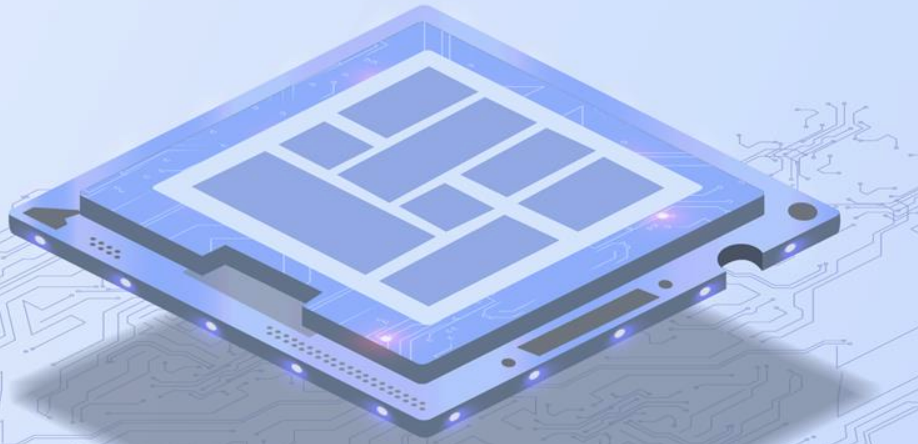
CacheCraft

Enhancing GPU Performance under Memory Protection through Reconstructed Caching



Soyoung Park†, Hojung Namkoong†, Boyeol Choi†, Michael Sullivan‡, Jungrae Kim†

†Sungkyunkwan University (SKKU), ‡NVIDIA

Outline

Demands for GPU Reliability

GPU reliability challenge

- A large-scale field study [1] indicates
  - NVIDIA A100 GPUs show 14,000-hour MTBF (Mean Time Between Failures)
  - A large-scale cluster (992 A100s) shows 14-hour MTBF

Major source of GPU failures: memory errors

- Another large-scale field study [2] indicates
  - HBM2 caused ~260,000,000 error events in 19 data centers over 2 years.

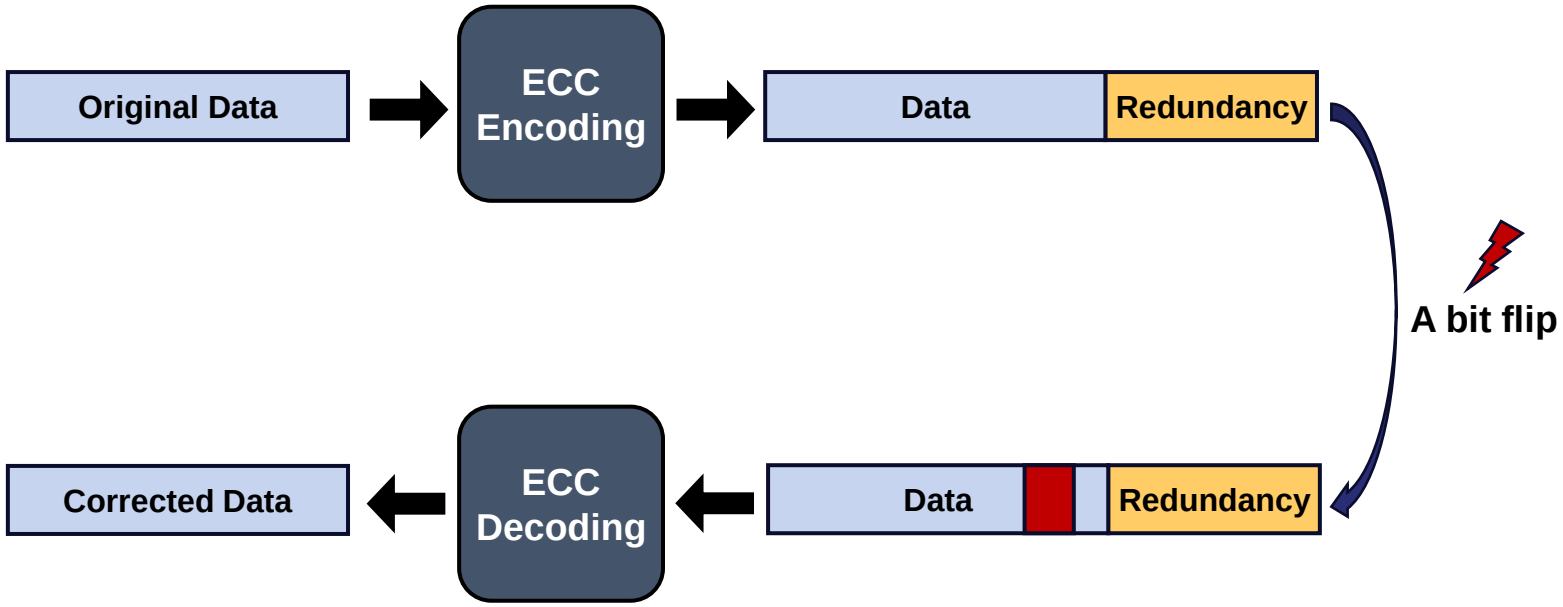
[1] Zhang, S., et al (2022). *OPT: Open Pre-trained Transformer Language Models*. in arXiv.

[2] Wu, R., et al (2024). "Removing Obstacles before Breaking Through the Memory Wall: A Close Look at HBM Errors in the Field." in *ATC*. USENIX Association.

Error Correcting Codes (ECC)

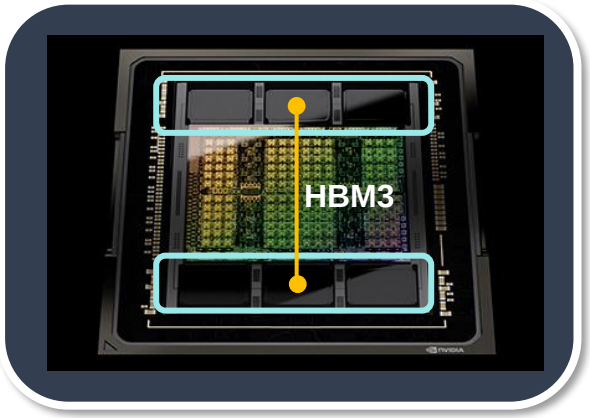
Modern GPUs utilize ECC to safeguard data

- ECC can detect and correct erroneous bits using redundancy.



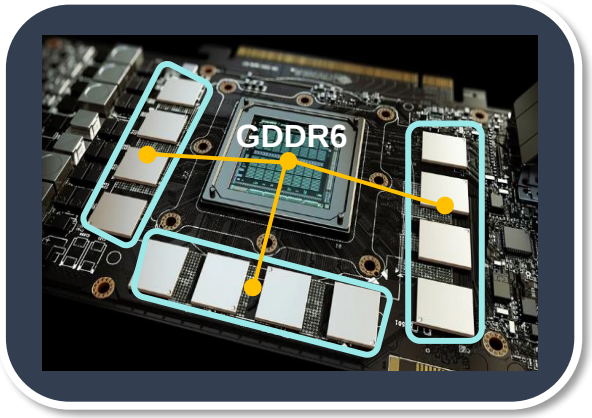
HBM and GDDR with different trade-offs

HBM



[SRC : NVIDIA]

GDDR

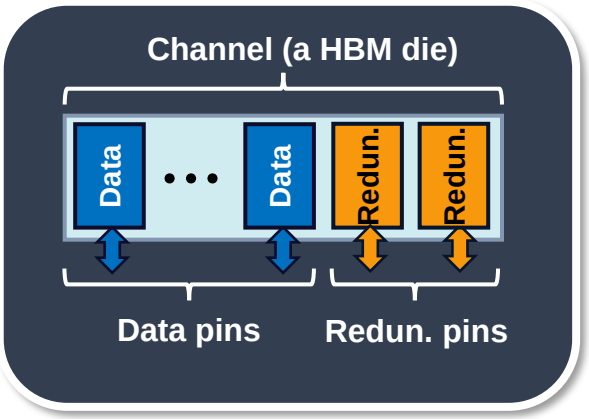


[SRC : NVIDIA]

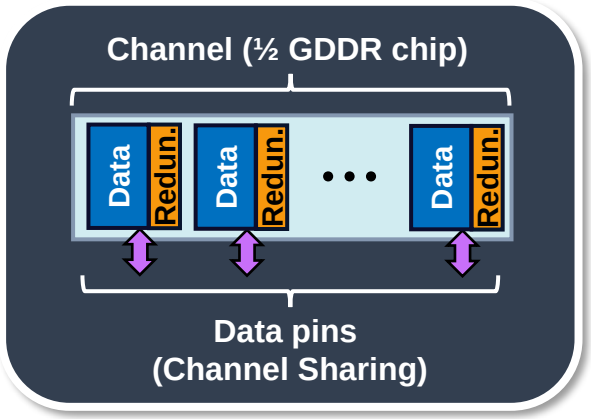
Target GPU	Flagship data center GPUs	Other data center GPUs Consumer GPUs
Target application	HPC and large-scale AI models	Medium- and small-scale AI models
Bandwidth	Provide state-of-the-art bandwidth (E.g. NVIDIA H100 with 3.3 TB/s)	Suitable for most mainstream apps (E.g. NVIDIA L40 with 0.86 TB/s)
Accessibility & costs	Limited supply due to high cost and packaging complexity	More accessible and cost-effective

Two types: Side-band ECC and in-band ECC

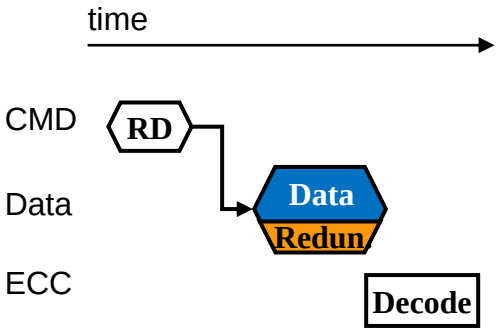
Side-band ECC (for HBMs)



In-band ECC (for GDDRs)

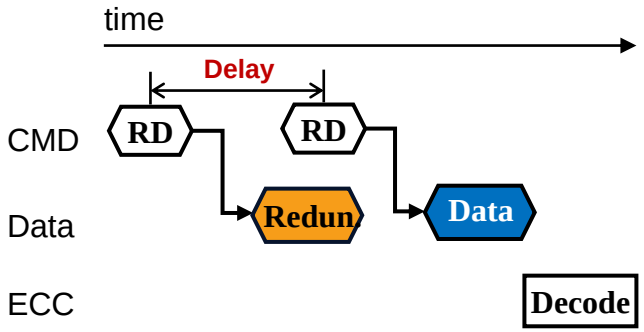


With built-in ECC support



Parallel transfer

Without built-in ECC support



Serial transfer

Costs of In-band ECC

The serial transfer of data and redundancy

- Consumes data throughput
  - Approximately **20% bandwidth reduction** (NVIDIA Documentation[1])
  - Up to **59.5% bandwidth reduction** (our field measurement)
- Degrades system performance
  - Up to **54.5% slowdown** on NVIDIA Tesla K40C[2]

We need a more efficient in-band ECC implementation

[1] NVIDIA Corp, “Tuning CUDA Applications for Pascal," [Online]. Available: https://docs.nvidia.com/cuda/pascal-tuning-guide/index.html.

[2] G. Juckeland et al., "Spec accel: A standard application suite for measuring hardware accelerator performance," in *HPCA*.

Outline

GPU Sector Cache

GPUs utilize

- 128B cache lines
  - Can serve a 128B coalesced request from 32 threads within a warp
  - Sometimes lead to memory over-fetching and bandwidth wastage

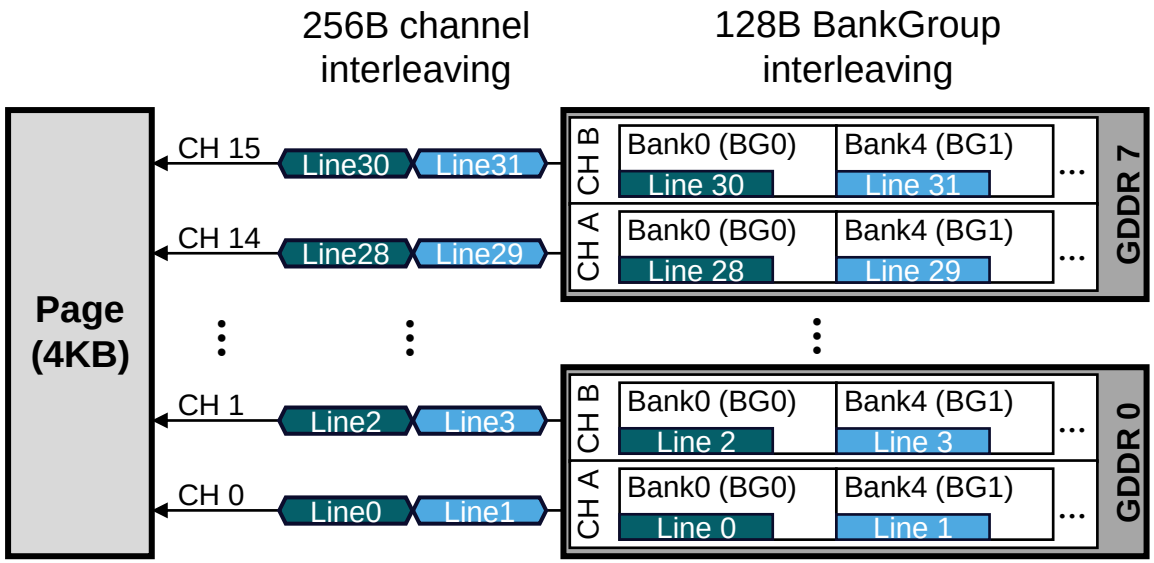
- Sector cache
  - Divides a cache line into smaller segments (sectors)
  - Reduces over-fetching while maintaining low tag overhead

A 128B cache line

Tag	Sector0	Sector1	Sector2	Sector3
A	32B	32B	32B	32B

GPU Memory Interleaving

- **GPUs employ fine-grained memory interleaving**
  - 256B channel interleaving
  - 128B BankGroup interleaving
- **Interleaving enhances load balancing but degrades row buffer locality**
  - **Each bank houses only a single cache line (128B) from a 4KB page (in GPUs with 8 GDDRs)**

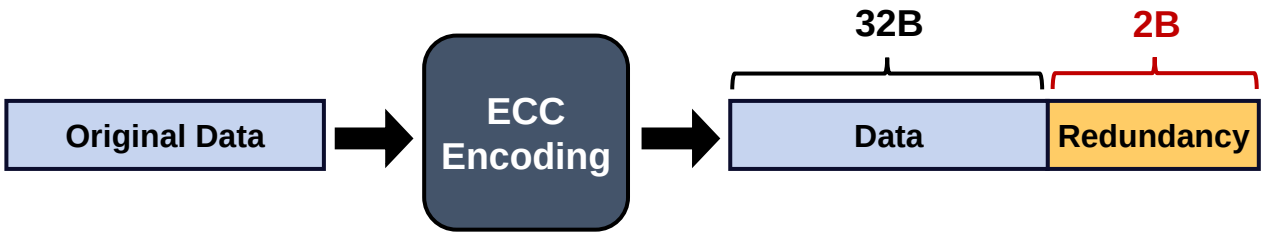


Outline

GPU Error Correcting Codes

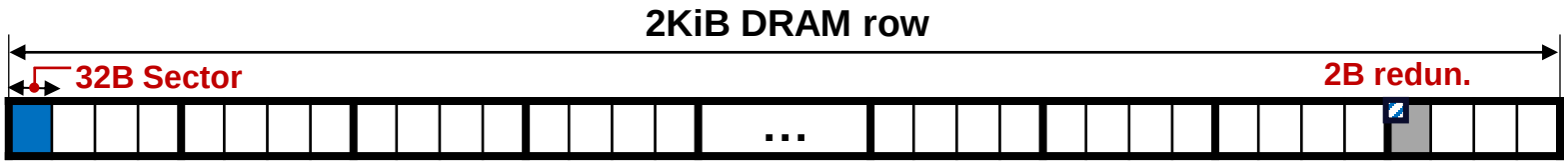
SEC-DED on a 32B block (sector)

- Single Error Correction – Double Error Detection
- 6.25% redundancy (2B redundancy for every 32B data)

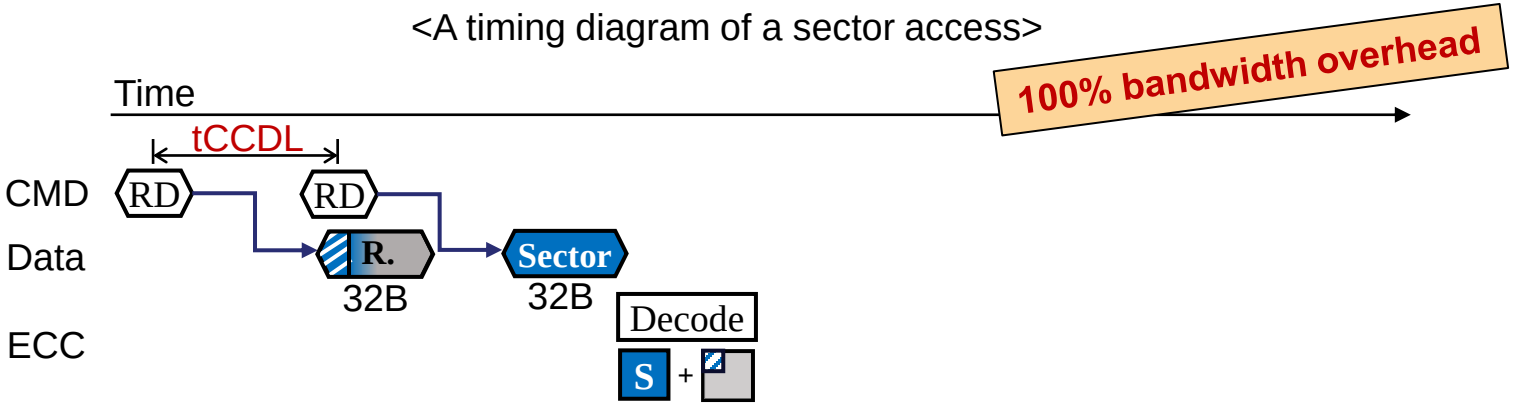


Single Sector Access with In-band ECC

- Data and redundancy in the same row



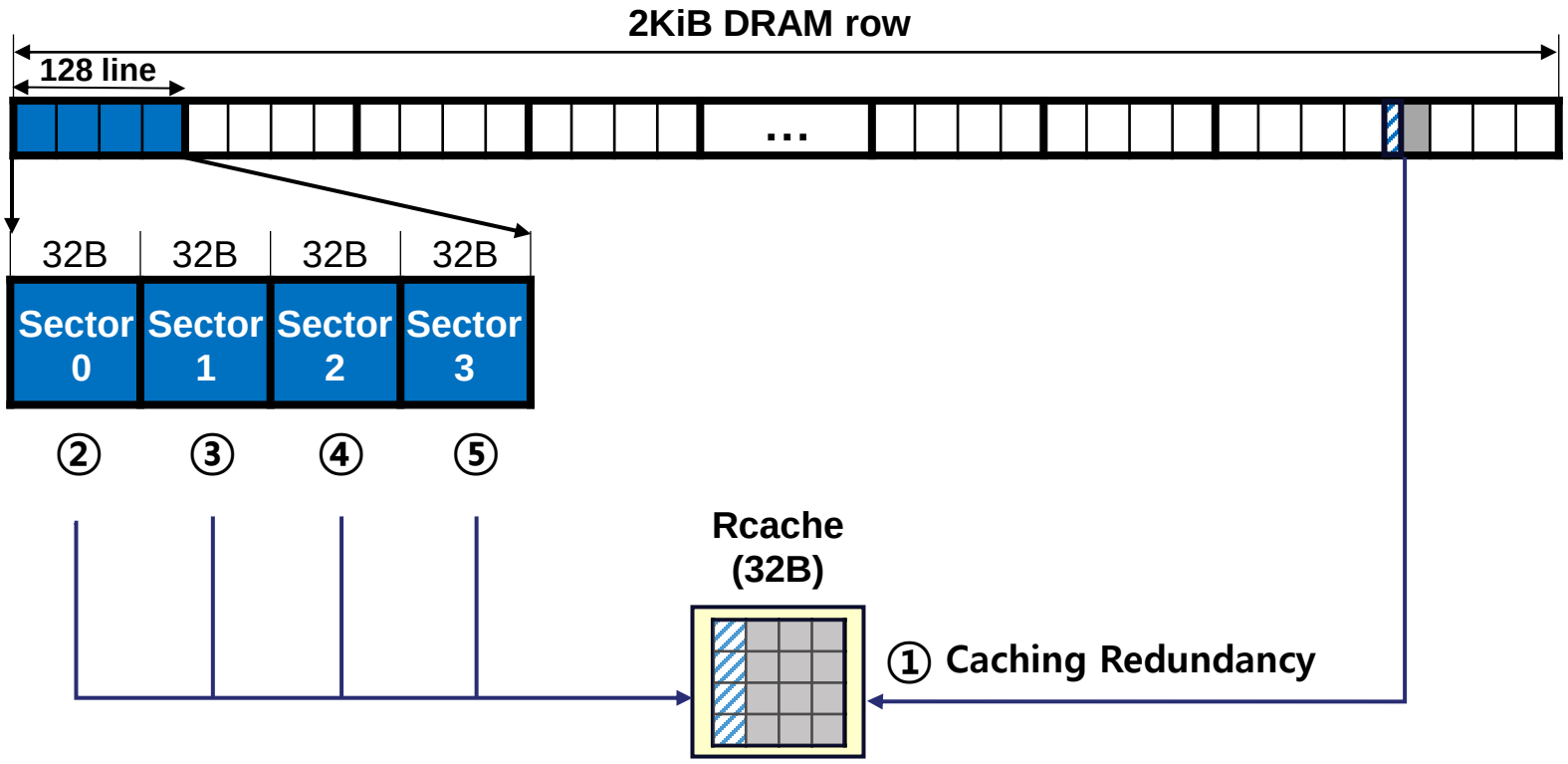
<A timing diagram of a sector access>



Single Line Access with In-band ECC

Rcaches (Redundancy Caches) can alleviate BW penalty

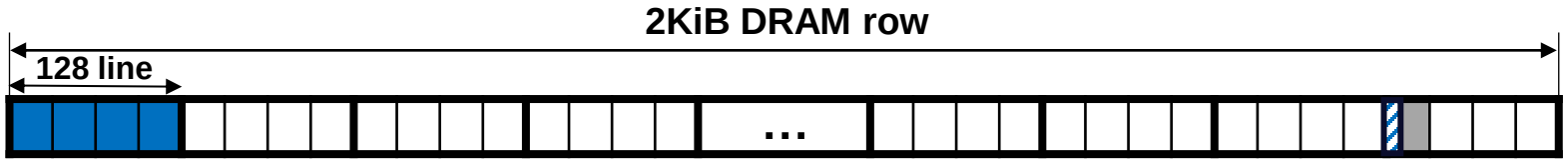
- Reduces # of memory accesses of full cache line requests (8 to 5)
  - Four for data sectors, and the other for the shared redundancy blocks



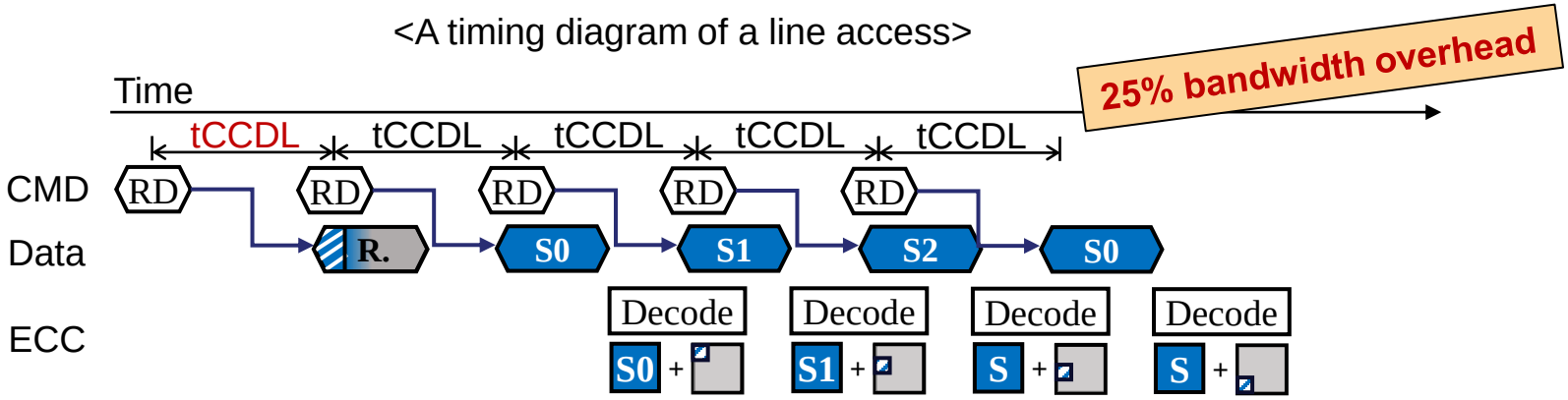
Single Line Access with In-band ECC

Rcaches (Redundancy Caches) can alleviate BW penalty

- Reduces # of memory accesses of full cache line requests (8 to 5)
  - Four for data sectors, and the other for the shared redundancy blocks



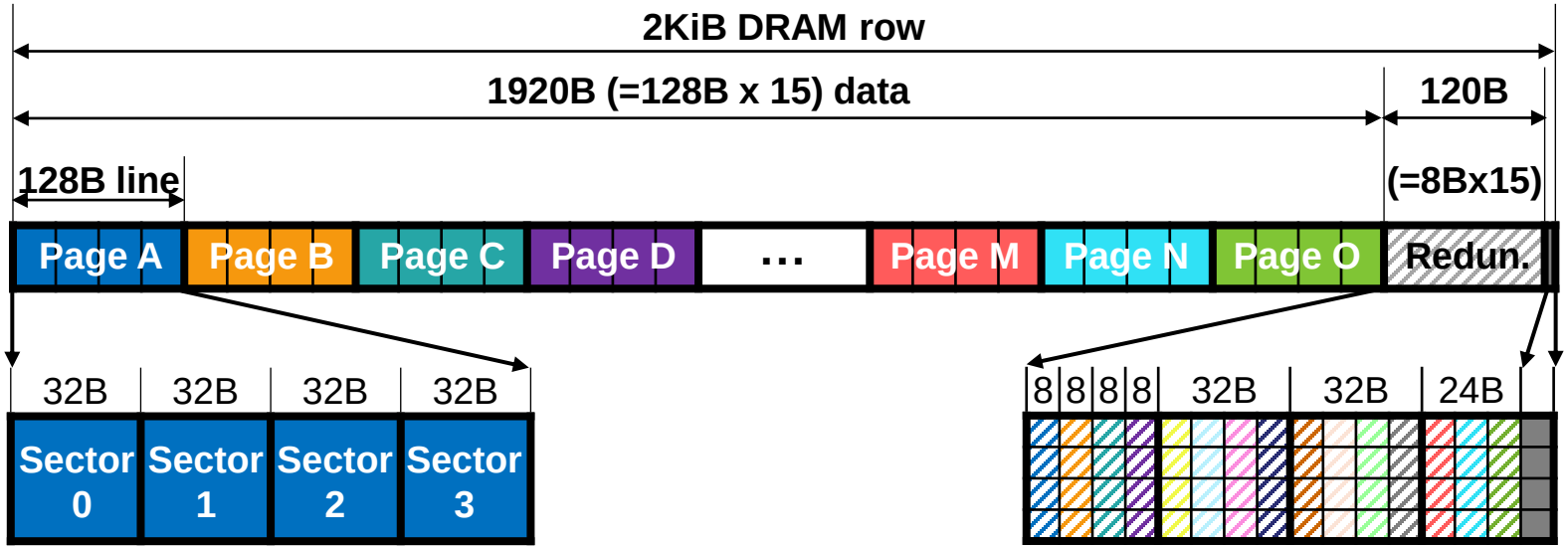
<A timing diagram of a line access>



Multiple Line Access with In-band ECC

Poor row buffer locality limits the efficacy of RCache

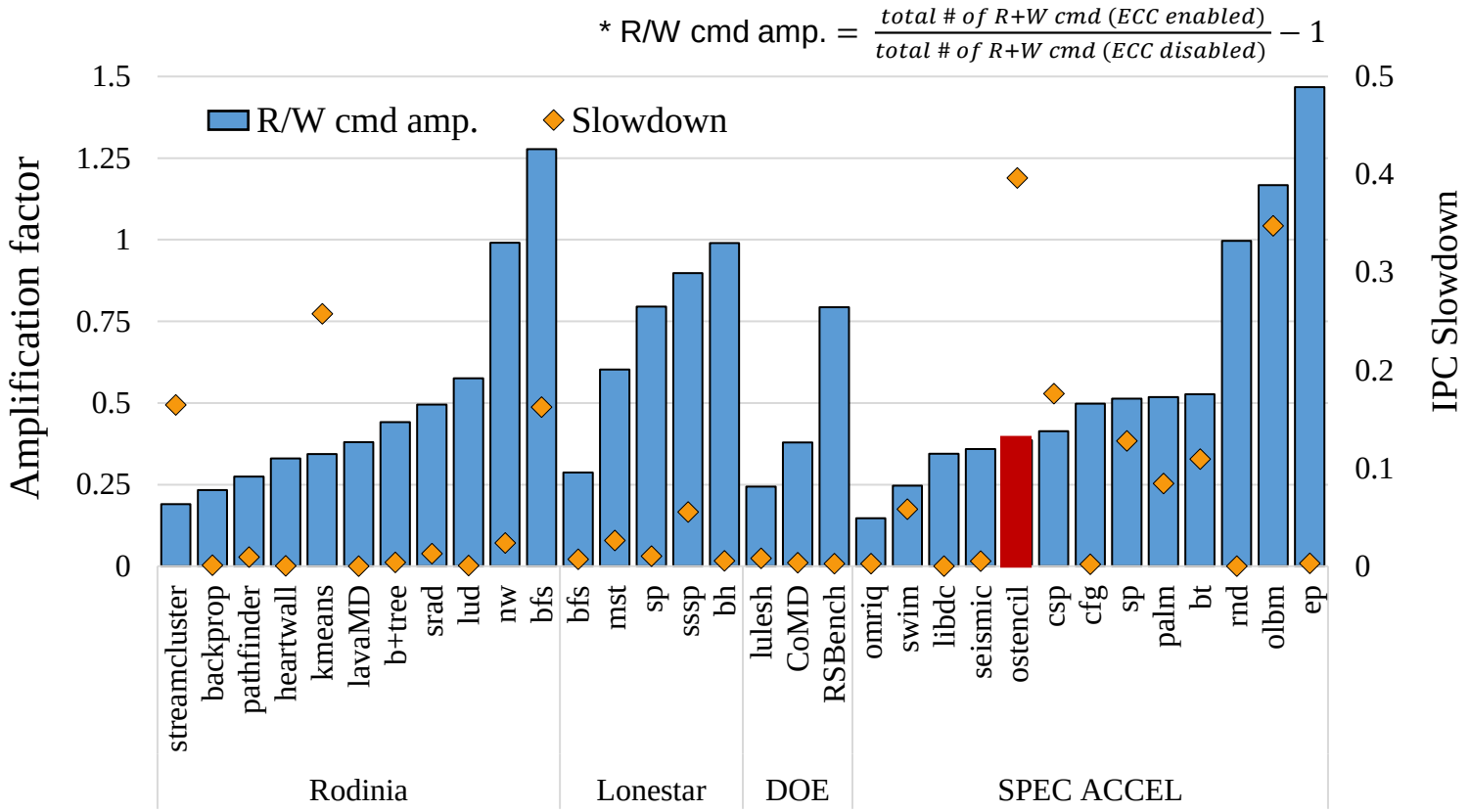
- Only 8 bytes in the redundancy block are adjacent to each other
- Other 24 bytes belong to other pages or are distant by at least 4KiB



Evaluation of In-band ECC Costs

Empirical evaluation on NVIDIA T4 GPU with ECC enabled/disabled

- Out of 32 benchmarks, 27 exceed 25% memory traffic amplification with 14 over 50%
- Impact of excessive transfers vary across applications

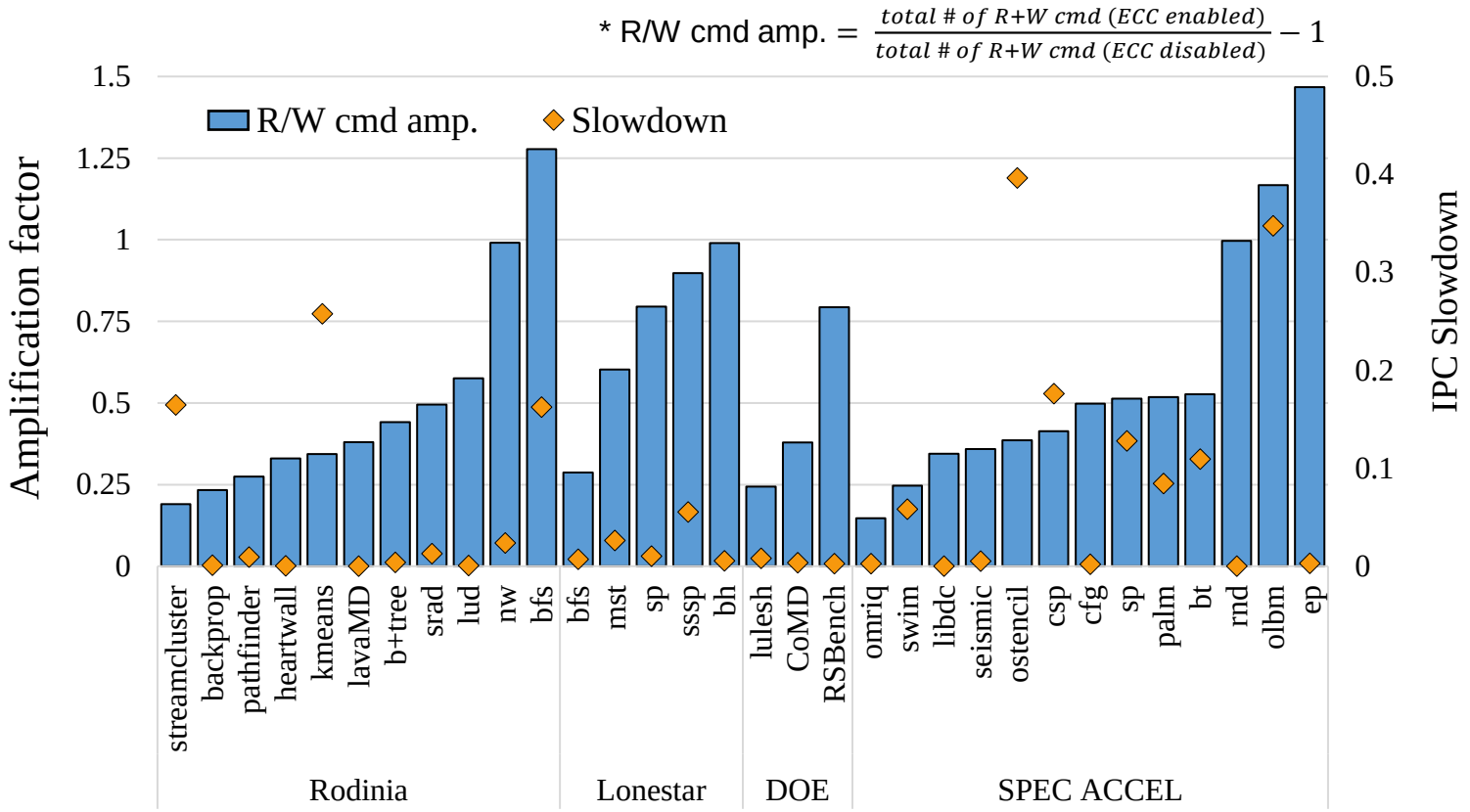


Evaluation of In-band ECC Costs

Empirical evaluation on NVIDIA T4 GPU with ECC enabled/disabled

Existing in-band ECC necessitates additional memory access for redundancy

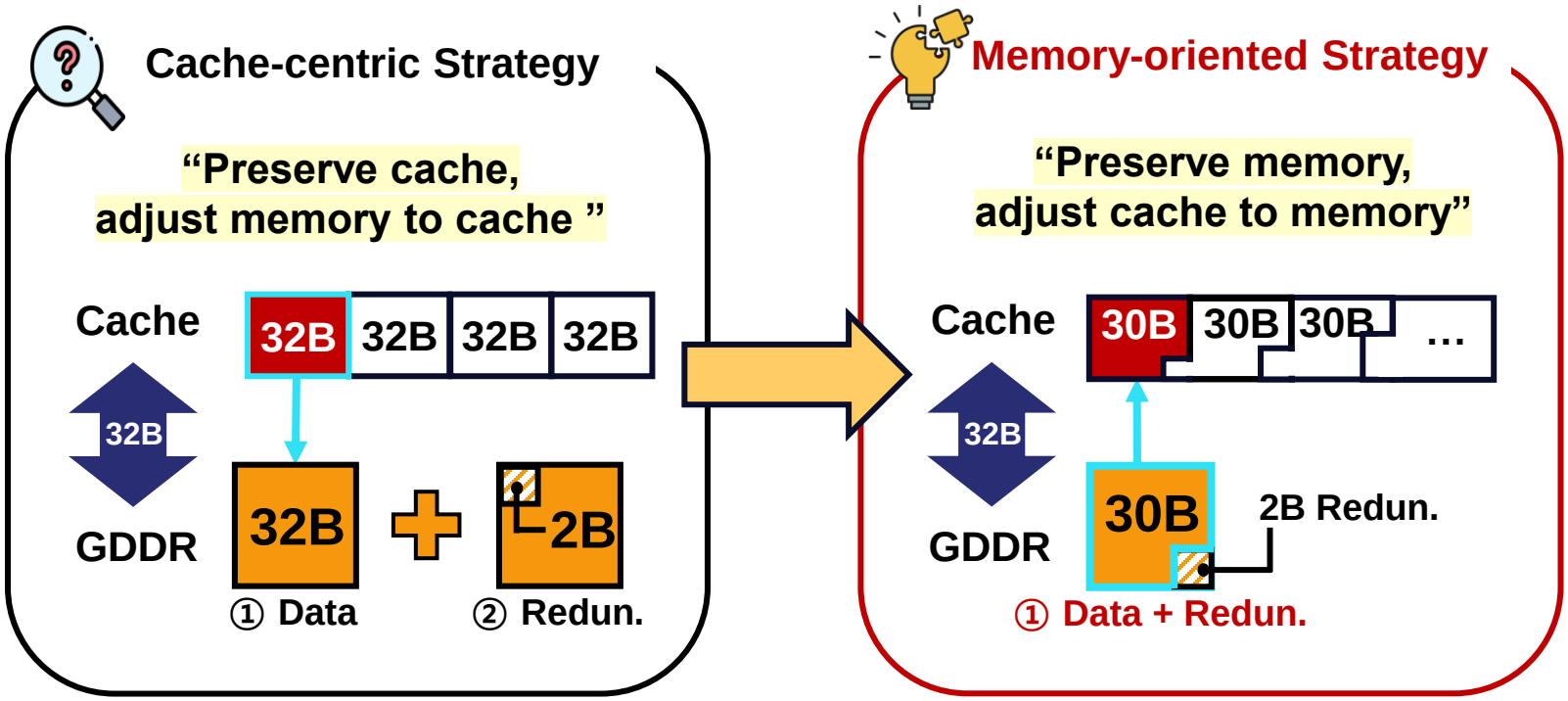
⇒ Bandwidth consumption ↑, Performance ↓



Outline

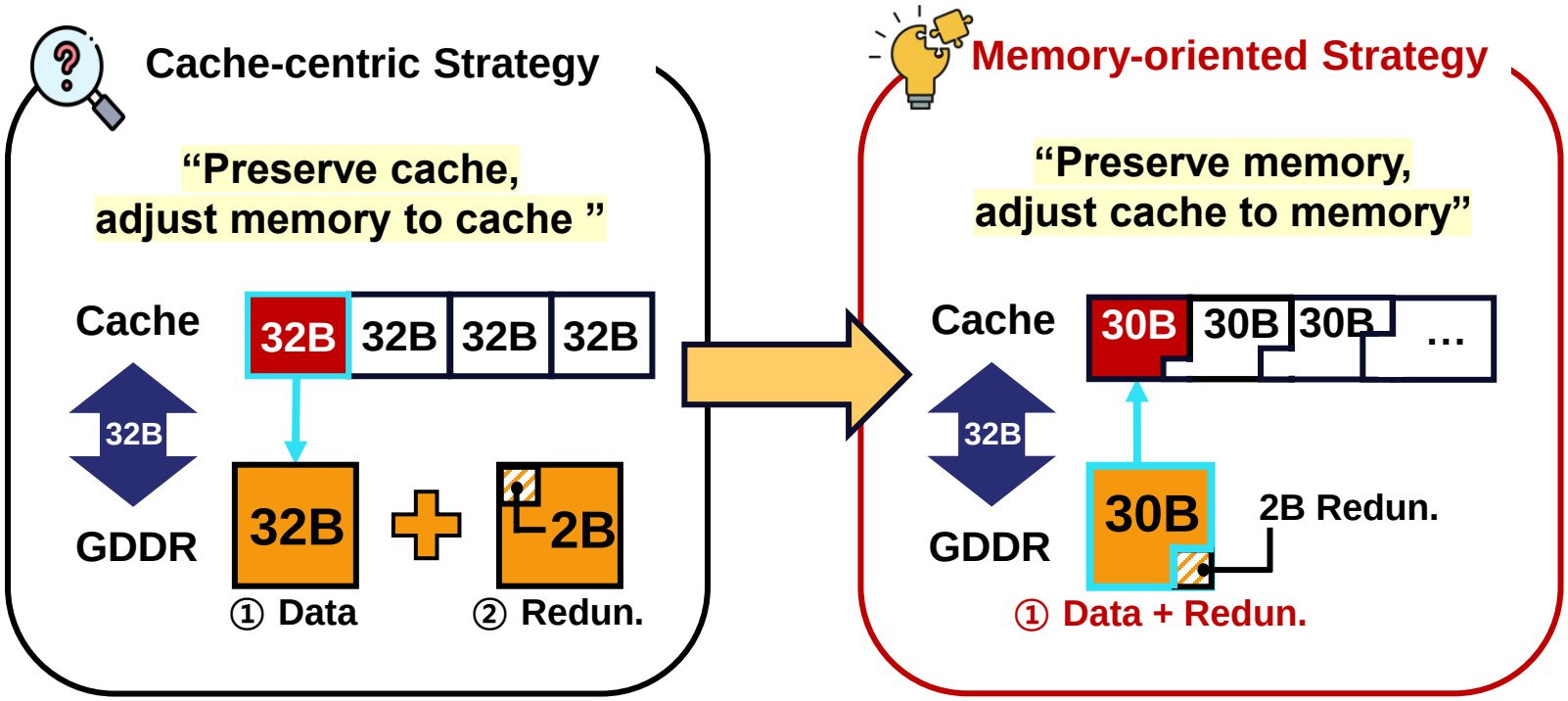
CacheCraft - Overview

Let's think outside the box



CacheCraft - Overview

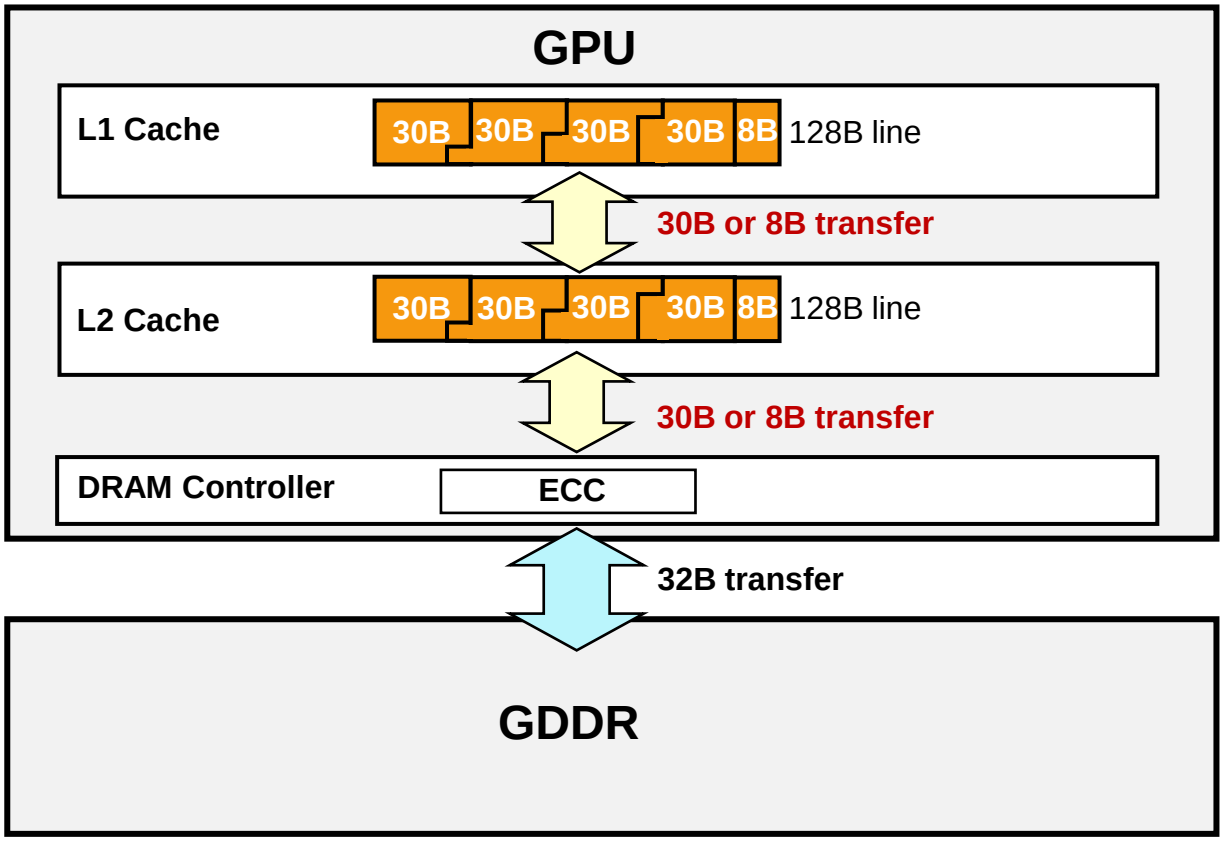
We can fetch both data (30B) and redun. (2B) in a single 32B memory access !



CacheCraft - Overview

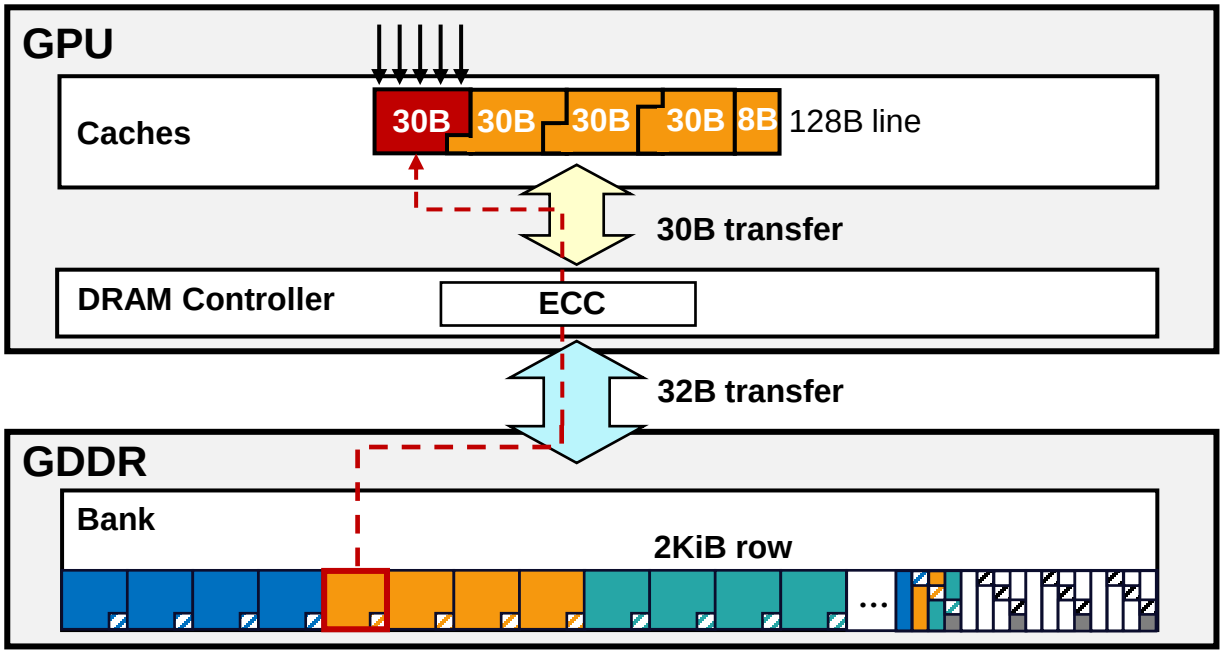
Reconfigures 128B cache lines to "30-30-30-30-8"

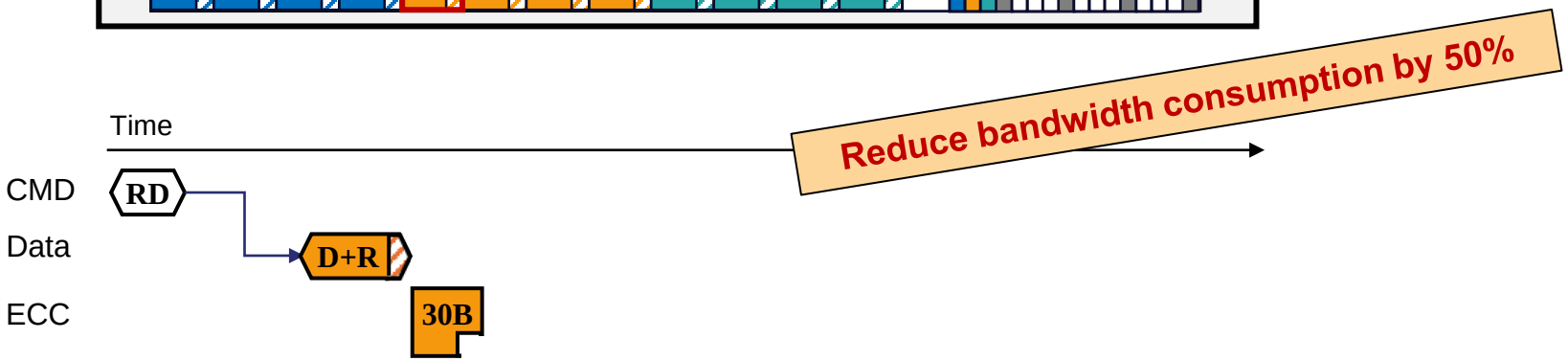
- **30B sectors:** A 32B memory chunk contains 30B data and the corresponding 2B redun
- **8B sectors:** Complements 30B sectors within the 128B line



CacheCraft – Single 30B Sector Access

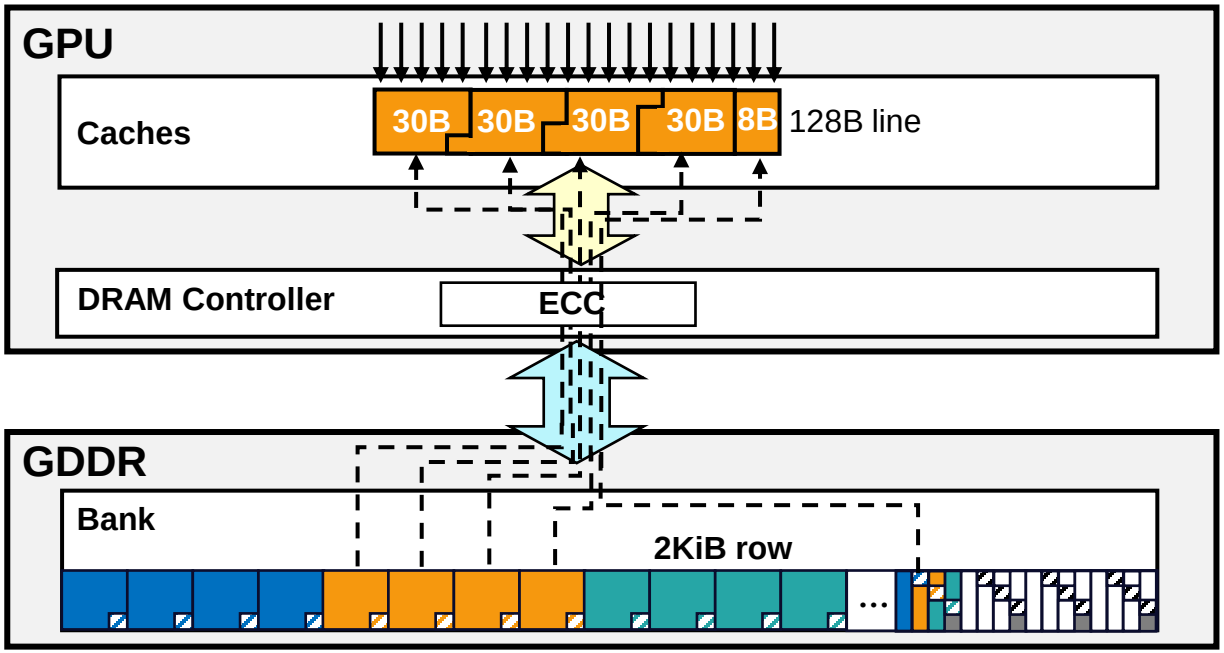
- For requests colaesced into a 30B sector

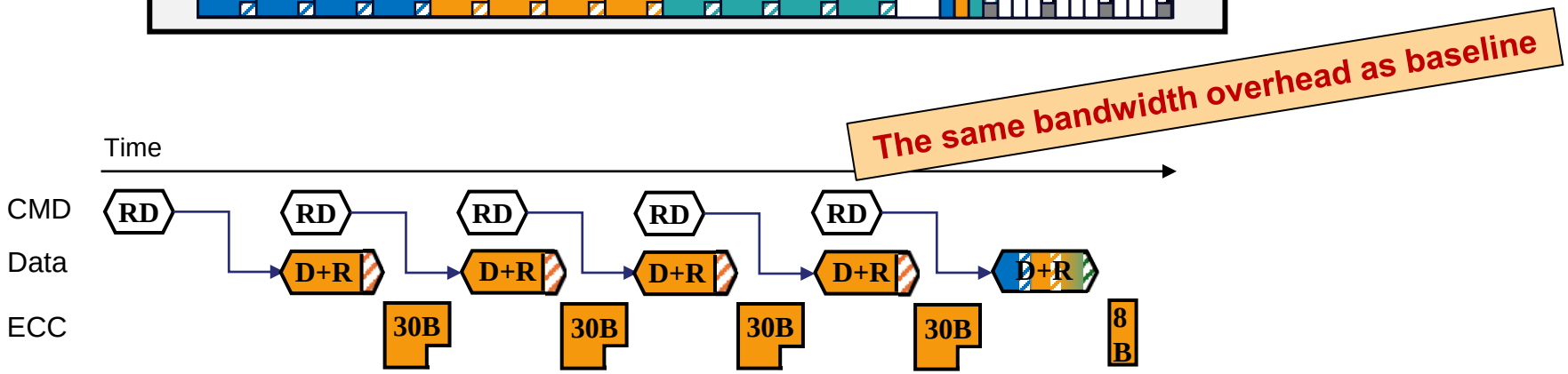




CacheCraft – Single Line Access

- For a full line access





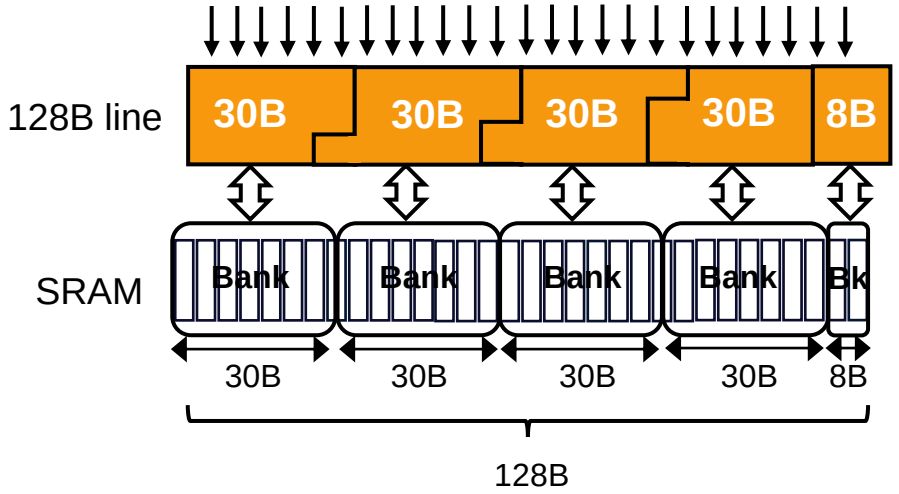
CacheCraft – Modifications

Software

- No change (the cache line size remains the same)

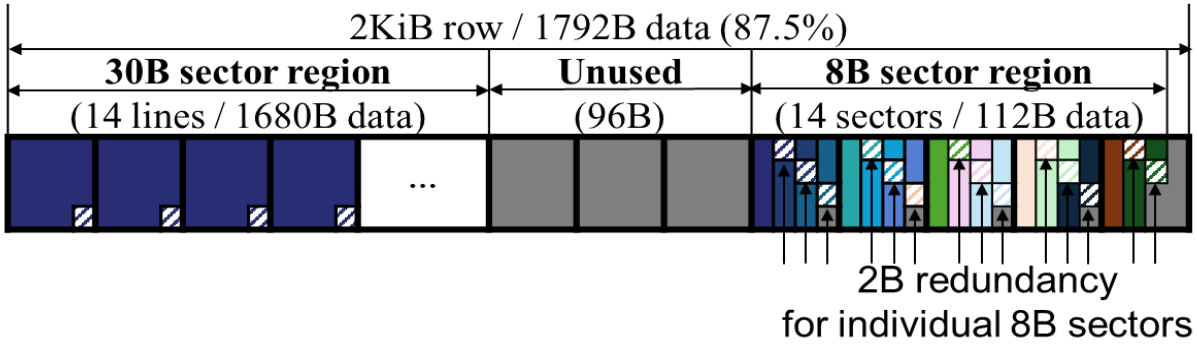
Hardware

- Memory Coalescer
  - To match with “30-30-30-30-8” sectoring scheme
- Caches
  - Sector matching logic
  - Adjusts SRAM banking

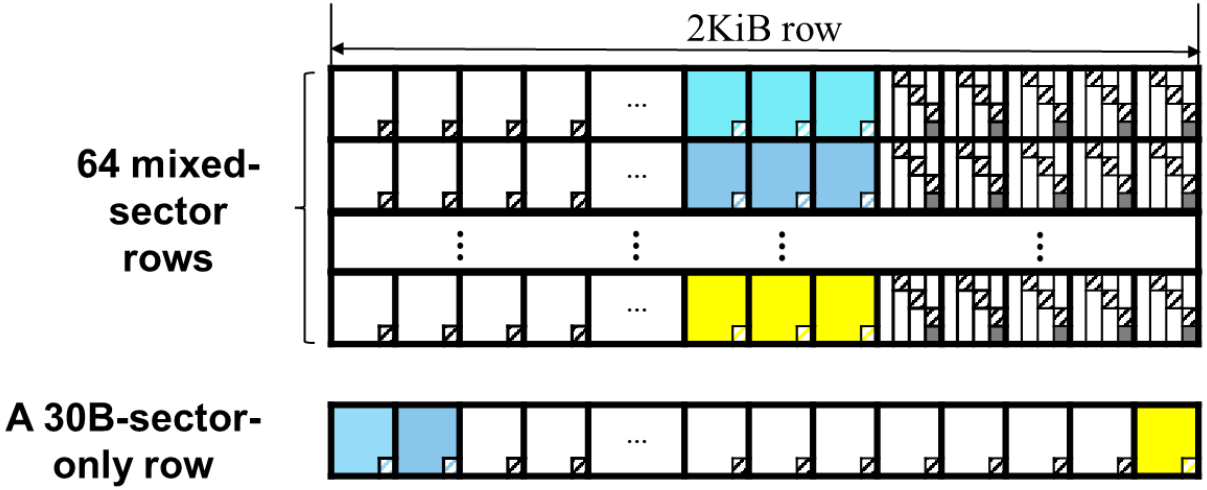


CacheCraft – Modifications

- Memory layouts
  - Simple: unused capacity to meet 30-30-30-30-8

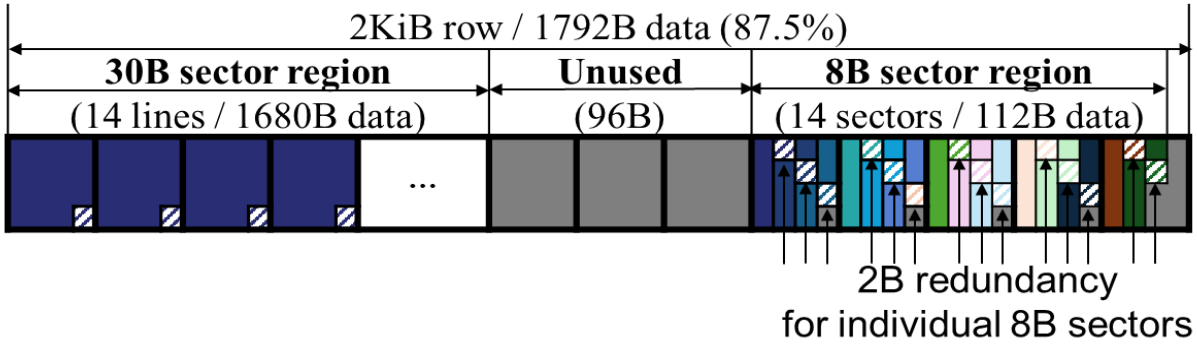


  - Balanced between capacity and bandwidth

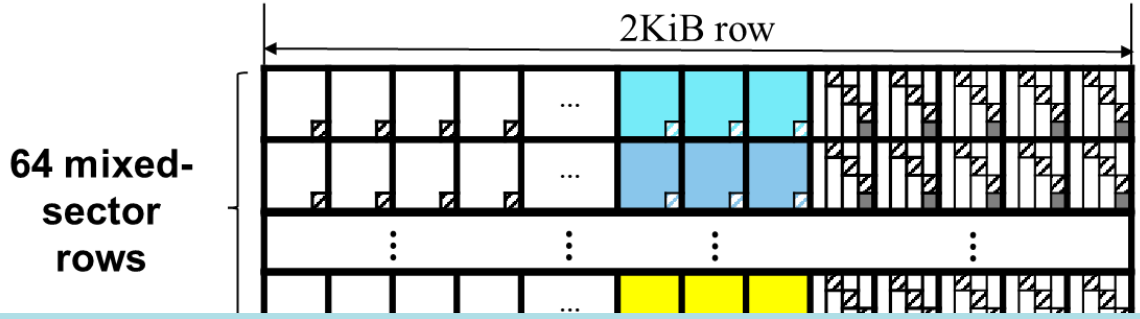


CacheCraft – Modifications

- Memory layouts
  - Simple: unused capacity to meet 30-30-30-30-8



  - Balanced between capacity and bandwidth



Further details are presented in the paper !

Outline

Evaluation Environment

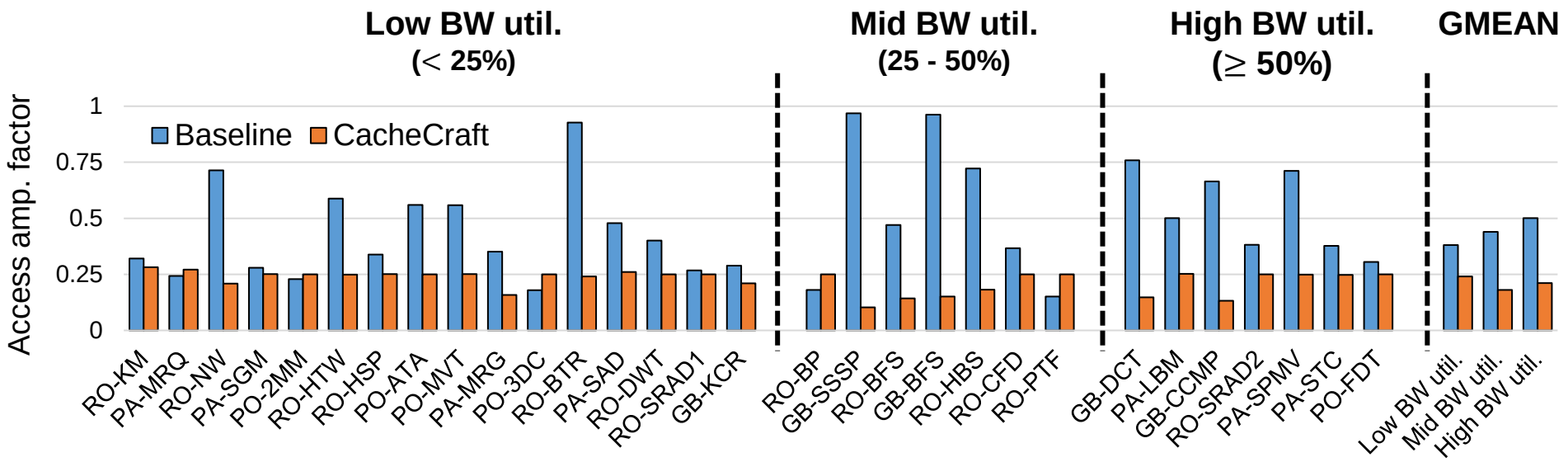
- Accel-Sim with NVIDIA RTX 3070 config
- Benchmarks: Rodinia (RO), Parboil (PA), Polybench (PO), GraphBig (GB)
- We analyze 3 schemes

Non-ECC	- No ECC performance overheads - Caches have 32-32-32-32 sectors
Baseline	- Current in-band ECC implementations in NVIDIA - Caches have 32-32-32-32 sectors - Includes one 32B RCache per GDDR bank
CacheCraft	- Caches have 30-30-30-30-8 sectoring schemes - Memory-layout: Balanced layout

Memory Access

Access amplification factor (compared to Non-ECC)

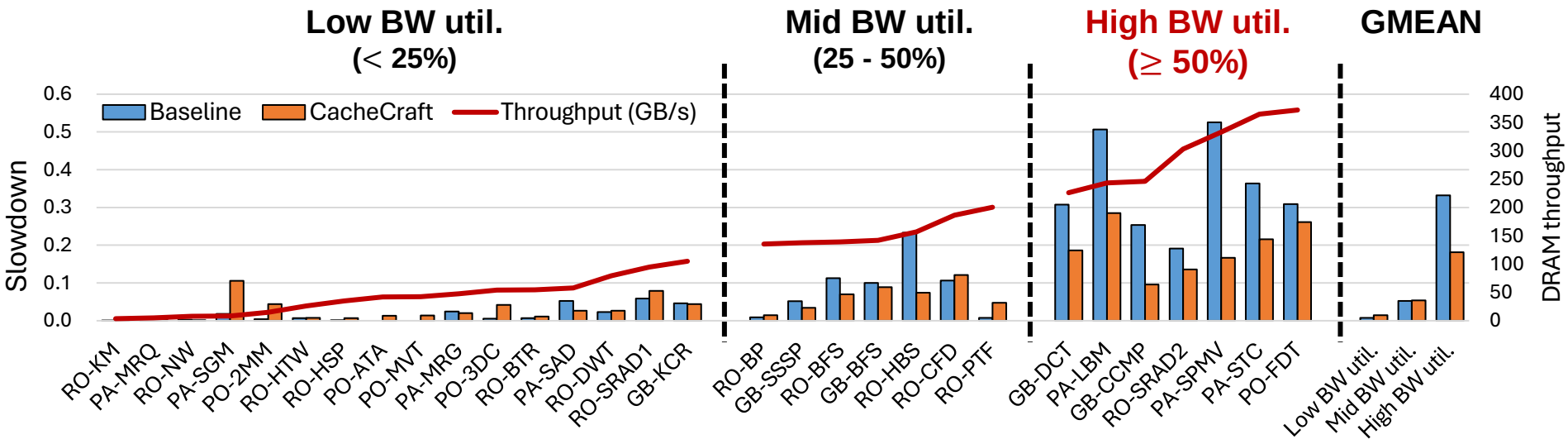
- Baseline: **overall 41.9%**, ranges from **15.2%** (RO-PTF) to **96.9%(GB-SSSP)**
- CacheCraft: **overall 21.9%**, ranges form **10.2%(GB-SSSP)** to **28.2%** (RO-KM)



⇒ **CacheCraft can reduce memory access overhead by up to 89.4% (GB-SSSP)**

IPC slowdowns (normalized to Non-ECC)

- Baseline: 33.2% (average), **52.6% (worst: PA-SPMV)**
- CacheCraft: 18.1% (average), **16.6% (worst: PA-SPMV)**



⇒ CacheCraft can enhance the performance of memory-intensive applications by up to 23.5% (PA-SPMV)

GPU in-band ECC

- Diminishes data throughput
- Degrades system performance

CacheCraft

- A novel GPU μ-architecture with new sectoring of “30-30-30-30-8”

Benefits

- Reduce memory access overhead by up to 89.4% (GB-SSSP)
- Improve performance of applications by up to 23.5% (PA-SPMV)

CacheCraft is a promising solution for GDDR-based GPUs, enhancing performance under memory protection

Thank you

Q&A