The Case for Heterogeneous HTAP

Raja Appuswamy, Manos Karpathiotakis, Danica Porobic, and Anastasia Ailamaki

Data-Intensive Applications and Systems Lab

EPFL

HTAP – the contract with the hardware

Hybrid OLTP & OLAP Processing

High-throughput OLTP Low-latency OLAP

HTAP DBMS

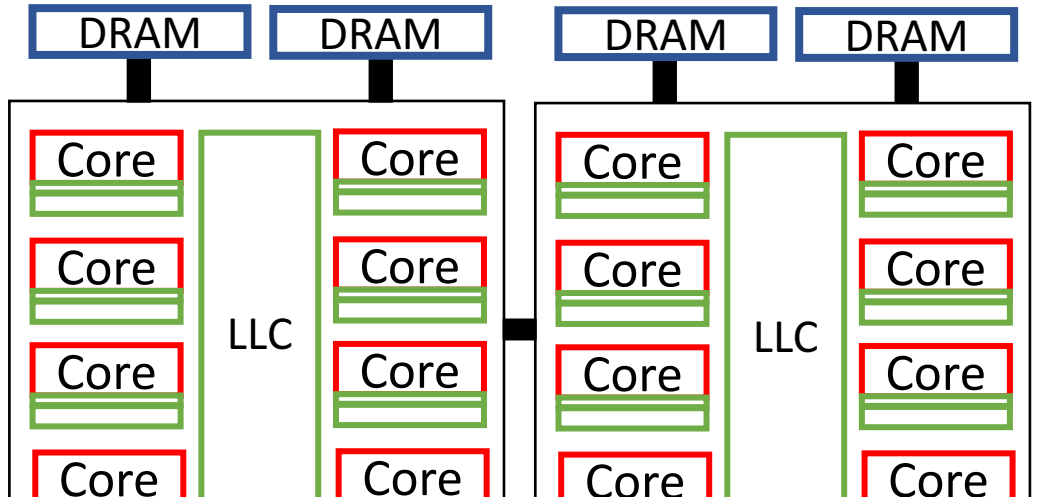
Database Fresh data

HTAP on multicores

Massive parallelism => high concurrency

Global shared memory => data sharing

System-wide coherence => synchronization

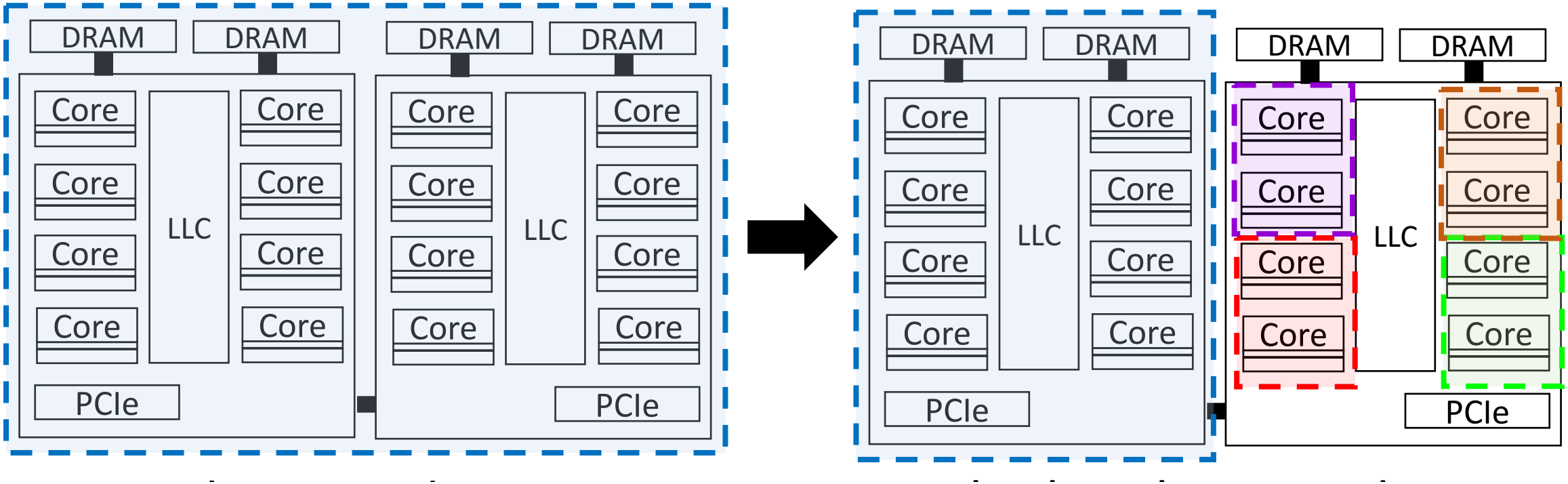


Necessary for current systems

Shifting hardware landscape (1): Specialization of CPUs

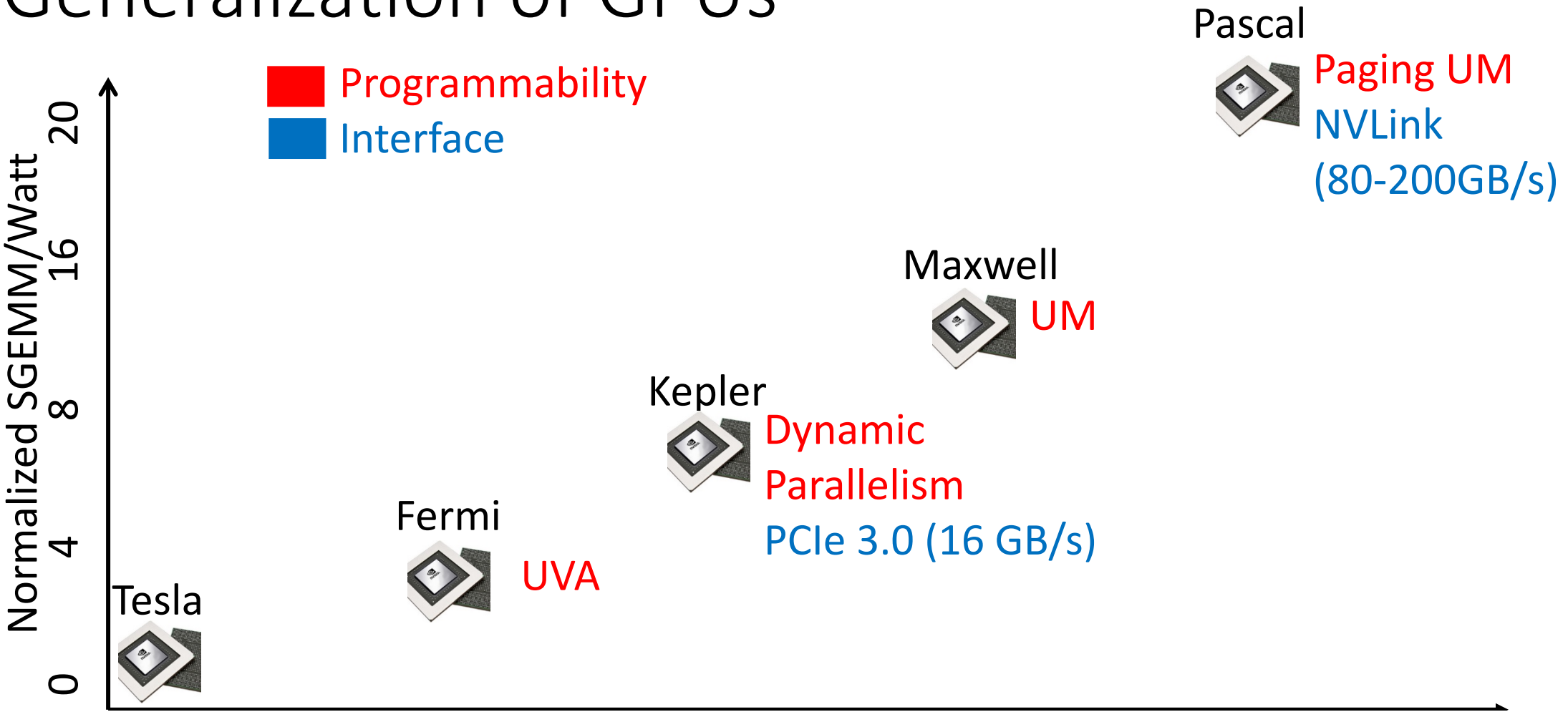
Multisocket multicores

Intel SCC, ARM v8, Cell SPE



CPUs: general-purpose → customizable features

Shifting hardware landscape (2): Generalization of GPUs



GPUs: Niche accelerators → general-purpose processors

Emerging hardware: Revisiting the contract

~~Current~~Emerging hardware

- ~~Homogeneous~~ Heterogeneous parallelism
  - Task-parallel CPUs
  - Data-parallel GPUs
- ~~System-wide~~ Relaxed cache coherence
  - OS (FOS), FS (Hare)
  - runtimes (Cosh)
- Global shared memory
  - Unified address space

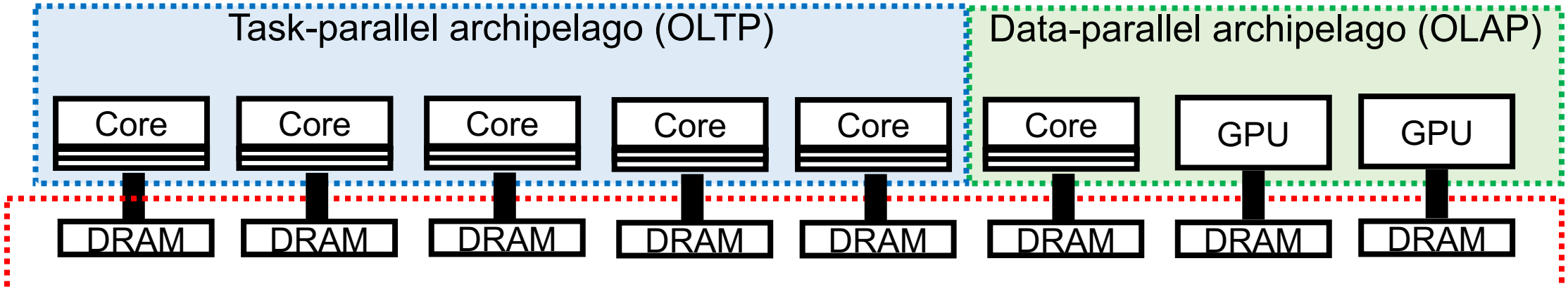
HTAP software

- Cannot exploit heterogeneity
  - HTAP across processors
- Shared-everything OLTP: N/A
  - No synch. sans coherence
- Server as distributed system
  - Fails to exploit shared memory

Clean slate redesign in order

Heterogeneous HTAP (H^2TAP): Caldera

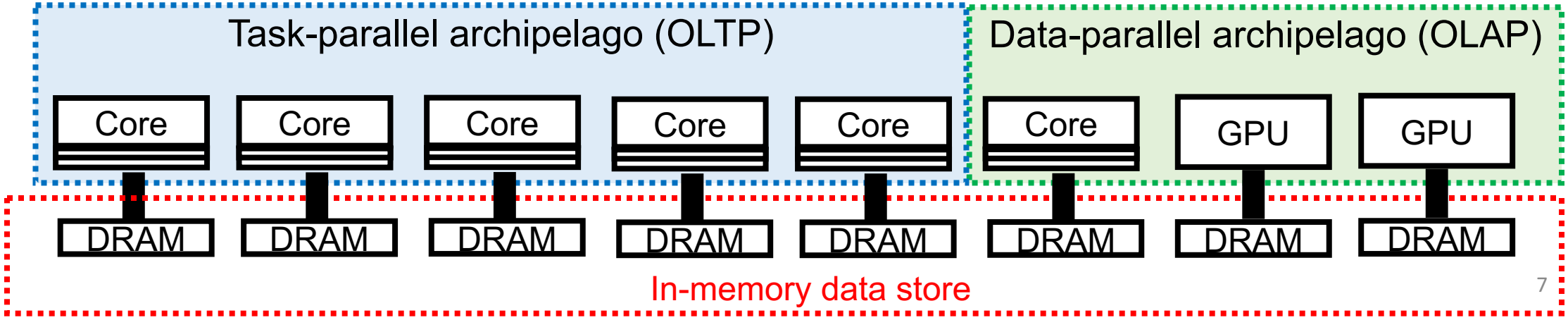
- Store data in shared memory
- Run OLTP workloads on task-parallel archipelago
- Run OLAP workloads on data-parallel archipelago



Loose job-to-core assignment exploits heterogeneity

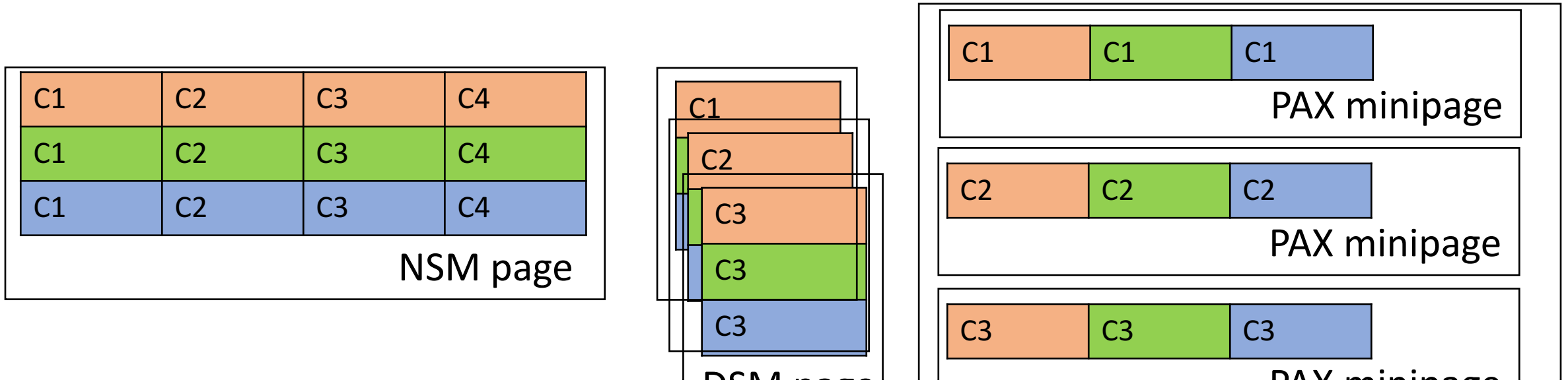
H^{2}TAP Challenges

- Store data in shared memory
  - **Choose optimal data layout**
- OLTP on task-parallel archipelago
  - **Make up for (lack of) cache coherence**
- OLAP on data-parallel archipelago
  - **Share transactionally-consistent snapshots across processors**



Data layout

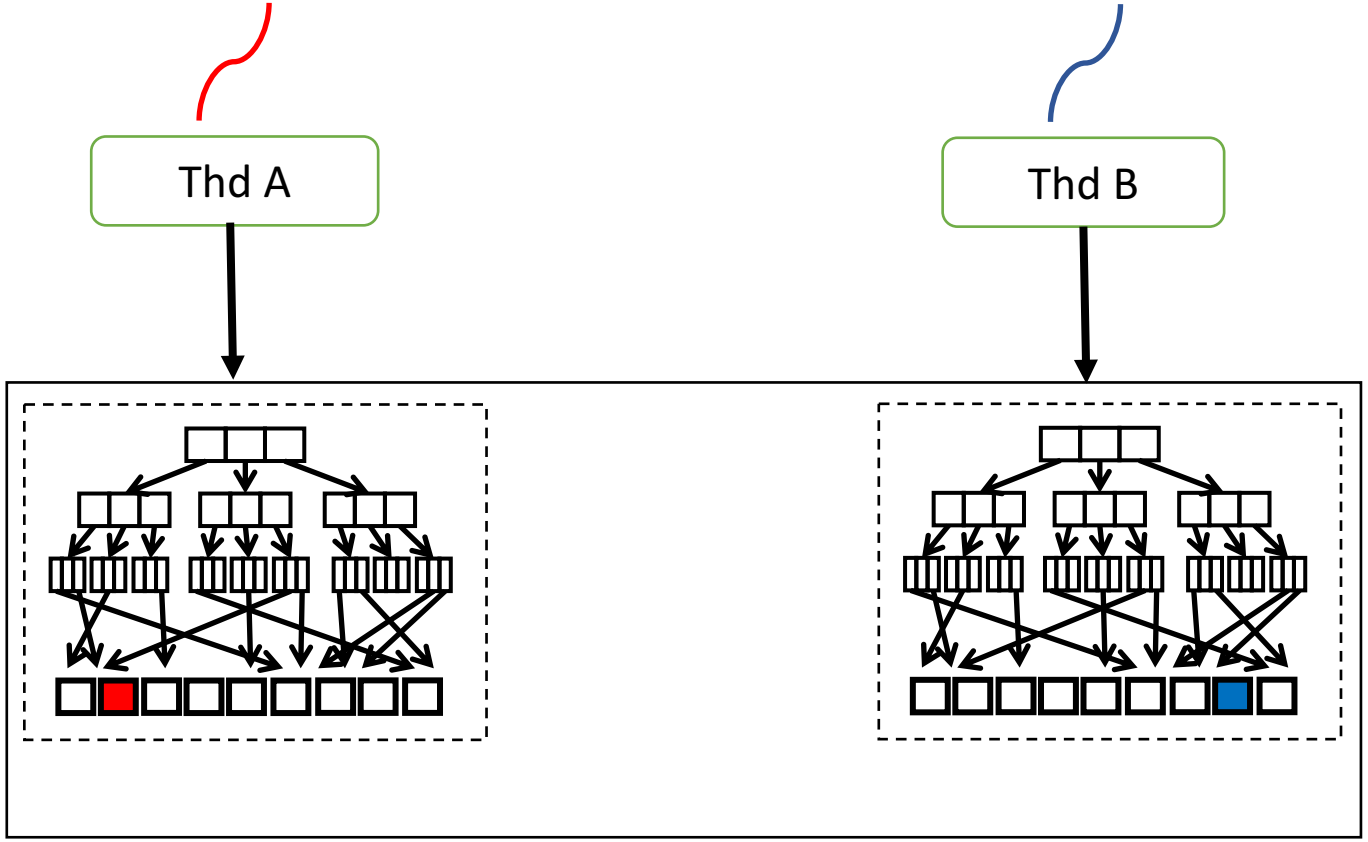
- Need to minimize PCIe data transfer to GPU
- Data access on GPU should be sequential to enable “coalescing”
- Caldera implements NSM, DSM, and PAX



PAX fits GPUs best (PCIe & coalesced accesses)

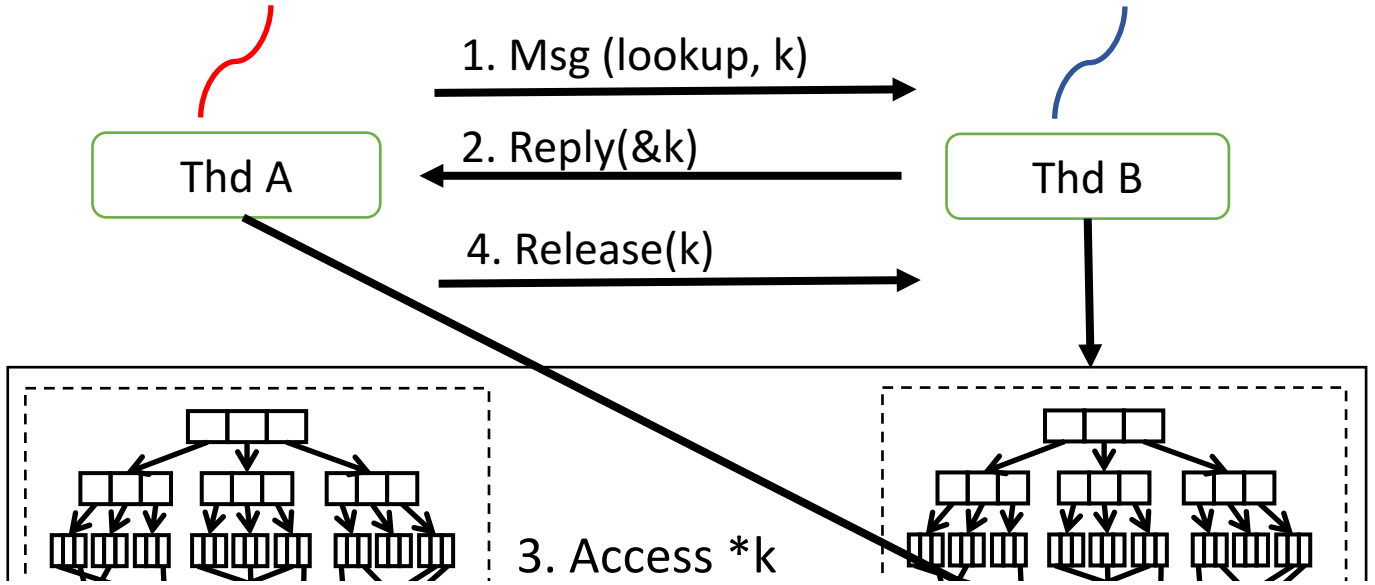
OLTP without cache coherence

- Use Data-Oriented Transaction Execution principles
  - Thread-to-data assignment leads to partitioned data, metadata (2PL, index)



OLTP without cache coherence

- Use explicit messaging instead of implicit latching
- Exploit shared memory by exchanging pointers instead of data



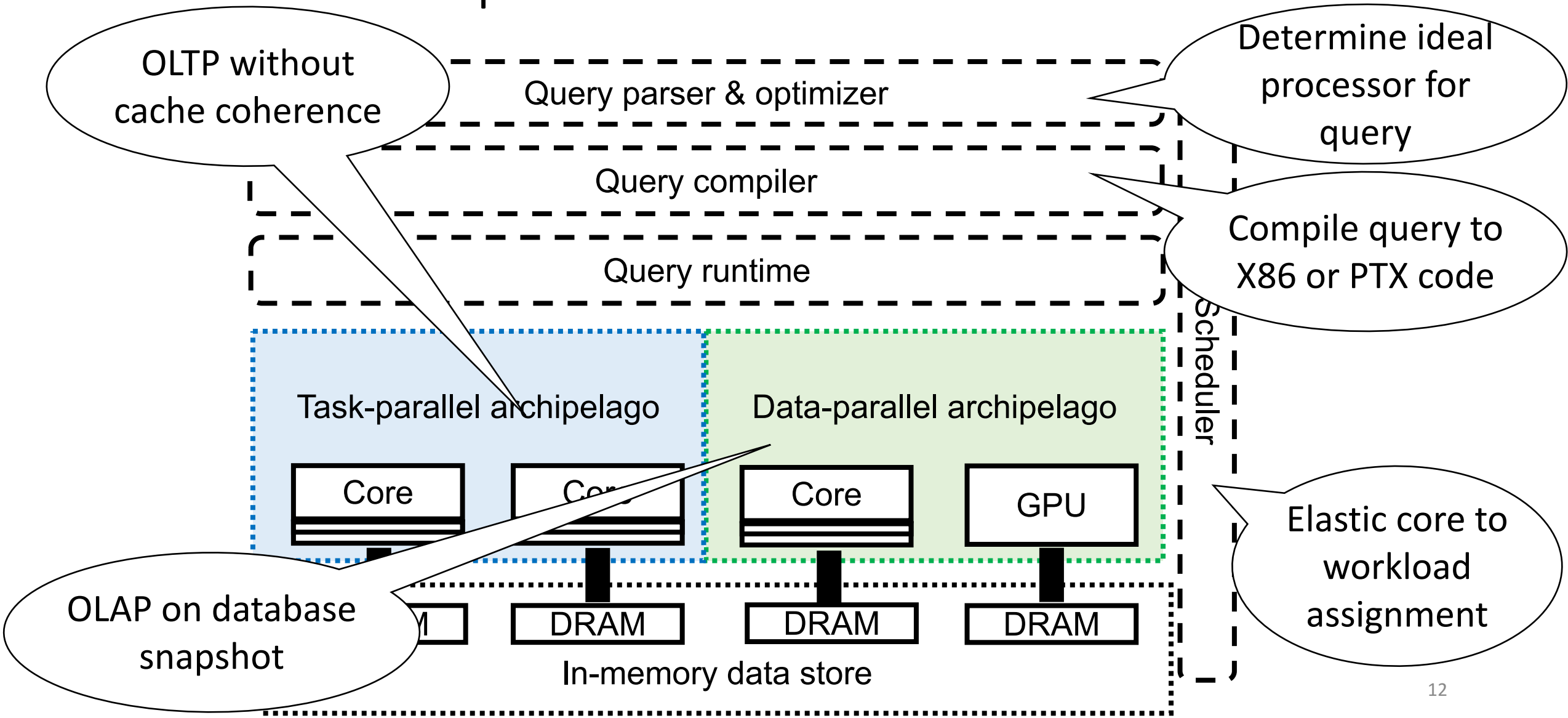
Enforce coherence in software

Transactionally-consistent data sharing

- Data sharing across workloads
  - Use Unified Virtual Addressing (UVA) for CPU—GPU sharing
- Consistent data sharing via hardware snapshotting (ex: Hyper)
  - CUDA runtime restricts use in H^2TAP context
- Caldera supports lightweight software snapshotting
  - OLAP queries run on immutable snapshot
  - Copy-on-write performed by update transactions

Snapshots across GPU-CPU archipelagos

Caldera blueprint



Experiments

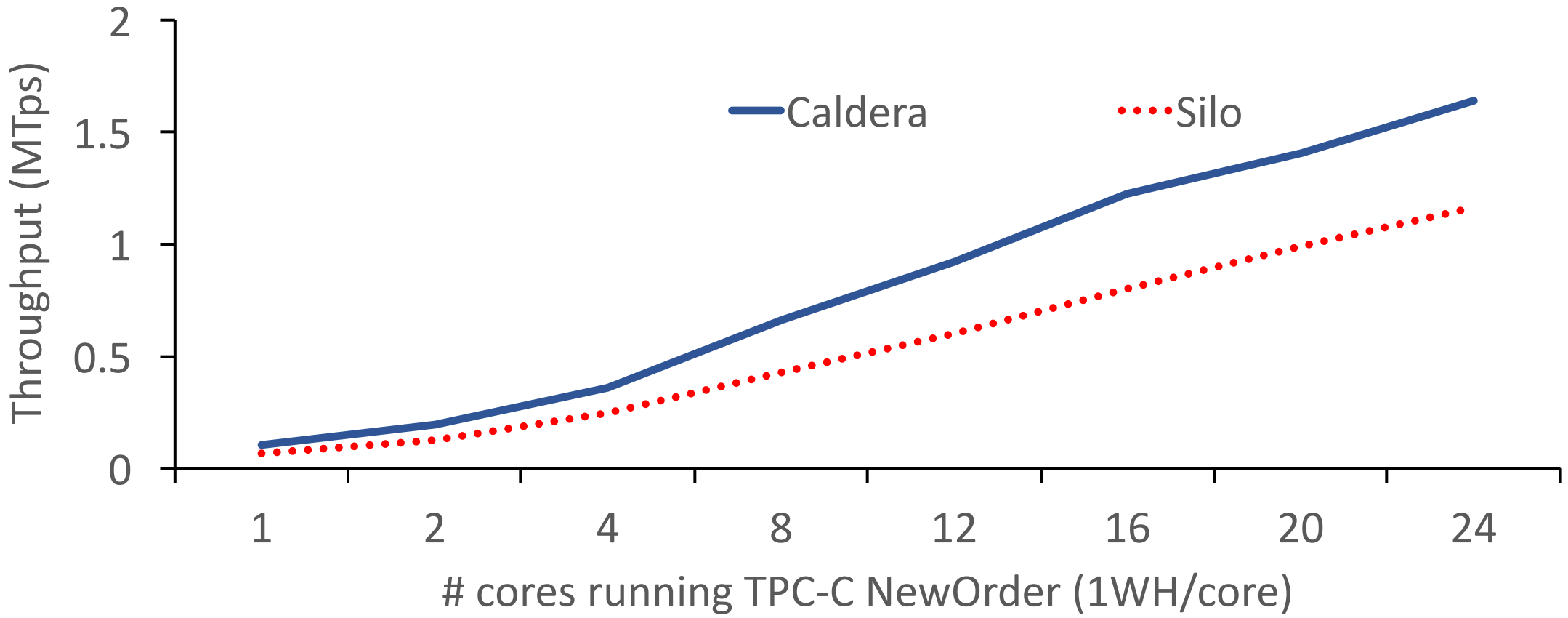
Setup

- Two 12-core Intel Xeon E5-2650L v3 CPUs, 256GB RAM
- GeForce GTX 980 GPU (PCIe 3.0) with 4GB memory
- TPC-C, TPC-H, YCSB in various scale factors
- Silo, MonetDB, DBMS-C

Goals

- Message passing and Software snapshotting overhead
- PAX performance compared to NSM and DSM on GPUs
- Caldera performance compared to state-of-the-art

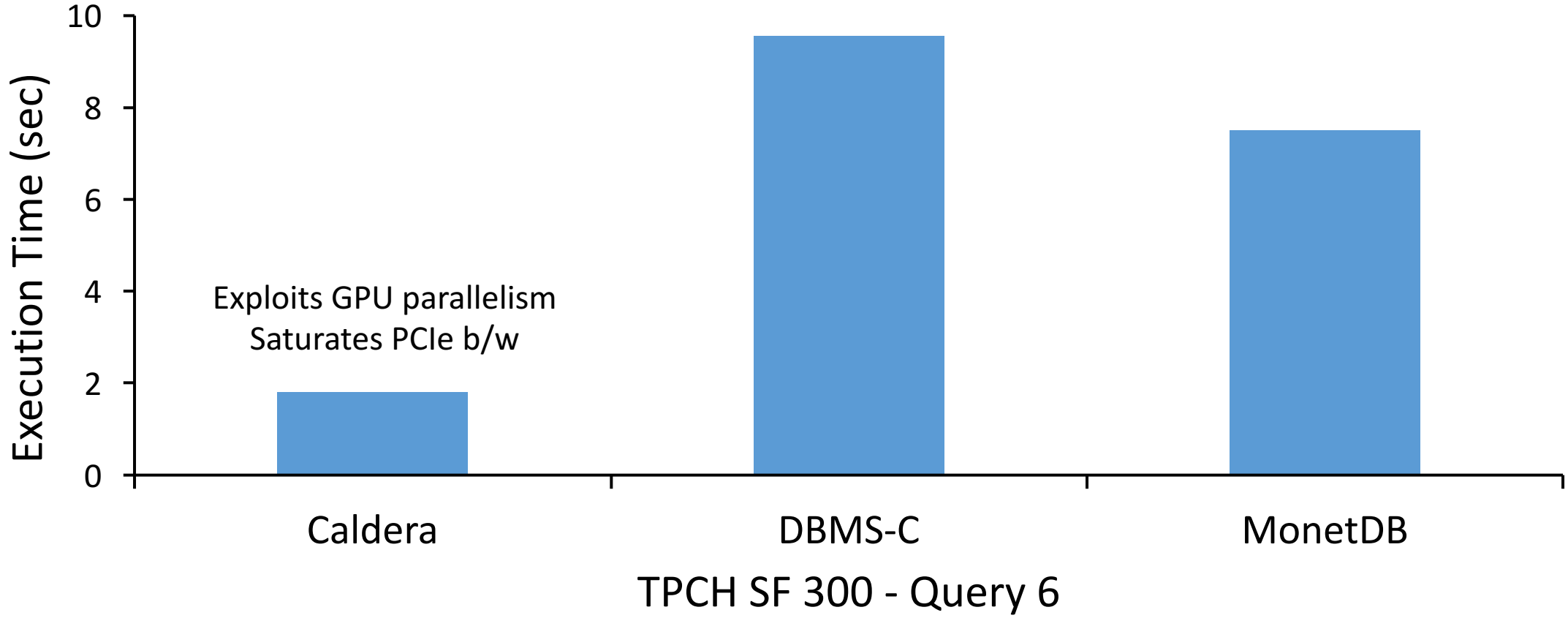
OLTP throughput



Message passing-based design scales well

Better code & data locality (partitioning), no synchronization overhead

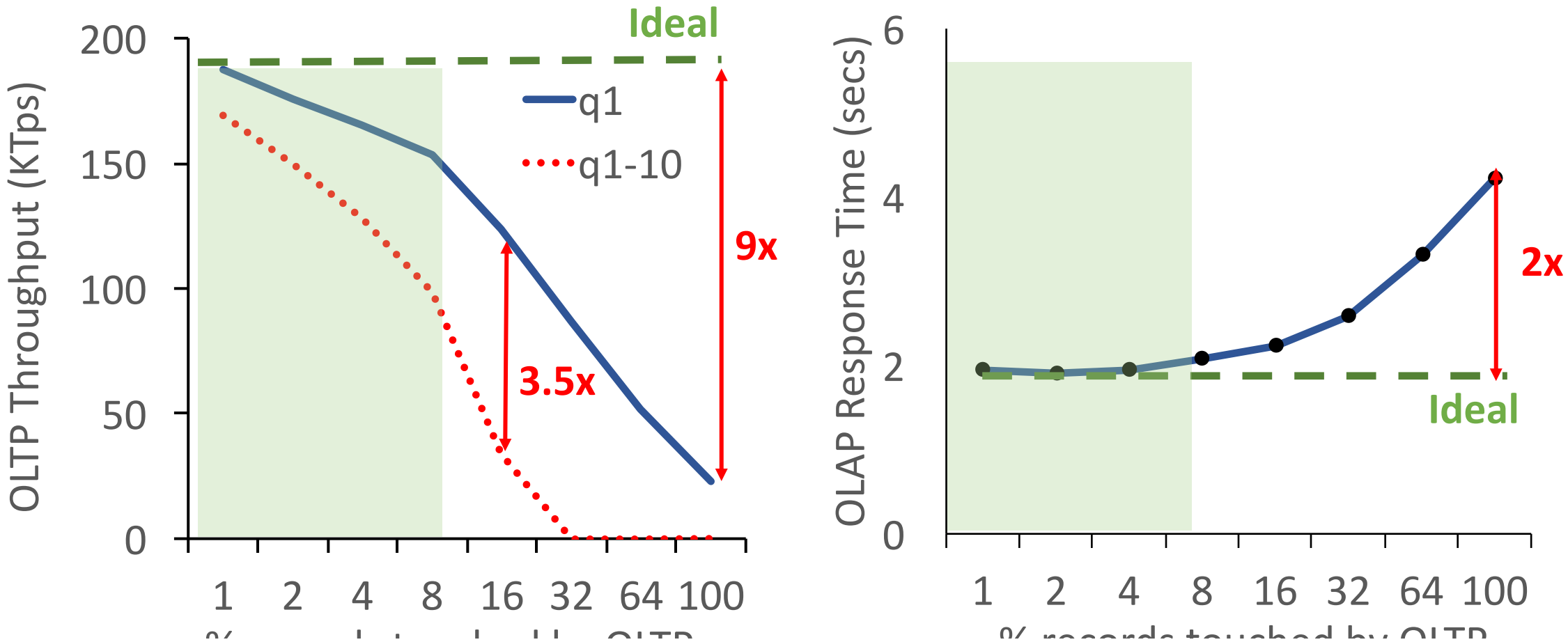
OLAP response time (incl. data movement)



Bounded by PCIe bandwidth (12GB/s)

Emerging interconnects (NVLink): 80-200 GB/s

Impact of snapshotting



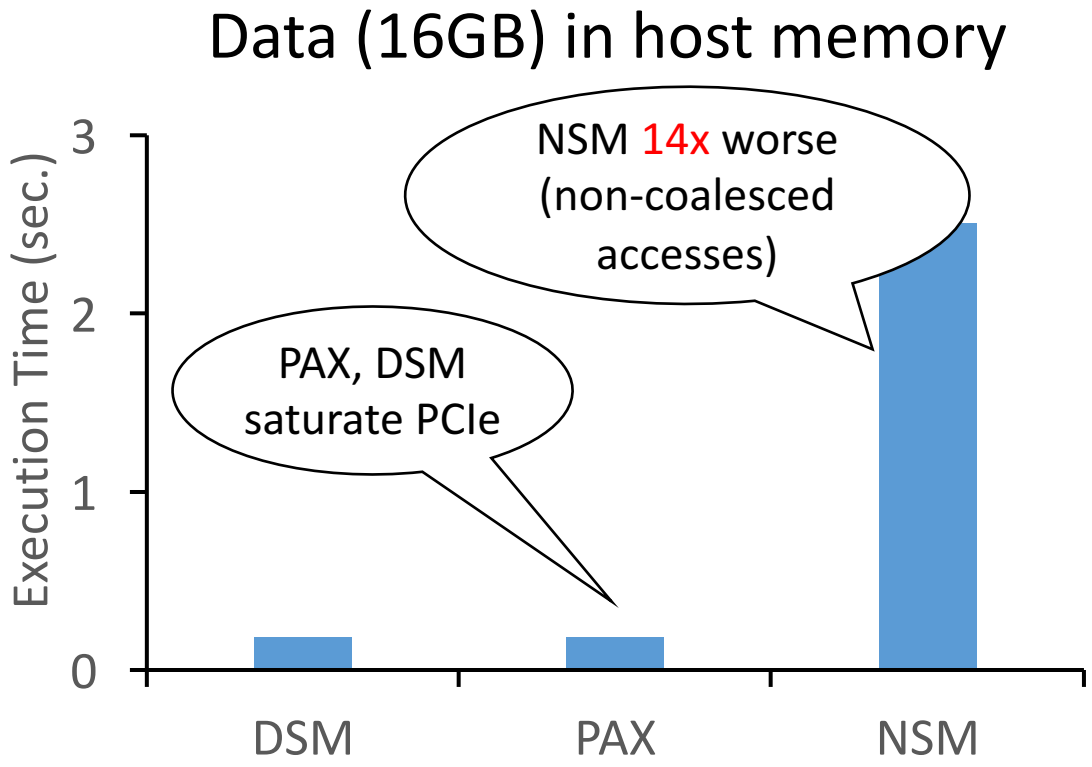
Limitation: Software shadow copying imposes a high overhead

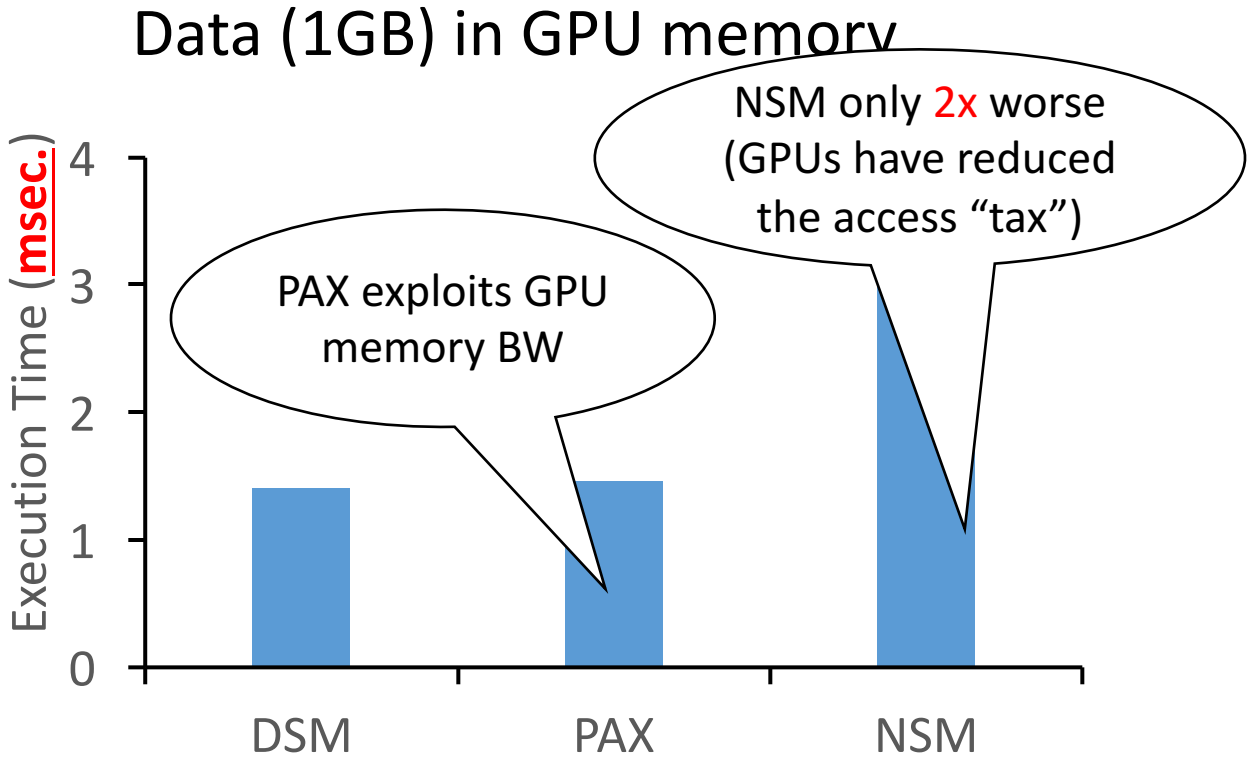
Possible fix: data classification, snapshot sharing, h/w acceleration

Impact of data layout

1 table (*i1 integer, i2 integer, …. i16 integer*)

SELECT SUM(colA + colB) FROM table





Hybrid layouts like PAX a good fit for H²TAP

Conclusion

- Hardware architecture is changing
  - New opportunities: massive parallelism, fast interconnects
  - New challenges: heterogeneity, relaxed coherence
- Databases can and should exploit hardware trends
  - Exploit hardware heterogeneity in their core architecture design
  - Decouple system-wide coherence from shared memory
- Time to move from HTAP to H^2TAP
  - H^2TAP architecture: revisit age-old h/w—s/w contract
  - Caldera: Preliminary prototype to prove that H^2TAP is possible