ATraPos: Adaptive Transaction Processing on Hardware Islands

Danica Porobic, Erietta Liarou, Pınar Tözün, Anastasia Ailamaki

Data-Intensive Application and Systems Lab, EPFL
Scaling up OLTP on multisockets

Multisocket servers are severely under-utilized
Multisocket multicores

Communication latencies vary by an order-of-magnitude...
OLTP on Hardware Islands

Shared-everything  Island shared-nothing  Shared-nothing
Scaling up on a 8-socket machine

Throughput (MTPS)

Number of sockets

- Shared-nothing
- Island shared-nothing
- Shared-everything

Islands significantly challenge scalability

8 socket x 10 core
800K row dataset
Probing one row
Physical partitioning for Islands

Throughput (KTPs)

- Shared-nothing
- Island shared-nothing
- Shared-everything

No configuration is optimal for all environments

4 socket x 6 core
240K row dataset
Updating 10 rows
OLTP on Hardware Islands

- **Shared-everything**
  - ✔ Stable
  - ✗ Not optimal

- **Island shared-nothing**
  - ✔ Robust middle ground

- **Shared-nothing**
  - ✔ Fast
  - ✗ Sensitive to workload

### Challenges
- Optimal configuration depends on workload and hardware
- Expensive repartitioning due to physical data movement

**ATraPos: hardware and workload-aware shared-everything adaptive system**
ATraPos: Adaptive Transaction Processing

• No unnecessary inter-socket synchronization

• Workload & hardware-aware partitioning

• Lightweight monitoring and repartitioning

ATraPos: hardware and workload-aware shared-everything adaptive system
Outline

• Impact of Hardware Islands on OLTP

• ATraPos
  – Avoiding unnecessary synchronization
  – Workload & hardware-aware partitioning & placement
  – Lightweight monitoring & repartitioning

• Summary
Critical path of transaction execution

Many accesses to shared data structures
Perfectly partitionable workload

Throughput (MTPS) vs. Number of sockets

- **Shared-nothing**
- **Centralized shared-everything**

8 socket x 10 core
800K row dataset
Probing one row

- Accessing centralized data structures limits scalability
PLP: Physiologically partitioned SE*

System state is still shared

Perfectly partitionable workload

Throughput (MTPS)

- Shared-nothing
- PLP
- Centralized shared-everything

Number of sockets

- 8 socket x 10 core
- 800K row dataset
- Probing one row

Inter-socket accesses to system state are a bottleneck
ATraPos: Island-aware SE

System state

threads

Core
Core
Core
Core

Core
Core
Core
Core
Perfectly partitionable workload

Throughput (MTPS)

- Shared-nothing
- ATraPos
- PLP
- Centralized shared-everything

Number of sockets

8 socket x 10 core
800K row dataset
Probing one row

Island awareness brings scalability
Outline

• Impact of Hardware Islands on OLTP

• ATraPos
  – Avoiding unnecessary synchronization
  – Workload & hardware-aware partitioning & placement
  – Lightweight monitoring & repartitioning

• Summary
Naive partitioning and placement

8 socket x 10 core
800K rows per table
Probing 1 row each from A and B

Cores are overloaded with contending threads
ATraPos partitioning and placement

8 socket x 10 core
800K rows per table
Probing 1 row each from A and B

Throughput (KTPS)

PLP
ATraPos HW-aware
ATraPos Load balanced

4.4x

Ignoring Islands -> synchronization overhead
ATraPos partitioning and placement

ATraPos: balanced load + reduced synchronization
Outline

• Impact of Hardware Islands on OLTP

• ATraPos
  – Avoiding unnecessary synchronization
  – Workload & hardware-aware partitioning & placement
  – Lightweight monitoring & repartitioning

• Summary
ATraPos monitoring

1. Initialize with naive scheme
2. Monitor the workload
3. Evaluate cost model:
   - Balance the load
   - Minimize synchronization

Probe A

Stats
ATraPos monitoring

1. Initialize with naive scheme
2. Monitor the workload
3. Evaluate cost model
4. Repartition

Probe A

Probe B

stats
Repartitioning Multi-Rooted B-trees

Splitting and merging B-trees accesses few pages
ATraPos repartitioning

Repartitioning of a table takes < 200ms

8 socket x 10 core
800K row table

Number of repartitioning actions

Repartitioning cost (ms)

- merge
- split
- rearrange (split+merge)
TATP - speedup over PLP

ATraPos improves performance of TATP by 3.1-6.7x
Adapting to workload skew

ATraPos detects skew and quickly adapts

Throughput (MTPS)

Time (s)

ATraPos

Static

Monitoring

Repartitioning

50% requests to 20% data

8 socket x 10 core
800K subscribers
TATP GetSubData
Adapting to changing workload type

ATraPos gracefully adapts to any change
Challenges

- Optimal configuration depends on workload and hardware
- Expensive repartitioning due to physical data movement

ATraPos

- Minimal inter-socket accesses in the critical path
- Workload & hardware-aware partitioning & placement
- Lightweight monitoring and repartitioning

Thank you!