needed
Metrics on resource usage, error rates, and device health, etc. are continuously reported to centralized monitoring systems



Transparent memory tiering. Minimal to no application changes. One fungible fleet, Simple to run

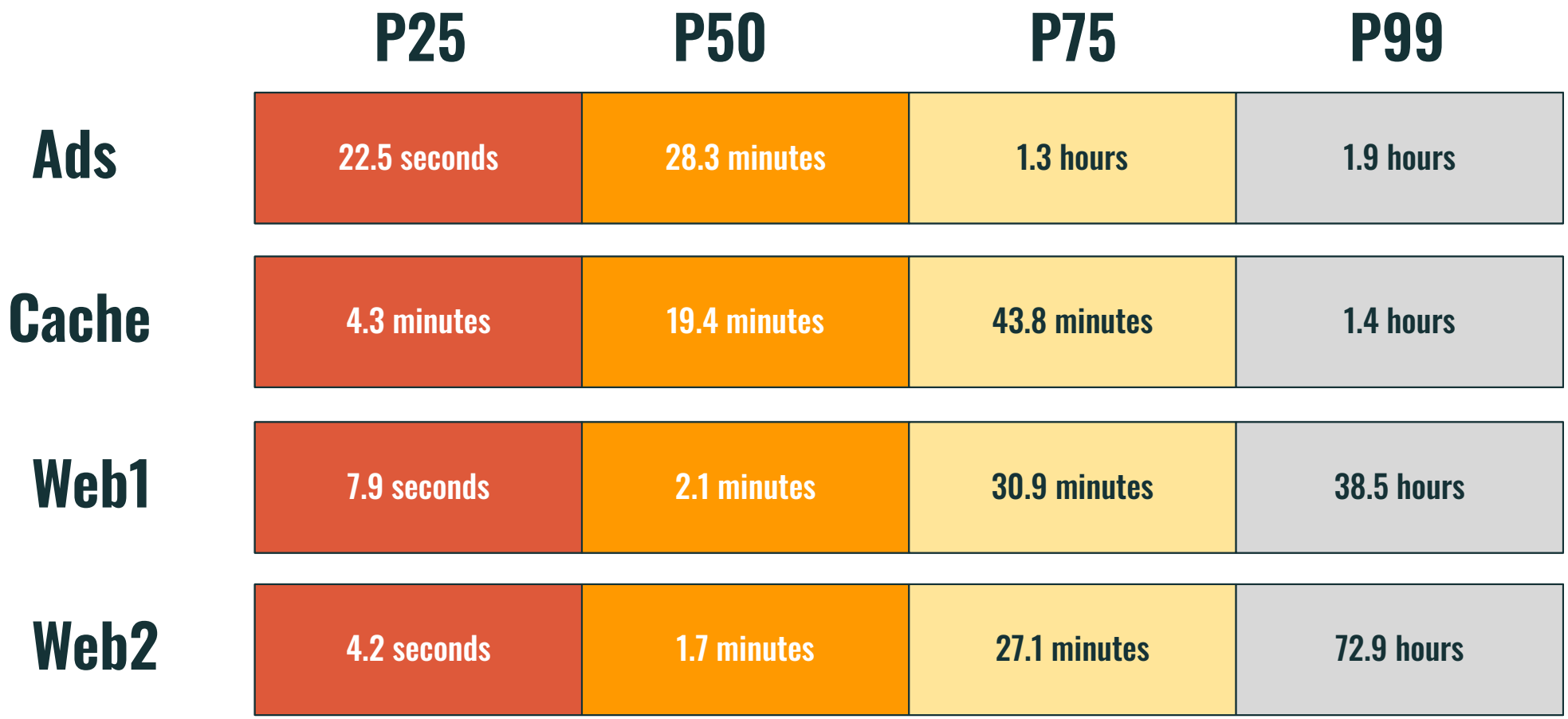
Local vs CXL Memory in Production

	Local	CXL
Capacity	768 GB	256 GB
BW	614 GB/s	58 GB/s
Power	132 W	30 W
Power/GB	1x	0.7x
Cost/GB	1x	0.13x



Practical trade-offs of deploying CXL memory shows its potential for cost-effective capacity expansion

Memory Idle Time in Production



When workloads have large cold memory (memory with high idle time), CXL tier's extra latency doesn't impact much

Already in Production

ML Parameter Server

Fewer servers, less fan-out, fresher models

-25%
fleet size

Cache

Bigger cache partitions cut backend traffic

-29%
avg. query processing time

+25%
throughput

-25%
fleet size

Development Infrastructure

More CI/CD jobs & VMs per server

-33%
fleet size

Data Warehouse

+33% executor packing, fewer OOMs

-10%
execution time

-30%
fleet size

Server count reduction & DRAM reuse cut cost and carbon footprint

Learnings from the Production

Stable CXL Latency

270–370 ns for steady production workloads; tail latency variance is inline local memory

Workload Impact

50–100 ns latency gap between local and CXL in production is imperceptible to the workloads

OS-Based Tiering

Simple OS-based tiering works; production design limits overheads under 0.5%

Hotness Detection

LRU-based hotness detection suffices to keep hot working set in local memory for a 3:1 setup

Vistara: Making CXL Real!

First at scale

The first reported CXL memory expansion deployed at scale.

Co-design wins

Custom ASIC + transparent OS tiering, built production-grade solution.

Broad impact

Delivers performance gains across a wide range of real-world workloads.

CXL Memory expansion with DRAM Reuse pays

High memory-to-compute ratio achieved with DDR4 reuse dramatically cuts cost and carbon footprint landing huge TCO wins!

Q&A

Thank You