UNLEASHING HETEROGENEOUS COMPUTE
LESSONS FROM REAL-WORLD SYSTEM DESIGN

PALLAVI SHARMA - IMAGINATION TECHNOLOGIES
ERIC NORIGE - BAYA SYSTEMS

ANDES RISC-V CON, APRIL 2025 | SAN JOSE,CA

IF YOU CAN'T FEED IT, YOU CAN'T USE IT

— a principle truer today than ever in heterogeneous compute.

Modern SoCs are built for abundance — more cores, more compute engines, more acceleration blocks. But in real-world workloads, performance isn't limited by how much you can compute. It's limited by how fast you can move data, synchronize execution, and reuse memory intelligently.

At Andes RISC-V Con 2025, Imagination Technologies partnered with Baya Systems to present a joint demo using CacheStudio to model and evaluate system behavior across CPU, GPU, and hybrid configurations. The cache analysis was not the subject — it was the lens, helping us visualize how system architecture choices affect coordination, bottlenecks, and throughput.

This paper captures the core insights from that work, applicable across any data-intensive heterogeneous platform.

HETEROGENEOUS COMPUTE REQUIRES MORE THAN JUST MORE COMPUTE

Adding CPUs and GPUs to the same die doesn't make a system heterogeneous. It just makes it crowded.

What matters is:

- How engines interact with shared memory
- How scheduling decisions influence latency
- How coherence policies, fabric layout, and cache sizing shape performance

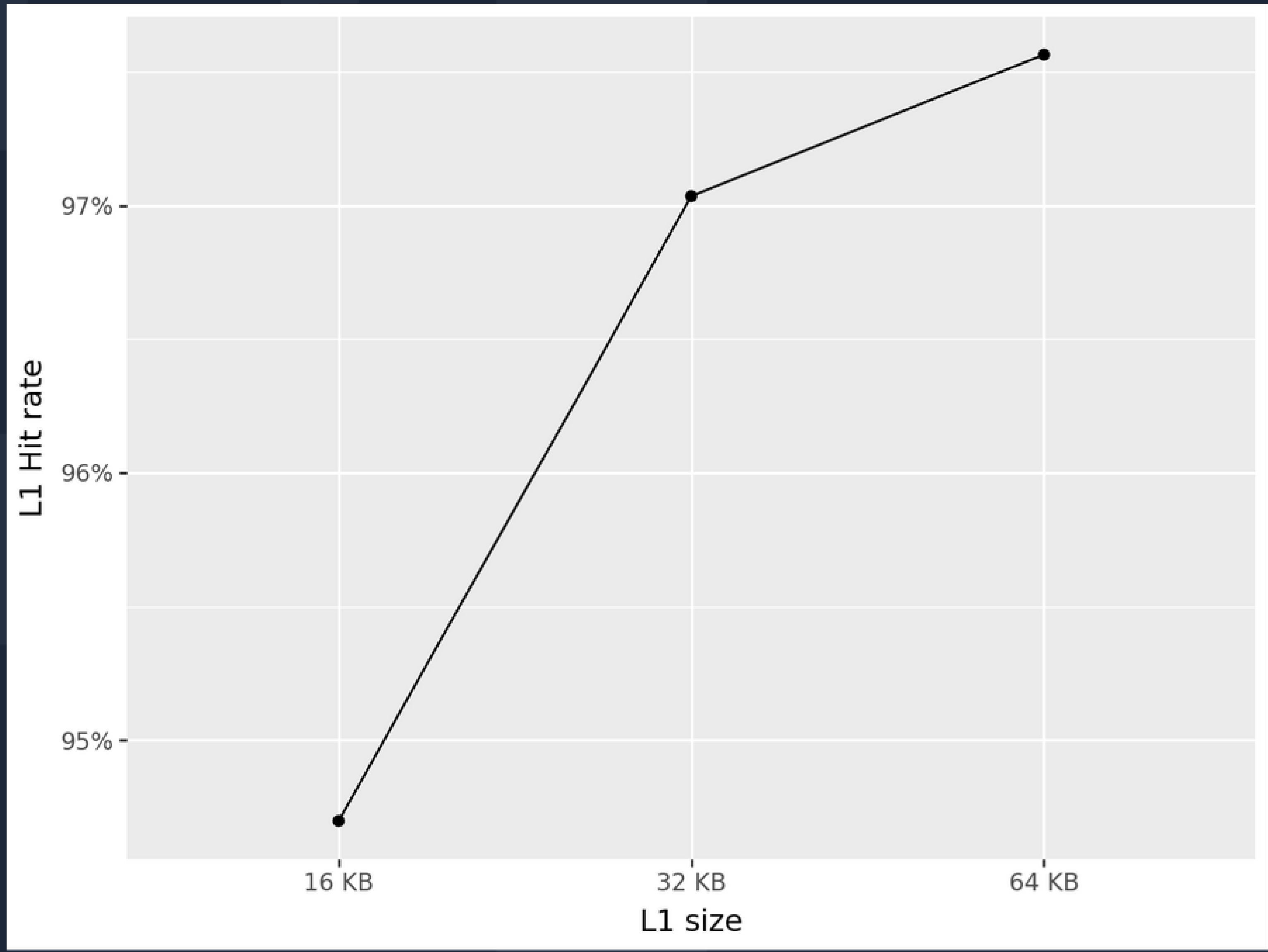
---

Real heterogeneity is about architectural cooperation — not just integration.

ANALYZING CACHE BEHAVIOR IN **HETEROGENEOUS WORKLOADS**

To ground our observations in practical data, we conducted a series of controlled profiling experiments that varied L1, L2, and L3 cache sizes across CPU-only, GPU-only, and mixed CPU+GPU workloads. By isolating the impact of cache configurations on hit rates and memory load, we aimed to understand how different system design choices influence performance across heterogeneous architectures. The goal wasn't just to benchmark absolute performance, but to expose behavioral patterns—what stalls, what scales, and what saturates.

CPU CACHE ANALYSIS - L1



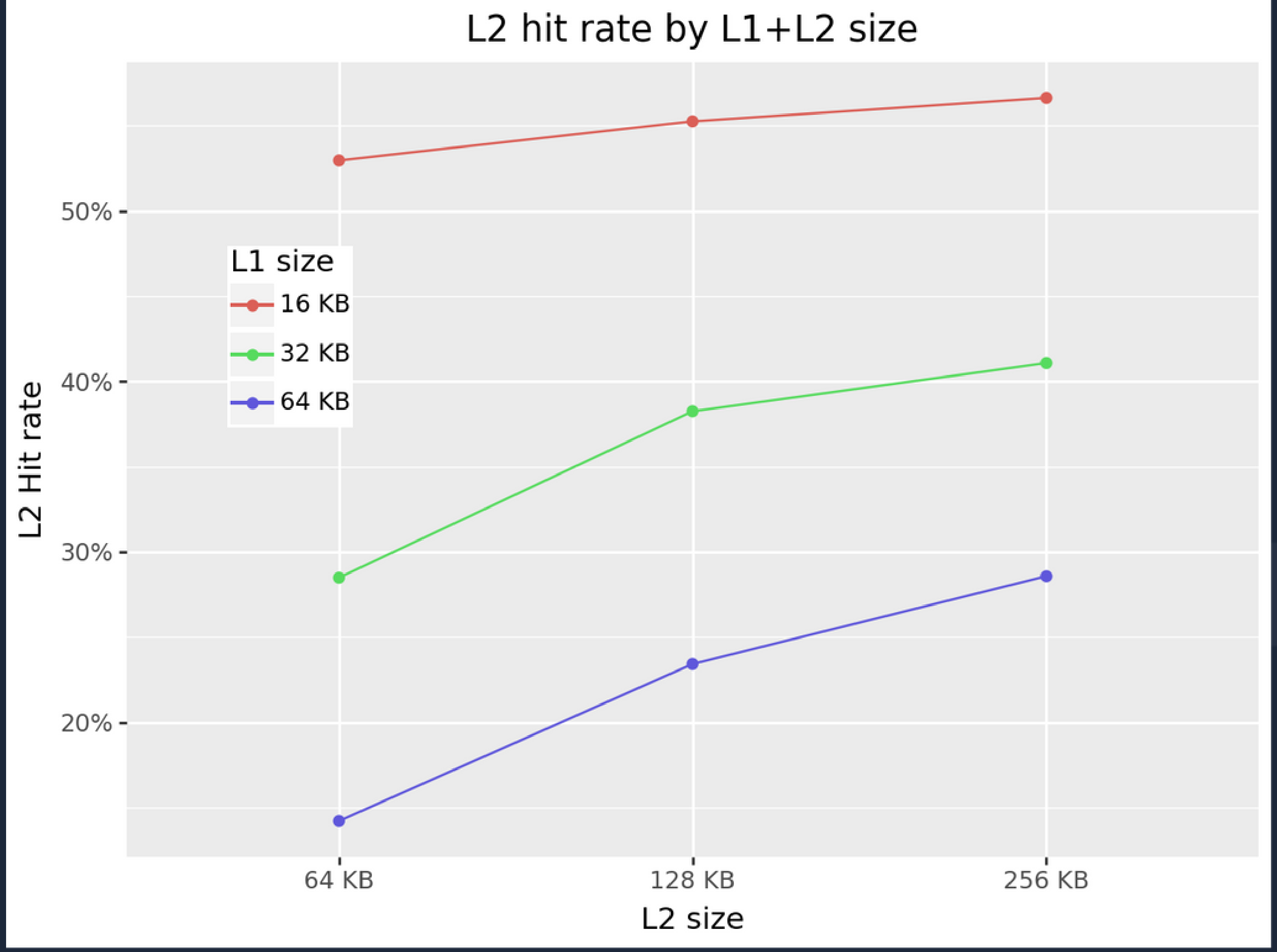
Increasing L1 size from 16 KB → 64 KB significantly improves hit rate:
~94.5% at 16 KB | ~97.1% at 32 KB | ~97.8% at 64 KB

Insights:
CPU workloads demonstrate strong temporal locality.

Even a moderate increase in L1 yields diminishing but consistent returns.

For CPU-bound systems, investing in a well-sized private L1 per core improves execution latency and reduces dependency on L2.

CPU CACHE ANALYSIS - L2



Insights:

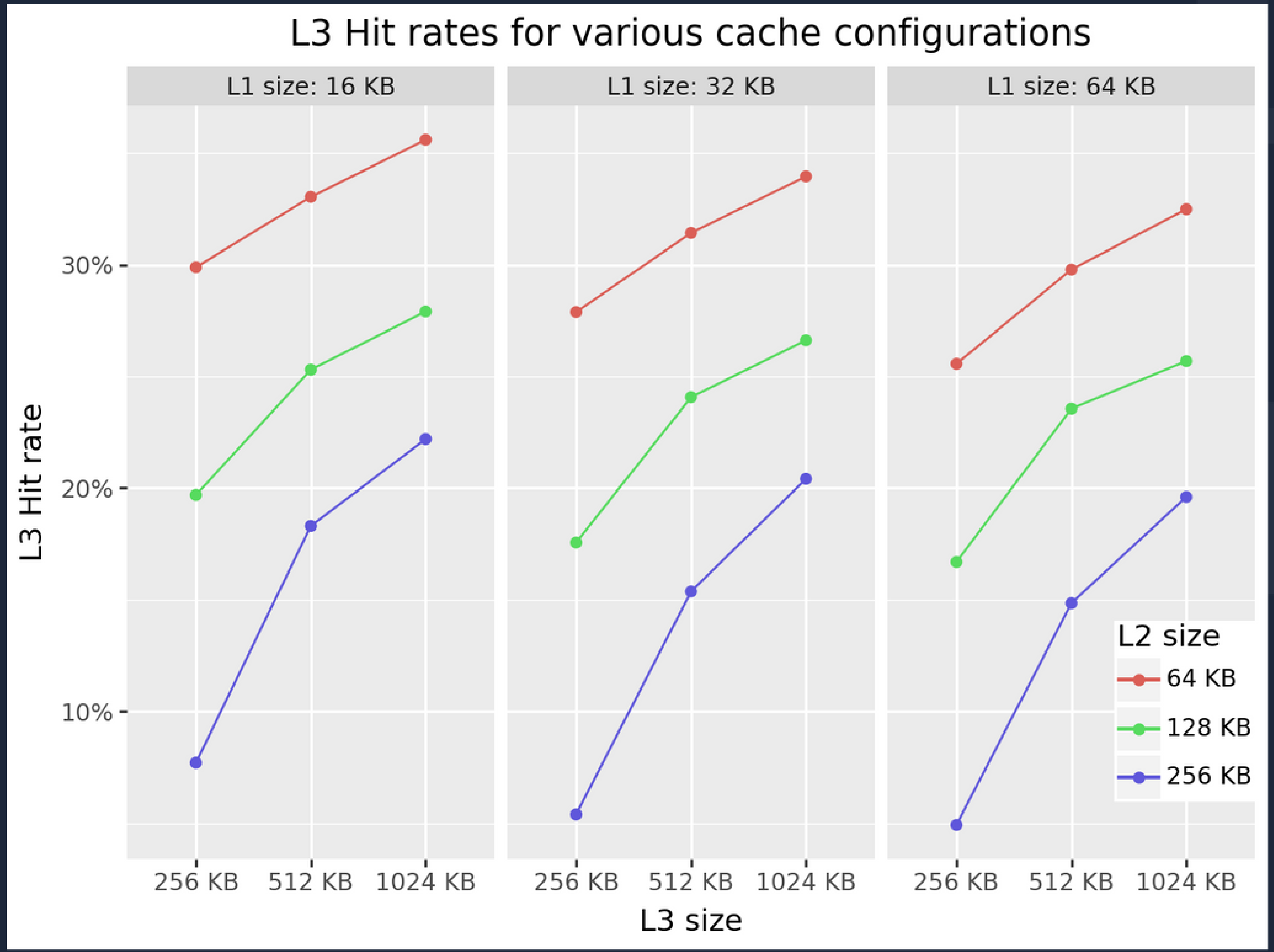
L1 and L2 caches operate as a complementary hierarchy.

Once L1 is well-sized, L2 should be tuned for fallback performance and coherence cost, not maximum hit rate.

Oversizing L2 may not provide linear gains.

L2 hit rate drops as L1 size increases. With small L1, L2 picks up the slack and achieves 50–56% hit rates. As L1 increases, L2 hit rate drops to ~14–28%. With a larger L1 absorbing more accesses, fewer requests fall through to L2 — which reduces L2 utilization

CPU CACHE ANALYSIS - L3

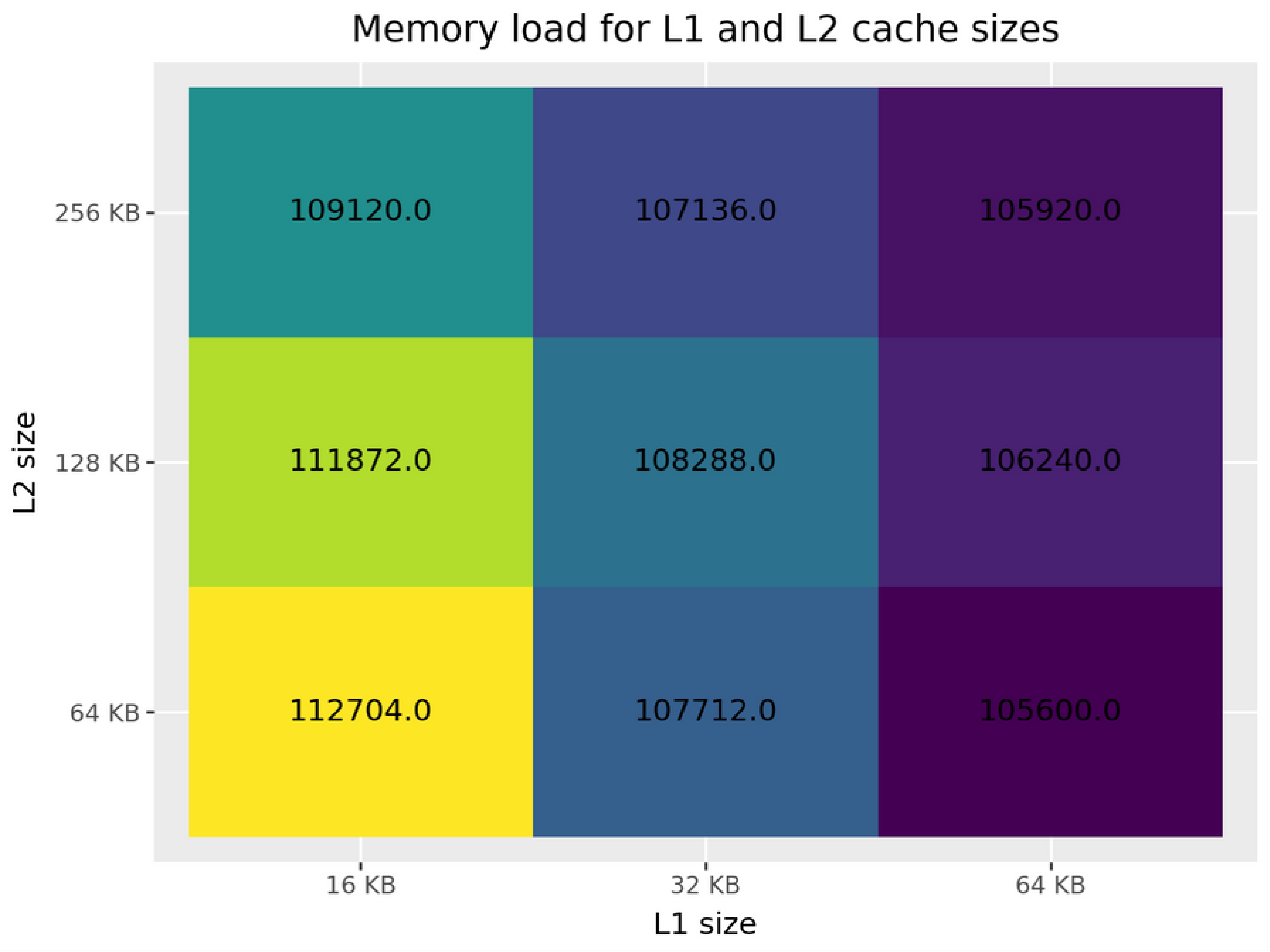


Insights:
L3 plays a secondary role in CPU-bound workloads.

Its primary function is absorbing misses from L2, supporting coherence across cores, and reducing system-level DRAM traffic - but its performance impact is not as sharp as L1 or L2.

L3 hit rate increases predictably with size - but it remains modest (~20–35%)

CPU - L1 & L2 SIZE VS MEMORY LOAD (DRAM TRAFFIC)



Insights:
For CPU-bound workloads, hierarchical cache tuning directly reduces DRAM pressure.

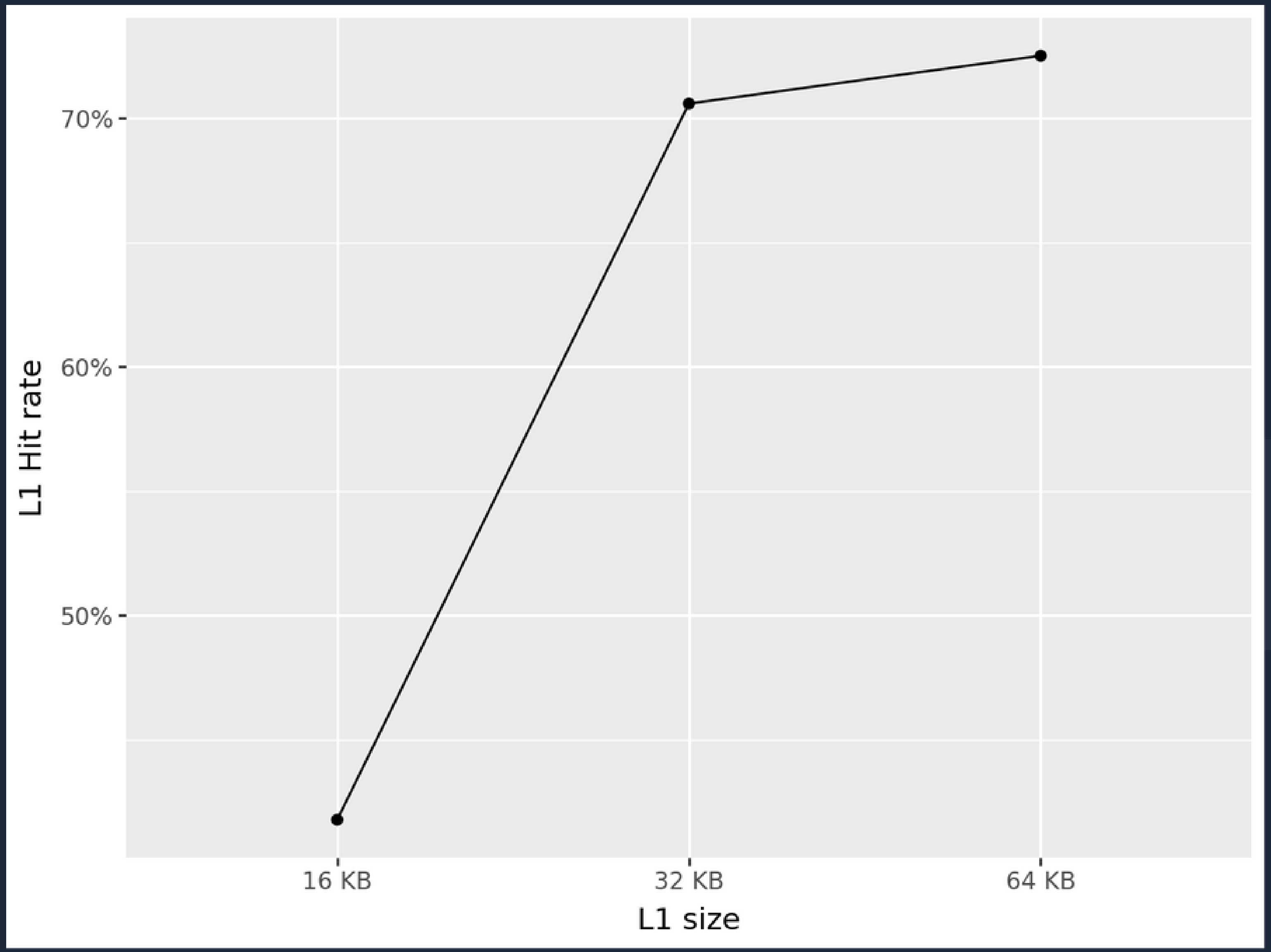
This matters for energy, bandwidth, and latency-sensitive tasks. It's especially useful in real-time or embedded systems where DRAM access is a power and timing bottleneck.

As both L1 and L2 sizes increase, memory load decreases consistently.
At smallest cache sizes: ~112K memory accesses | **At largest:** ~105K

FINDINGS ON
CPU-BOUND WORKFLOWS

CPU workloads benefit most from a well-tuned private L1 and a right-sized L2. Once those are optimized, the system will only see significant gains from increasing L3 in multi-core or coherence-sensitive scenarios.

GPU - L1 CACHE ANALYSIS



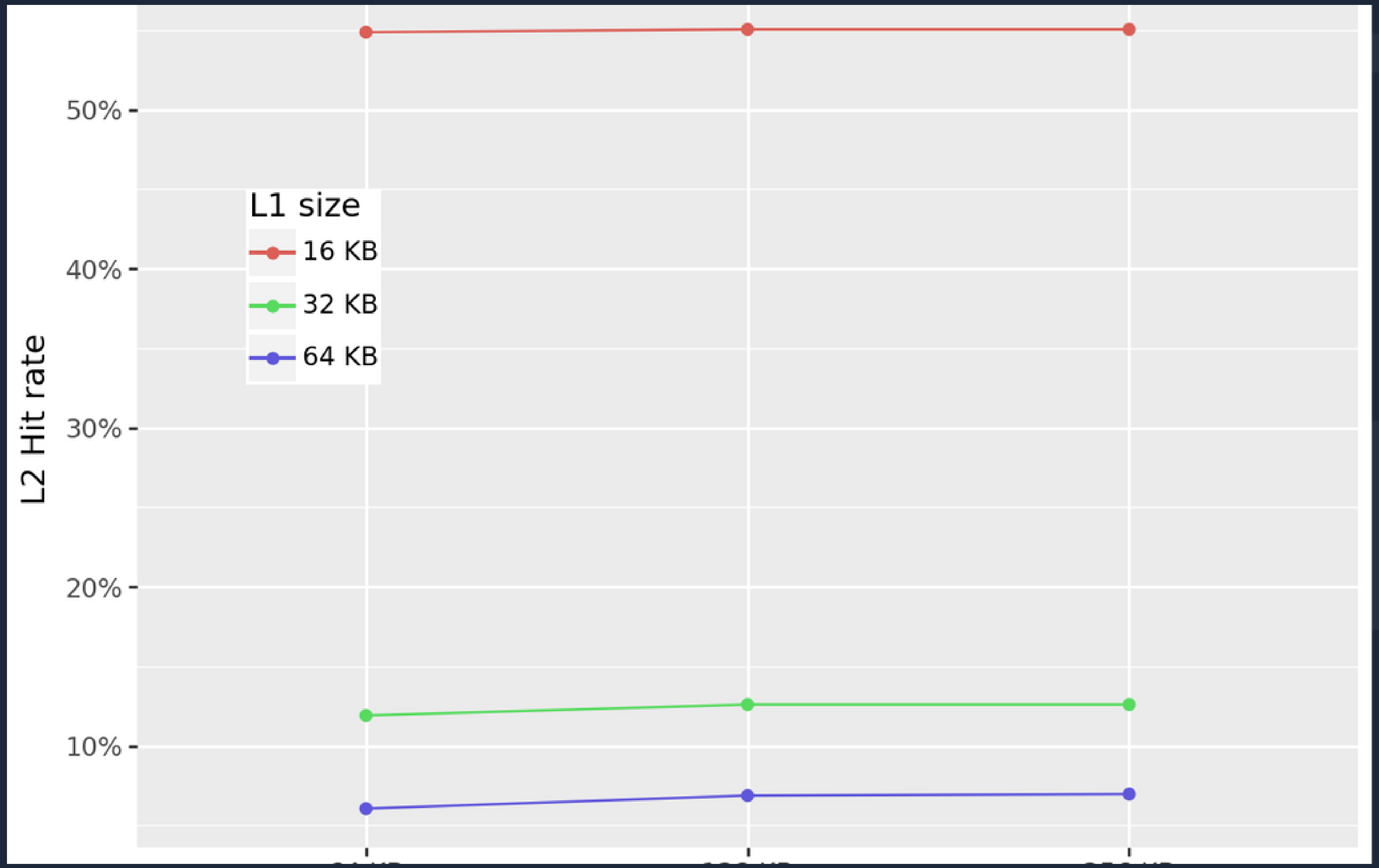
L1 hit rate improves with size, but is much lower than CPU:
54% at 16 KB | 70% at 32 KB | 73% at 64 KB

Insights:
GPU workloads show high parallelism but weaker temporal locality at the L1 level compared to CPUs.

L1 is beneficial, but because GPU threads often access diverse memory regions simultaneously (scattered loads), the cache quickly saturates or suffers from thrashing.

This reinforces the need to optimize memory access patterns in GPU software – not just hardware sizing.

GPU - L2 CACHE ANALYSIS

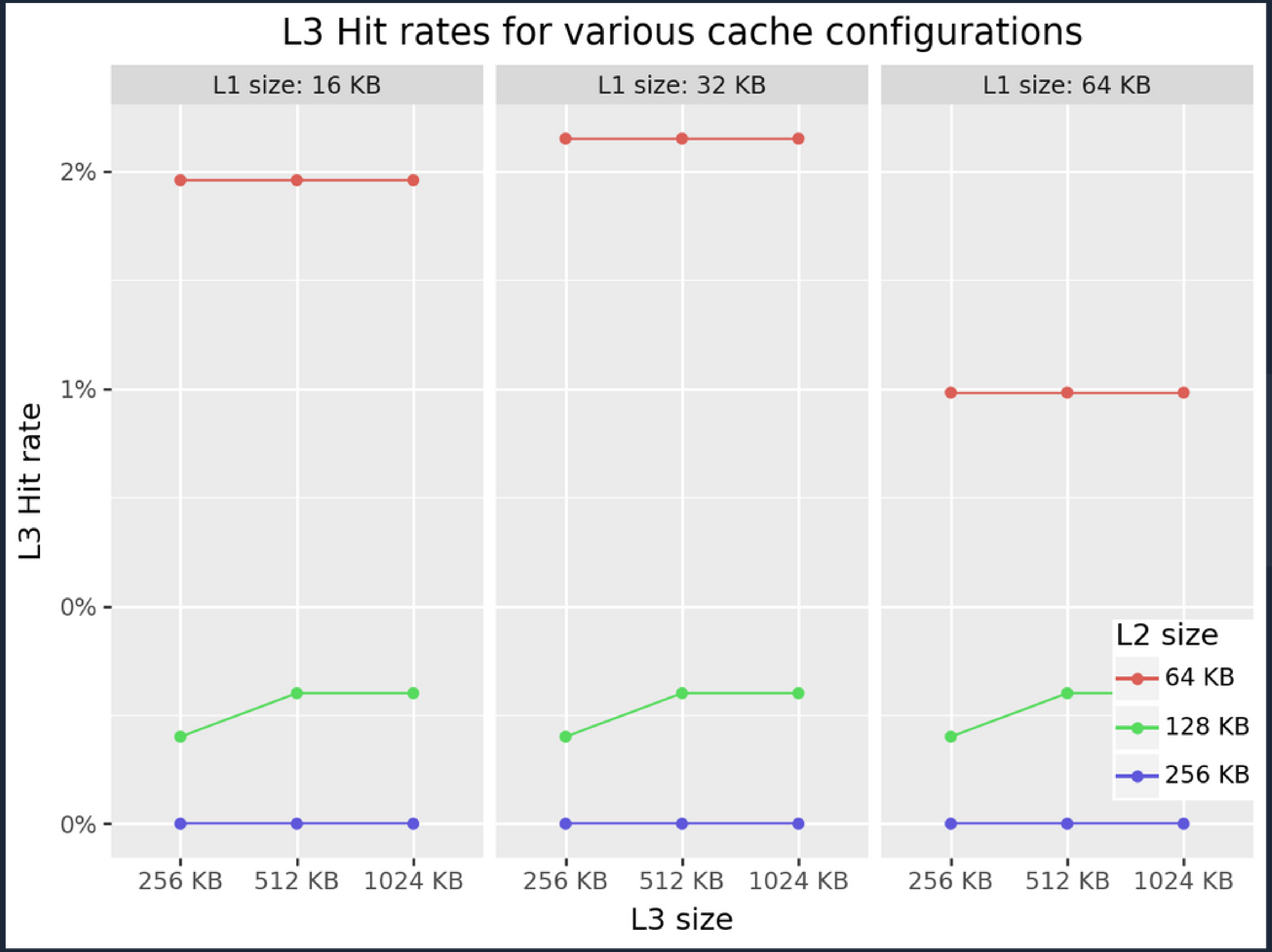


Insights:
GPU L2 effectiveness is not just a matter of size — it's a coherence and access pattern problem.

This shows the importance of shared memory tuning, local store use, or software-managed tiling strategies to reduce unpredictable loads.

We observe very low L2 hit rates unless L1 is very small: At 16 KB L1: ~55% ; At 64 KB L1: ~6-7%. Increasing L2 size has very limited benefit when L1 is large because larger L1 caches absorb most of the spatial locality. But unlike CPUs, GPUs suffer when memory accesses are irregular, causing coherence overhead or bypassing cache tiers altogether

GPU - L3 CACHE ANALYSIS



L3 hit rates are very low across the board — peaking at ~2.2%

Insights:

- L3 is nearly ineffective for GPU traffic in this setup. This could likely be caused due to lack of locality across thread groups, poor caching of streaming workloads or heavy bandwidth pressure that forces L3 bypass or invalidation.

- This demonstrates the need to architect GPU memory flows around bypass behavior and direct-to-DRAM access optimization.

FINDINGS ON
GPU-BOUND WORKFLOWS

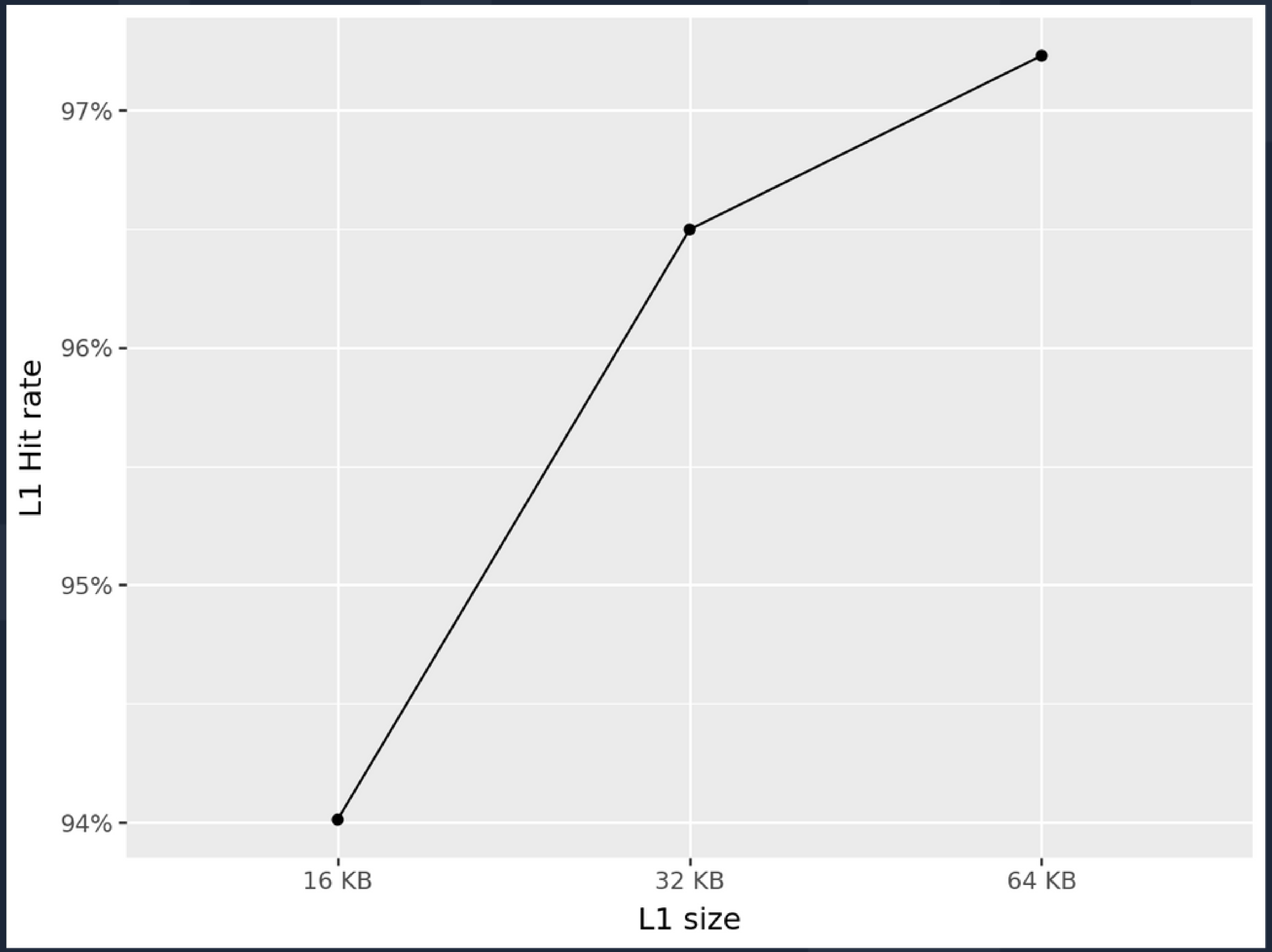
For GPU-bound workloads, L1 cache helps, but not dramatically: Hit rate improves from ~54% to ~73% as size increases from 16 KB to 64 KB — suggesting some local reuse, but limited compared to CPUs.

L2 hit rate drops as L1 gets larger: Larger L1 caches reduce L2 effectiveness. With a 64 KB L1, L2 hit rates drop to ~6–7%, indicating diminishing returns.

L3 cache shows minimal benefit: Across all cache sizes, L3 hit rates remain under 3%

For GPU-bound workloads, optimization needs to go beyond hardware sizing to how data flows through the system.

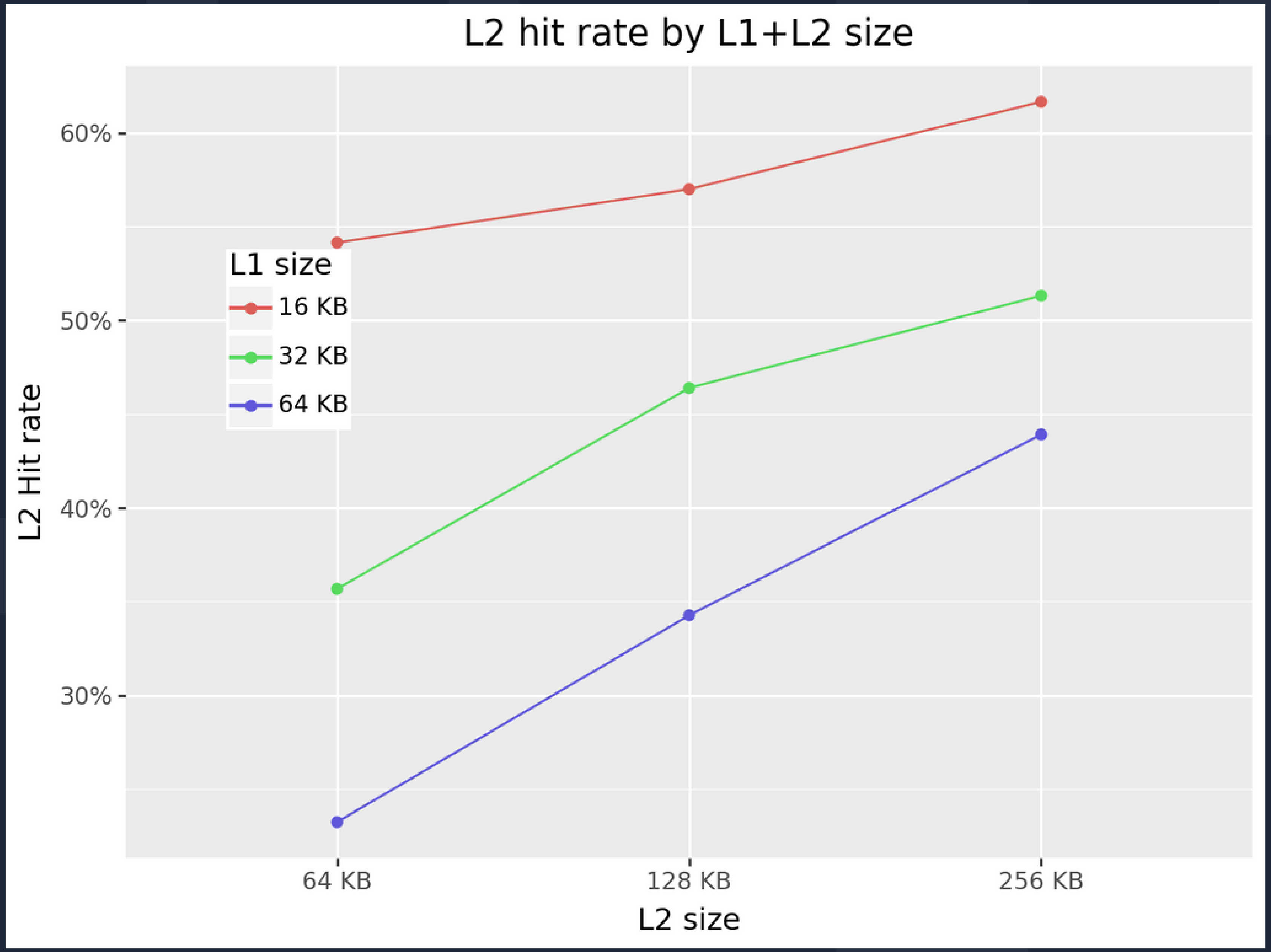
CPU + GPU MIXED WORKLOAD:
L1 CACHE ANALYSIS



Insights:
Even in mixed workloads, L1 locality matters. Both CPU control logic and GPU streaming data benefit from larger L1 caches.

As L1 cache size increases from 16 KB to 64 KB, L1 hit rate improves from ~94% to over 97%.

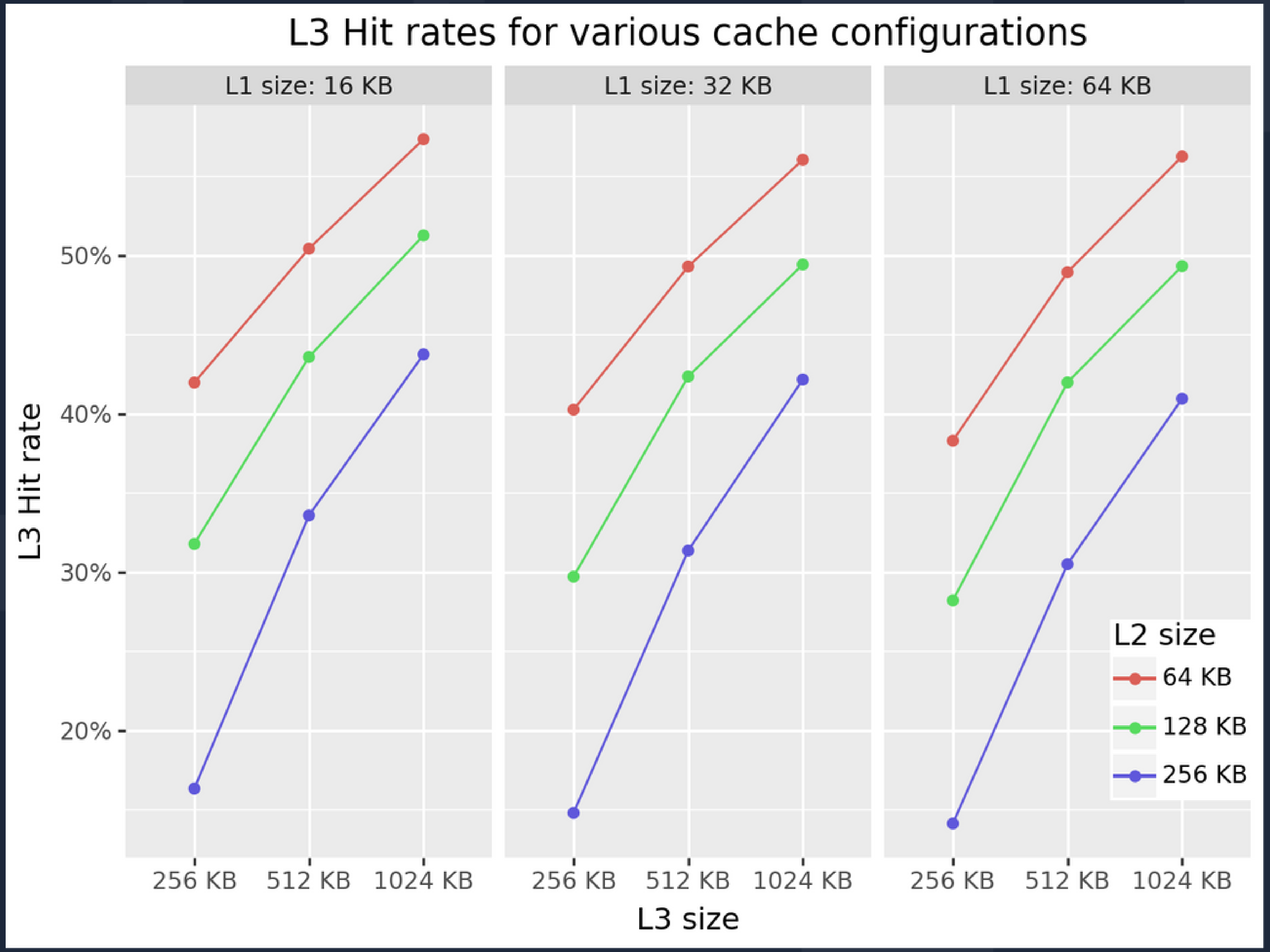
CPU + GPU MIXED WORKLOAD:
L2 CACHE ANALYSIS



Insights:
Larger L1 caches alone don't guarantee better L2 performance. In fact, beyond a point, larger L1s may reduce L2's reuse window due to eviction patterns.
Balanced sizing is key.

L2 hit rates increase significantly as both L1 and L2 sizes grow.
At 256 KB L2 and 16 KB L1, L2 hit rate is 61.7%, but drops sharply to 23.2% with 64 KB L1 and 64 KB L2.

CPU + GPU MIXED WORKLOAD: L3 CACHE ANALYSIS



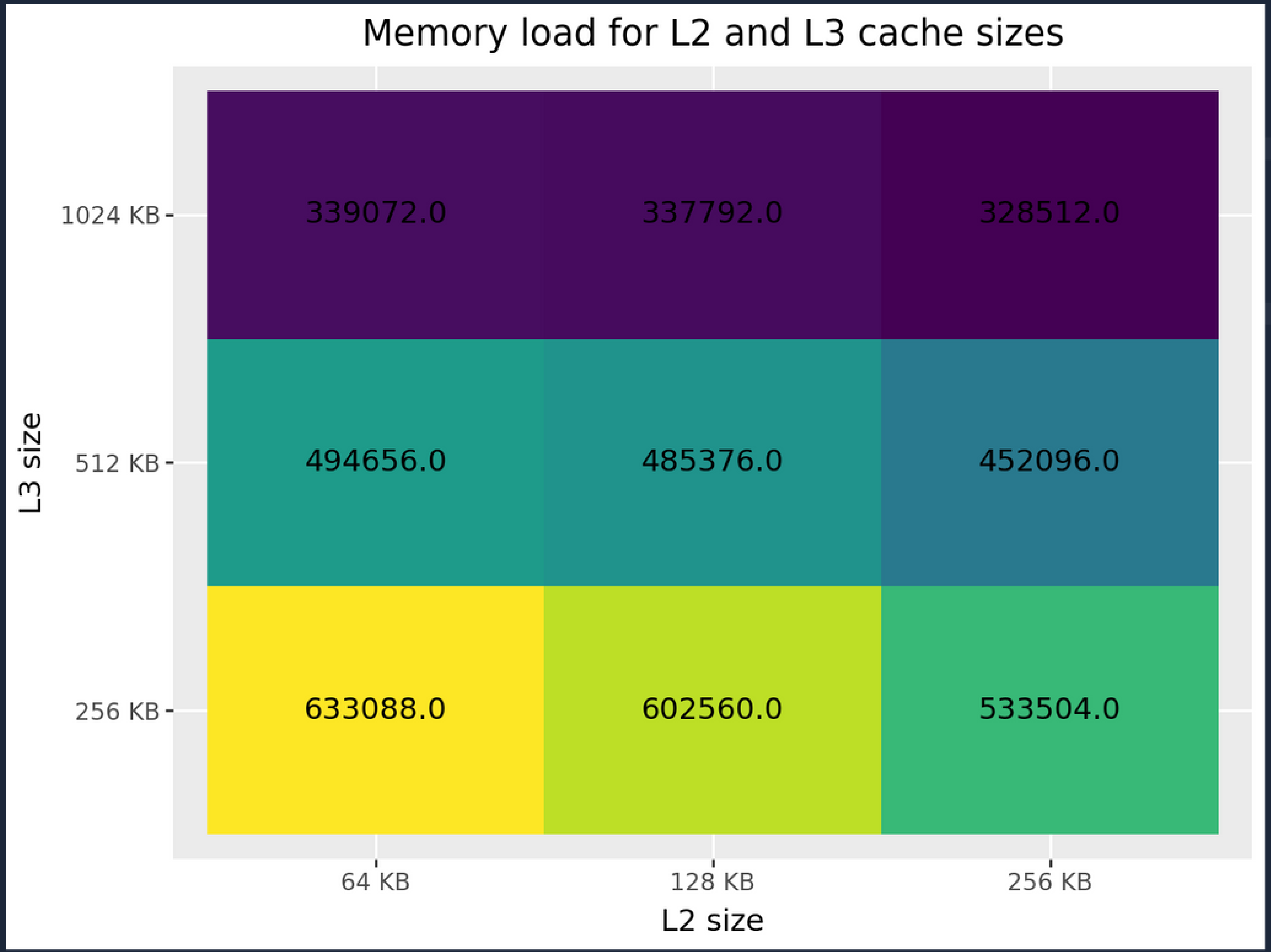
Insights:

L3 hit rate increases steadily with larger L3 sizes across all L1 and L2 configurations.

L3 acts as a useful last-level cache for this mixed workload. Its role becomes more prominent when L1 and L2 can't hold the working set.

At 1024 KB L3, L3 hit rate approaches 55–57% for small L1/L2 combos, but stays lower for larger ones.

CPU + GPU MIXED WORKLOAD:
DRAM ANALYSIS



Insights:
Right-sizing all three cache layers together reduces DRAM pressure in mixed compute scenarios.

Memory load decreases as cache sizes increase, similar to previous sets.

At 64 KB L1 + 256 KB L2, memory load is ~328K, compared to over 385K at 16 KB L1 and 64 KB L2.

CACHE DESIGN LEARNINGS FOR **HETEROGENEOUS SYSTEMS**

- L1 cache size increase always helps, even when CPU and GPU share compute.
- L2 benefit is conditional - co-tuning with L1 is important
- L3 becomes the unsung hero when both engines are active—it shields the system from DRAM access bursts
- Cache design can act as a force multiplier, not just a buffer, in heterogeneous systems

---

This demo-driven analysis reinforces a core message from our Andes RISC-V Con 2025 session:
in heterogeneous compute, system design choices - not just core counts -dictate real-world performance.

By profiling cache behaviour across CPU, GPU, and mixed workloads, we showcased how decisions around cache sizing and memory hierarchy directly influence data locality, bandwidth utilization, and ultimately execution efficiency. While this represents a single trace, it served as a powerful foil for highlighting the need to architect for data movement-not assume it-especially in hybrid compute scenarios.

DESIGN FOR DATA FLOW, NOT JUST FLOPS

Imagination's GPUs are optimized for:

- Uniform, parallel workloads
- Shared memory performance
- Bandwidth-aware execution

But even the best GPU stalls without well-architected data flow.

Heterogeneous systems must be built to:

- Minimize cross-engine contention
- Optimize coherence where sharing is required
- Align compute engines with the memory topology they actually use

CLOSING **INSIGHTS**

CACHE ANALYSIS IS A PROXY, NOT THE POINT...

We used cache modeling because it made invisible things visible:
contention, reuse, locality mismatch, and interconnect behavior.

But the real lesson is broader:
Use system-level profiling early - not to size caches, but to shape architecture.

In heterogeneous compute, performance doesn't come from adding more engines.
It comes from designing how they interact.

So Remember... **If you can't feed it, you can't use it.**

THANK YOU_

baya Systems® | Imagination

PRESENTERS

Pallavi Sharma

Director of Product Management
Imagination Technologies

Dr. Eric Norige

Chief Software Architect
Baya Systems

ABOUT IMAGINATION TECHNOLOGIES

Imagination is a UK-based tech company that designs and licenses advanced GPU IP that powers next generation products across autonomous, edge computing, robotics, IoT, servers, and consumer electronics.

Its parallel GPU solutions combine high-performance graphics with on-device AI. Its programmability and ability to securely manage diverse workloads make Imagination's IP ideal for the evolving demands of edge systems. In automotive, its innovations in functional safety support cutting-edge IVI and ADAS systems.

Used in billions of devices across smartphones, cars, homes, and workplaces, Imagination empowers partners with the tools and technology to build groundbreaking products.

For more information, please visit www.imaginationtech.com

baya Systems® | Imagination

ABOUT BAYA SYSTEMS

Baya Systems is driving the next wave of foundational chiplet-based, high-performance, and modular semiconductor systems, with a focus on best-in-class on-chip fabric and communication technologies. By enabling seamless interoperability across a diverse array of industry-standard protocols, Baya Systems empowers best-of-breed compute, communication, and I/O solutions to work together.

For more Information please visit: www.bayasystems.com