

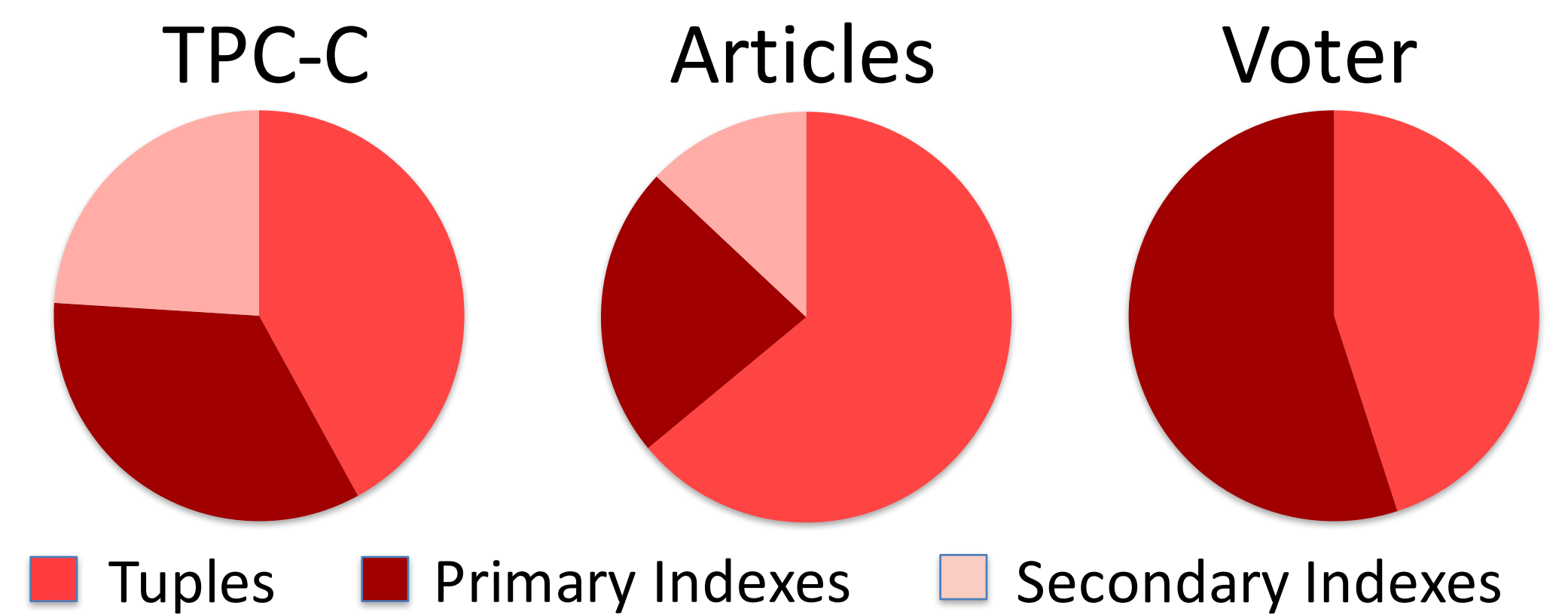
Reducing the Storage Overhead of Main-Memory

OLTP Databases with Hybrid Index

Huanchen Zhang, David G. Andersen, Andrew Pavlo, Michael Kaminsky (Intel), Rui Shen, Lin Ma

Motivation

- Memory is still a limiting resource
- The high throughput and low latency of main-memory databases is only available if the working set fits in memory



Problem: index structures are too LARGE

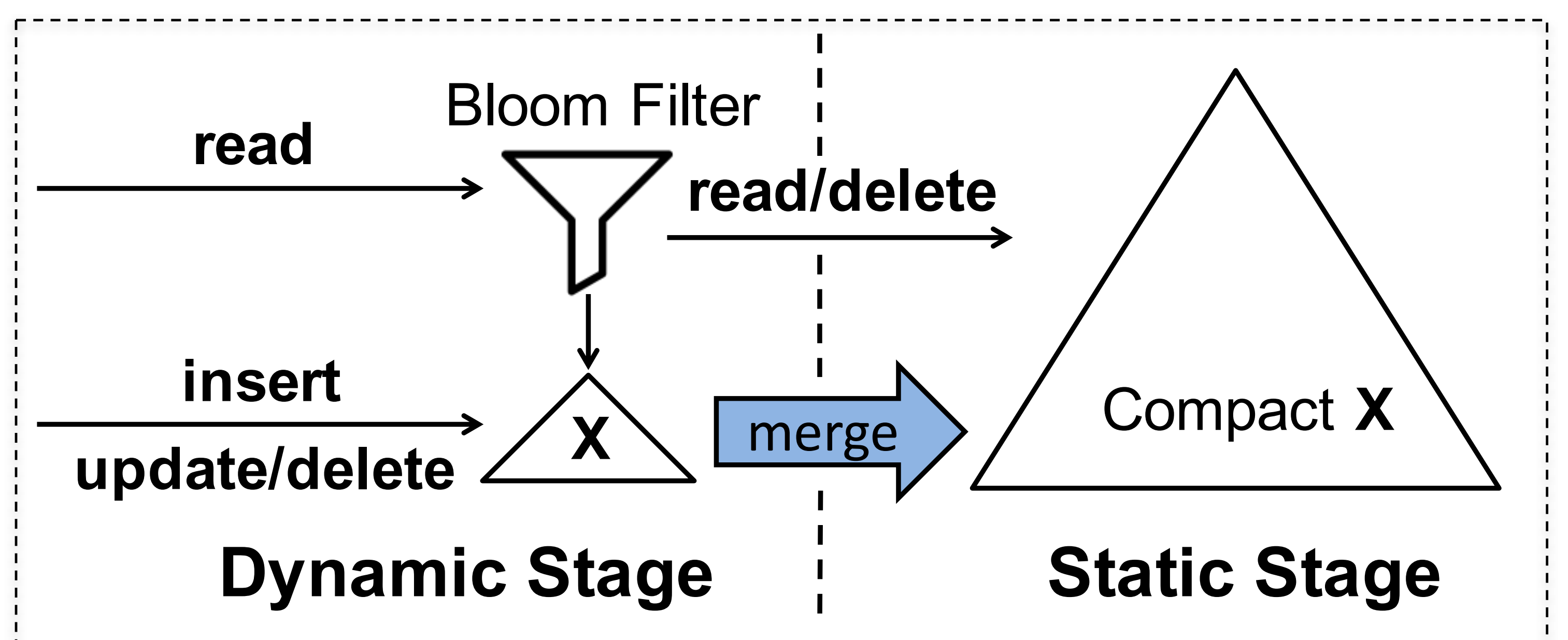
Hybrid Index: a Dual-stage Design

Benefit 1: Space-Efficient

Most entries are stored in the more compact static-stage structure

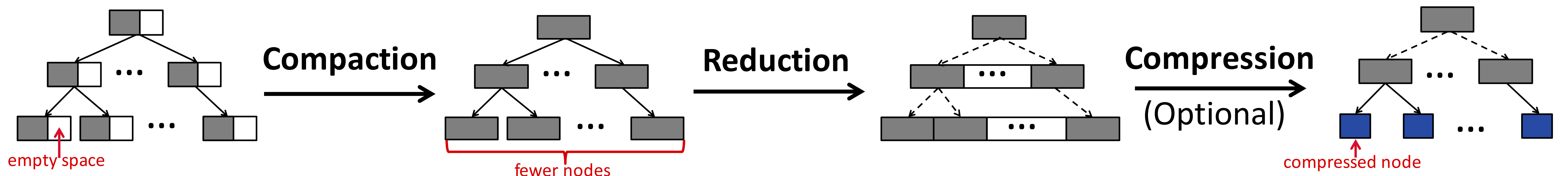
Benefit 2: Skew-Aware

Hot and fresh entries are stored in dynamic stage for fast accesses while aged entries are migrated to static stage for occasional look-ups



Dynamic-to-Static Rules

Convert a dynamic data structure to a compact, read-optimized structure



Evaluation: Resource Saving + Capacity Expansion

Resource Saving

YCSB-based Micro-benchmark

Key: email (average length = 30B)

Value: 64-bit int (pointers)

50M initial entries,

10M queries (Zipf skewed)

Single-threaded

Machine

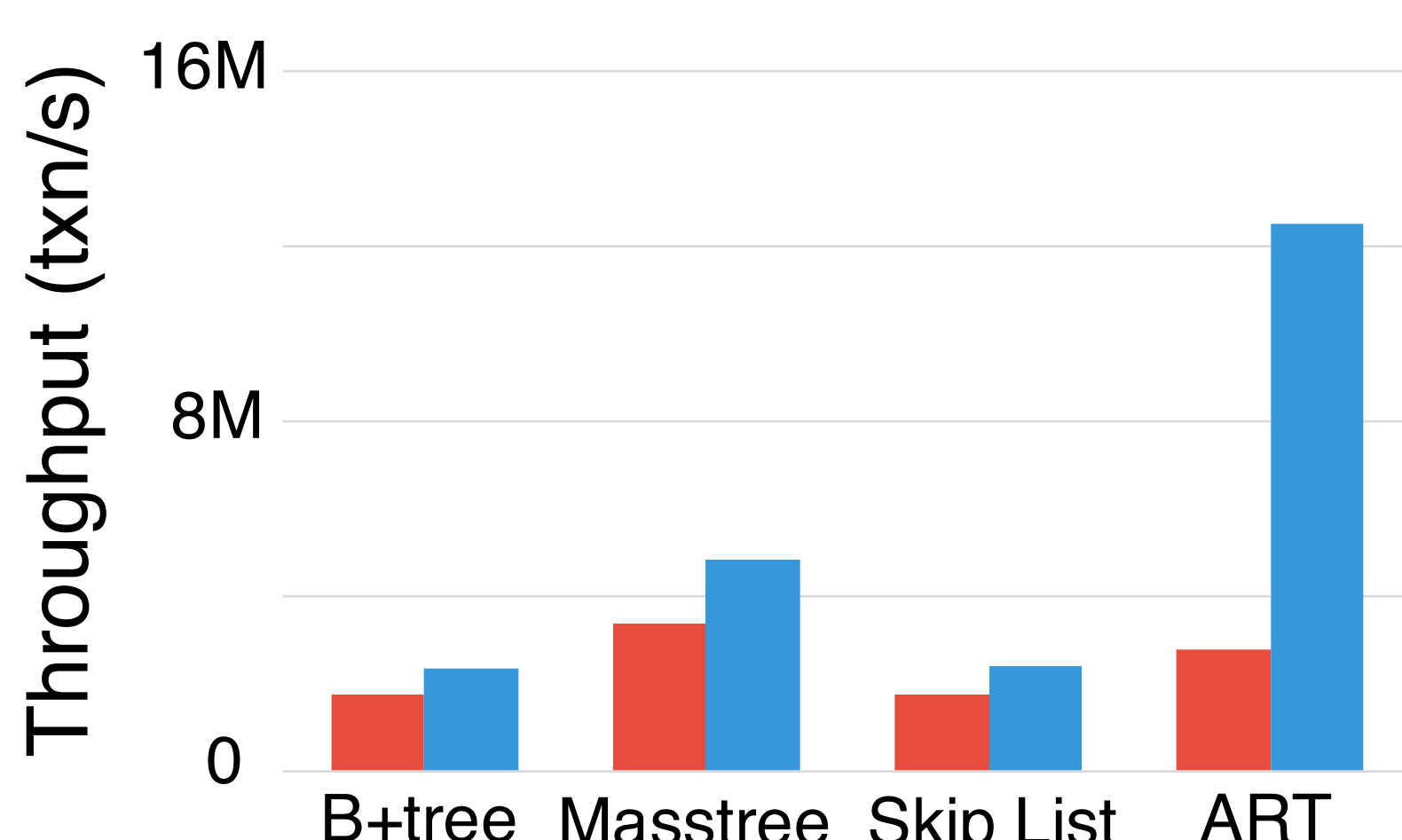
2xIntel Xeon E5-2680 v2 @ 2.80 GHz

4x32 GB DDR3 RAM

256 KB L2, 26 MB L3

500 GB, 7200 RPM, SATA

Read/Update (50/50)



Insert-only



Capacity Expansion

Full DBMS Evaluation

H-Store: an open-source, main-memory OLTP DBMS

Anti-caching: a database-managed tuple-level paging technique [VLDB'13]

Benchmark: TPC-C

8 server threads, 8 client threads

Memory Limit = 5GB

The benchmark executes for 12 minutes each run

