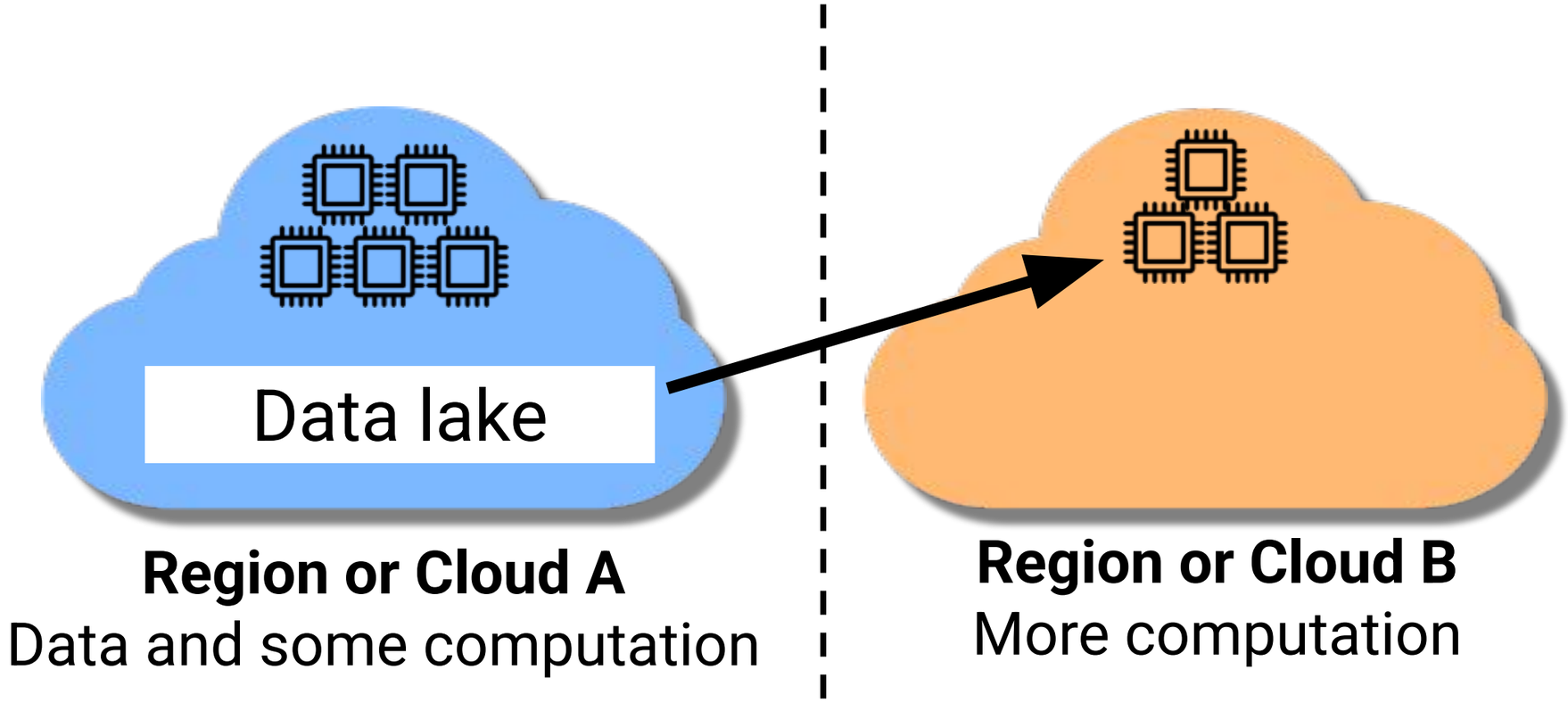
Reducing **Cross-Cloud/Region Costs** with the Auto-Configuring MACARON Cache

Hojin Park
Ziyue Qiu, Gregory R. Ganger, George Amvrosiadis

Carnegie Mellon University

PARALLEL DATA LABORATORY
CARNEGIE MELLON UNIVERSITY

Public cloud deployment



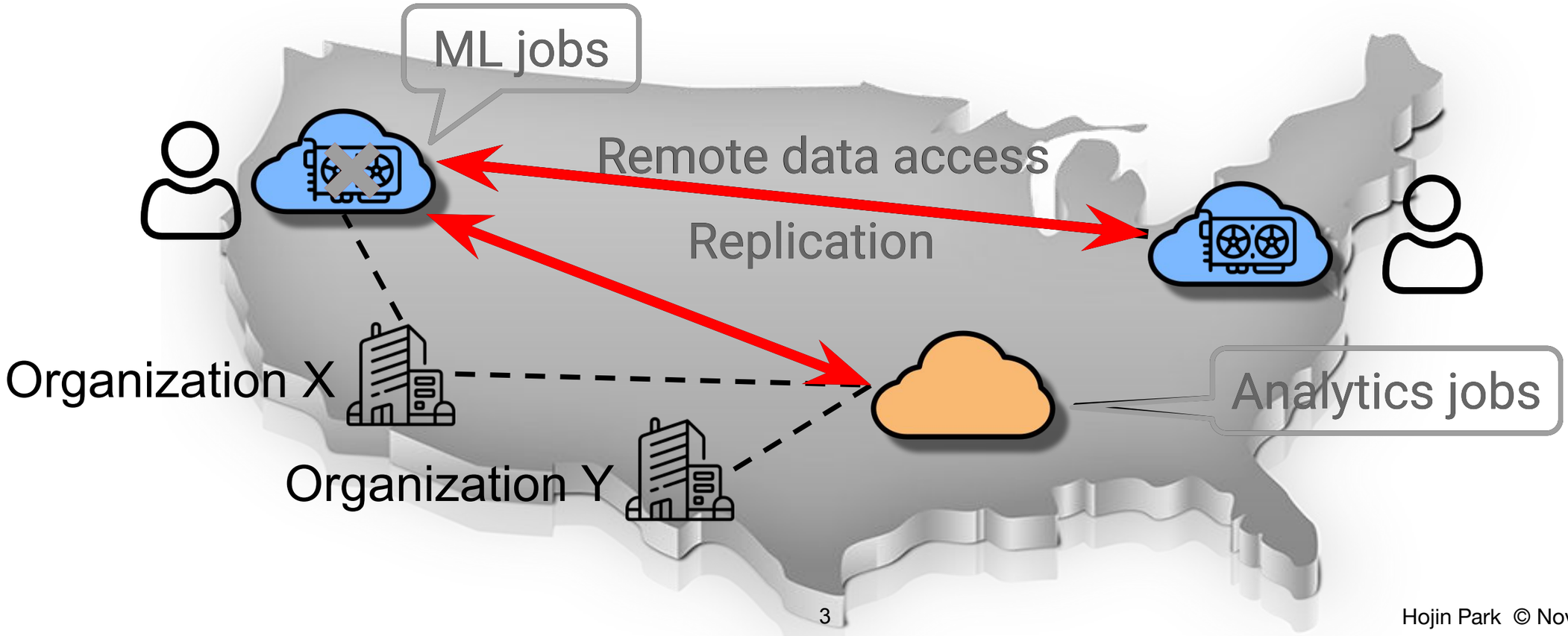
Cross-cloud/region data access

Cross-region

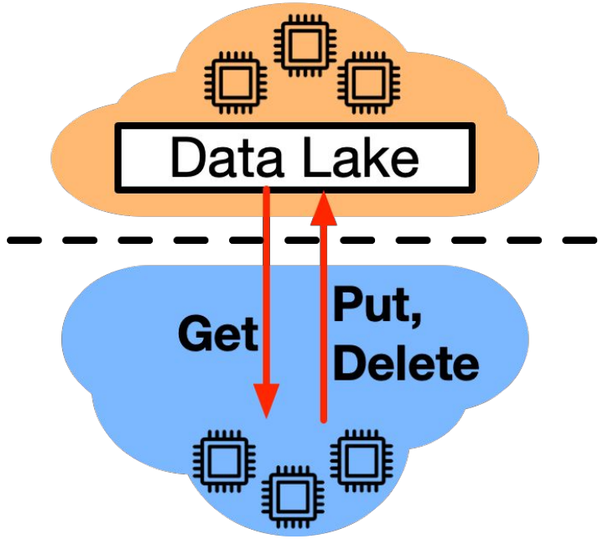
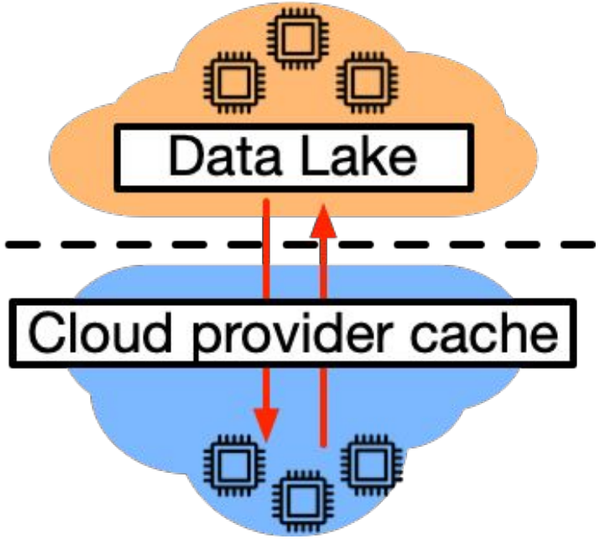
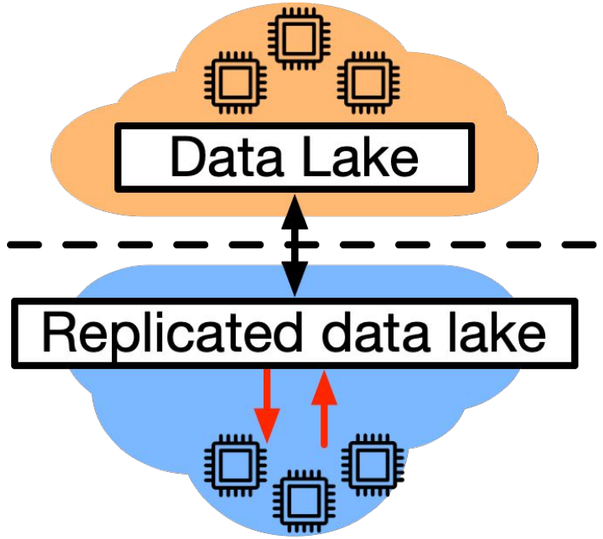
- Resource shortages in one region
- Collaborators in different regions

Cross-cloud

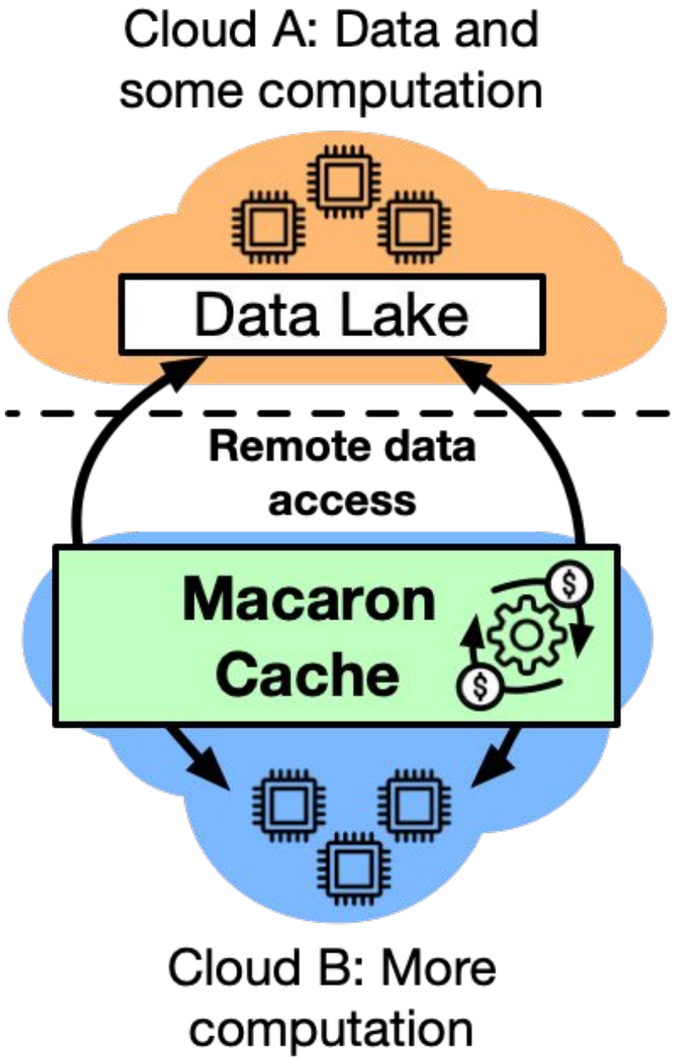
- Different features and service offerings
- Collaboration between organizations



Current approaches and downsides

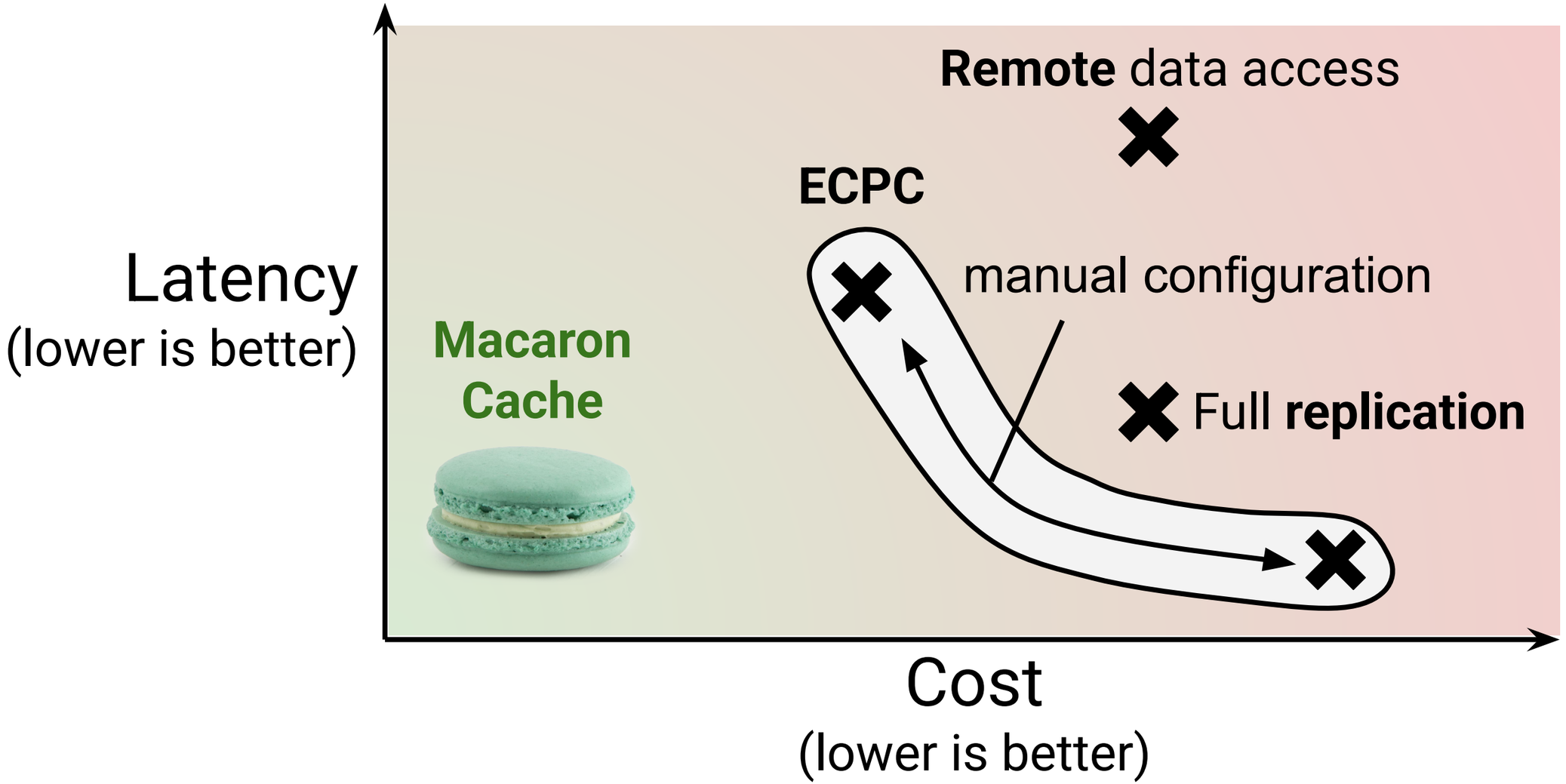
	Remote data access	**ECPC** (Elastic Cloud Provider Caching)	Full **replication**
Region or Cloud A / Region or Cloud B			
Downside	– Egress cost – Access latency	– DRAM capacity cost – Manual configuration	– Synchronization cost – Excess capacity cost

Macaron: cost-aware auto-configuring cache



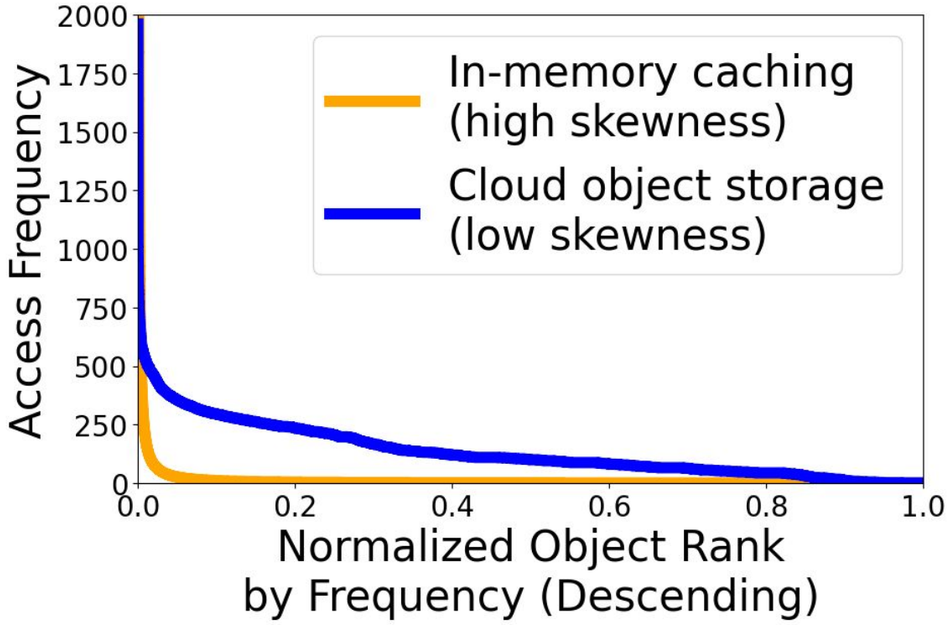
- Auto-configures cache to minimize remote access costs
  - Macaron reduces cost by up to 99%
- Can achieve both cost-efficiency and low latency
  - On average, 61% lower latency
- Adapts to workloads' data access pattern changes

Goal: performance and cost-efficiency

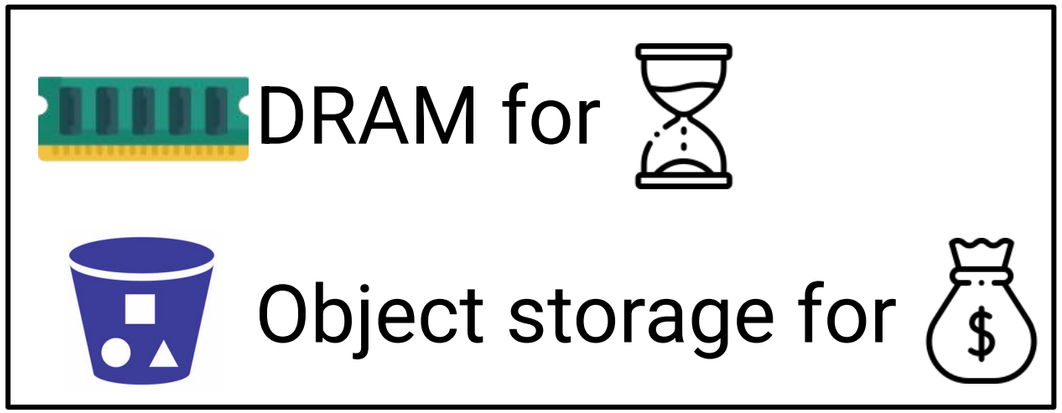


Design #1: two-level caching

Observation: Cloud object storage workloads have large objects and high spread of accesses



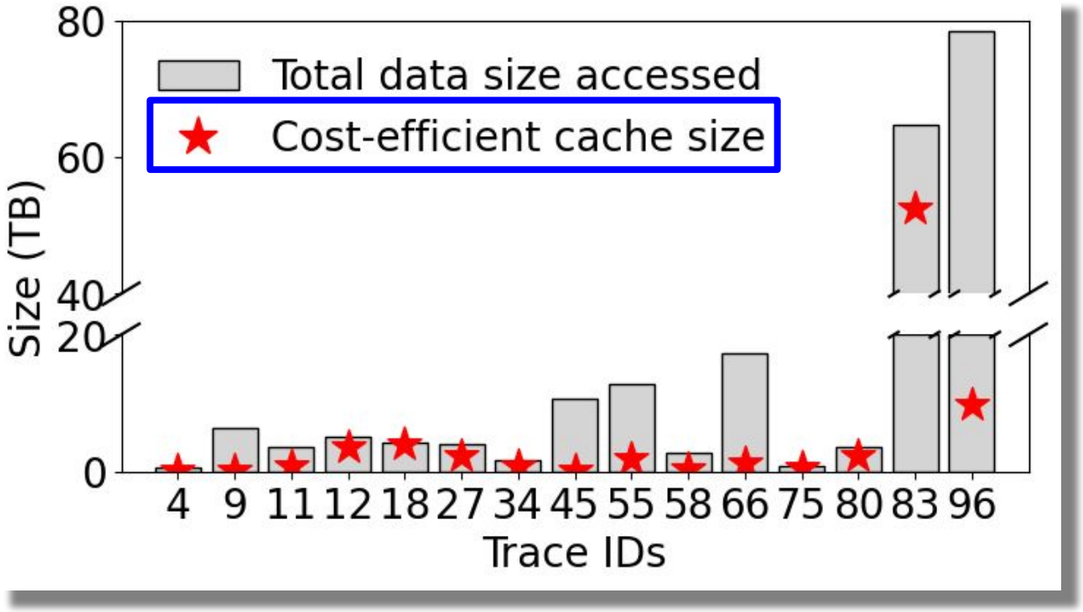
Macaron's hybrid cache

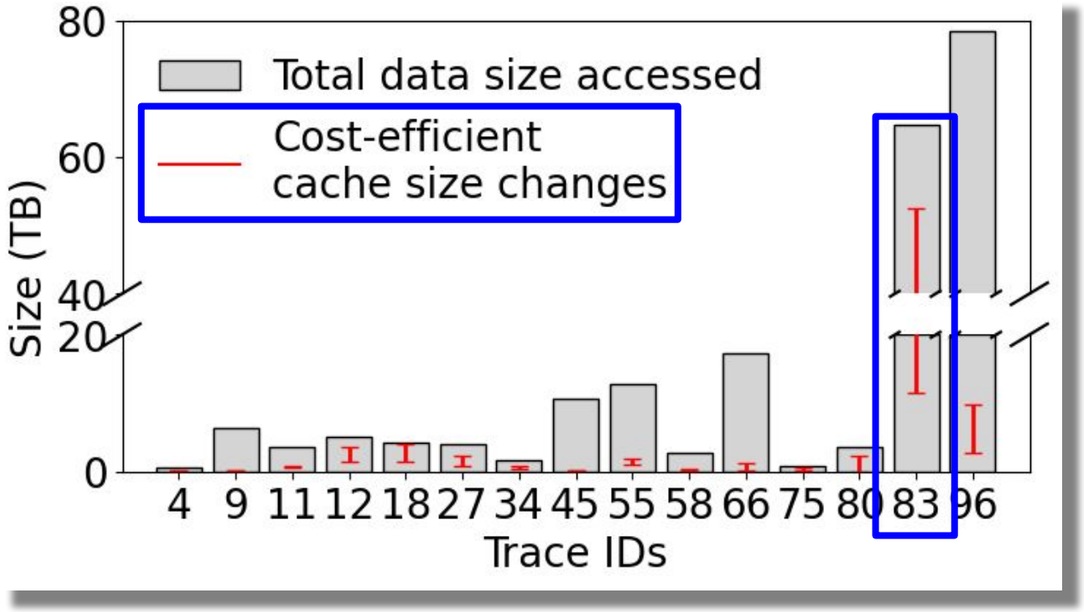


➡ Need **large cache capacities** to reduce **data egress cost**

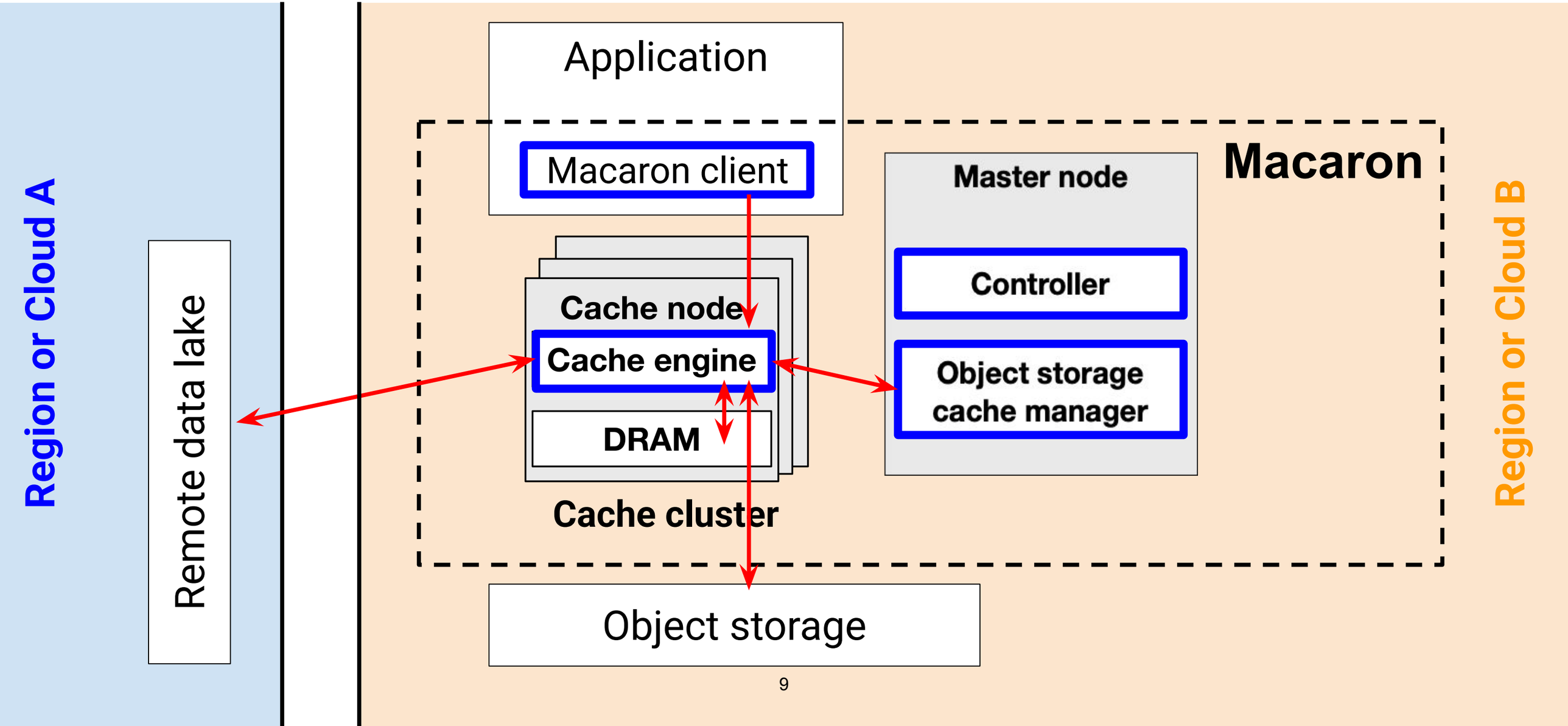
Design #2: adaptive cache sizes

- Cost-efficient cache sizes **vary across workloads**
  - Ranges from 62GB-52TB (1-81% of data size accessed)
- Cost-efficient cache sizes **changes over time**
  - Without adaptivity, costs can be up to 6.7x higher.

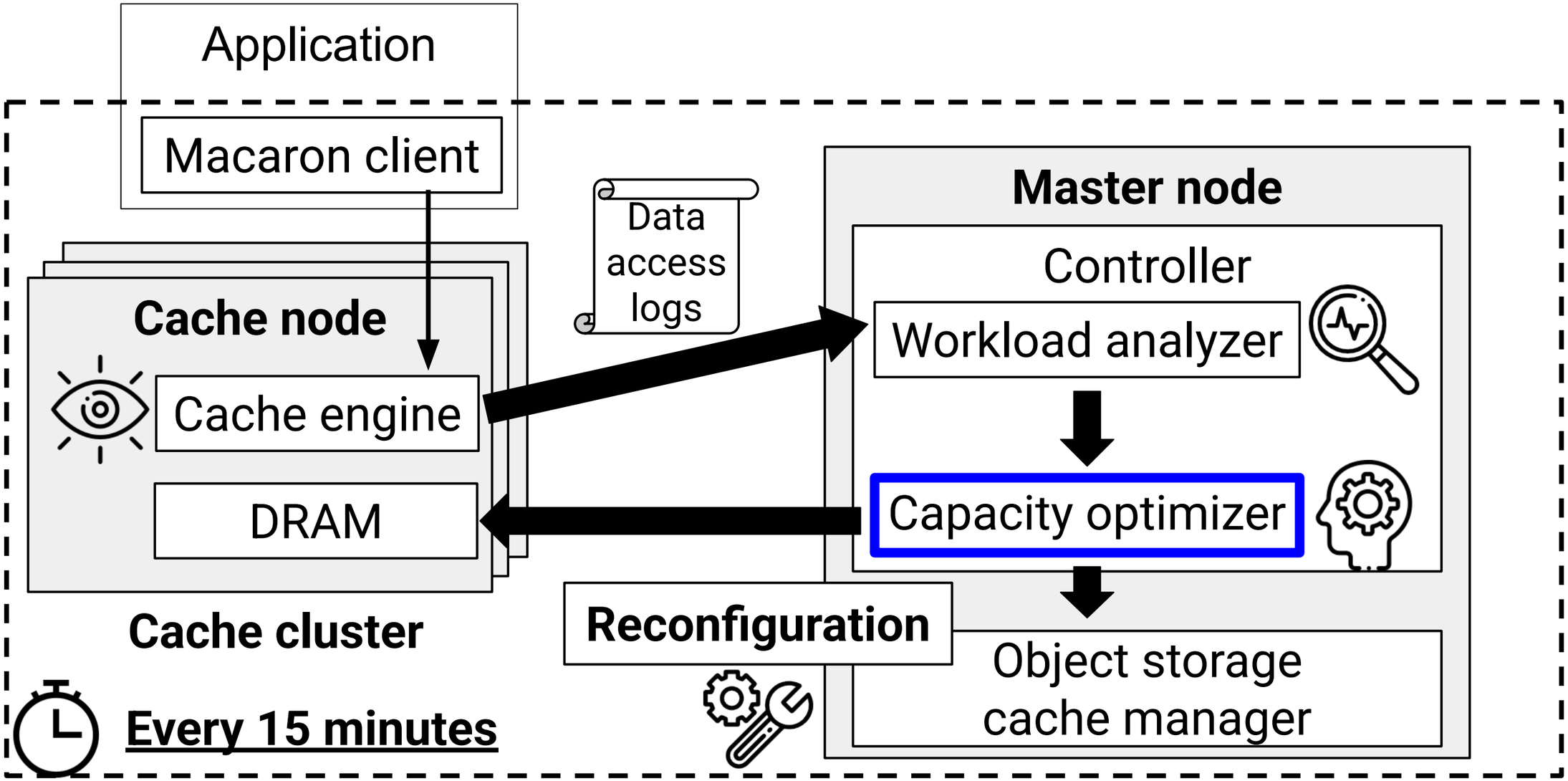




Architecture

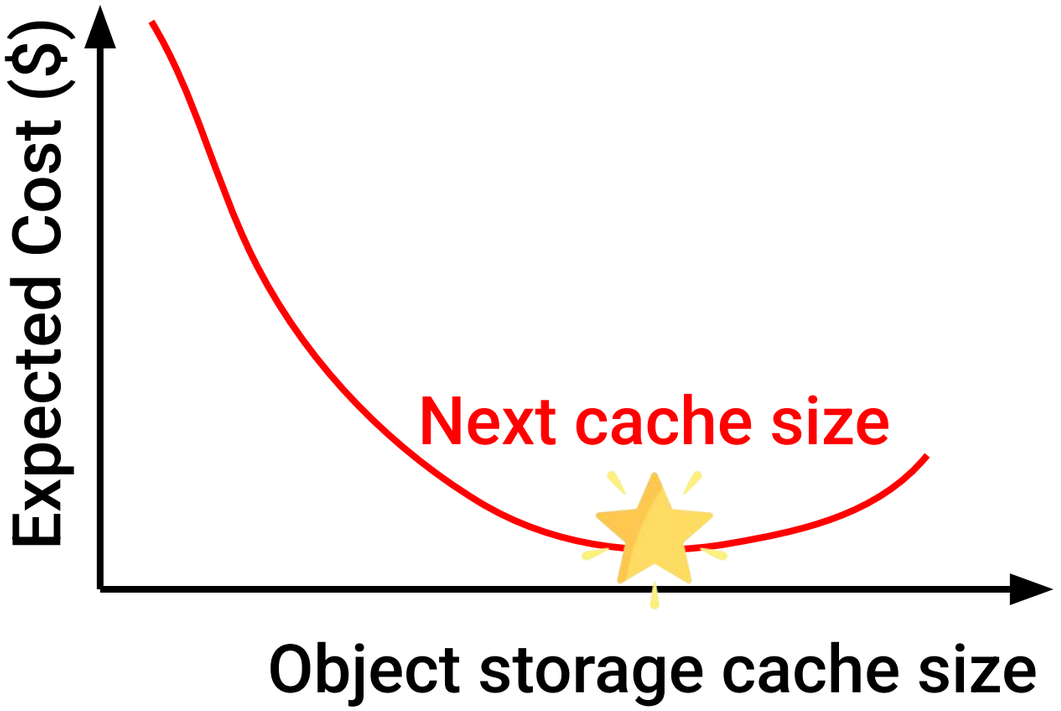


Optimization loop



Optimizing object storage cache

Expected cost curve: How much cost is expected during the next time window for each object storage cache size?

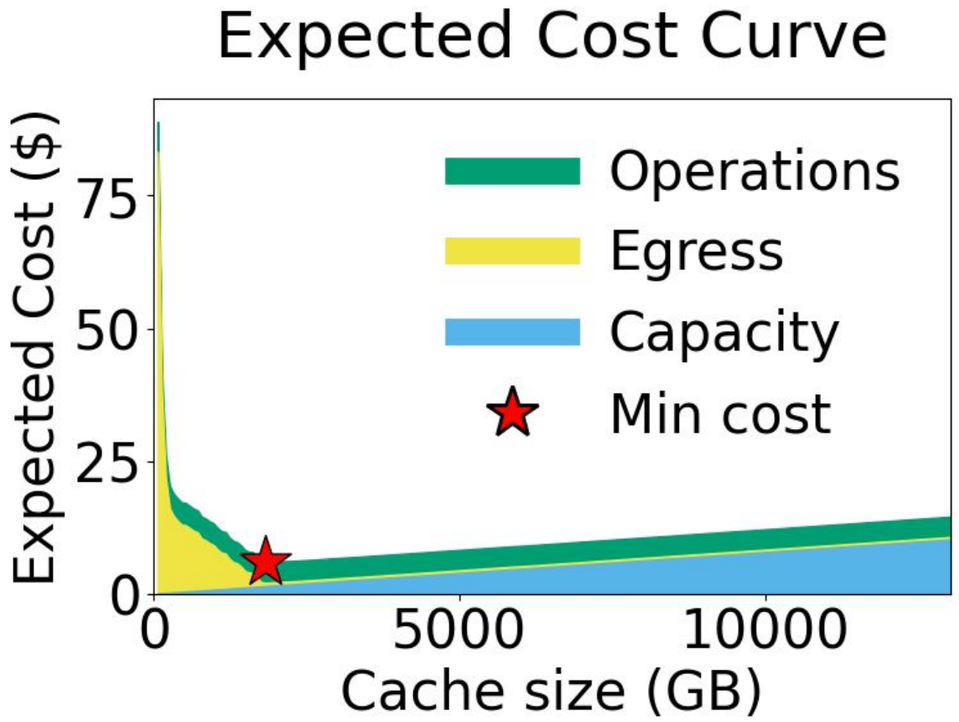


Optimizing object storage cache

Expected cost curve: How much cost is expected during the next time window for each object storage cache size?

Cost breakdown

- Egress $\propto$ Cache miss bytes
- Capacity $\propto$ Cache size
- Operation $\propto$ Miss ratio, # of requests



More in the paper

- Optimizing the DRAM cache cluster
- Extending miniature simulation* for workload analysis
- Prototype implementation insights
- Mechanism for adaptivity

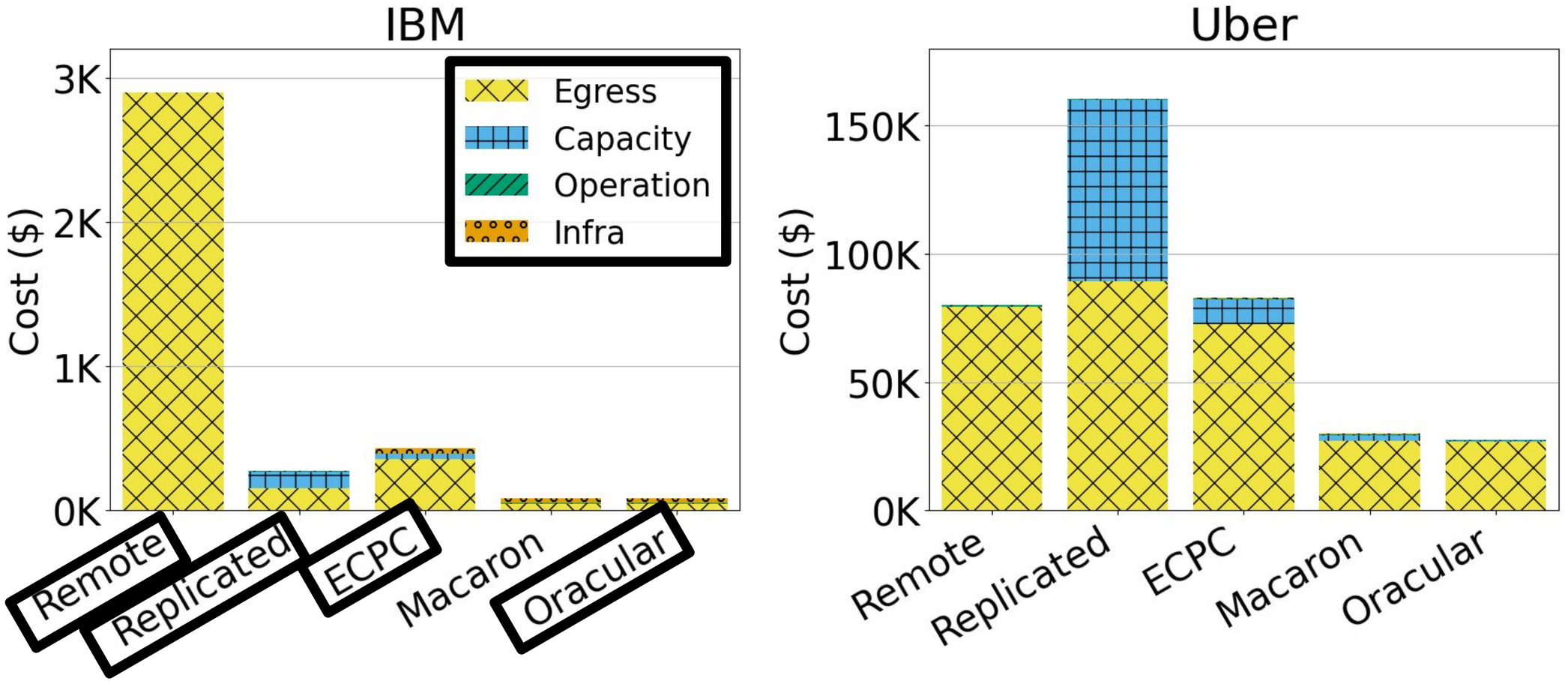
* Waldspurger et al. "Cache Modeling and Optimization using Miniature Simulations" ATC‘17

Evaluation setup

- Data: 15 IBM traces* and 4 traces collected from VMware and Uber
- Configurations: AWS pricing model, N. California and N. Virginia regions
- Trace-based Macaron simulator
  - Macaron prototype used for cloud validation (details in paper)
  - Cost of running prototype will be more than $1MM

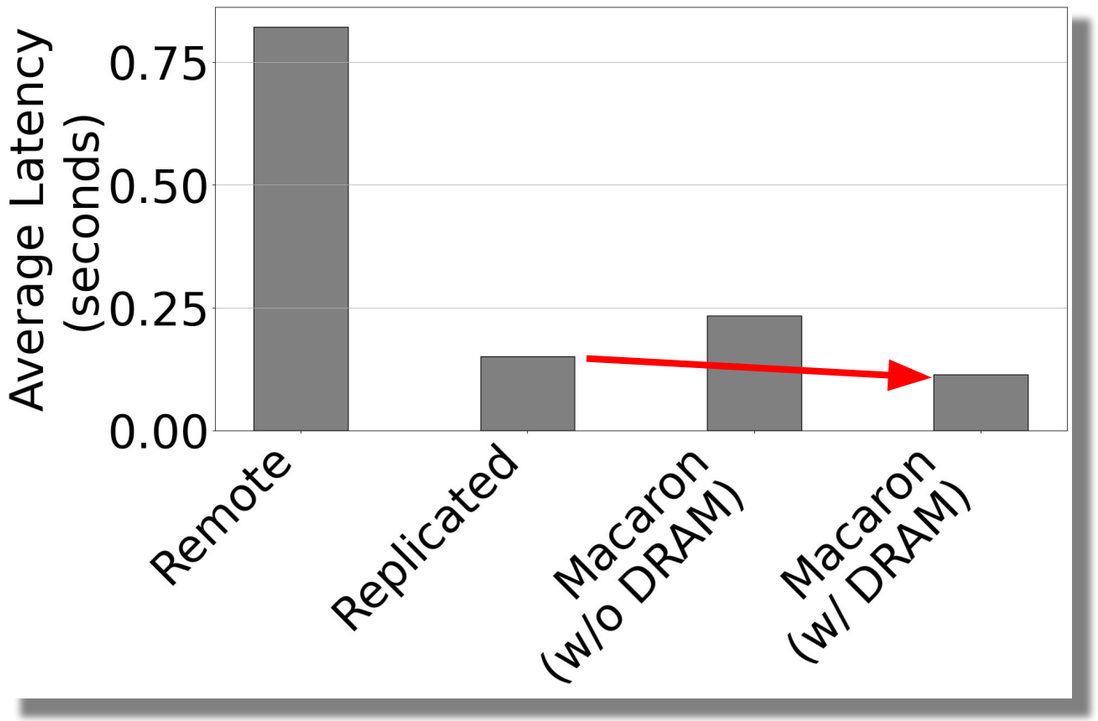
* Eytan et al. "It’s Time to Revisit LRU vs. FIFO" HotStorage’20

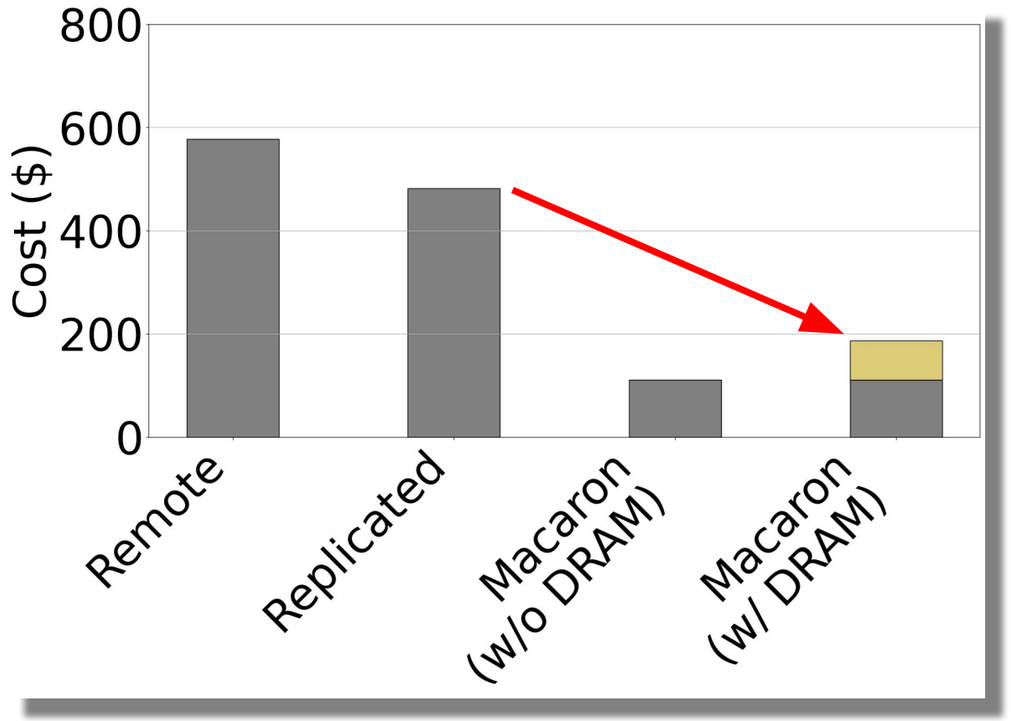
Macaron significantly reduces costs



Macaron reduces costs by up to **89%** compared to **ECPC**

Macaron is cost-efficient and performant





On average, Macaron achieves **61% lower latency** and **64% cost savings** than **Replicated** with its dynamic cache cluster

Conclusion

- Auto-configures cache to minimize remote access costs
  - Macaron saves costs by up to 99%
- Can achieve both cost-efficiency and low latency
  - On average, 61% lower latency
- Adapts to workloads' data access pattern changes

More info at https://github.com/hojinp/macaron_simulator