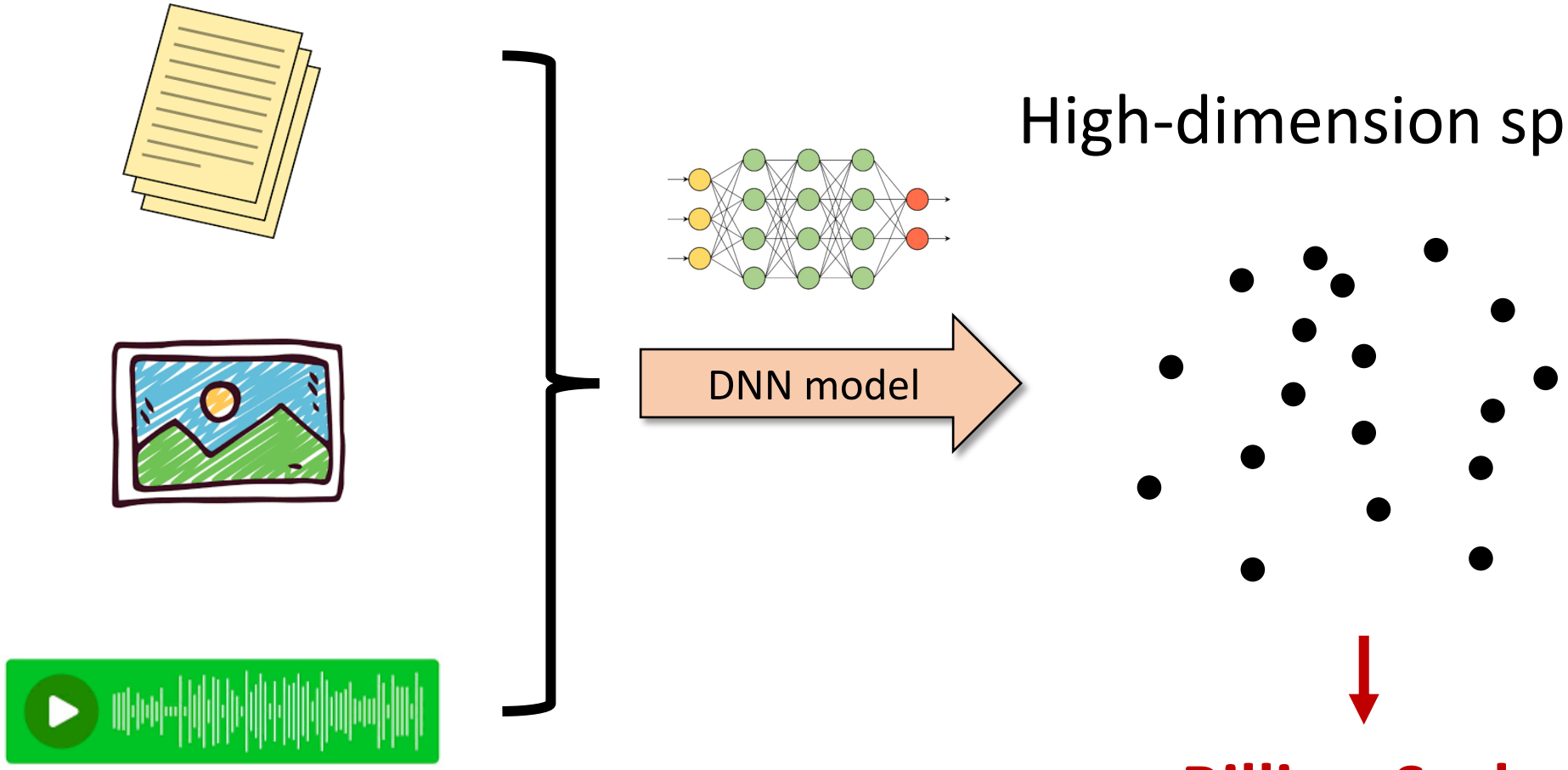
Fast Vector Query Processing for Large Datasets Beyond GPU Memory with Reordered Pipelining

Zili Zhang, Fangyue Liu,
Gang Huang, Xuanzhe Liu, Xin Jin

北京大学
PEKING UNIVERSITY

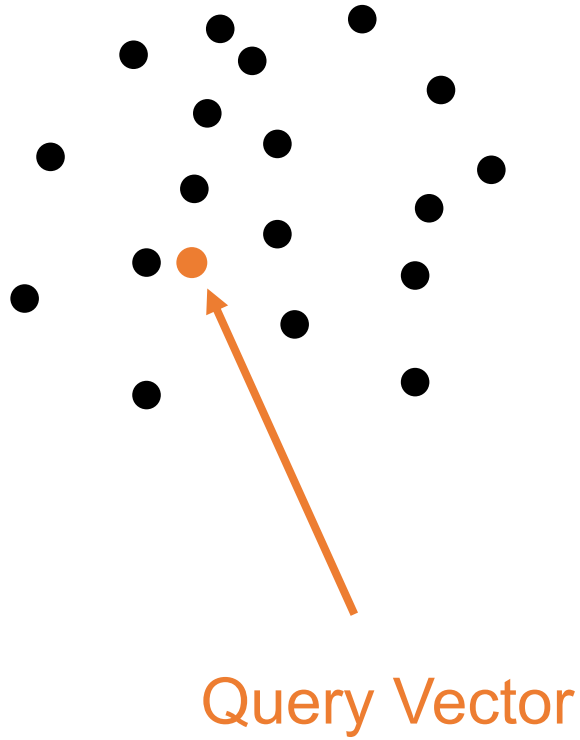
Vector Query



Vector Query

- What is Vector Query ?

Given a query vector

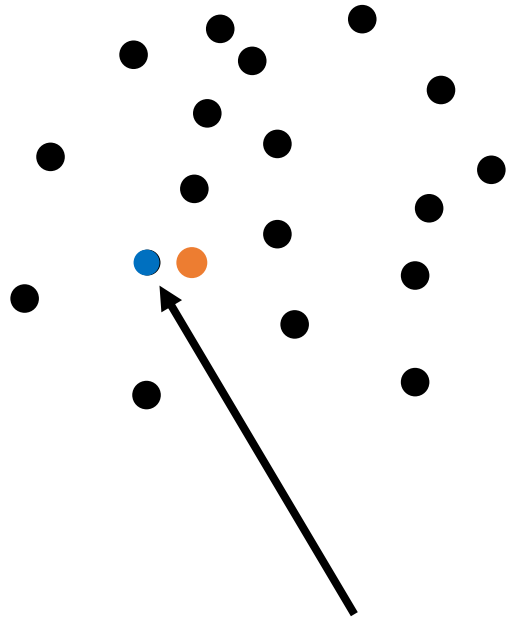


Vector Query

- What is Vector Query ?

Given a query vector

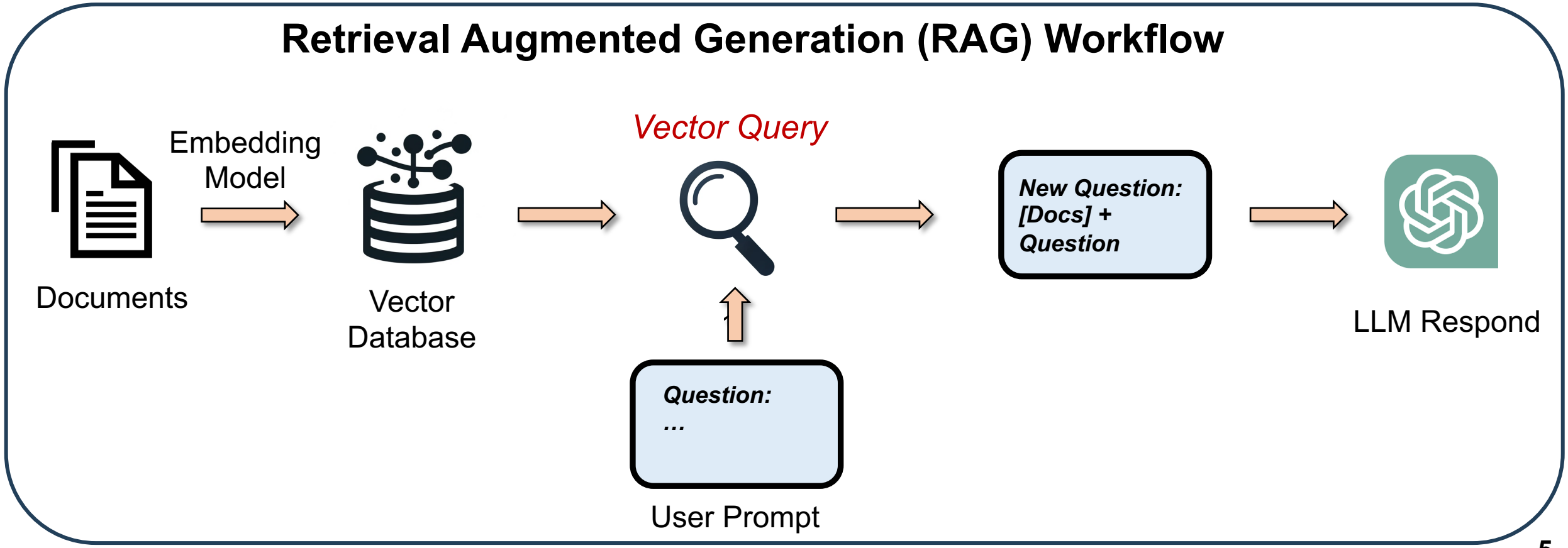
Return Top-k Nearest vectors



Top-1(k=1) Nearest Vector

Vector Query

- Vector Query in **Real-World Applications**



Vector Query

Milvus

Pinecone

Faiss

Qdrant

Enumeration (KNN)

- Low throughput
- High latency

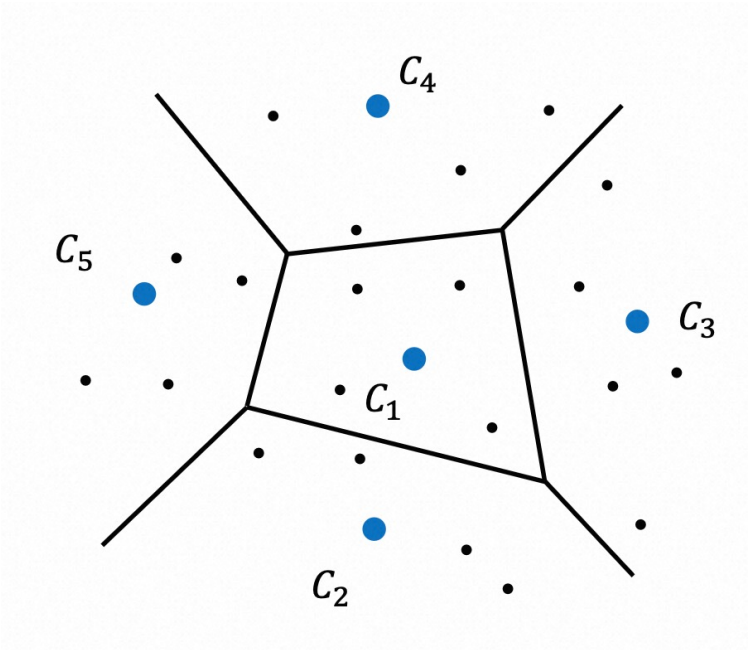
ANN

- High throughput
- Low latency
- Accurate enough

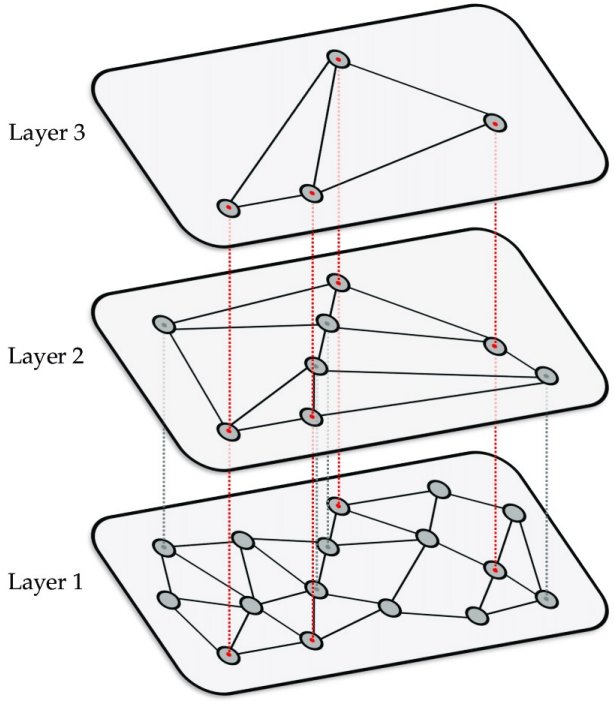
Approximate Vector Query

- Two Representative Vector Query Algorithms

Inverted Index



Graph Index



Approximate Vector Query

- Two Representative Vector Query Algorithms

Inverted Index

- Simple computation pattern
- Less memory footprint

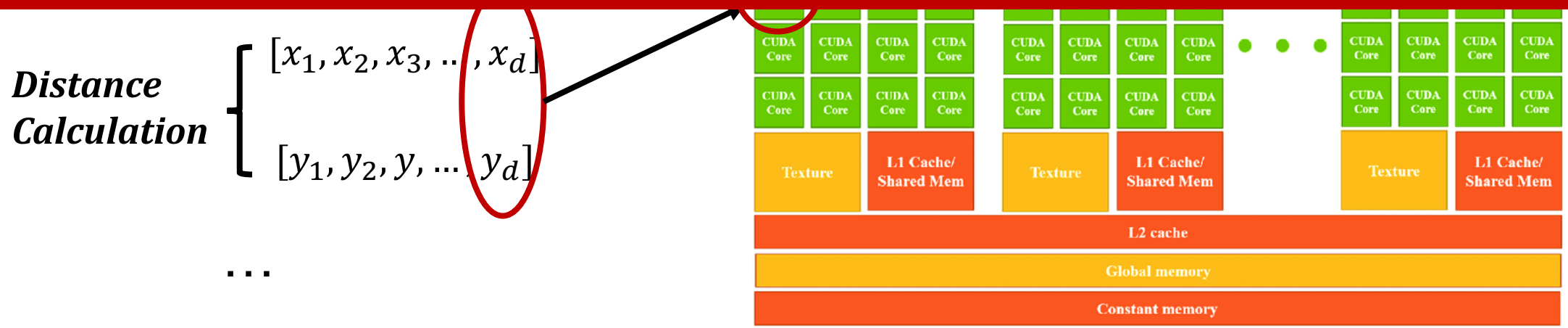
Graph Index

- Complex computation pattern
- More memory footprint

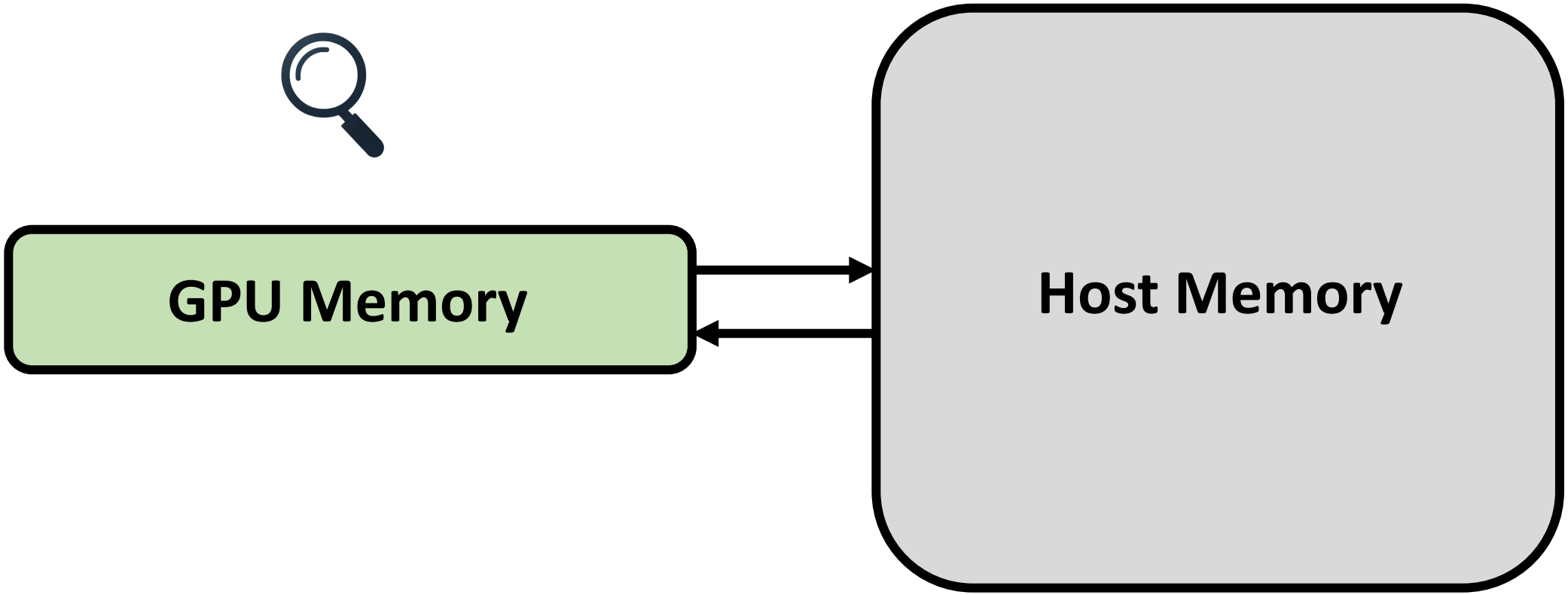
Vector Query on GPU

- GPUs are natively designed for vector operations

Limited GPU Memory is a key bottleneck!



Vector Query on GPU



Vector Query on GPU

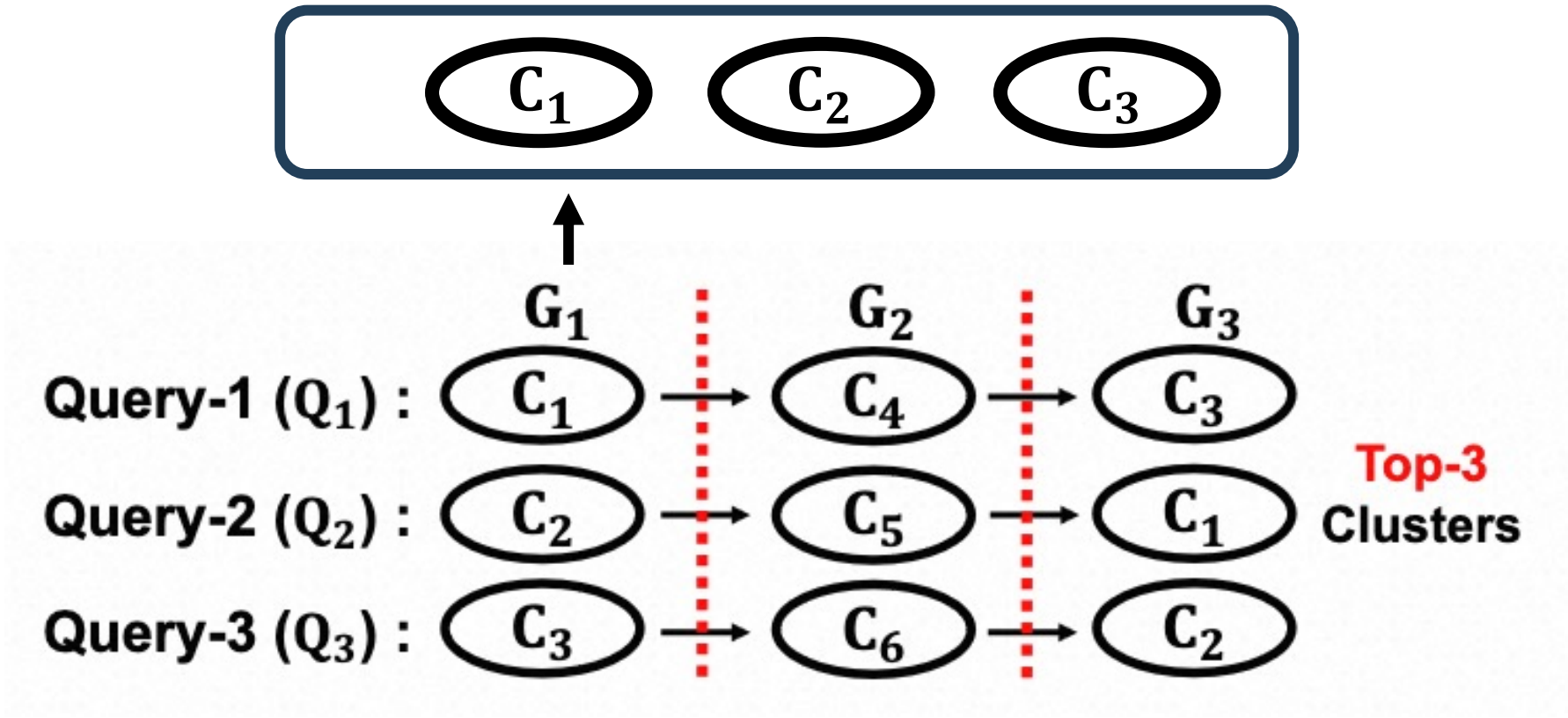
Challenge 1: How to reduce redundant data **transmission**?

Challenge 2: How to maximize GPU utilization for **computation**?

Challenge 3: How to maximize the efficiency of **pipeline**?

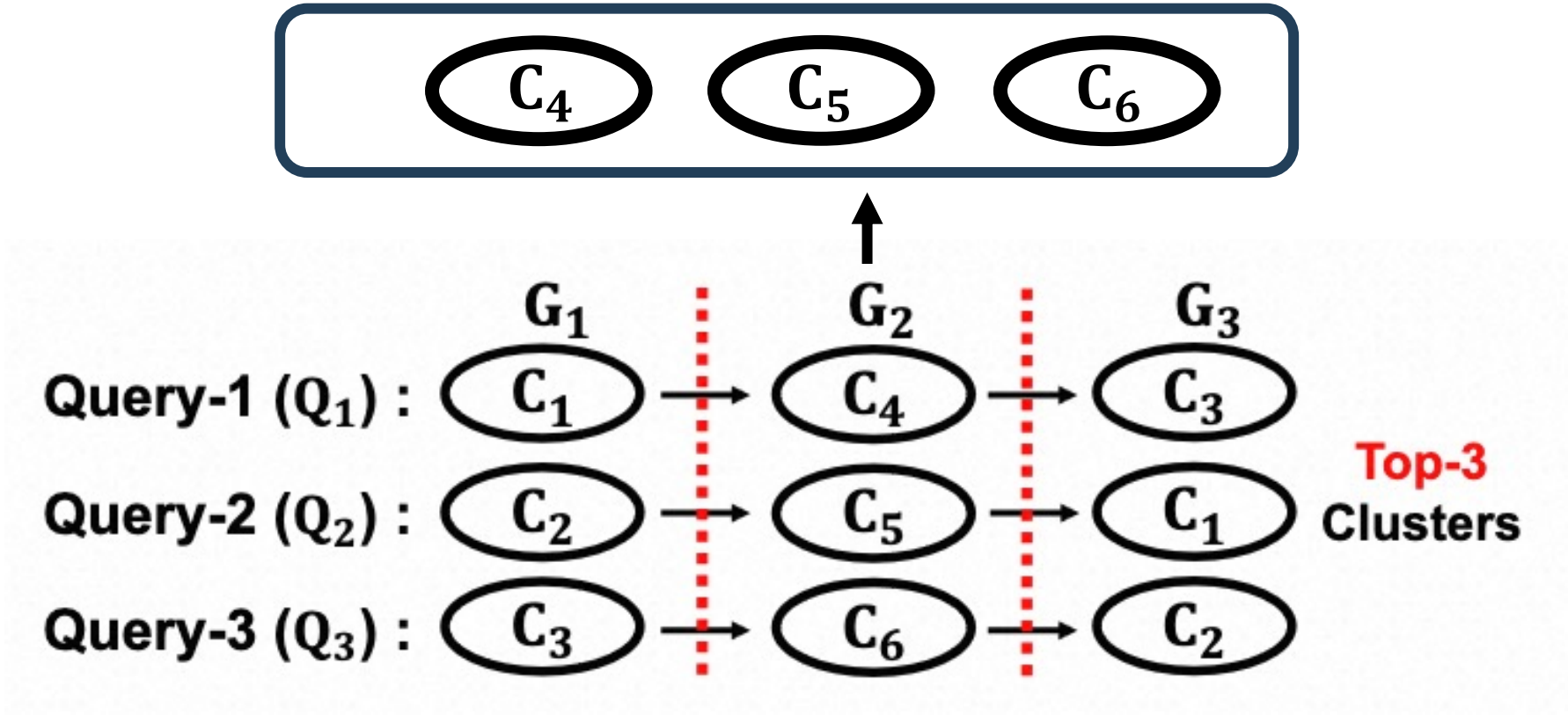
Vector Query on GPU: Challenge

- Transmission Challenge



Vector Query on GPU: Challenge

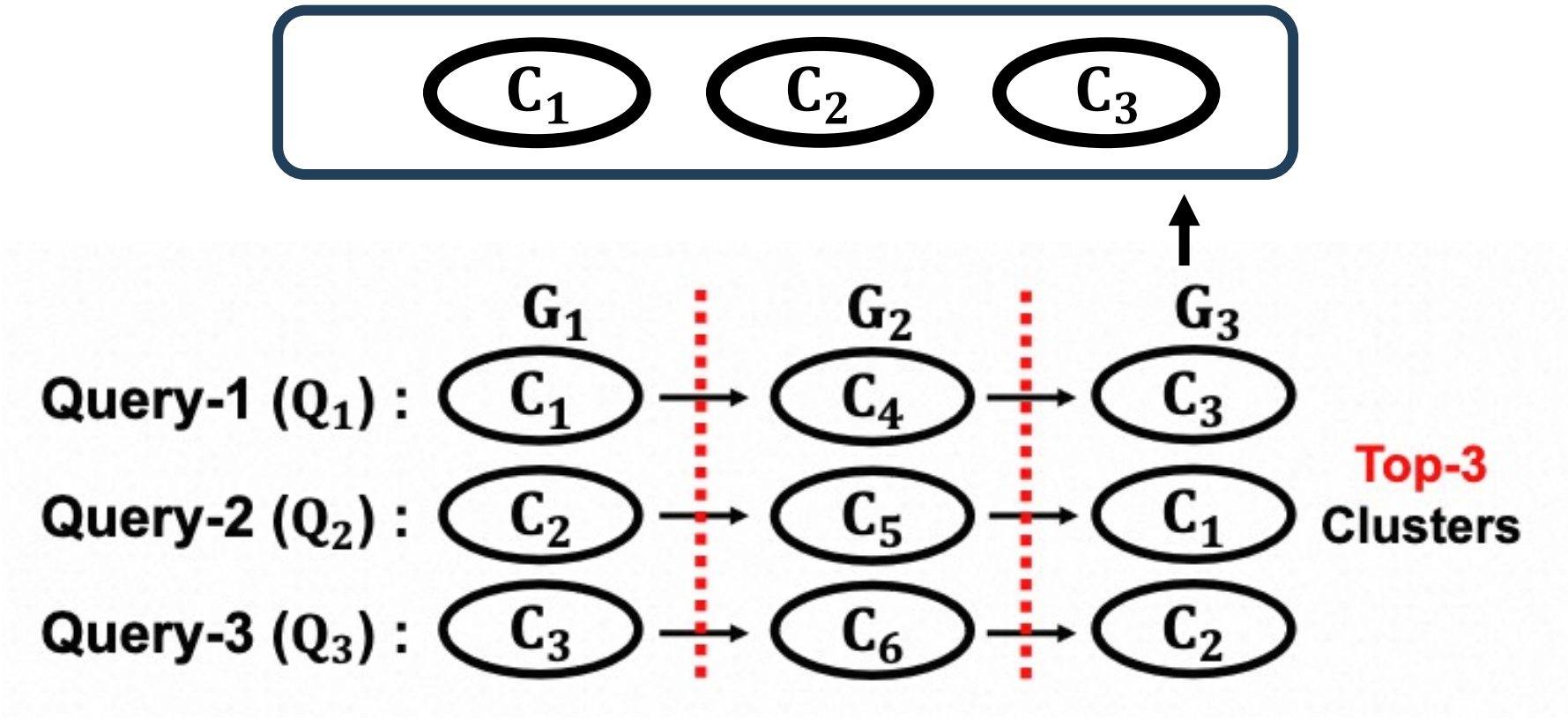
- Transmission Challenge



Vector Query on GPU: Challenge

Redundant Transmission !

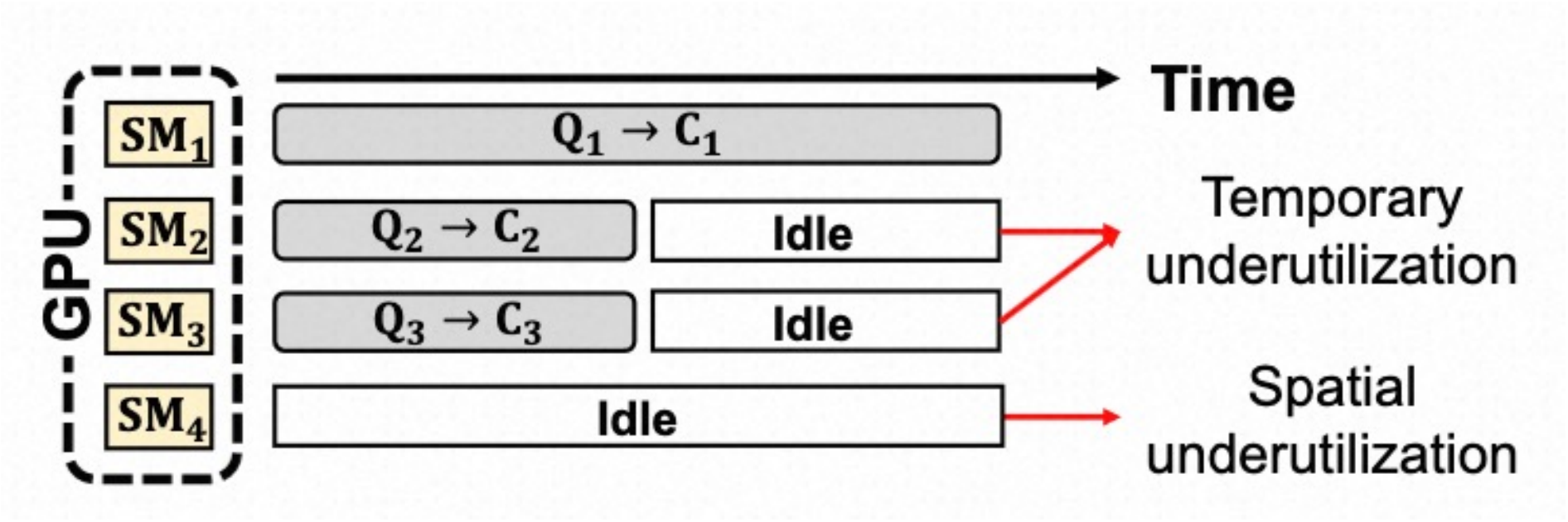
- Transmission Challenge



Vector Query on GPU: Challenge

GPU Underutilization !

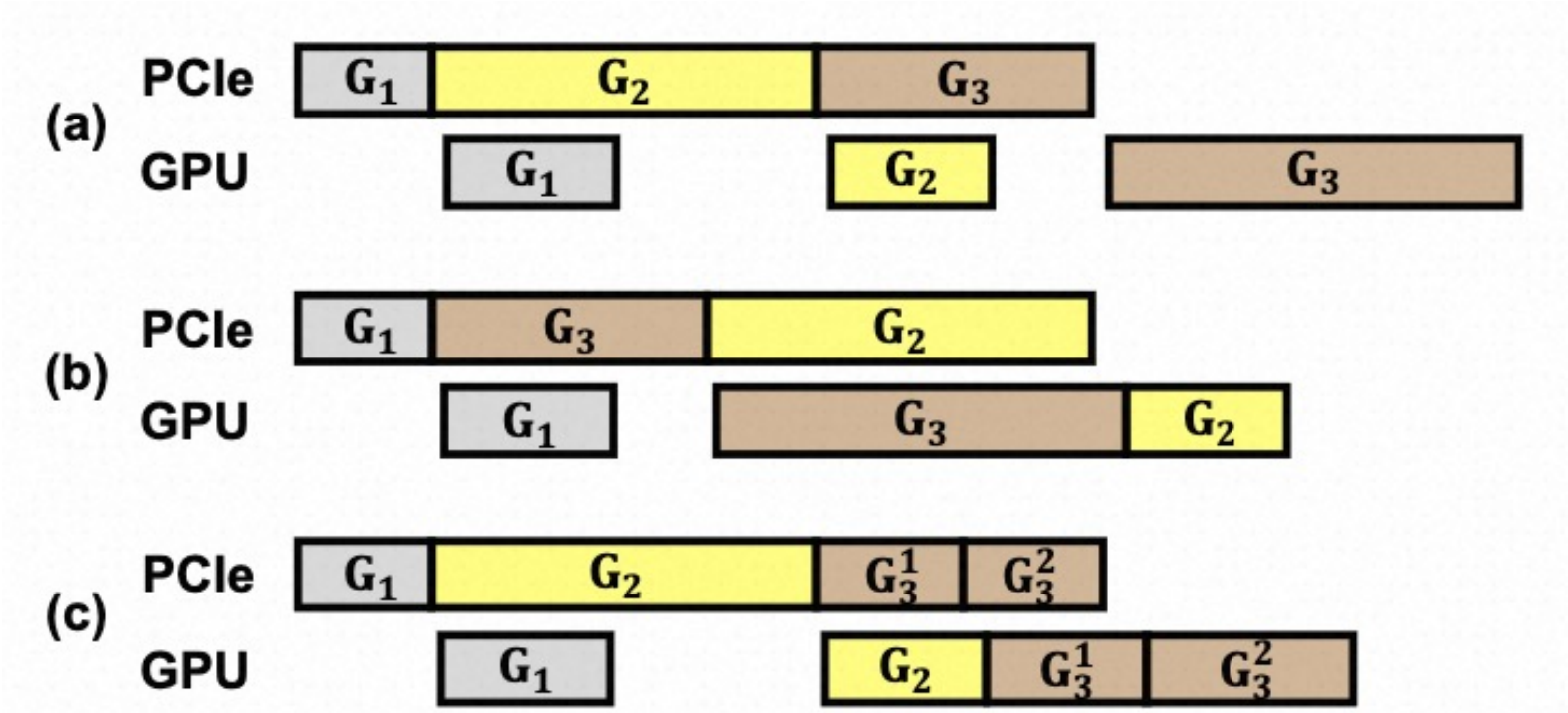
- Computation Challenge



Vector Query on GPU: Challenge

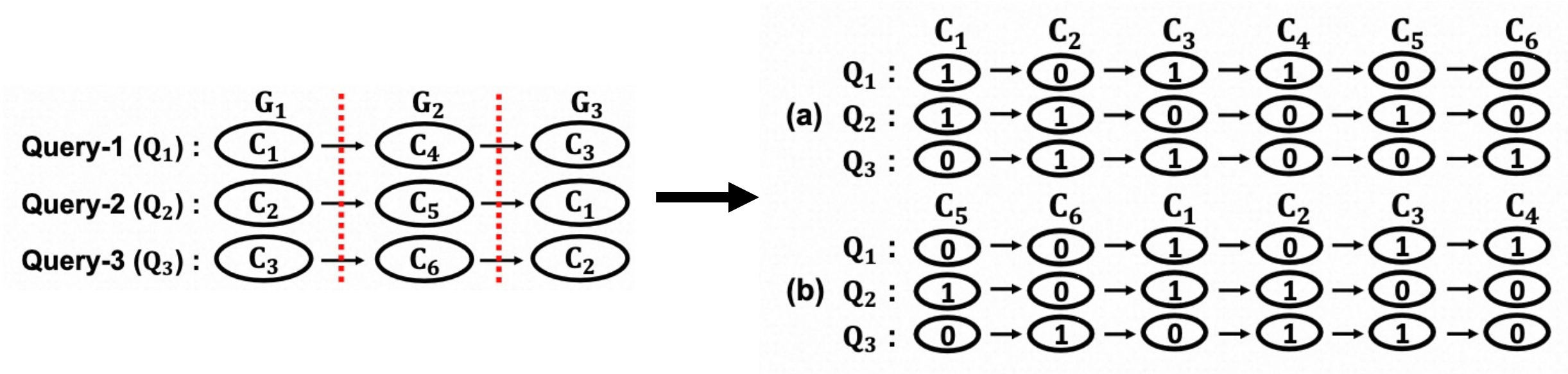
Overlap PCIe and GPU !

- Pipeline Challenge



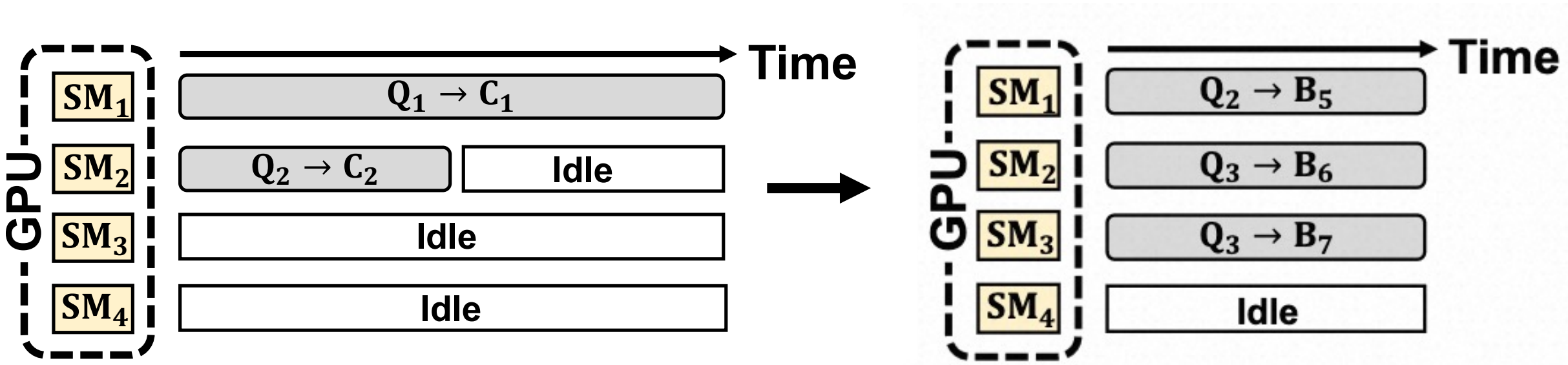
Rummy Design

- Transmission: **Cluster-based Retrofitting**



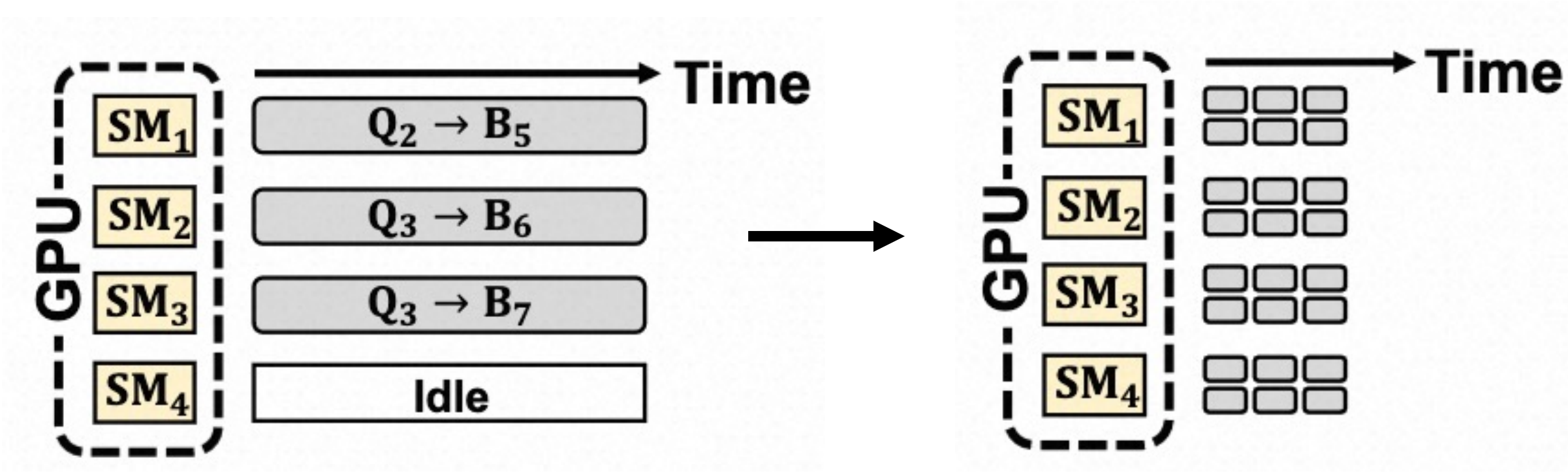
Rummy Design

- Computation: **Cluster Balancing**



Rummy Design

- Computation: **Dynamic kernel padding**



Rummy Design

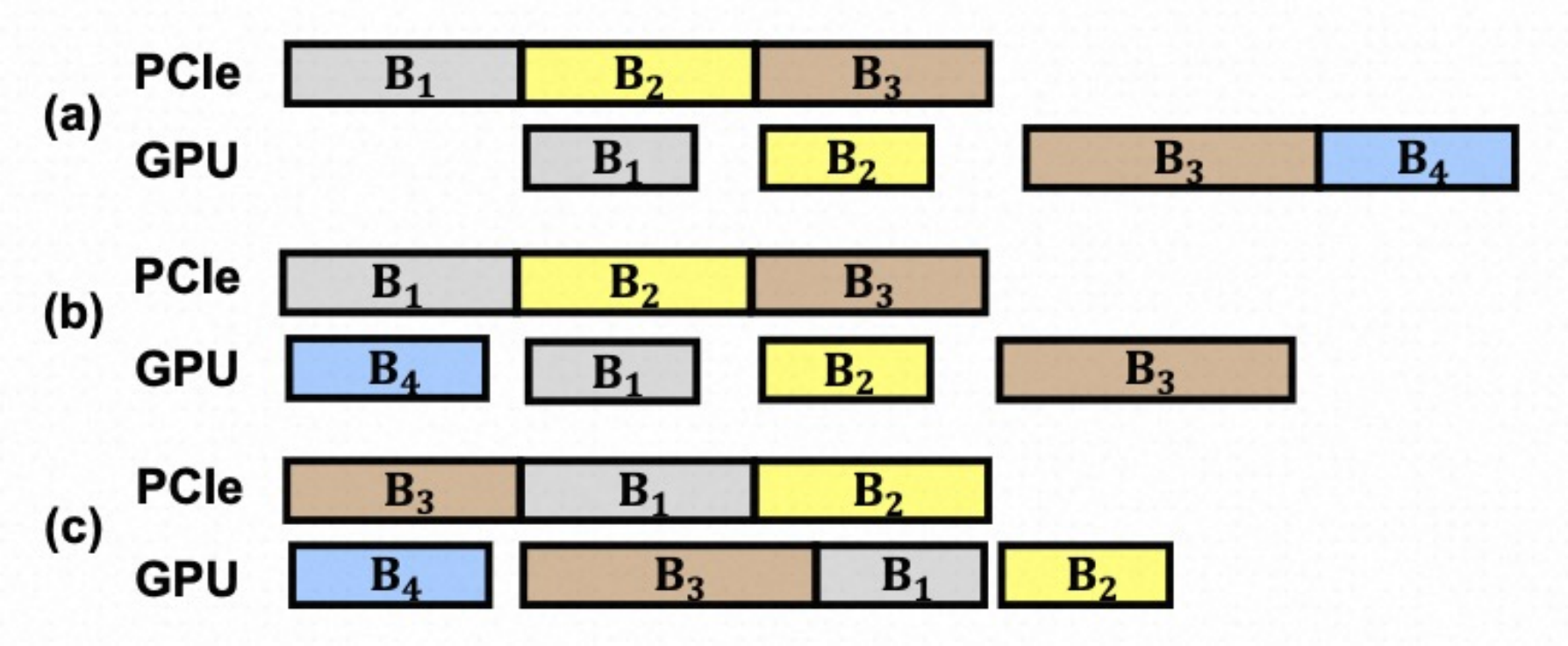
- Pipeline: **Reordering** + **Dynamic Programming**

Optimal processing order of clusters

Optimal groups of clusters

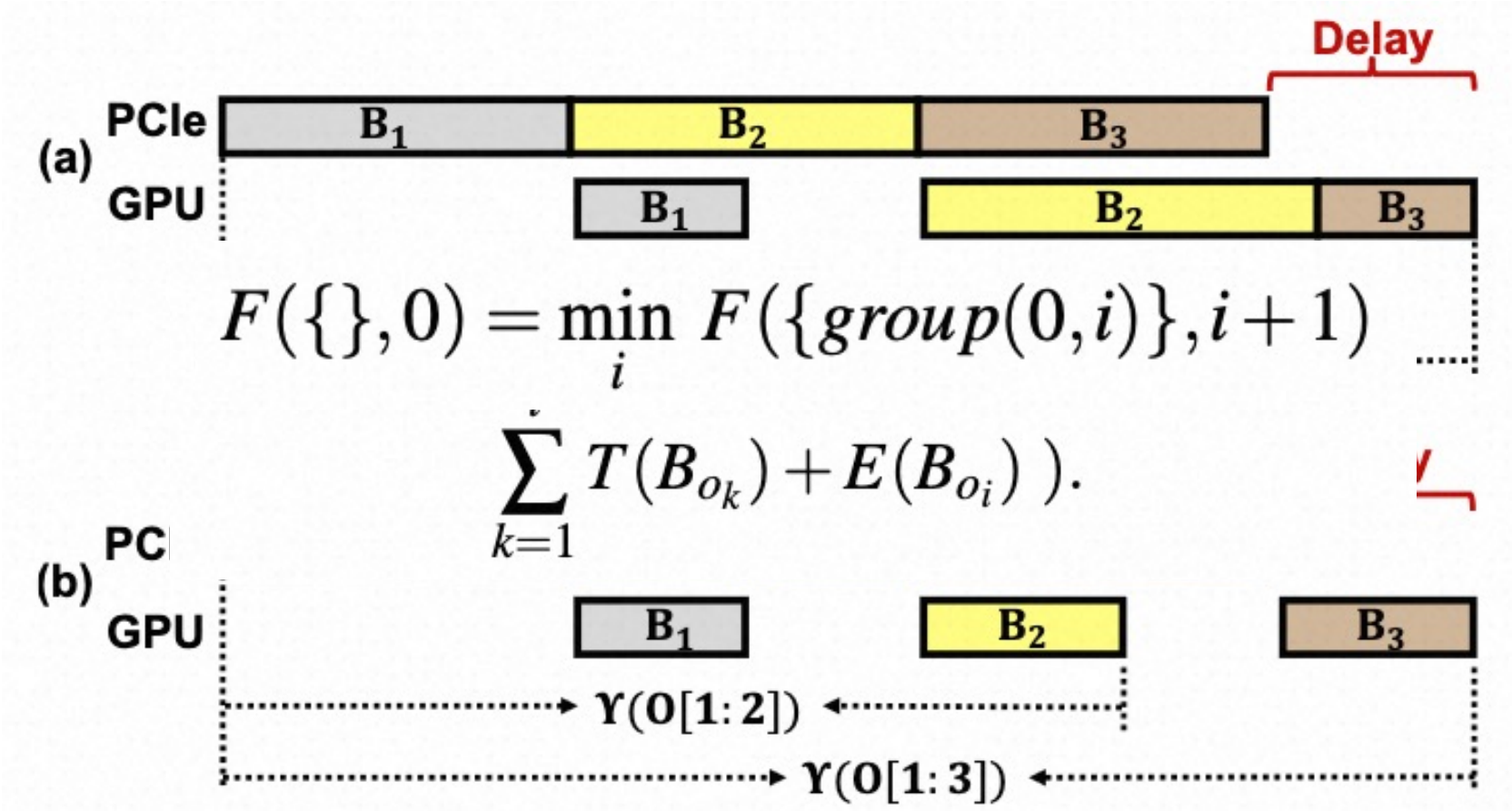
Rummy Design

- Pipeline: **Reordering**



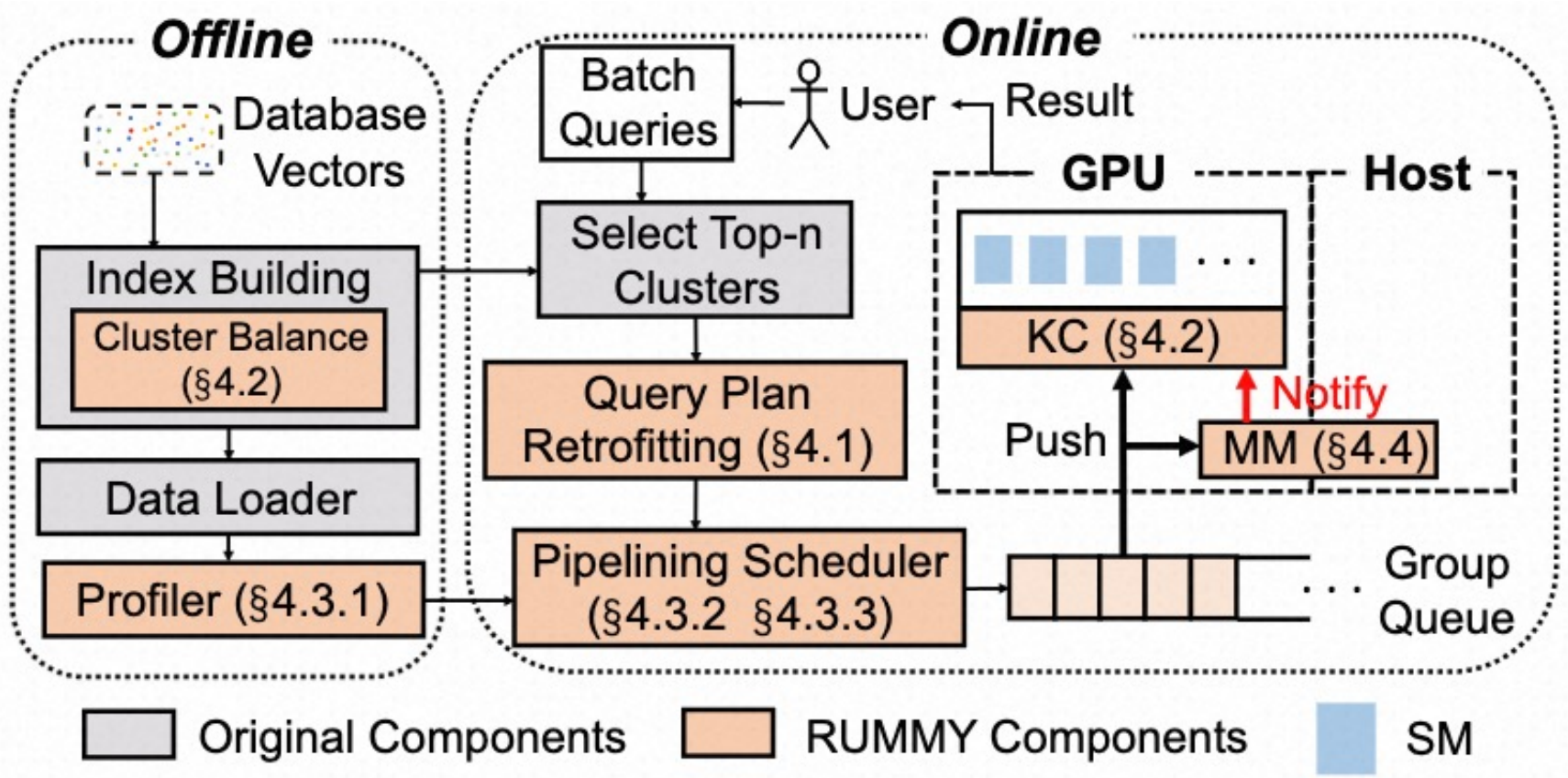
Rummy Design

- Pipeline: **Dynamic Programming**



Rummy System

- A GPU-based vector query system beyond GPU memory

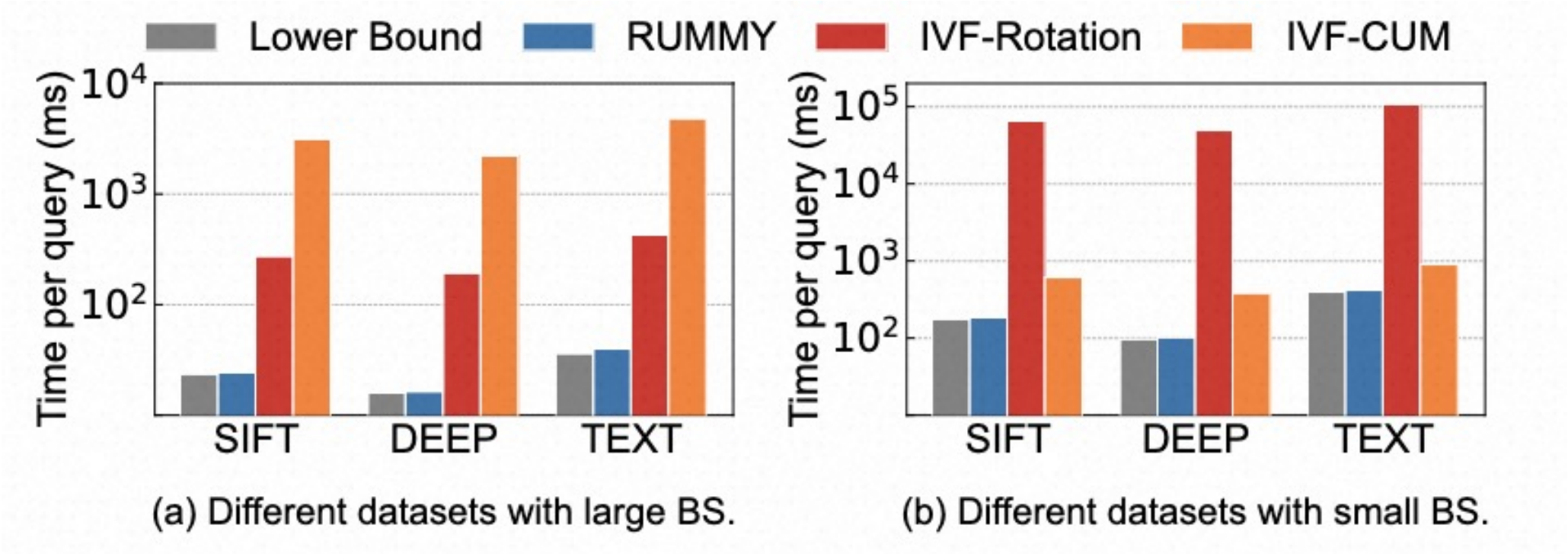


Implementation and Evaluation Step

- Implementation
  - ~12K LoC C++
  - Faiss
- TestBed
  - A100, V100, T4 GPU (for GPU)
  - AWS C5X instance (for CPU)
- Datasets
  - SIFT, DEEP, TEXT (one billion items)

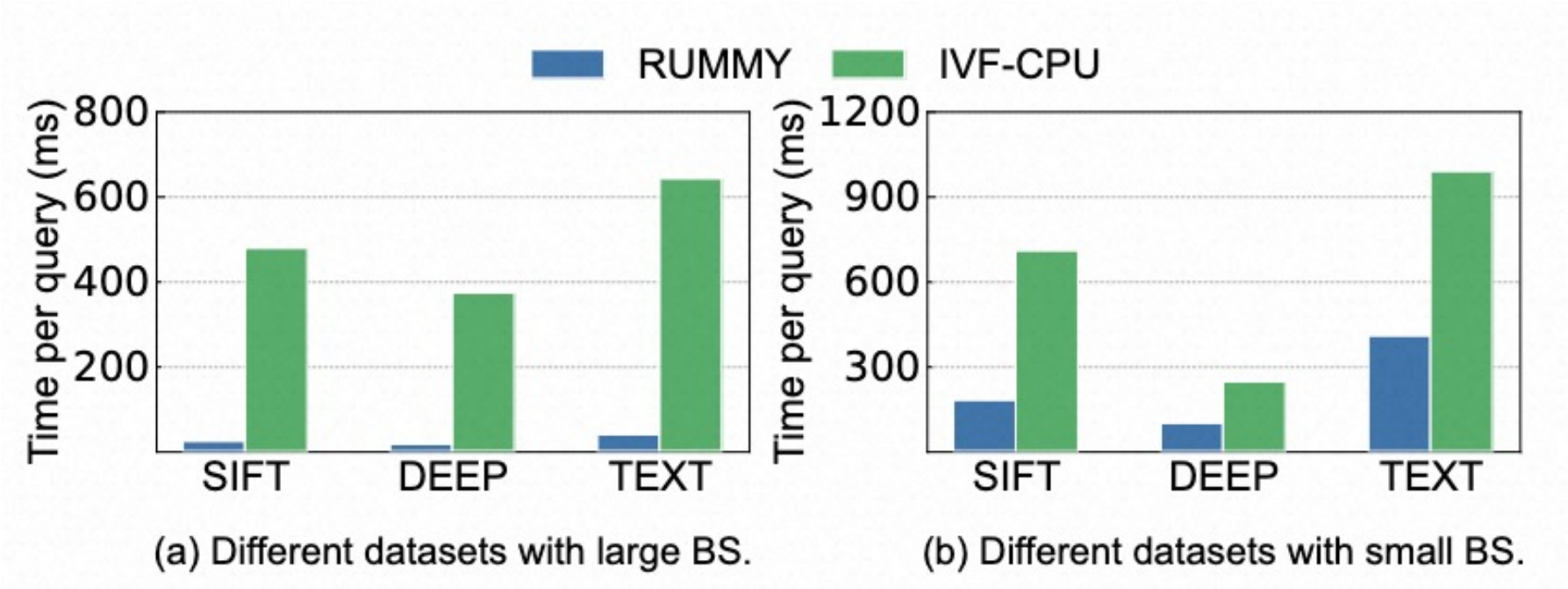
Evaluation

- Rummy is able to achieve near-optimal performance



Evaluation

- Rummy is more efficient than CPU-solutions



Conclusion

- Rummy:

  The first GPU-based system for billion-scale vector query processing beyond GPU memory

zzlcs@pku.edu.cn

Thanks!