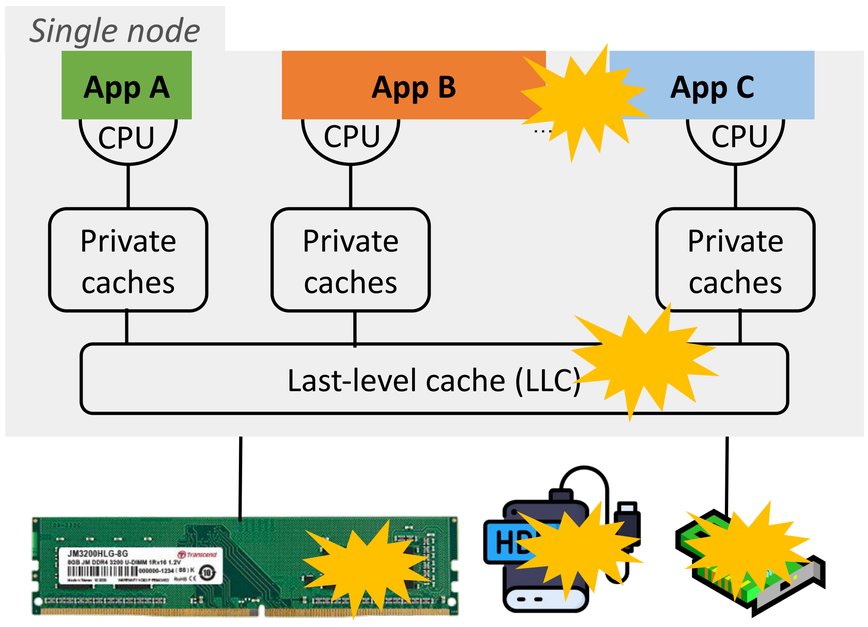
Efficient Scheduling for Multiple Database Systems on Shared Hardware

Yi Jiang, Anastasia Ailamaki
firstname.lastname@epfl.ch

1. Cloud economics demands multi-tenancy



Performance

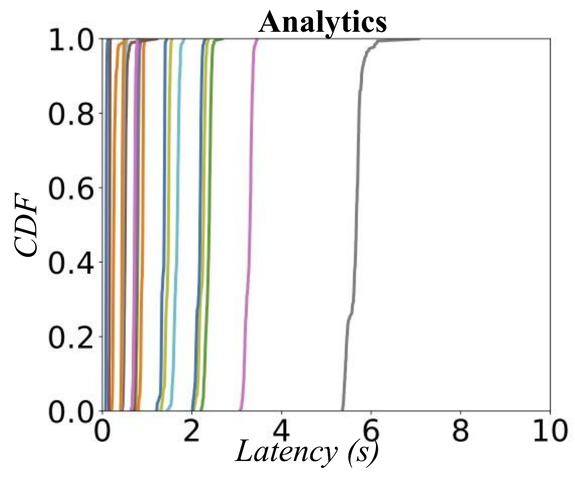
Tradeoff

Resource Efficiency

Goal:
- Improve resource efficiency
- Meet SLAs

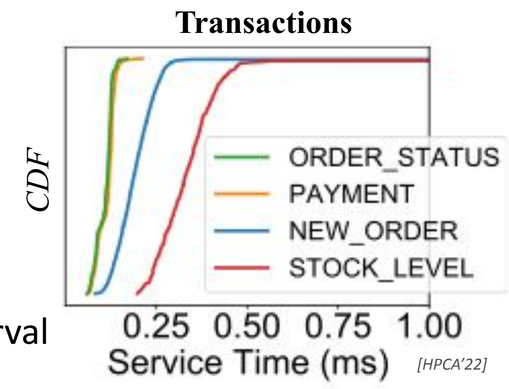
Under co-location

2. Database workloads are diverse



TPC-H in DuckDB, sf=30
Intel Xeon E5-2680L v4, 2x14 cores
70MB LLC, 256GB memory

- Analytics
  - **.1s - hours**
  - Highly variable
  - ms-scale scheduling interval
- Transactions
  - **10 - 100's us**
  - us-scale scheduling interval



≥ 3 orders of magnitude difference in service time

3. Co-located database systems cannot fully utilize the underlying hardware

Existing resource schedulers
- Avoid application co-location at potential interference
  - *Bolt, Quasar, Borg, Heracles*
- Partition shared resources at runtime to reduce interference
  - *Ubik, Rubik, PARTIES, Caladan*

However
- Fixed decision interval *(PARTIES, CLITE, Aurora Serverless)*
- Not considering the relative importance:
  - Task latency and resource partitioning overhead
- Not adjusting the full resource spectrum *(Caladan)*

Co-located database systems cannot fully utilize the underlying hardware with existing resource schedulers

Resource partitioning mechanisms have various overheads

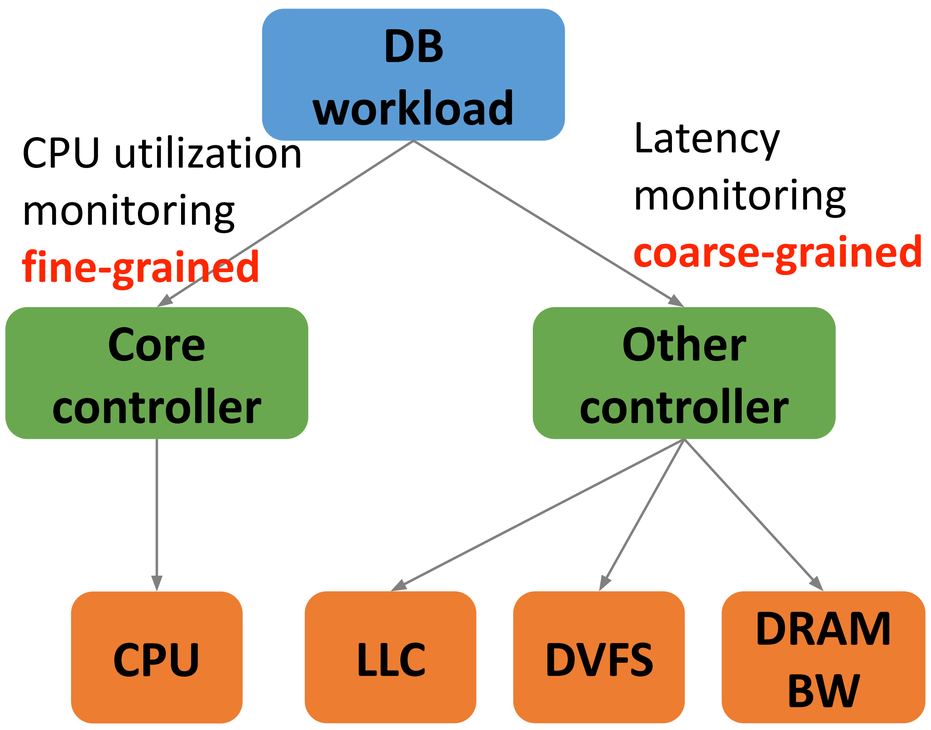
Resource	Isolation Tool	Time to take effect
CPU core affinity	taskset	10-100's us
CPU core frequency	ACPI frequency driver	100's us
LLC ways	Intel CAT	ms-scale (cache eviction/refilling)
Memory capacity	Linux's memory cgroups	ms-scale (memory refilling)
Memory bandwidth	Intel MBA	ms-scale

Fast and adaptive core allocation for Tx

+

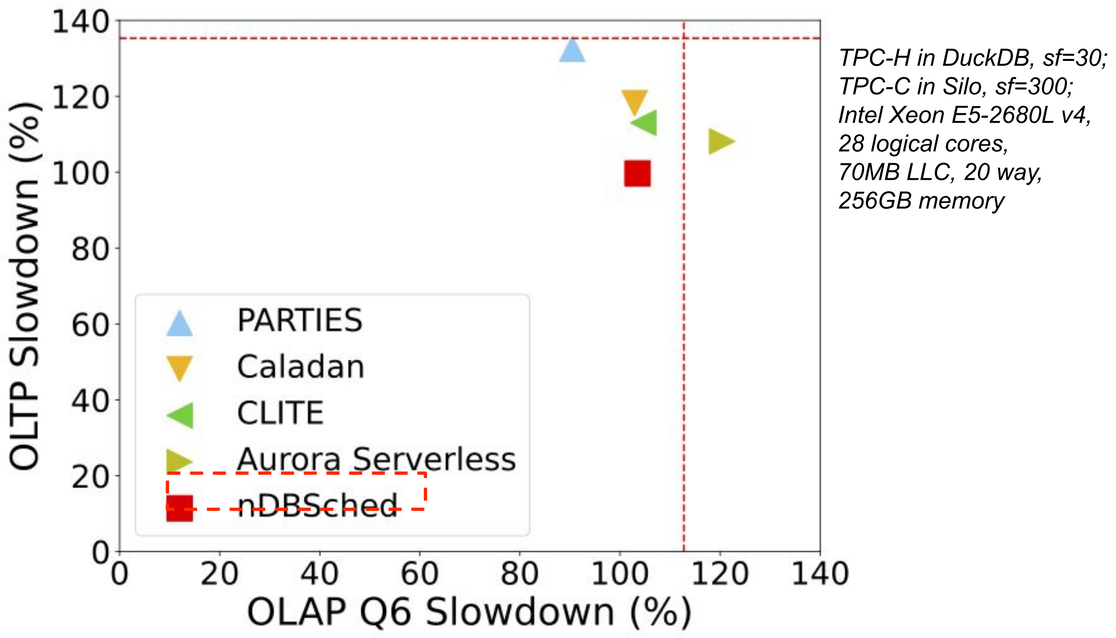
Adjust the full resource domain adaptively

4. Separate control signals/loops



Adjust core allocation with separate control loops

5. Better runtime resource allocation



TPC-H in DuckDB, sf=30;
TPC-C in Silo, sf=300;
Intel Xeon E5-2680L v4,
28 logical cores,
70MB LLC, 20 way,
256GB memory

6. Efficient resource scheduling for DB systems under co-location

- Separate control loop for multiple resources with various decision intervals
- Relative importance between task latency & resource adjustment overhead

DATA-INTENSIVE APPLICATIONS AND SYSTEMS

EPFL