Dynamic and **Transparent** Data Tiering for In-Memory Databases in Mixed Workload Environments

Relevance Based Data Partitioning

Martin.Boissier & Carsten.Meyer @HPI.de

Motivation

- In enterprise systems data is typically kept up to 10 years
- High data access skewness (transactional & analytical workloads)
- Performance (by limiting query execution on hot data only)
- Cost (reduce allocation of main memory as required)

Preconditions

- Analysis of a big, real-life database systems incl. workload traces
- EMC cooperation provided 2nd storage device and dedicated API
- HYRISE - open source, hybrid, main-memory data storage engine

Relevance Based Partitioning

1. Individually split table columns,
2. Periodically (offline) into a hot and cold data segment
3. Based on workload characteristics
4. Using different storage classes for hot / cold data
5. Providing deterministic and transparent data access

Challenges

1. Data classification for mixed-workload: no blocks, no caching mechanism for hot data

2. Deterministic data access (e.g. hot only column scans)

Hot Data Views (HDV)

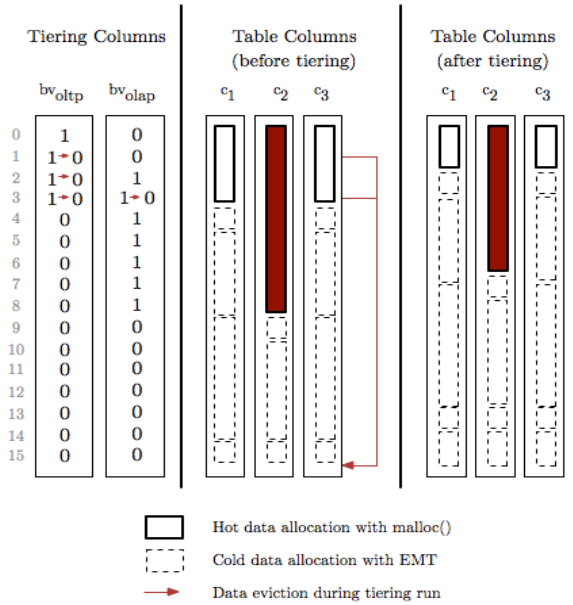
- Defined periodically on sampled historical workload statistics, e.g.
  - SELECT * FROM <tbl> WHERE id (8873, …)
  - SELECT a, b, c FROM <tbl> WHERE b > 2014-05-25
- Classify hot data (horizontally and vertically)
- Used to determine query execution (data access) strategy

EMT API / NAND Flash

- PCIe NAND Flash device
- EMT API (linux kernel module)
  - alternative to mmap, bypassing OS
  - optimized for flash device
  - deterministic caching (coloring), read-ahead strategies
- Used for cold data segments only

Implementation

- **Tiering Run:** Periodical optimization of the data allocation
- **Tiering Columns:** Persisted information of data allocation
- **Tiering Check:** View matching agains HDVs before query execution



Evaluation / Benchmark

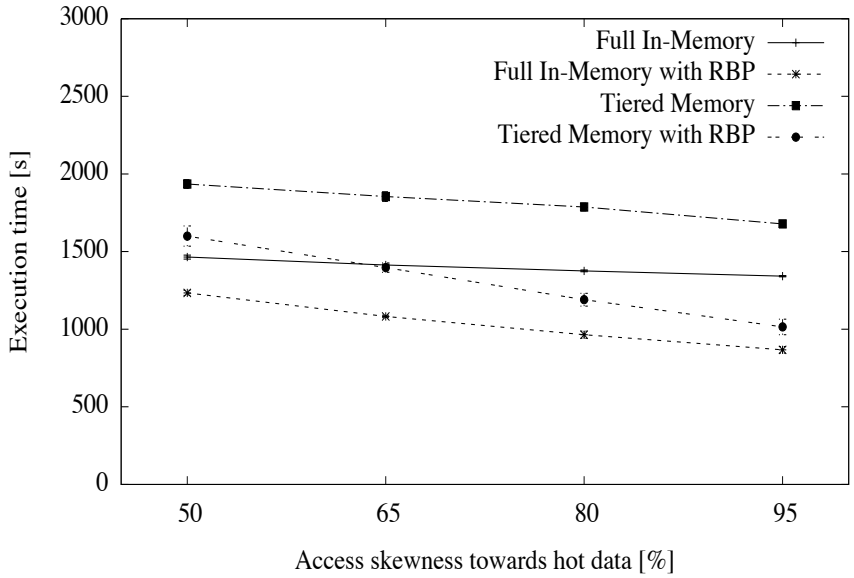
- 3x OLAP queries,100x OLTP queries
- Different hot query ratio (50%, 65%, 80%, 95%)
- 20% hot data

- **Full In-Memory:** 1 GB data allocated in main memory - no memory pressure
- **Full In-Memory with RBP:** query optimization due to RBP
- **Tiered Memory with RBP:** 0,2 GB hot data in main memory + 10% cache size for remaining 0,8 GB data = 0,28 GB main memory
- **Tiered Memory:** 0,28 GB main memory for cache

Results / Insights

While workload skewness towards hot data increases:

- There is no performance impact for *Full In-Memory* setup
- *Tiered Memory* leverages increasing skewness
- *Relevance Based Partitioning* improves Full In-Memory and Tiered Memory setup



Ongoing research

- Hot Data Views:
  - Automated generation based on workload statistics
  - Definition using application knowledge
- Separate dictionaries for hot and cold data
- Tiering check for join queries

Related Work

- R. Stoica and A. Ailamaki. Enabling efficient os paging for main-memory OLTP databases. In Proceedings of the Ninth International Workshop on Data Management on New Hardware, DaMoN '13, pages 7:1–7:7, New York, NY, USA, 2013. ACM.

- A. Eldawy, J. J. Levandoski, and P. Larson. Trekking through siberia: Managing cold data in a memory-optimized database. PVLDB, 7(11):931–942, 2014.

- B. Höppner, A. Waizy, and H. Rauhe. An approach for hybrid-memory scaling columnar in-memory databases. In International Workshop on Accelerating Data Management Systems Using Modern Processor and Storage Architectures - ADMS 2014, Hangzhou

Hasso Plattner Institut

Thank you