

Paper Discussion

H-Store & Anti-Caching

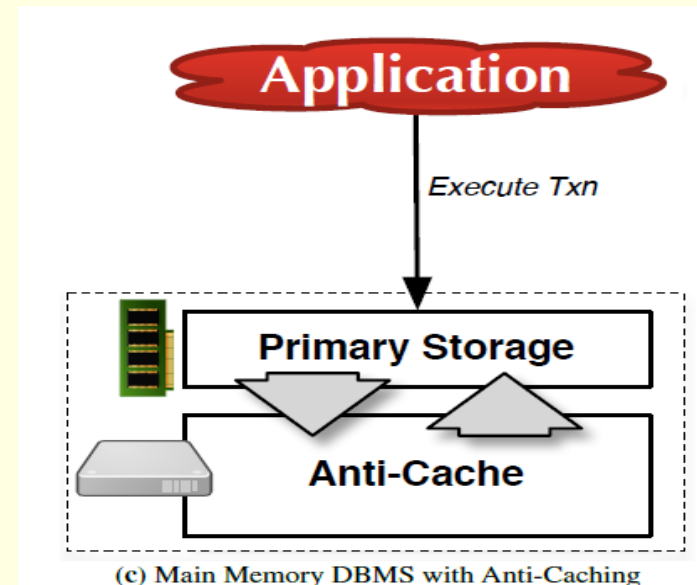
Discussant: Liyun Zhang(lz23)

Quick Review of H-Store

- H-Store is a new in-memory database prototype for **OLTP** with high throughput.
 - Single-threaded execution model, no lock-based concurrency control
 - Distributed cluster of shared-nothing machines with data partition
 - Optimized for OLTP transactions which are mostly short-lived and light-weight

Quick Review of Anti-Caching

- A new architecture to extend H-Store to support large databases which cannot fit into memory
 - Primary storage is main memory
 - Cold data is moved to disk and hot data remains in memory
 - High performance for highly skewed OLTP application



H-Store:

- All database accesses must be performed through pre-defined store procedures.
- Therefore not well-suited for ad-hoc applications.

H-Store:

- Best suited for short-lived light weight transactions
- Not well-suited for applications with complex transactions such as OLAP transactions
 - OLAP transaction often involve full-table scan, which needs data From multi-partition
 - But Multi-partition transaction is relatively slow in H-Store

H-Store:

- It requires applications designers to partition the data in advance in such a way that conflicts are minimized
- This requires application designer have good understanding of their data access patterns in order to avoid conflicts
- It need a knowledgeable human to carefully code the transaction class

H-Store:

- What if the workload is not partitionable?
- If the workload partitions poorly, the transactions need to touch several partitions and performance of transactions will deteriorate considerably

Anti-Caching:

- Anti-Caching performs best if data access is skewed
- If data is not skewed, the performance reduces dramatically
- So Anti-Caching is not suitable for applications with balanced data access pattern

Anti-Caching:

- Anti-Caching system assumes that the scope of queries fit in main memory
- So what if some queries requires data larger than main memory?

Anti-Caching:

- Anti-Caching system assumes that all indexes fit in main memory
- This restricts use of large second index!

Anti-Caching:

- In pre-pass phase of Anti-Caching, the transaction executes as normal. When pre-pass has finished execution, the DBMS rolls back any changes that the transaction made at any partition.
- Since DBMS will roll back any changes after pre-pass phase, why not no change in the first place?

Anti-Caching:

- Sample rate of LRU chain has significant impact on system performance
- The sample rate may be hard to decide in advance
- It needs to be tuned for specific database application