Flash Memory Summit

AIAP-302-2: Storage for AI Part 2

35 YEARS OF NAND FLASH MEMORY INNOVATION

Agenda: AIAP-302-2: Storage for AI Part 2

Flash Memory Summit

- Accelerating Big Data and AI with a Remote Persistent Memory Pool
  - Jack Zhang – Intel
- Solving AI Workload's Storage Problem with CXL and Tiered Memory
  - Rekha Pitchumani - Samsung Semiconductor
- Enabling sophisticated Endpoint AI by leveraging SPOT and MRAM
  - Carlos Morales - AIMBQ
- Introductions, question wrangling and other things of limited value
  - Howard Marks – VAST Data
    - @deepstoragenet Howard@VASTdata.com

Flash Memory Summit

Accelerating Big Data and AI with a Remote Persistent Memory Pool

Jack Zhang, System Architect, Intel

Jian Zhang, AI Software Engineering Manager, Intel

35 YEARS OF NAND FLASH MEMORY INNOVATION

Agenda

- Background and Motivation
- Persistent Memory
- Remote Persistent Memory Pool
- Performance
- Summary

Background and motivation

Background and Motivation

BOUNDED Storage and Compute resources and **SEPARATED** bigdata and AI cluster brings challenges

Data Capacity

Silos

Costs

Performance
& efficiency

Typical Challenges

Dedicated Bigdata and AI cluster

Data/Capacity Scalability
Space, Power, Utilization
Upgrade Cost

High Data movement cost

Multiple Storage Silos
Inadequate Performance
Provisioning and Configuration

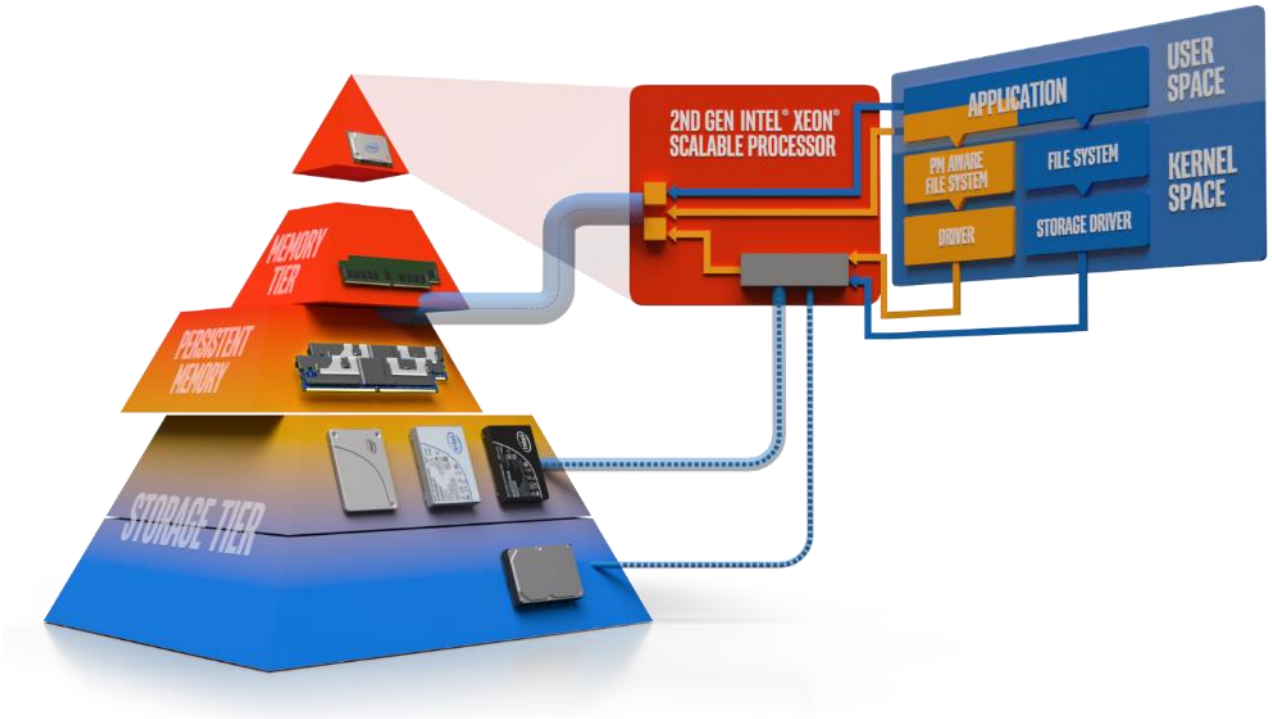
Background and Motivation

- Bigdata and AI convergence
  - Converged bigdata and AI cluster – Bigdata analytics and AI on the same CPU/GPU cluster
    - Improved the balance of hardware resources utilization across different nodes
    - Removed data movement issues – training on where the data stores
- Compute and Storage disaggregation
  - Recent development in software and hardware
    - Modern datacenter is evolving - high speed network between compute and disaggregated storage and tiered storage architecture makes local storage less attractive
    - New storage technologies are emerging, e.g., storage class memory (or PMem)
    - Compute and storage disaggregation become a key trend, diskless environment becoming more and more popular *
  - Disaggregating compute and storage in the converged cluster brings more benefits
    - Improved intermediate data performance with state-of-the-art hardware and software technologies
    - Improved reliability – as intermediate data are replicated on the disaggregated storage

1. https://www.slideshare.net/databricks/improving-apache-spark-by-taking-advantage-of-disaggregated-architecture
2. https://databricks.com/session/cosco-an-efficient-facebook-scale-shuffle-service
3. http://apache-spark-developers-list.1001551.n3.nabble.com/Enabling-fully-disaggregated-shuffle-on-Spark-td28329.html
4. https://databricks.com/session/optimizing-performance-and-computing-resource-efficiency-of-in-memory-big-data-analytics-with-disaggregated-persistent-memory

Persistent Memory

PMem - A New Memory Tier



- IDC reports indicated that data is growing very fast
  - Global datasphere growth rate (CAGR) 27%**
  - But DRAM density scaling is becoming slower: from 4X/3yr to 2X/3yr to 2X/4yr*
  - A new memory system will be needed to met the data growth needs for new cases
- PMem: new category that sits between memory and storage
  - Delivers a unique combination of affordable large capacity and support for data persistence
- Two operational Mode
  - Memory Mode: Enlarge system Memory size
  - App Direct Mode: exposes two set of independent memory resources to OS and applications

Access Remote Persistent Memory over RDMA

Flash Memory Summit

Remote Persistent Memory offers

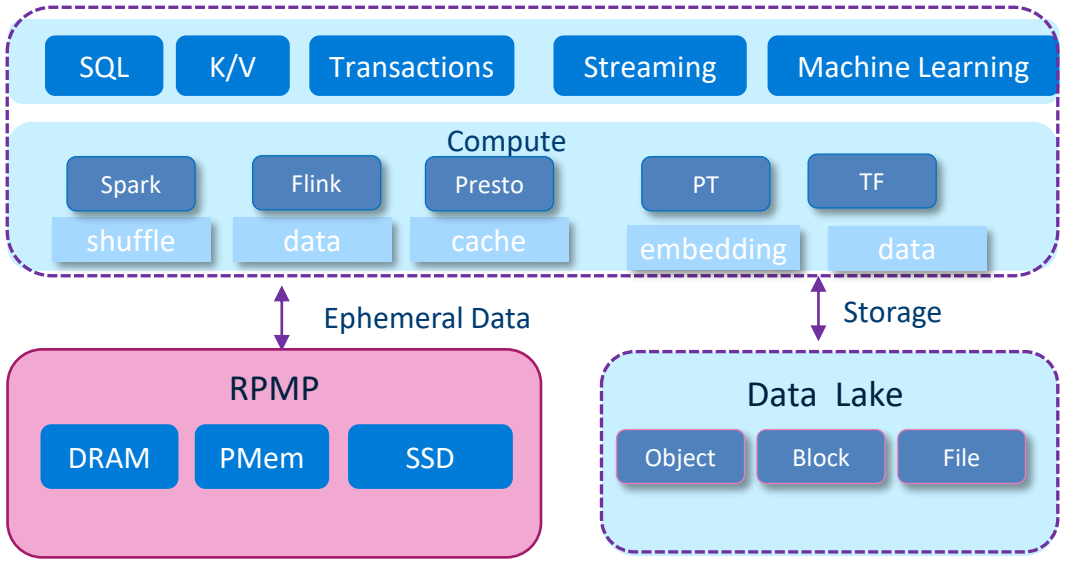
- PM offers remote persistence, without losing any of characteristic of memory
- PM is really Fast
- Needs ultra low-latency networking
- PM has very high bandwidth
- Needs ultra efficient protocol, transport offload, high BW
- Remote access must not add significant latency
- Network switches & adaptors deliver predictability, fairness, zero packet loss

RDMA offers

- Moving data between (zero-copy) two system with Volatile DRAM, offload data movement from CPU to NIC
- Low latency
- Latency < uses
- High BW
- 200Gb/s, 400Gb/s, zero-copy, kernel bypass, HW offered one side memory to remote memory operations
- Reliable credit base data and control delivered by HW
- Network resiliency, scale-out

Remote Persistent Memory Pool

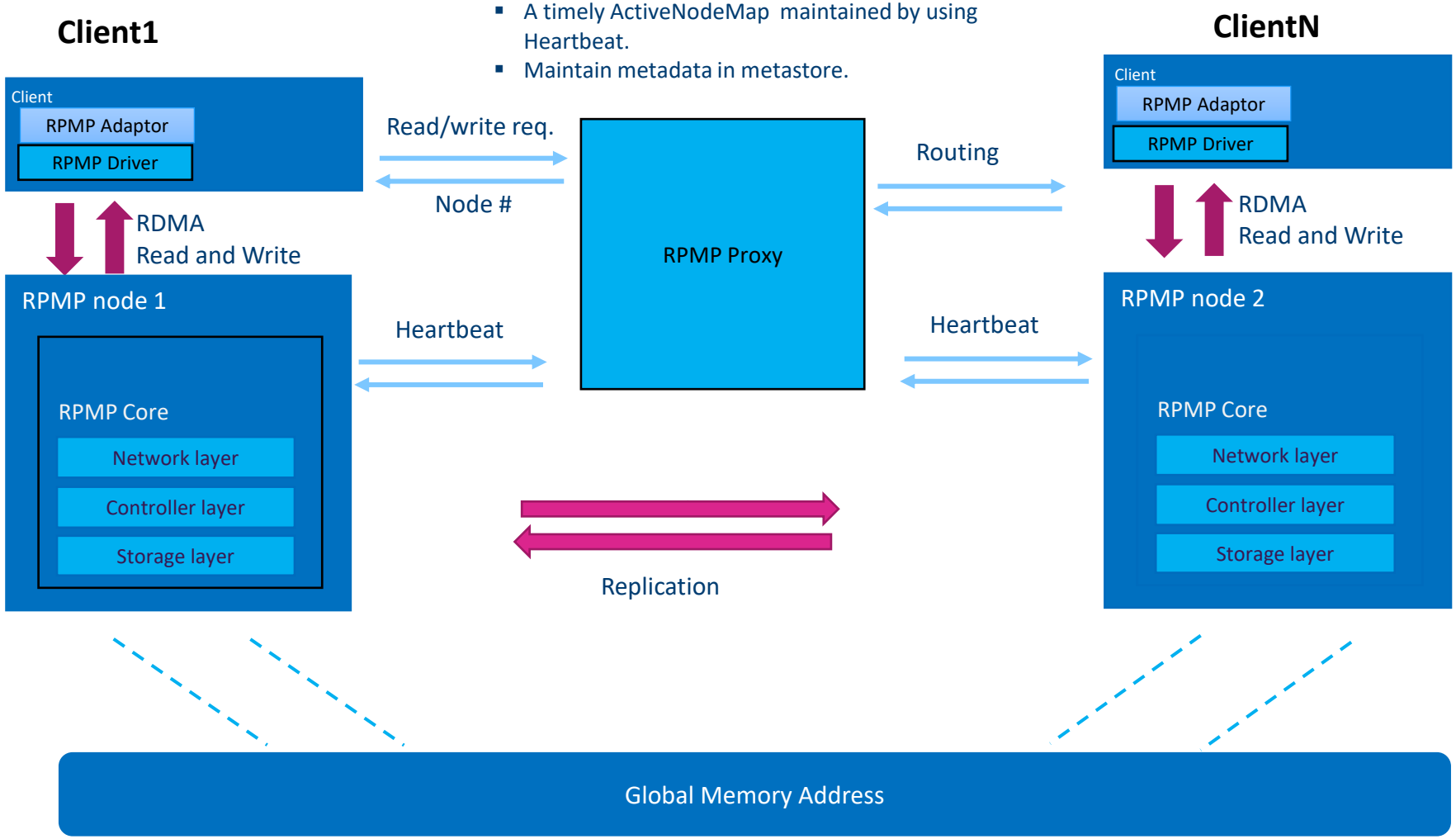
Remote Persistent Memory Pool



- **Remote Persistent Memory Pool:** a new fully distributed ephemeral storage solution that leverage state-of-art hardware technologies including persistent memory and RDMA for bigdata and AI converged cluster
  - A persistent memory based distributed storage system for ephemeral data storage: shuffle/cache/high performance data / embedding tables
  - An RDMA powered network library and an innovative approach to use persistent memory as both shuffle/cache media as well as RDMA memory region to reduce additional memory copies and context switches.
- Features
  - Provides allocate/free/read/write APIs on pooled PMem resources – can be used in other domain
  - Data will be replicated to multiple nodes for High availability
  - Can be extended to other usage scenario such as PMem based database, data store, cache store
- Benefits
  - Improved bigdata analytics workloads' scalability by disaggregating shuffle from compute node to a high-performance distributed storage
  - Improved ephemeral storage's performance with high-speed persistent memory and low latency RDMA network
  - Improved reliability by providing a manageable and highly available shuffle service supports data replication and fault-tolerant.

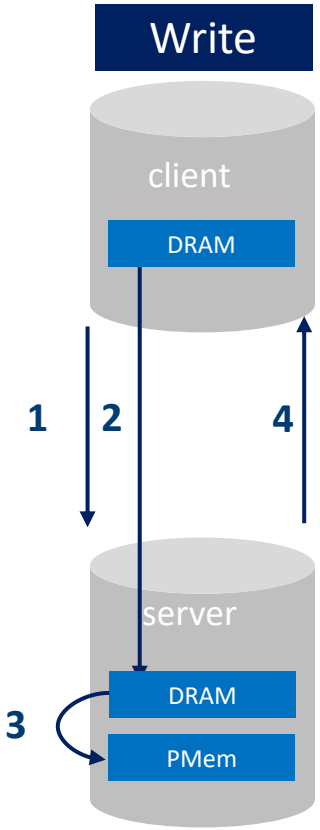
RPMP Cluster architecture

- Routing client requests to corresponding RPMP nodes
- A timely ActiveNodeMap maintained by using Heartbeat.
- Maintain metadata in metastore.

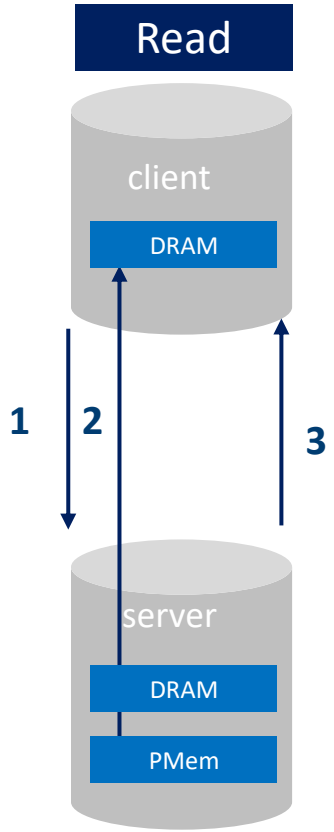


- Client uses RDMA w/ 100Gb NIC for high performance
- Once driver node goes down, worker node is still writable and readable with multiple replica.
- Data will be replicated to a worker node from driver node over RDMA

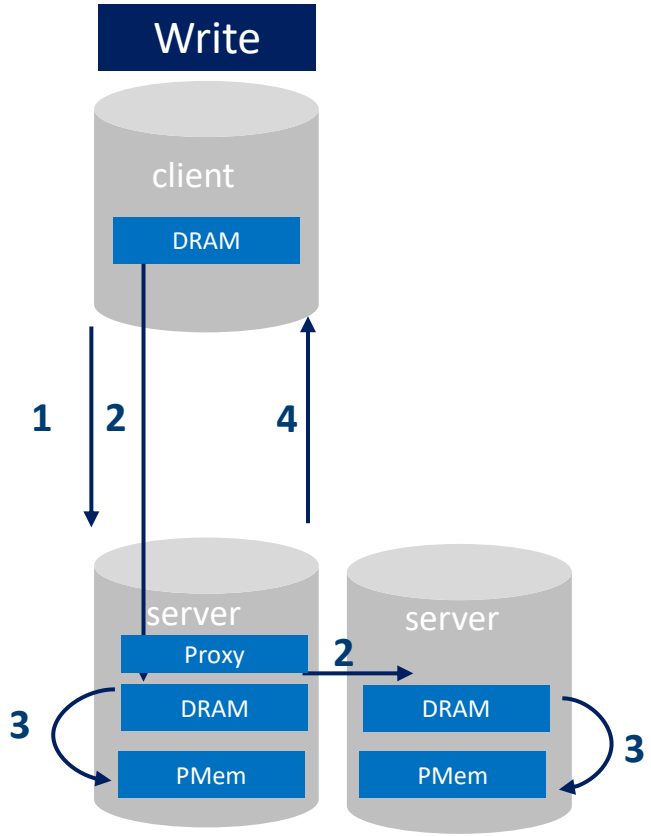
Read/Write Workflow



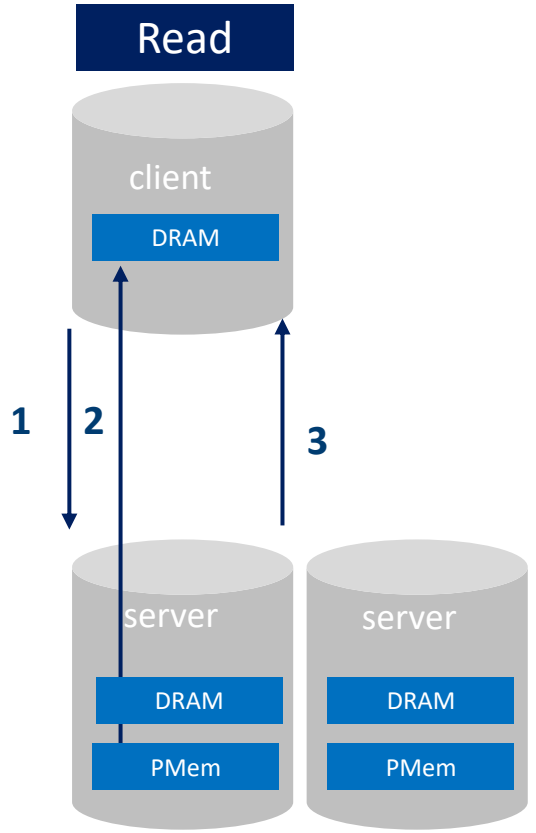
1. Write data to specific address.
2. Server issue RDMA read (client DRAM -> server DRAM).
3. Flush (DRAM -> PMEM).
4. Request ACK.



1. Read data from specific address.
2. RDMA write (server PMEM -> client DRAM).
3. Request ACK.



1. Write data to specific address.
2. RDMA read (client DRAM -> server DRAM), Secondary Node DRAM -> Primary Node DRAM)
3. Flush (DRAM -> PMEM).
4. Request ACK.



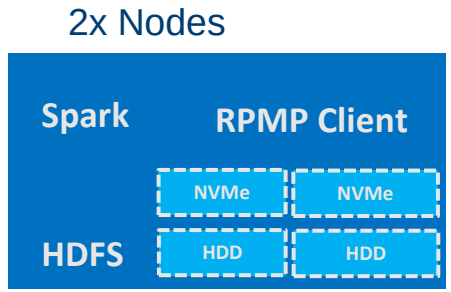
1. Read data from specific address.
2. RDMA write (server PMEM -> client DRAM).
3. Request ACK.

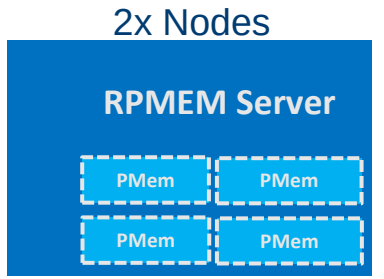
Serve side operation to reduce unnecessary network transfer

Performance

Test Cases

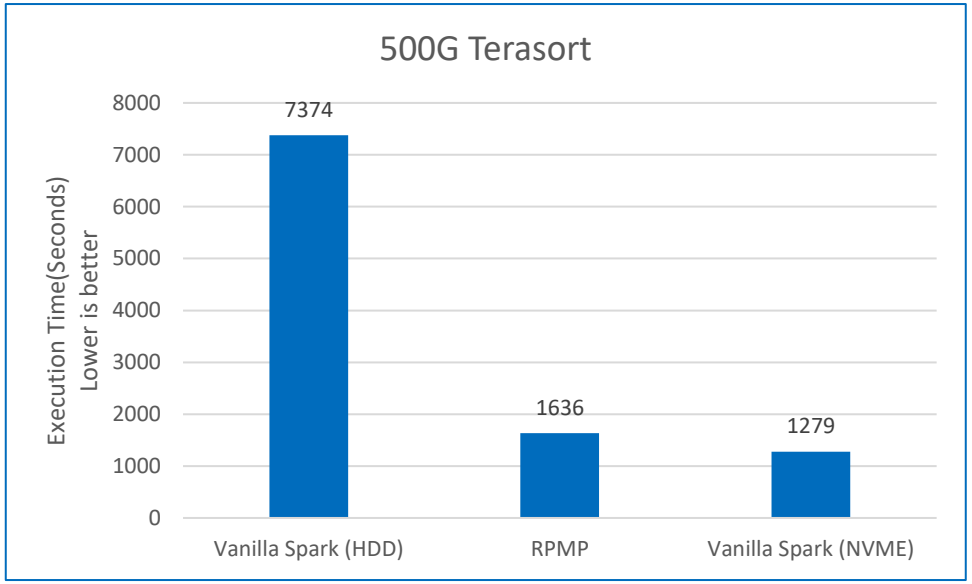
	Vanilla Spark (HDD)	Vanilla Spark (NVMe)	RPMP
Nodes	2	2	2+2
CPU	Intel Xeon Gold 6240 Processor	Intel Xeon Gold 6240 Processor	Intel Xeon Gold 6240 Processor
Memory/PMem	12x 32GB DRAM	12x 32GB DRAM	12x 32GB DRAM 4x 128GB PMem
NIC	X722 10Gb NIC	X722 10Gb NIC	Mellanox 40Gb RDMA NIC,MT27700 family
Storage - boot	1x INTEL 400GB SSD	1x INTEL 400GB SSD	1x INTEL 400GB SSD
Storage - Data	3x 1TB NVMe for HDFS data node 1x HDD for shuffle	3x 1TB NVMe for HDFS data node 1x 1TB NVMe for shuffle	3x 1TB NVMe for HDFS data node
OS	Fedora 29 5.3.11-100.fc29.x86_64	Fedora 29 5.3.11-100.fc29.x86_64	Fedora 29 5.3.11-100.fc29.x86_64
Workloads	Terasort	Terasort	Terasort
Test cases	Normal Shuffle Failover	Normal Shuffle Failover	Normal Shuffle Failover
SW	Hadoop 2.8.1 Spark 3.1.1	Hadoop 2.8.1 Spark 3.1.1	Hadoop 2.8.1 Spark 3.1.1 RPMP 1.0





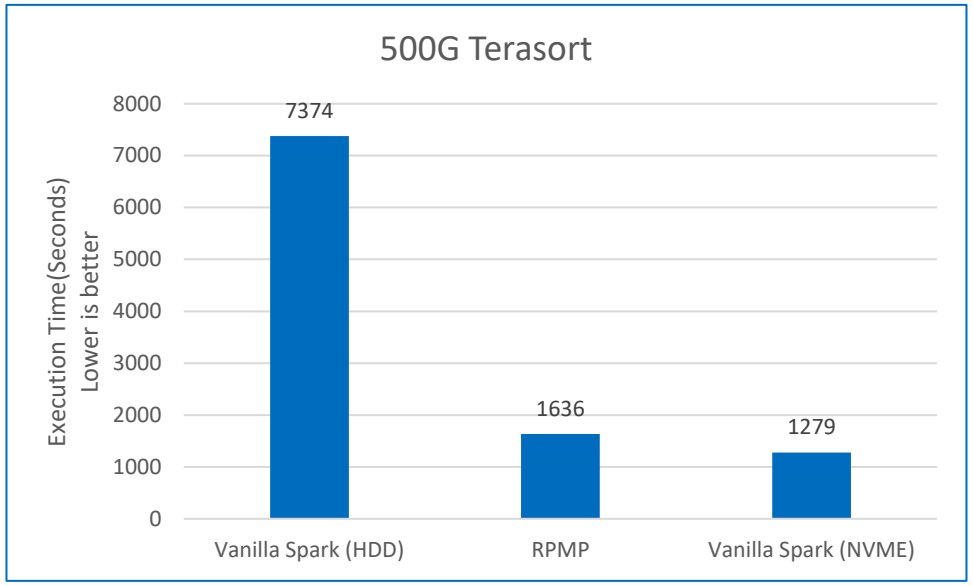
- Three configuration
  - Vanilla Spark with HDD shuffle
  - Vanilla Spark with NVMe Shuffle
  - Vanilla Spark with RPMP shuffle
- Tera sort workloads
  - Normal case without shuffle failure
  - Shuffle failover case

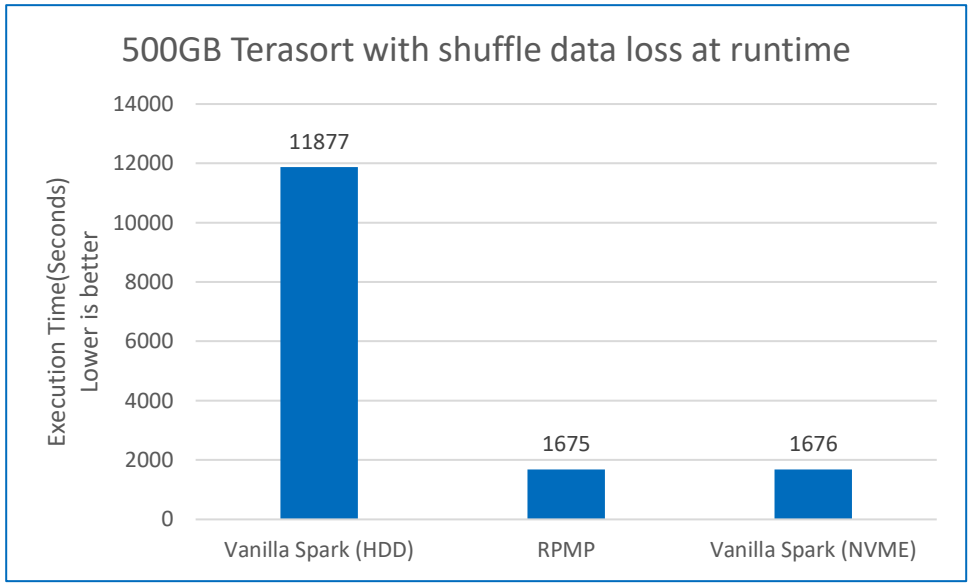
RPMP Normal Case Performance



- RPMP delivered 4.5x speedup over Vanilla Spark over HDD; but 27% lower compared against Vanilla Spark shuffle over local NVMe due to additional network transactions and replication overhead
- Besides the perf gain, RPMP provides better reliability by providing a manageable and highly available disaggregated storage supports data replication and fault-tolerant, e.g. shuffle data to avoid recompute.

RPMP Fail-over Performance

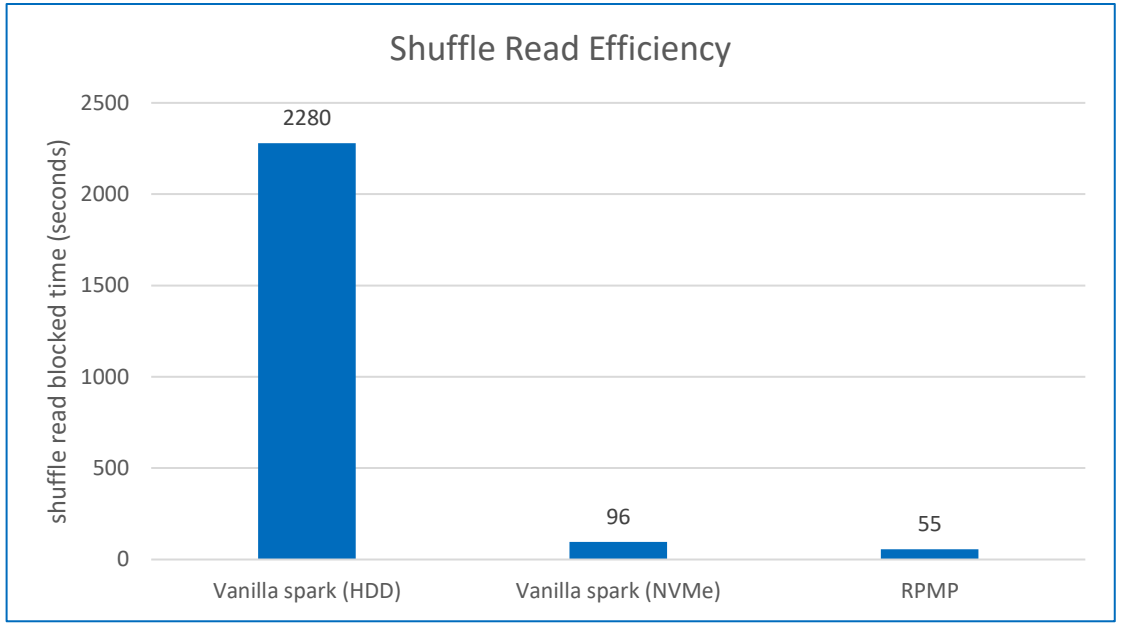




- Failover simulation
  - For Vanilla Spark, shuffle data of one node is removed when Terasort stage 1 completed to simulate one shuffle node failure issue. For RPMP, shuffle data of one RPMP Server is removed when Terasort stage 1 completed to simulate shuffle nodes failure issue.
  - Without manual intervention, Vanilla Spark **fails without successful completion**. To get a measurable performance result, an automatic script is introduced to launch Terasort again right after its failure.
- RPMP detected the failure and suggest Spark to fetch data from other RPMP Server. RPMP delivers same performance compared against Vanilla Spark over NVMe and **7.09x** speed up compared against Vanilla Spark over HDD.
- For Failover, Vanilla spark runtime increased by **61%** for HDD and **31%** for NVMe in fail-over scenario, but RPMP only increased by **2%** of runtime.

Workloads Metrics

- With RPMP as Shuffle storage
  - The compute node spends less time waiting for shuffle data to be read from remote machines.
  - The remote shuffle blocks read efficiency (shuffle read blocked time) improves 41x compared against Spark on HDD and 1.7x compared against Spark on NVMe



Summary

- Compute and storage disaggregation poses new challenges in bigdata and AI cluster, ephemeral data(shuffle, cache) access is critical for bigdata and AI workloads' performance
- Remote Persistent Memory extending PM new usage mode to new scenarios
  - RDMA being the most acceptable technology used for remote persistent memory access
- Remote persistent memory pool enables a fully disaggregated, high performance, low latency ephemeral data access for bigdata and AI workloads
  - Using Spark OLAP as example, it improved OLAP workloads’ scalability by disaggregating shuffle from compute node to a high-performance distributed storage
  - Improved shuffle performance with high speed persistent memory and low latency RDMA network
  - Improved reliability by providing a manageable and highly available shuffle service supports shuffle data replication and fault-tolerant.

Flash Memory Summit

SAMSUNG

Solving AI Workload's Storage Problem with Memory-Semantic SSDs and Tiered Memory

Rekha Pitchumani

PhD, Senior Manager,
Memory Solutions Lab,
Samsung Semiconductor Inc.

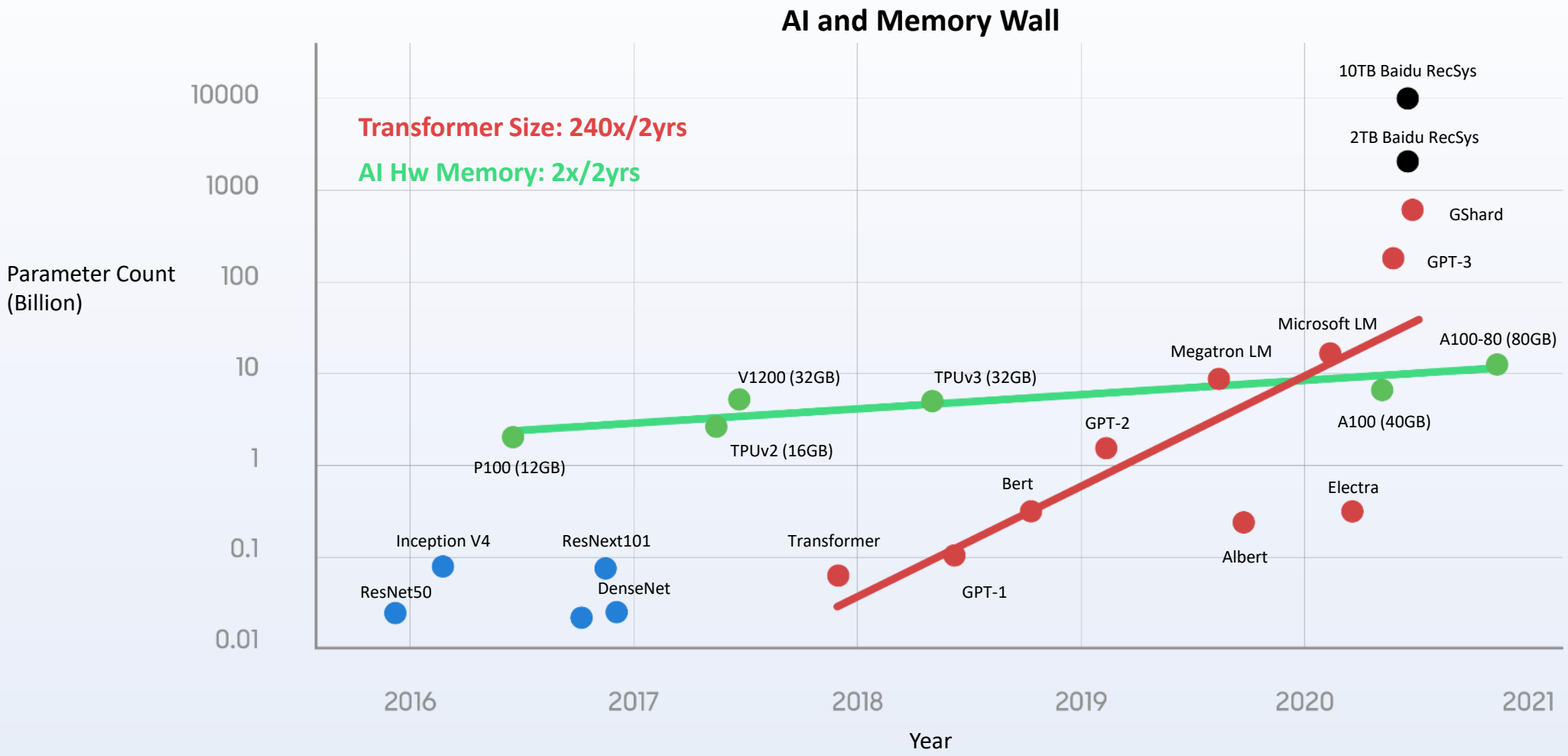
August 4th 2022

Agenda

Flash Memory Summit

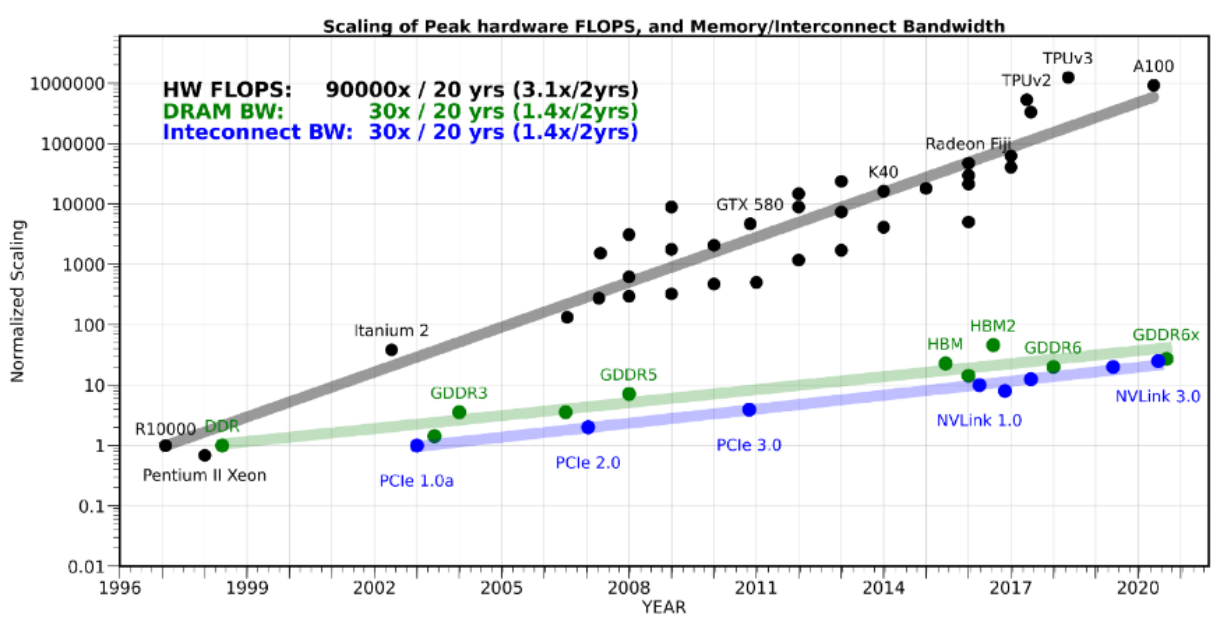
1. Growing AI Memory Needs
2. Samsung Memory Semantic SSD (MS SSD)
3. MS SSD for AI Memory/Storage
4. Memory Tiering for AI
5. Summary

Growing AI Models & Memory Needs



* Source: https://medium.com/riselab/ai-and-memory-wall-2cb4265cb0b8

Accelerator Growth & AI Memory Wall



* Source: https://medium.com/riselab/ai-and-memory-wall-2cb4265cb0b8

Memory capacity and bandwidth lagging behind compute growth. To keep up with workload demands,

- Rethink cache hierarchy
- Scale memory separately with caching, tiering and new interconnect technologies
- Compute near memory where applicable

Samsung's Memory-Semantic SSDs

Flash Memory Summit

With CXL, Memory, and Storage occupy the same physical slot

- Interchangeability means room for Memory-Storage convergence

Memory-Semantic SSD (MS SSD) supports dual (Memory/Storage) mode via the CXL.mem/CXL.io protocols

- Access the same data at a smaller granularity (64B) in memory mode than in IO mode (4KB)

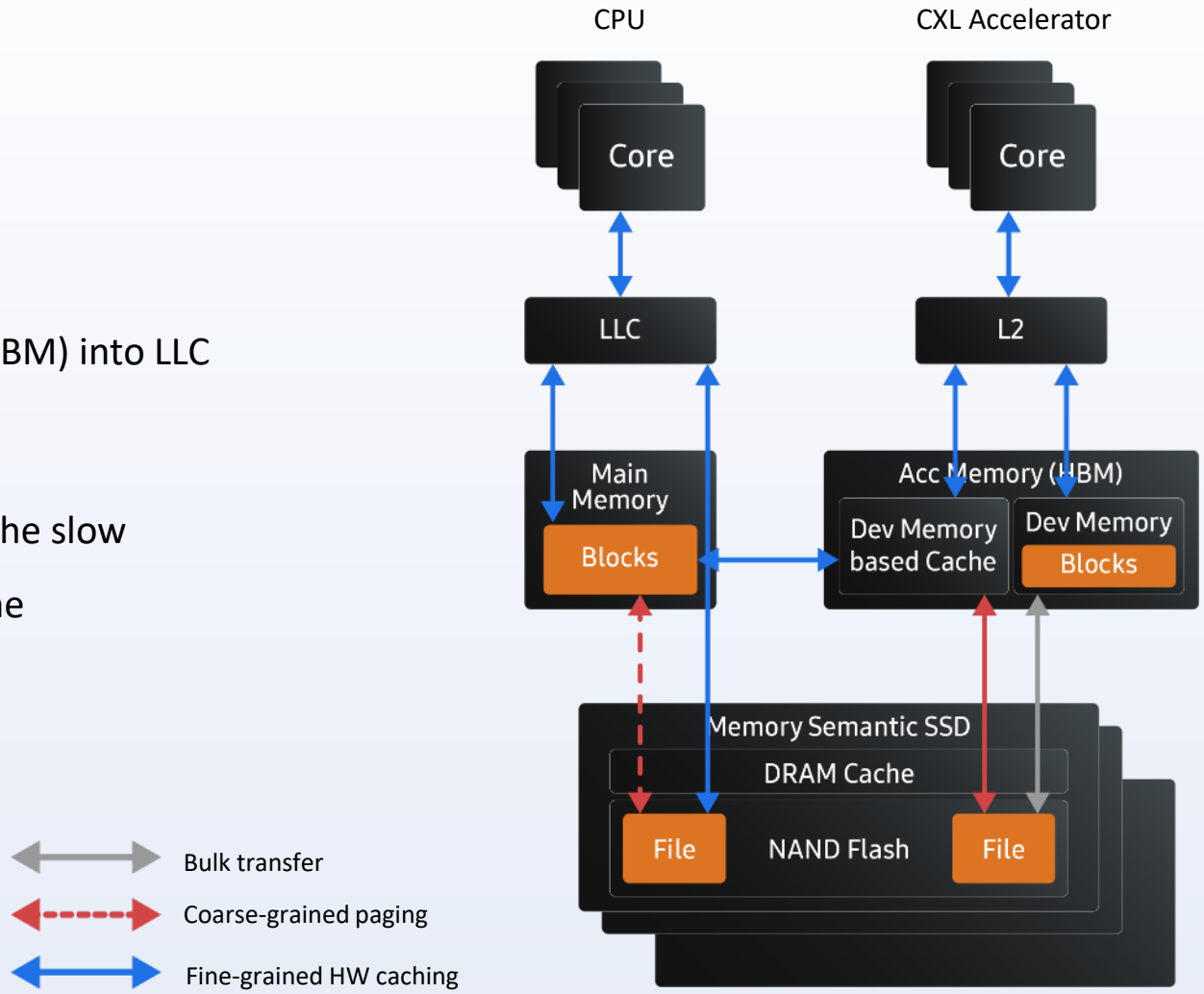
To learn more, check out 'Controller Design Considerations for Memory-Semantic SSD' talk at FMS '22.

CXL Based Solution with MS SSD

Flash Memory Summit

CXL Based Accelerator with

- Redesigned on-chip cache hierarchy
- Ability to partition accelerator memory (e.g., HBM) into LLC or device memory
- CXL based MS SSDs, accessible as memory, as the slow memory tier, with multiple MS SSDs to meet the bandwidth needs of the tier

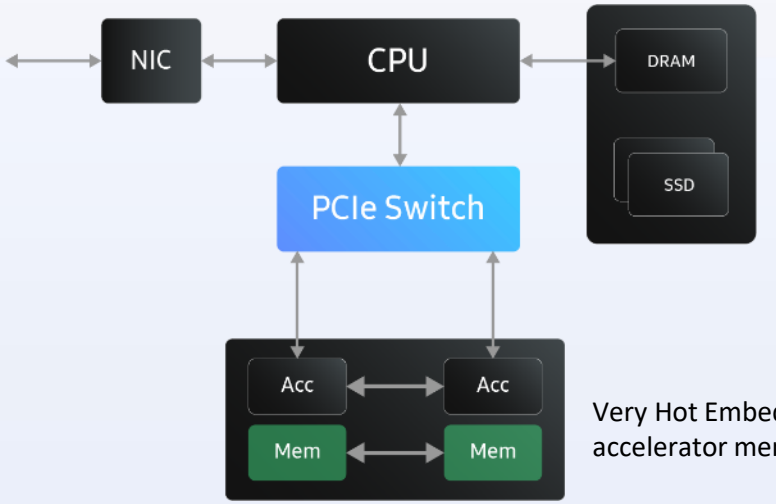


Recommendation Systems (RecSys)

Flash Memory Summit

Meta's Deep Learning Recommendation Model (DLRM) to represent huge RecSys models

NVMe SSD based Software Defined Memory exploration for DLRM models at Meta



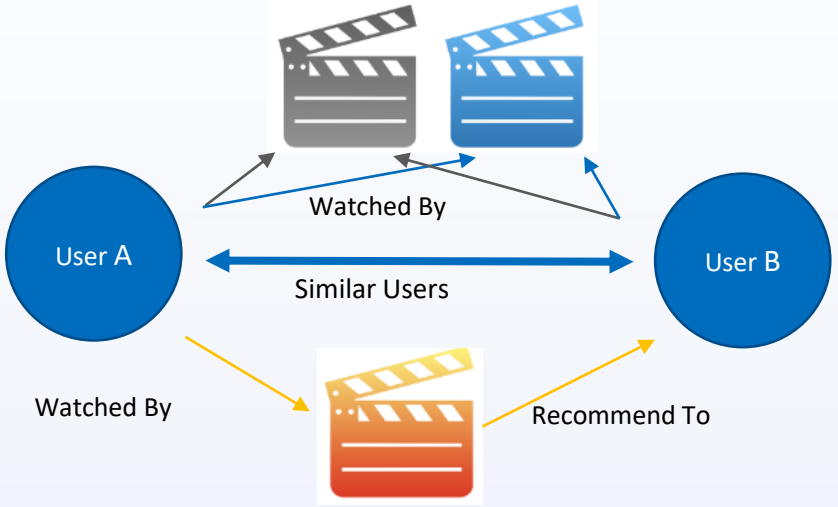
Small Hot Embeddings

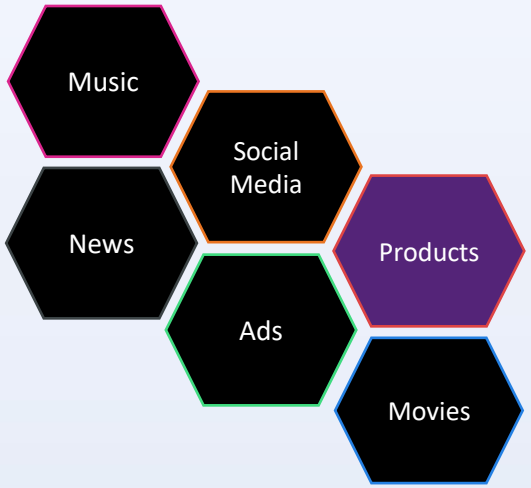
Cache for embeddings in SSD

Embeddings in SSD

Very Hot Embeddings in accelerator memory

Source: "Memory Requirements of Meta AI Workloads"

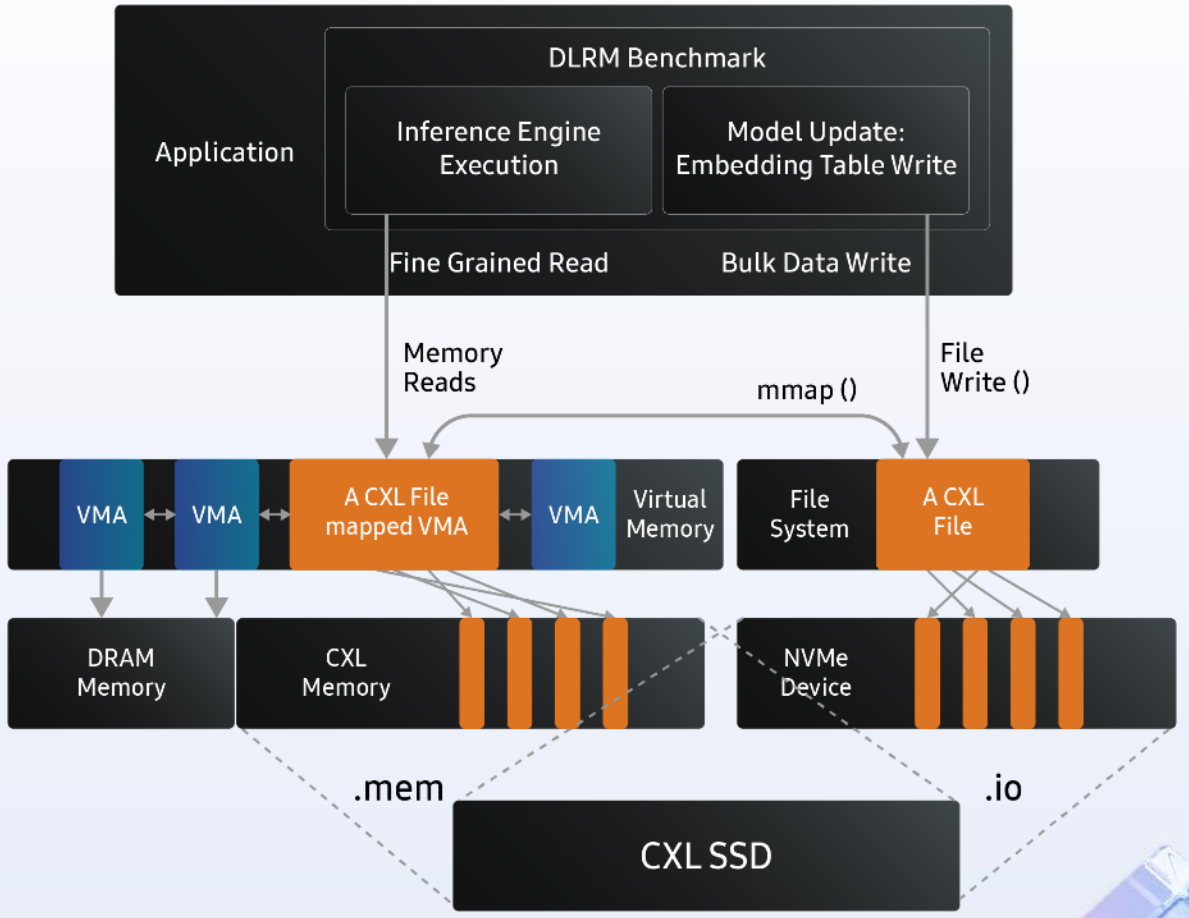




MS SSD for DLRM Workload

Dual-Mode Access

- Use as block device via CXL.io (file-system based access)
- Use as byte addressable memory via CXL.mem with load/store for memory-mapped files



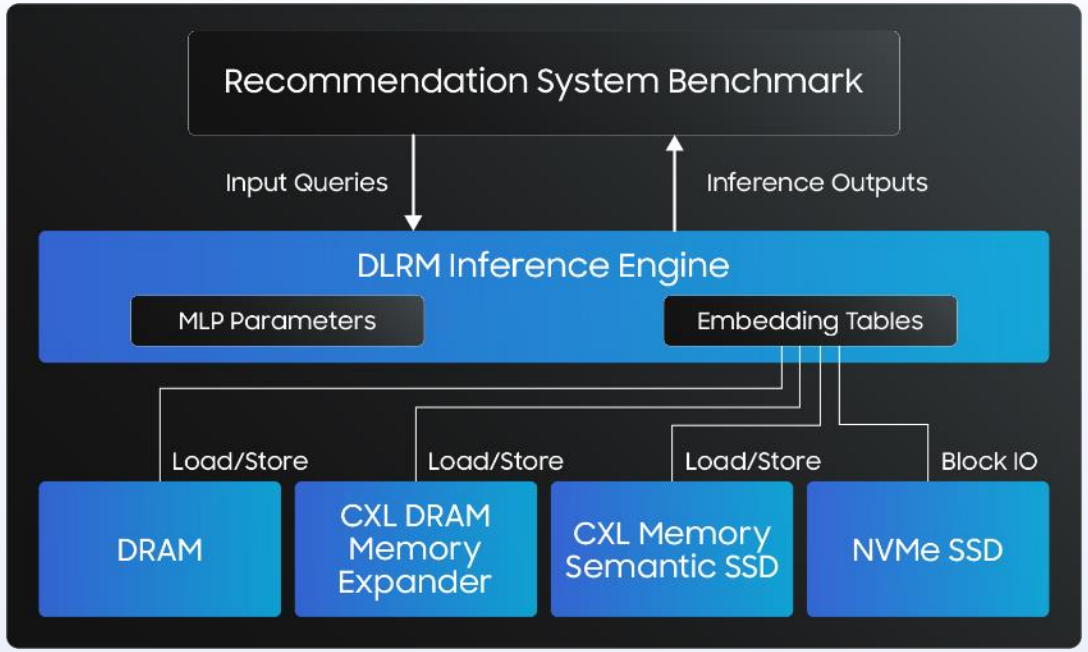
Visit Samsung demo booth (#407) for functional demo of Movie RecSys using MS SSD

RecSys Benchmark

Flash Memory Summit

Started with open recommendation system benchmark for a variety of RecSys including DLRM

- Added options to store and serve embedding tables from CXL Memory Expander, CXL based MS SSD, and NVMe SSD
- Added DRAM cache option and static partitioning of tables
- Adding memory tiering options



https://github.com/harvard-acc/DeepRecSys

https://github.com/facebookresearch/dlrm

Memory Tiering for AI

Flash Memory Summit

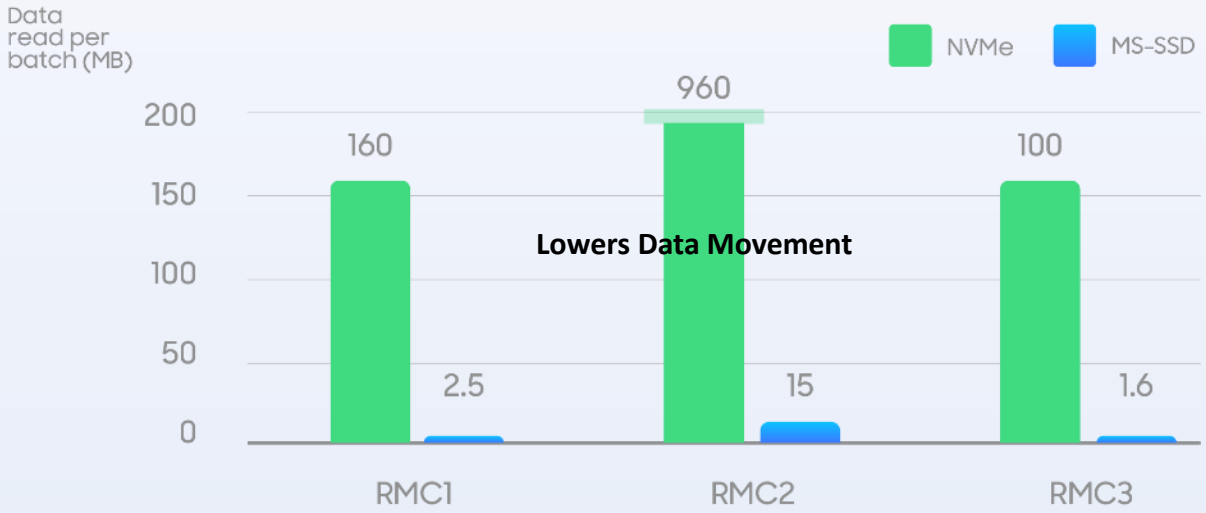
Has to operate at tensor (or embedding vector in case of DLRM) level

- Both static and dynamic options
- Needs and what works change for different AI domain such as Computer Vision, NLP, RecSys, etc.,
  - E.g., Hot/Warm/Cold embedding vector in different tiers in case of RecSys
- Call for more research and more AI framework level tiering support

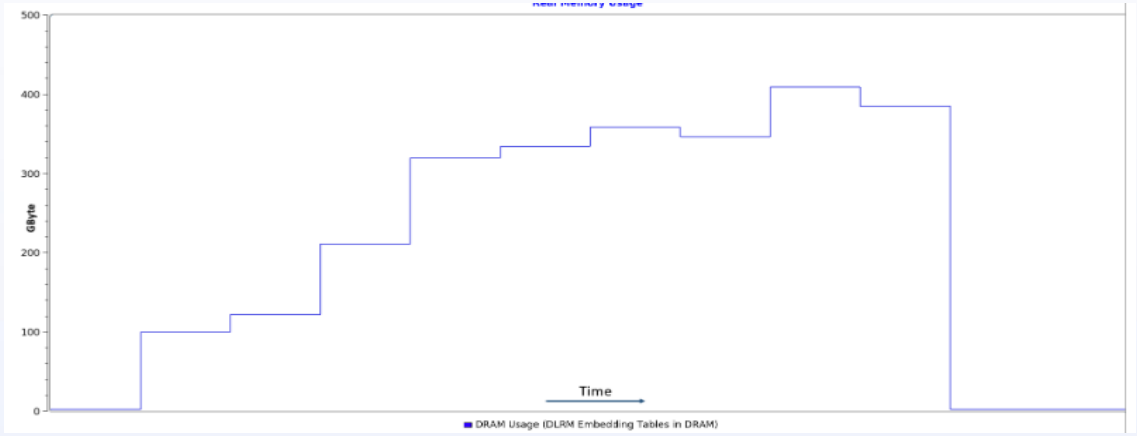
MS SSD Benefit

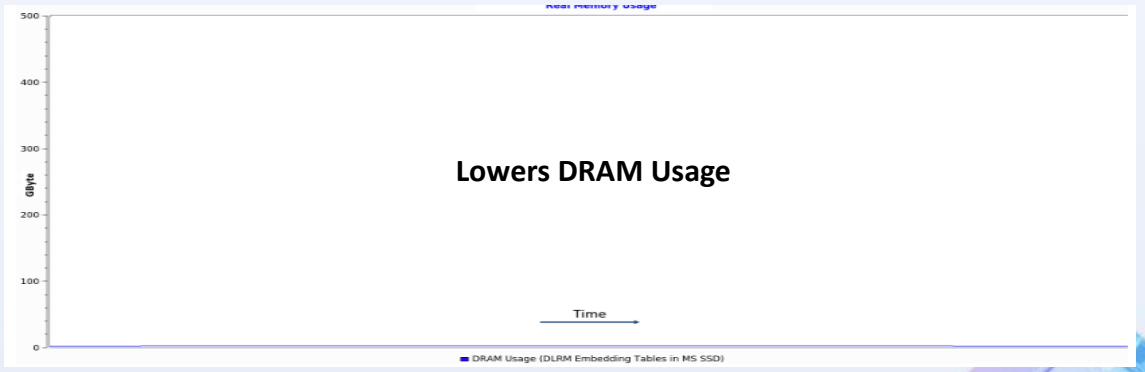
Finer-Granular access leads to less data movement from the SSD

Amount of data read for 3 DLRM models if 4K block is read for every embedding vector read



Amount of DRAM used to run same model with embedding tables on DRAM vs MS SSD





Summary

AI model sizes are fast growing

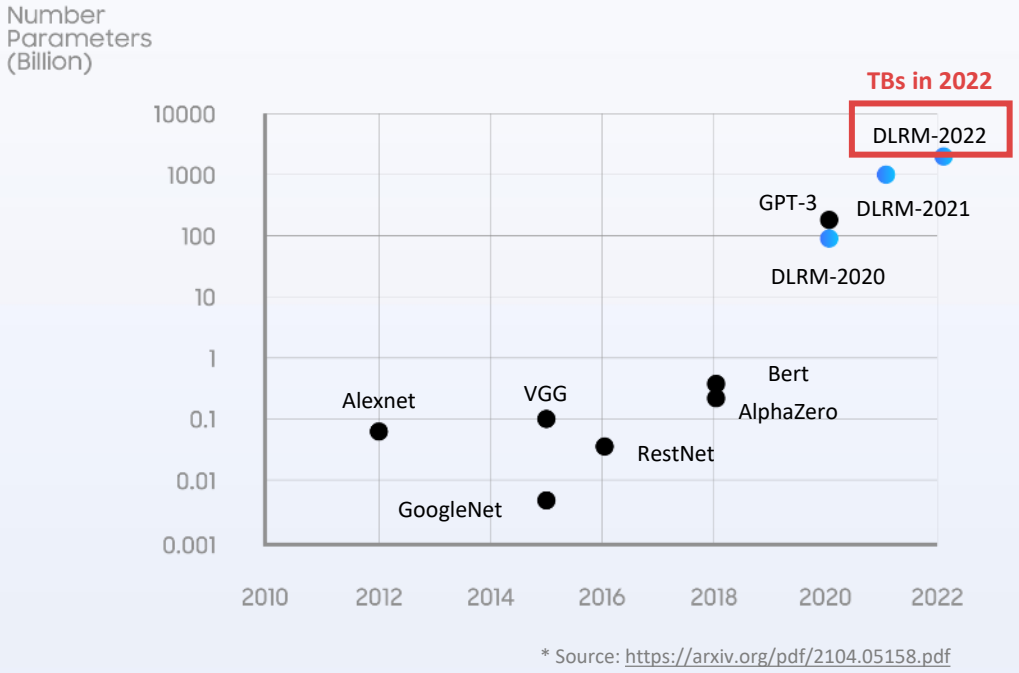
- RecSys, NLP, Video, Medical Imaging, etc.,

NAND flash SSD have better TCO compared to DRAM to store large models

- But NVMe SSDs have higher I/O stack overhead and unnecessary block data movement and copies to DRAM

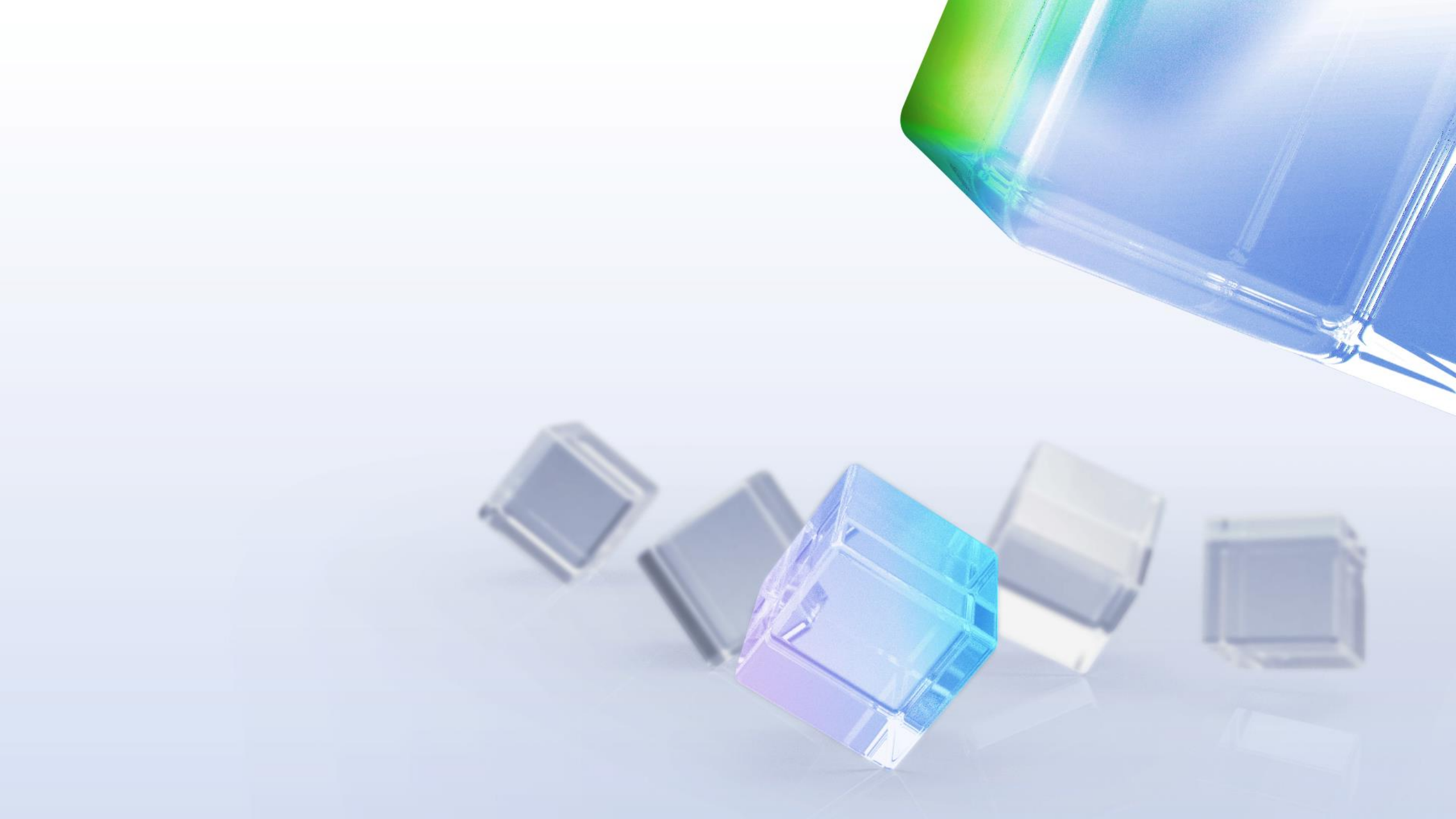
Tiered Memory in AI framework with MS SSD can help solve AI's Storage/Memory Problem

- Enables fine-grained access and hardware caching, reduces data movement costs, simpler software stack, no copies to DRAM necessary to access non-temporal data



* Source: https://arxiv.org/pdf/2104.05158.pdf

Please visit Samsung demo booth (#407) to learn more!



endpoint intelligence

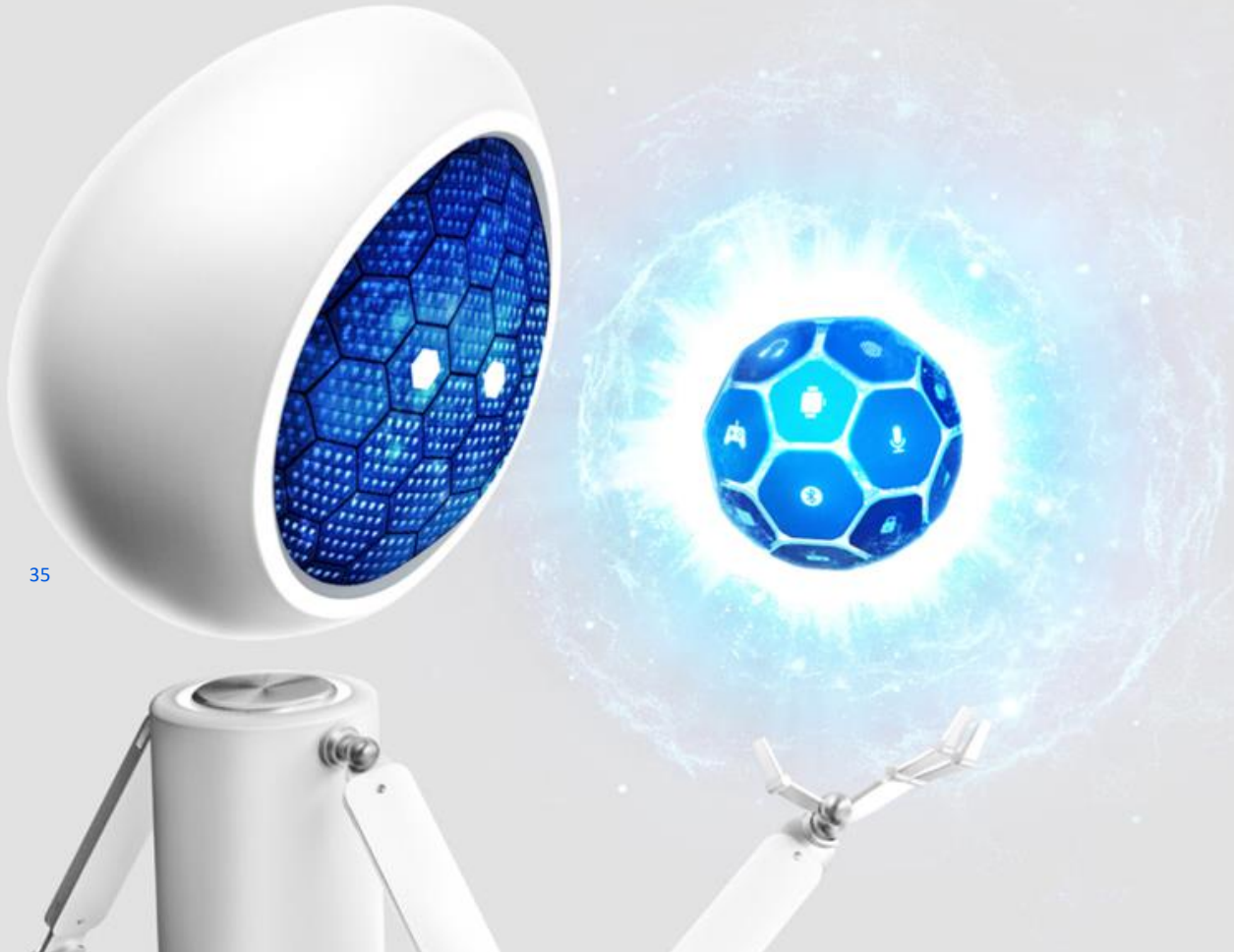
Driving The Future of Intelligent IoT Endpoints

Carlos Morales

Ambiq at FMS 2022

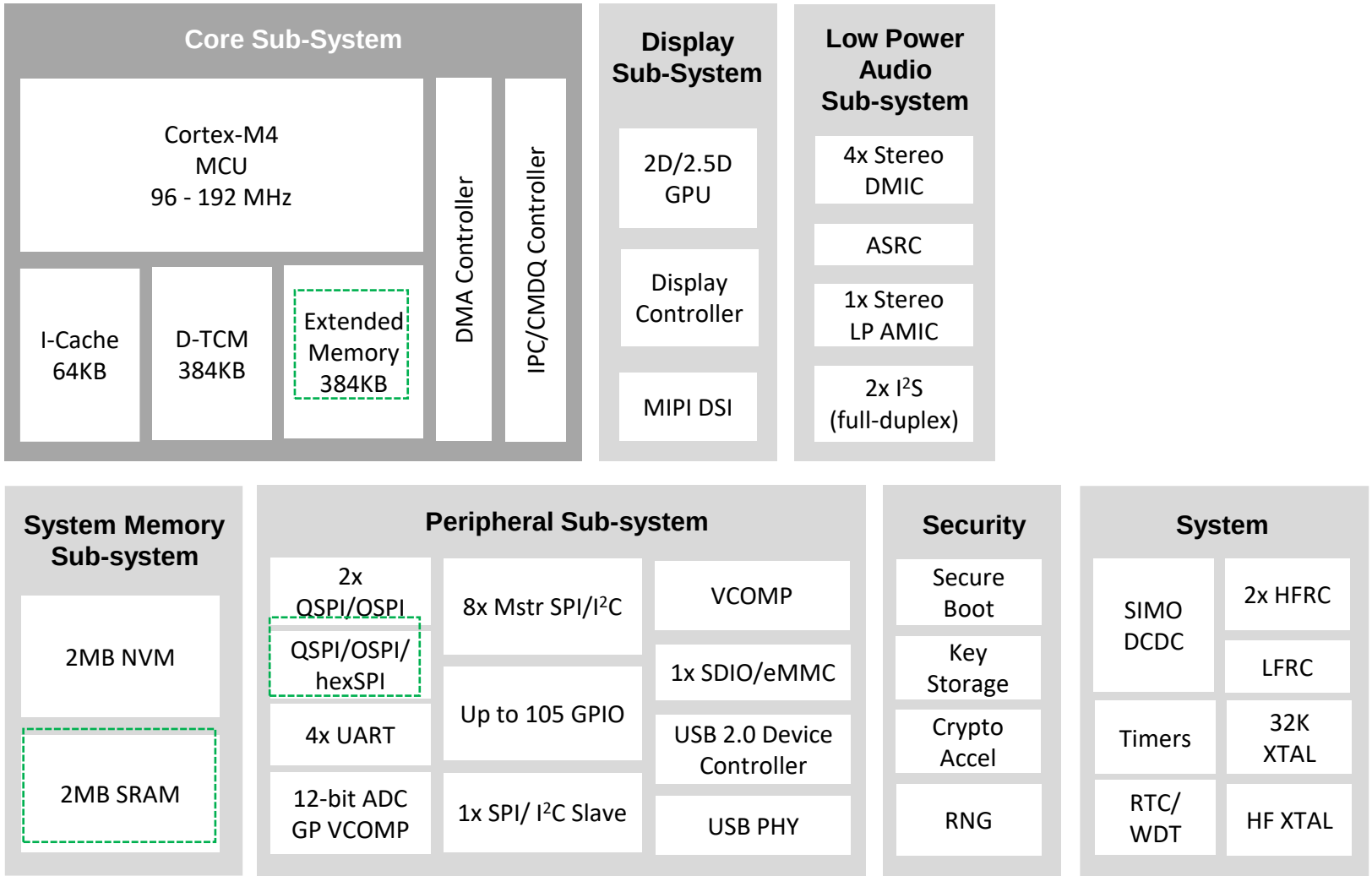
August 4, 2022

The World's Most **Energy Efficient** Solutions



- A pioneer of ultra-low-power semiconductor solutions built on our **patented** and **proprietary** Subthreshold Power Optimized Technology (SPOT®) platform
- SPOT provides a game-changing, **multi-fold improvement** in energy efficiency
- Our integrated SoCs are designed to act as the **brains** for our end-customers' offering
- SPOT fuels further growth by defining and enabling **endpoint AI**

Apollo4 Plus



Feature Highlights

- 4µA/MHz executing from MRAM
- Up to 192 MHz clock frequency with TurboSPOT
- Improved data caching; up to 32 line buffers with programmable read/write data persistence
- 2MB MRAM, 2.75MB SRAM
- 2D/2.5D Graphics Accelerator with anti-aliasing and dithering
- MSPI enhancement with HexSPI
- Display Controller supporting MIPI DSI 1.2 with up to two lanes at 500Mbps
- 4-Layer alpha blending, up to 500x500 resolution with 60fps
- Ultra-low power analog microphone for truly always on voice processing
- 4 PDM stereo channels, 2 full duplex I2S channels with ASRC, and ultra-low power ADC for analog mics

Case Study: Speech Interfaces

KEYWORD SPOTTING (KWS)

- "Ok watch, tell me the time"
- clunky interface, must use specific phrases
- 40KB of memory, 2MOPS
  (mega-ops per second)

SPEECH TO INTENT (S2I)

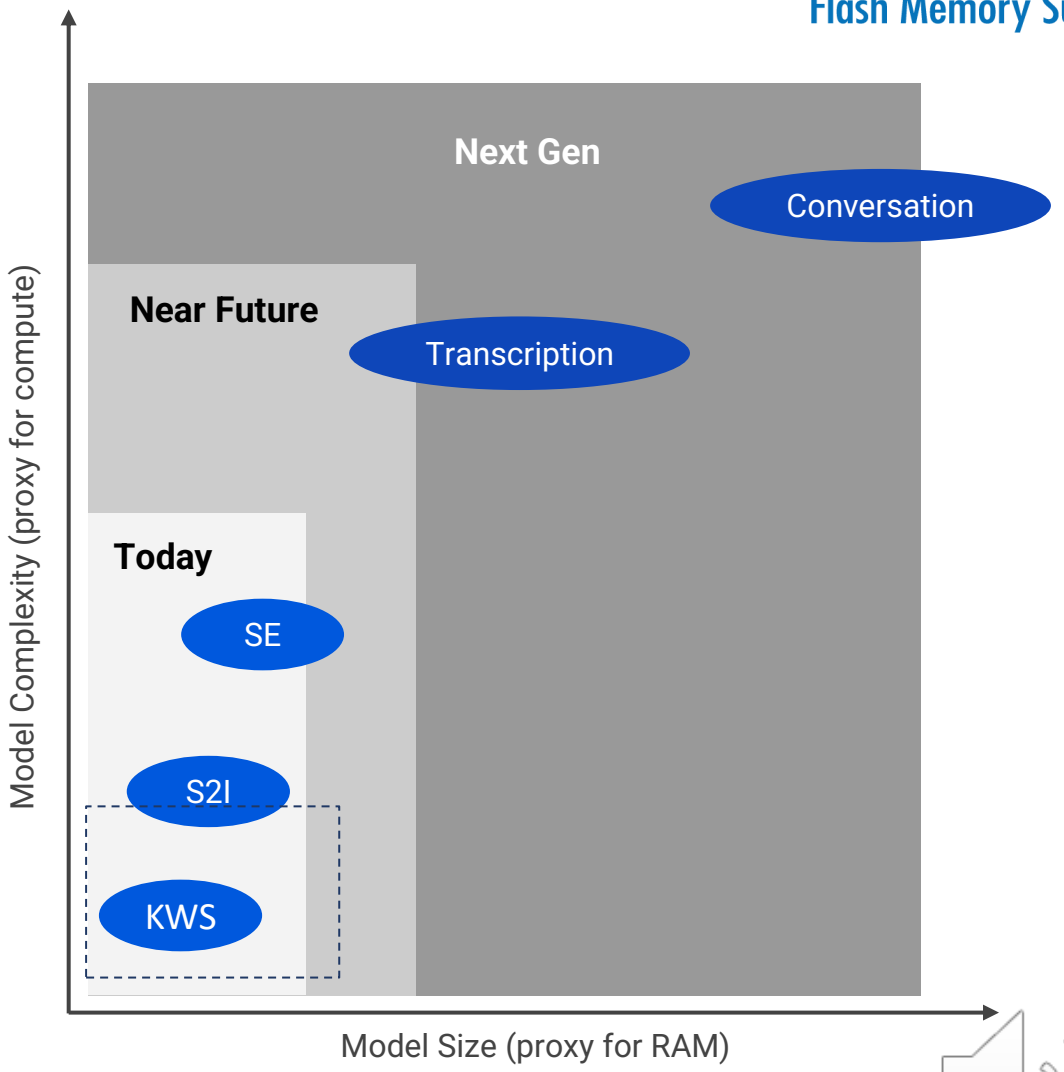
- "What time is it?", "watch, what is the time?", "what is the hour?" in many languages
- 100KB of memory, 3.5MOPS

SPEECH ENHANCEMENT (SE)

- Noisy speech to clean speech
- 310KB of memory, 5MOPS

NEAR FUTURE: NATURAL LANGUAGE UNDERSTANDING (NLU and NLP)

- "Schedule a meeting from 5pm to 7pm on Tuesday, and invite Sara"
- Dictation, translation, conversations
- 25MB-100MB of memory, 6-10**G**OPS



37

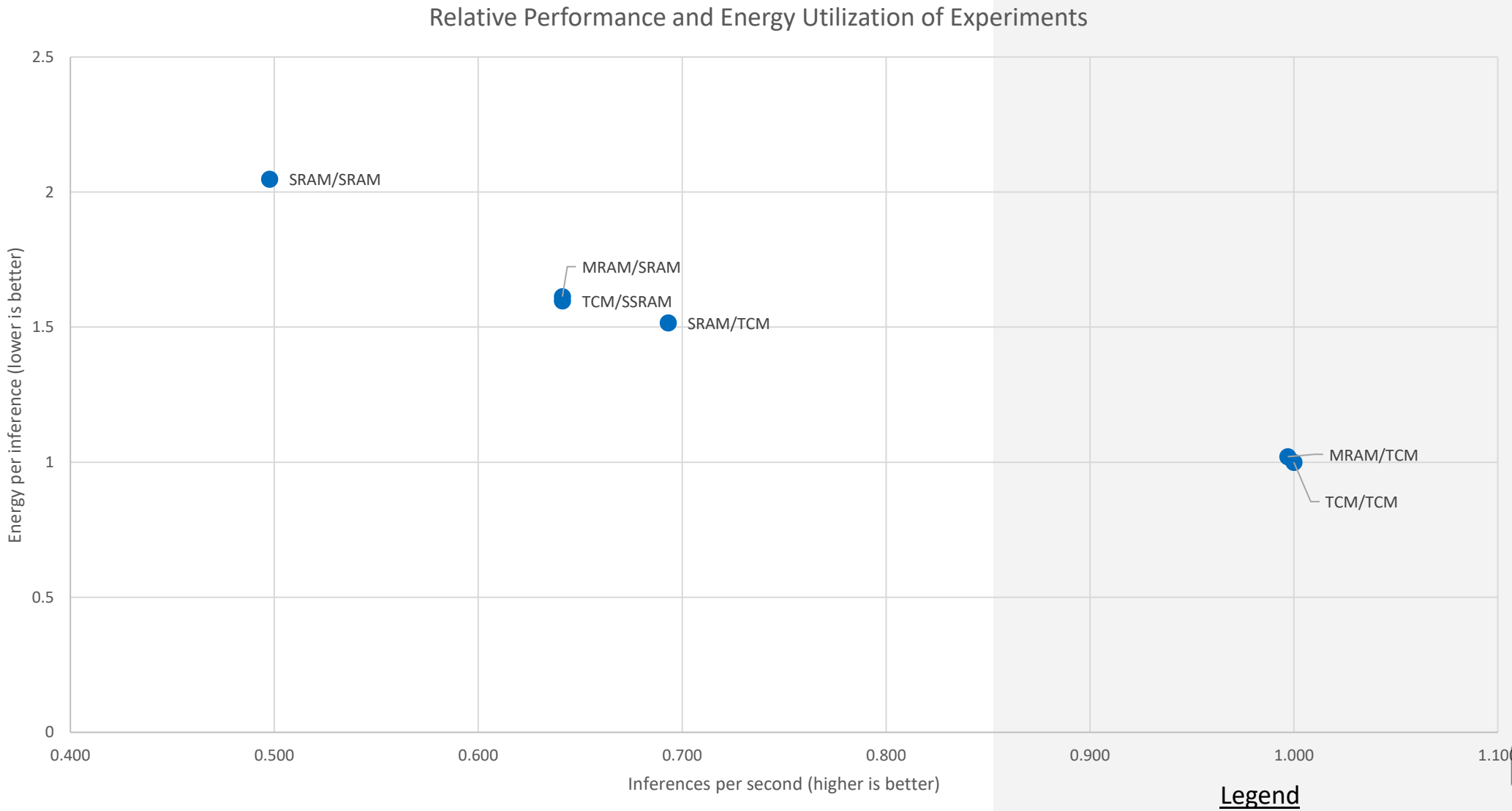
AI Developers Have Options

TCM	MRAM	SSRAM
Tightly Coupled Memory	Magnetoresistive RAM	Shared SRAM
Fast SRAM close to the CPU	Used as RO memory	Higher latency shared RAM

For each combination, our experiment measured
latency and **energy**
using the MLPerf Keyword Spotting Benchmark

Weights Stored In…	Activations Stored In…	Optimization Options
TCM	TCM	Turn off SSRAM
TCM	SSRAM	Turn off half of TCM
MRAM	TCM	Turn off SSRAM
MRAM	SSRAM	Turn off half of TCM
SRAM	TCM	
SRAM	SSRAM	Turn off half of TCM

Bottom Line – MRAM is As Good As TCM



Legend

"weight/activation"

My Conclusions From This Experiment

- **It's good to have options!**
  - Maybe you have a lot of models swapping in and out: TCM or SSRAM are good options
  - Maybe MRAM and TCM are reserved for other tasks: SSRAM is more than sufficient for most AI
- **I use MRAM and TCM when I can – best power and performance**
- **AmbiqSuite makes experimenting easy – usually 1-2 lines of code is all it takes**

endpoint intelligence

Thank You!


Questions

Q&A

Flash Memory Summit

Please take the Session Survey
Thank you!

Tuesday, August 2

Wednesday, August 3

Thursday, August 4

35 YEARS OF NAND FLASH MEMORY INNOVATION