The Memory Performance of DSS Commercial Workloads in Shared-Memory Multiprocessors

Pedro Trancoso, Josep-L. Larriba-Pey[*],
Zheng Zhang, and Josep Torrellas

I-ACOMA project, University of Illinois at Urbana-Champaign
trancoso,zzhang,torrella@cs.uiuc.edu
http://www.csrd.uiuc.edu/iacoma

[*] Universitat Politecnica de Catalunya
larri@ac.upc.es

Motivation

- Database applications: common workload;
- Memory performance studies: very few;
- Memory resident: cheaper RAM;

What is the shared-memory multiprocessor memory hierarchy behavior for complex database queries ?

Outline

- Background information;
- Memory access pattern;
- Experimental evaluation;
- Conclusions;
- Future work.

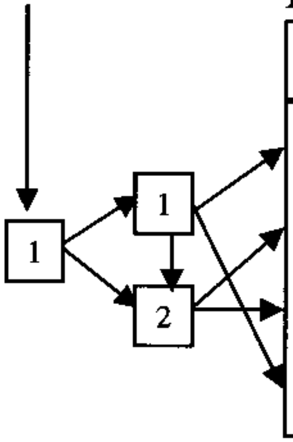
Basic Operations

DeptLoc

Dept	Location
1	Illinois
2	Texas

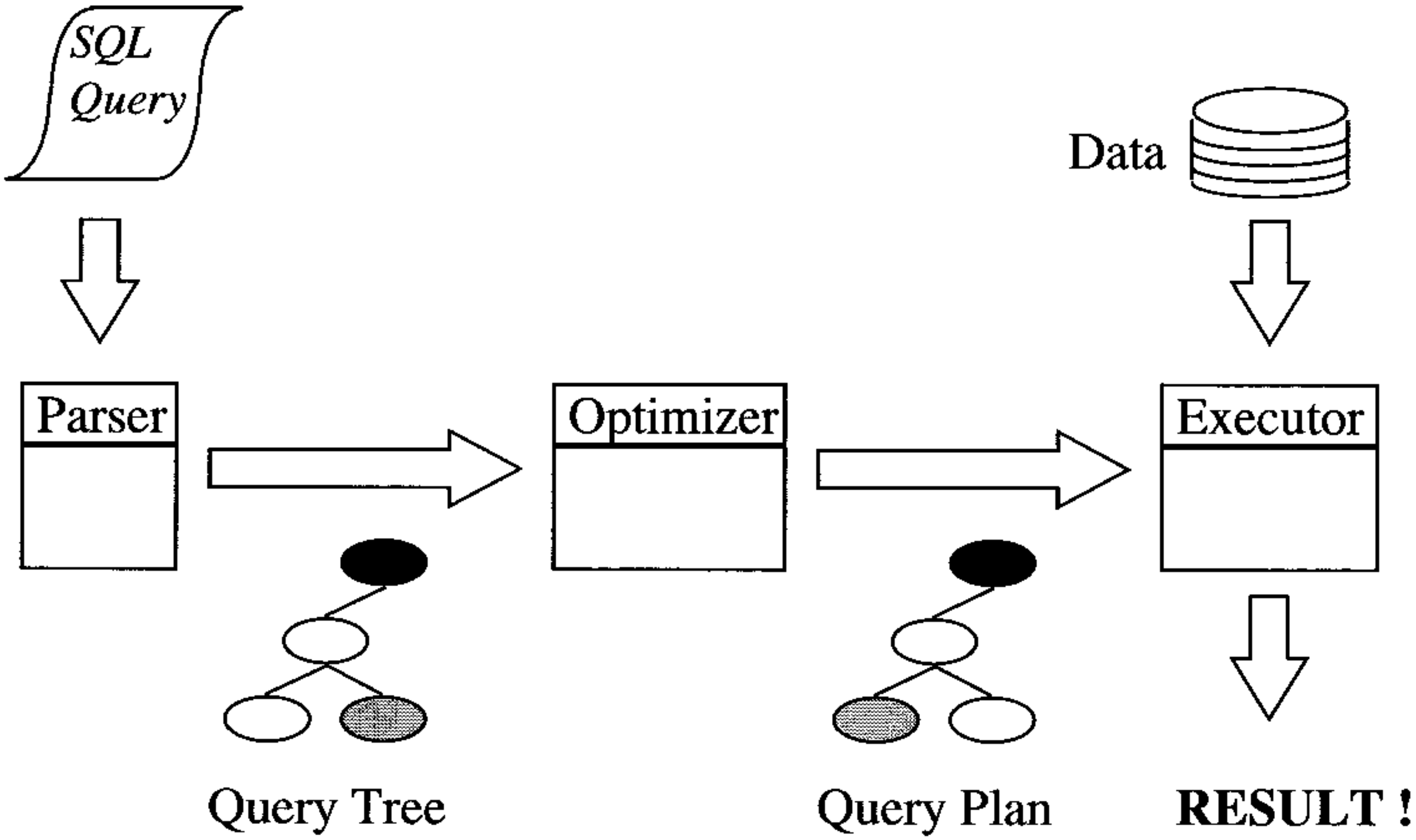
Personnel

Name	Dept
Helen	1
John	2
Julie	2
Peter	1



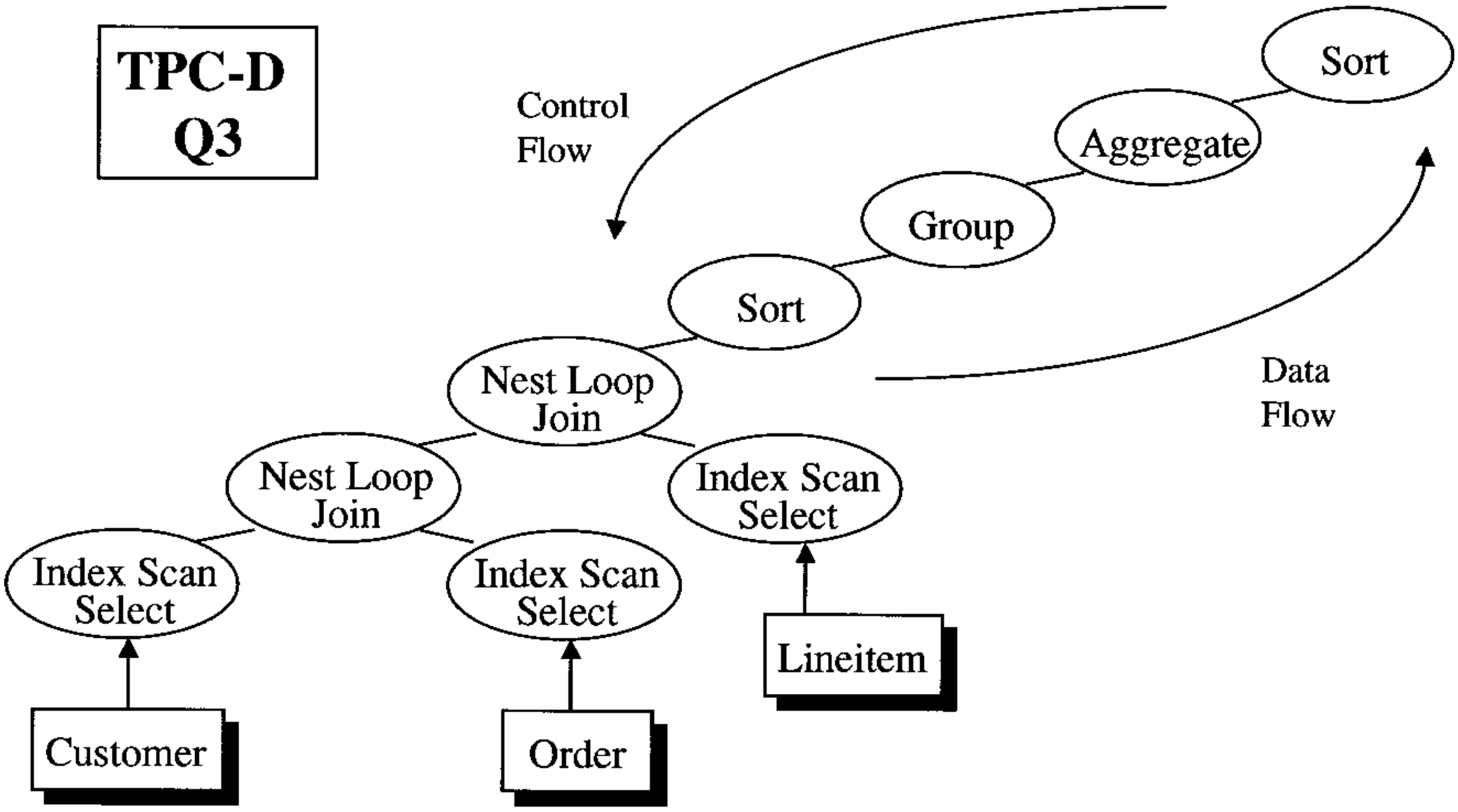
- Select:
  - Example: *Who works in Dept 1 ?*
  - Implementation:
    - » Index Scan
- Join:
  - Example: *Who works in Texas ?*
  - Implementation:
- Other: Sort, Group, Aggregate;

Query Execution

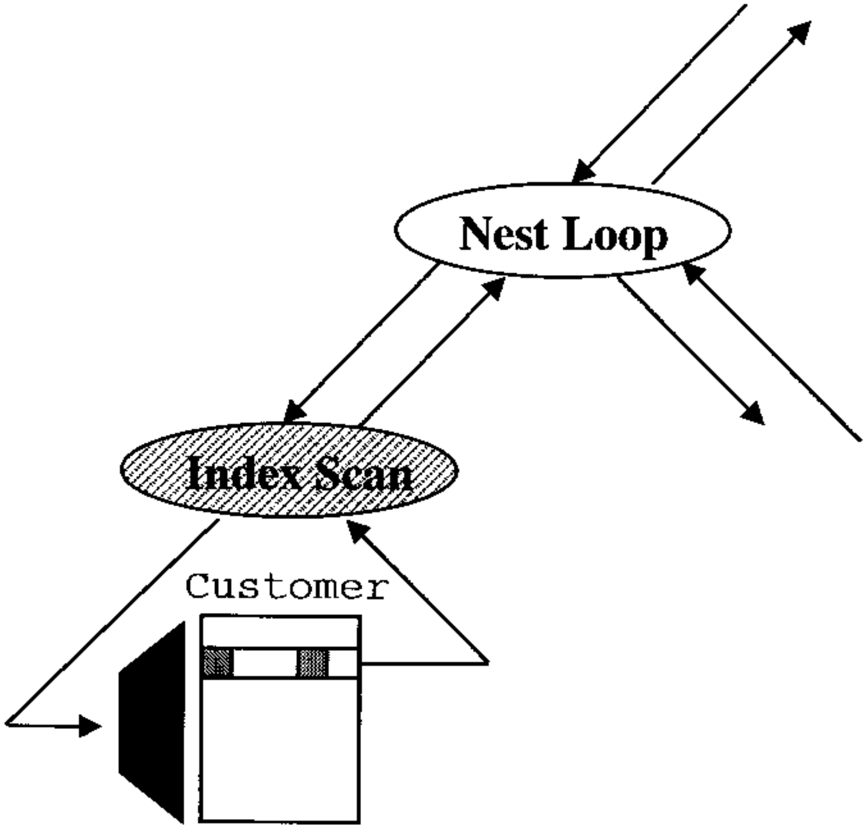


Query Plan Tree

TPC-D Q3

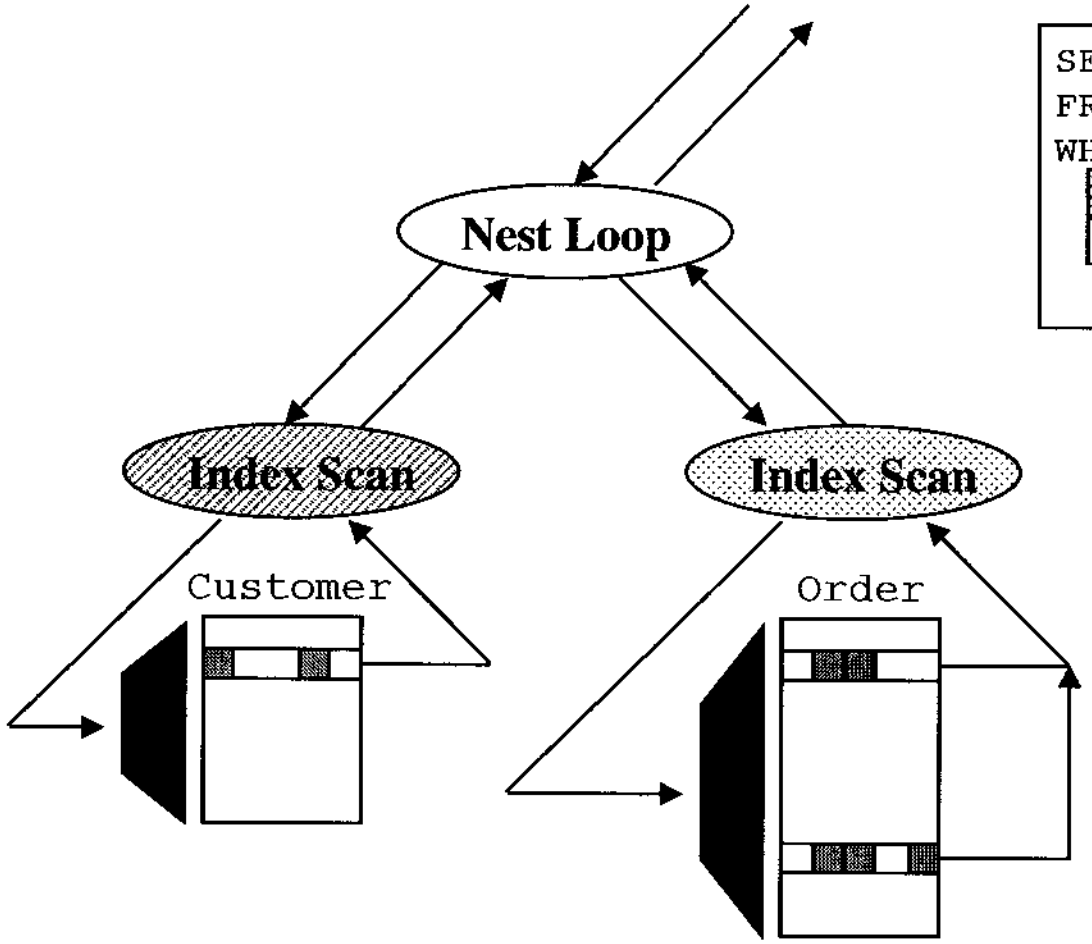


Memory Access Pattern



```
SELECT Order.price
FROM Order, Customer
WHERE
  Customer.age < 20 AND
  Order.data > 01-01-96 AND
  Customer.id = Order.custid;
```

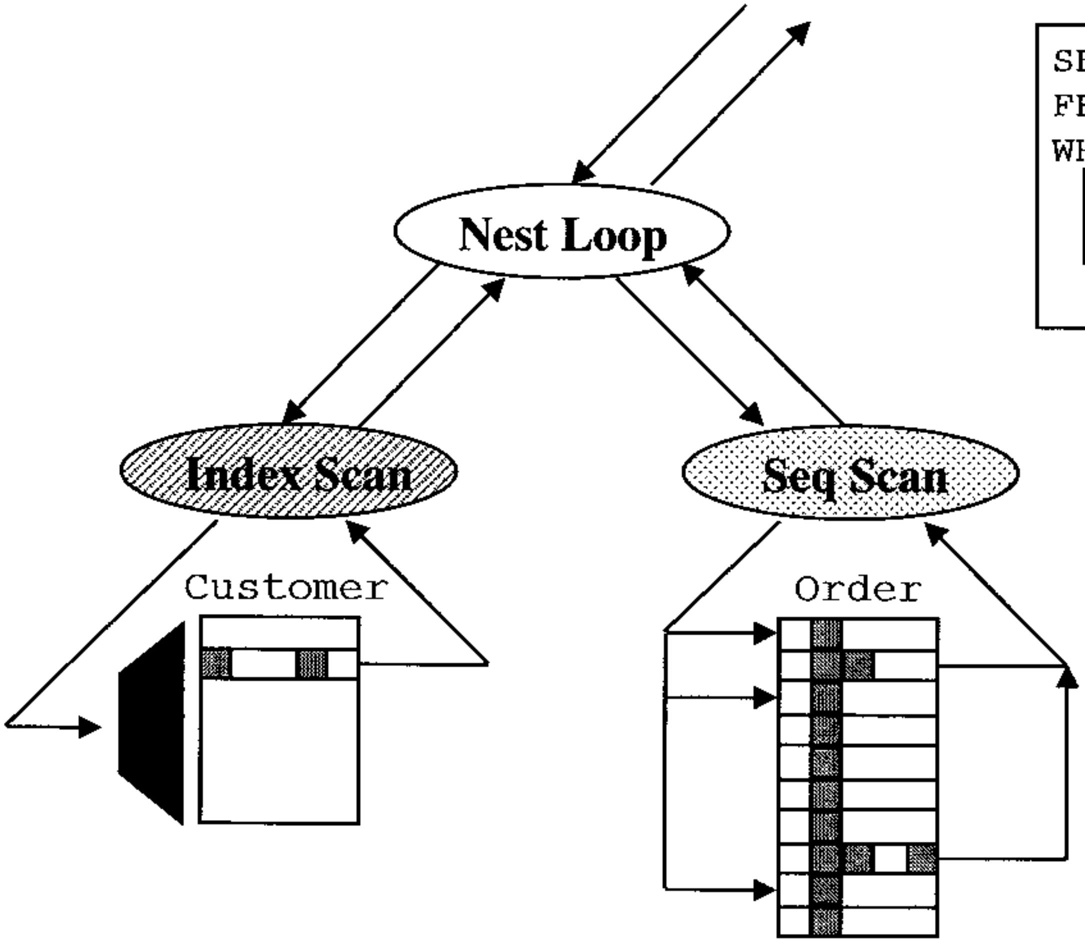
Memory Access Pattern



```
SELECT Order.price
FROM Order, Customer
WHERE
  Customer.age < 20 AND
  Order.data > 01-01-96 AND
   Customer.id = Order.custid;
```

- DBase:
  - Spatial Locality;
  - No Temporal Locality;
- Index:
  - Spatial and Temporal Locality;

Memory Access Pattern



```
SELECT Order.price
FROM Order, Customer
WHERE
   Customer.age < 20 AND
   Order.data > 01-01-96 AND
    Customer.id = Order.custid;
```

- DBase:
  - More Spatial Locality;
  - Temporal Locality with large caches;

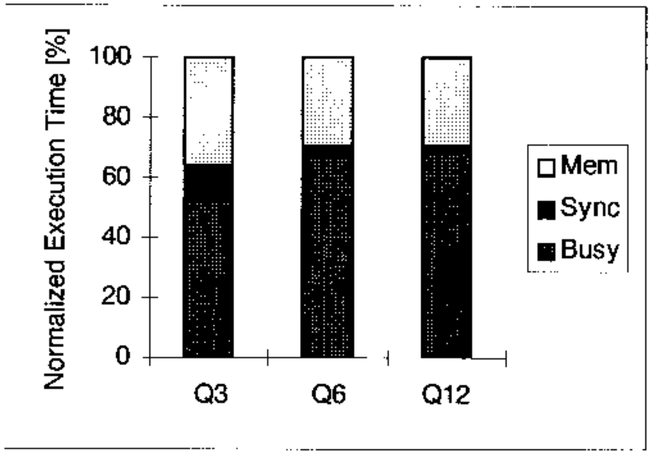
Workload

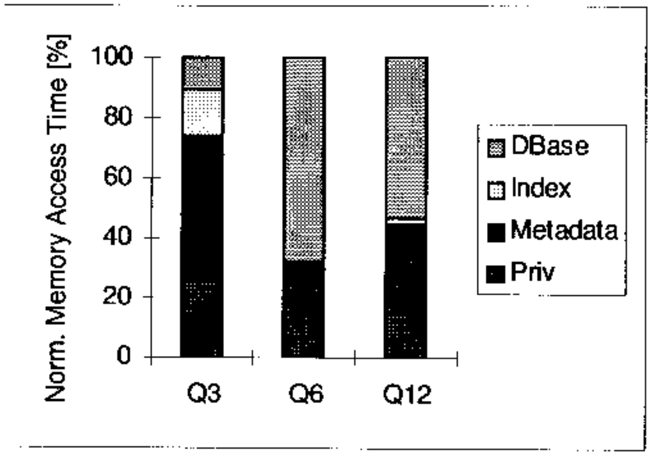
- TPC-D:
  - Decision Support System Benchmark;
  - 17 read-only and 2 update queries;
- Representative queries:
  - Index Scan Select: Q3;
  - Sequential Scan Select: Q6;
  - Index and Sequential Scan Select: Q12;

Evaluation

- Setup:
  - System: 4-node directory-based cache-coherent NUMA shared-memory multiprocessor;
  - Cache: 32-byte line 4-Kbyte primary and 64-byte line 128-Kbyte secondary;
  - Simulator: Mint + Zibra (tracing shared and private data references);
  - DBMS: Postgres95;
  - Workload: TPC-D Q3, Q6, and Q12;
  - Data set: data size aprox. 20 Mbyte.

Overall Memory Behavior

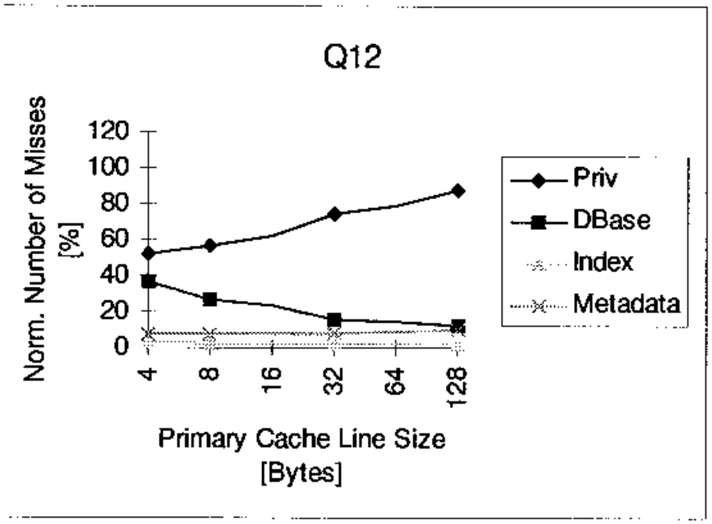


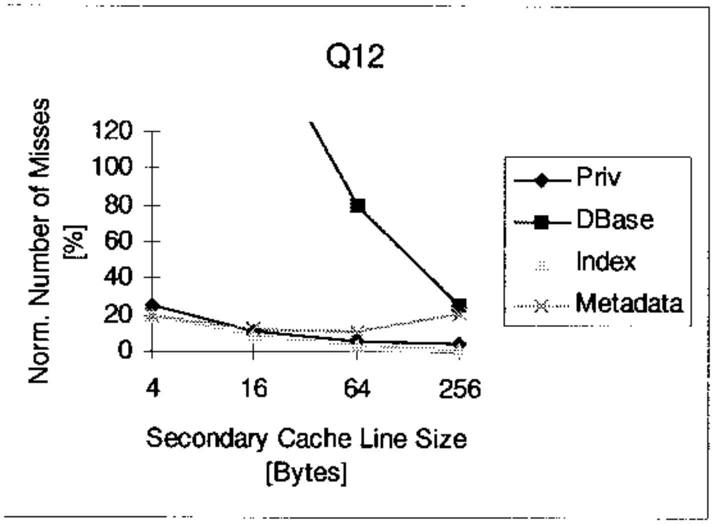


- Execution Time:
  - Large *Busy*: 50-70%
  - Significant *Mem*: 30-35%
- Memory Access Time:
  - Q3: *Metadata+Index* dominate;
  - Q6 and Q12: *DBase* dominates;

Index and *Sequential* queries

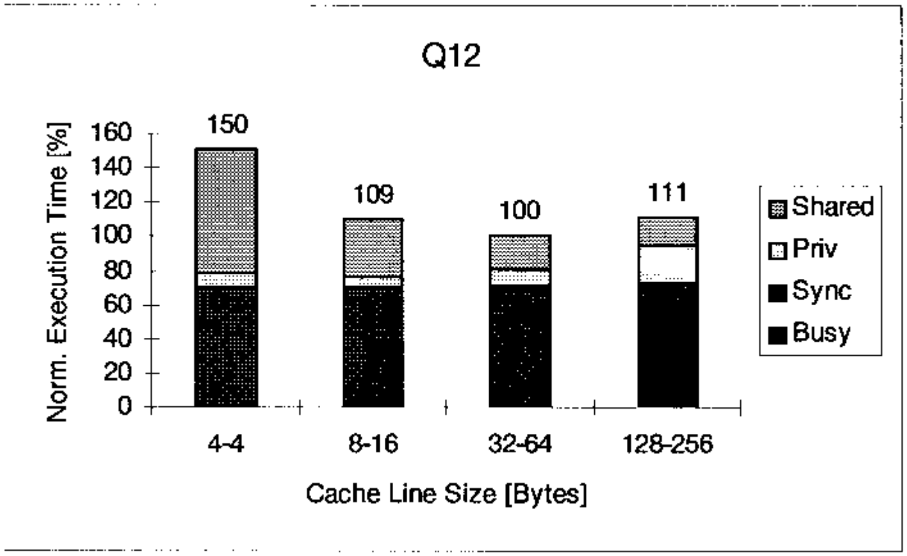
Spatial Locality: Good





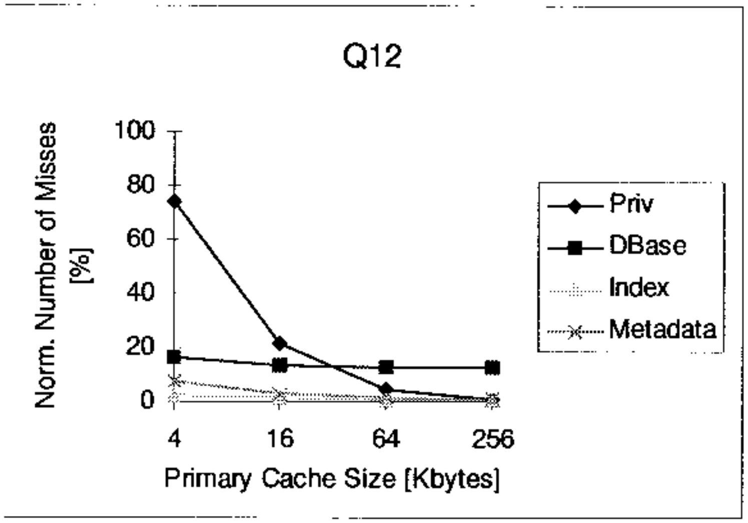
- *DBase* misses decrease with larger cache line size;

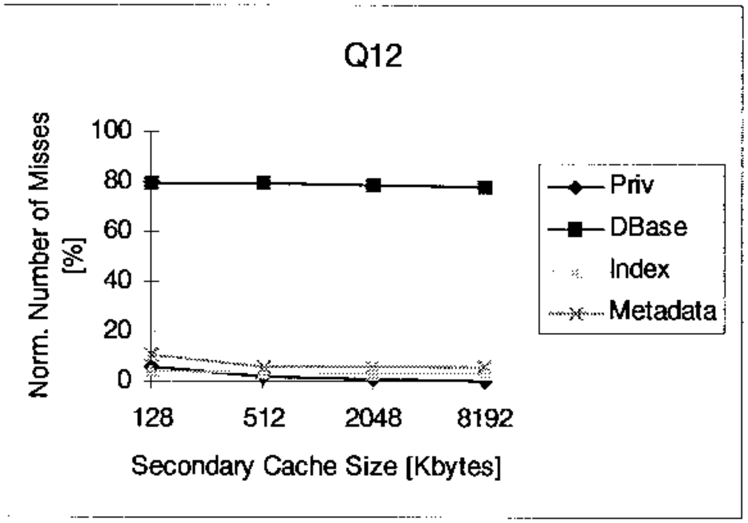
Spatial Locality: Good



- With larger cache line size:
  - *Shared* decreases;
  - *Priv* increases;
- Best results: relatively long cache lines (64 bytes);

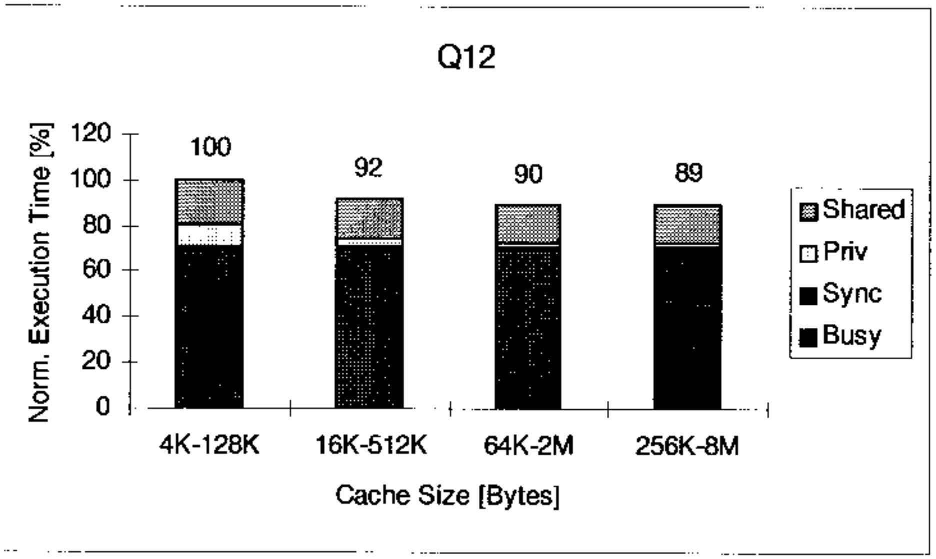
Intra-Query Temporal Locality: Bad





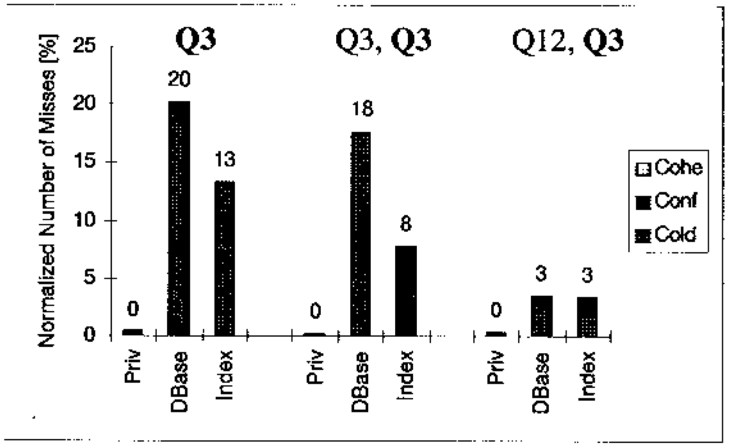
- *DBase* misses: maintain constant with different cache size;

Intra-Query Temporal Locality: Bad

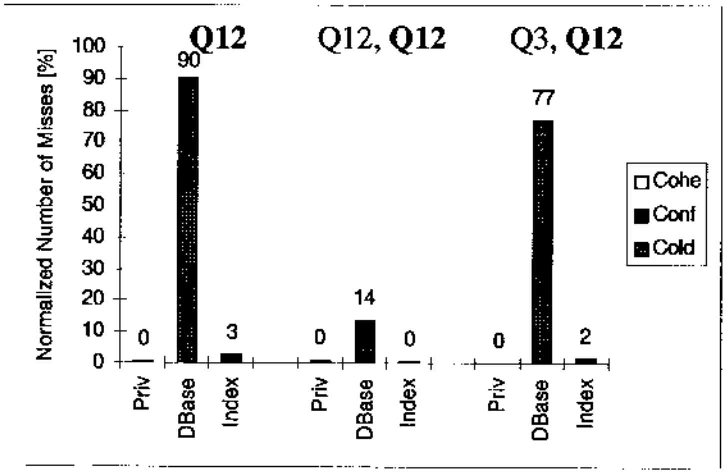


- With larger cache size:
  - *Shared* maintains the same;
  - *Priv* decreases;
- No change: no temporal locality;

Inter-Query Temporal Locality: Maybe



- Q3:
  - Q3, Q3: reuse of *Index*;
  - Q12, Q3: reuse *Index+DBase*;



- Q12:
  - Q12, Q12: large reuse of *DBase;*
  - Q3, Q12: very little reuse;

Conclusions

- Queries are different:
  - *Index* and *Sequential* queries;
- Caches can work:
  - Spatial locality: index and sequential queries;
  - Temporal locality: mostly across sequential queries;
- Simple optimizations:
  - Data prefetching for sequential queries;

Future Work

- Further memory analysis:
  - More complex queries;
- More complex optimizations:
  - Achieve better resource usage.

More information:

http://www.csrd.uiuc.edu/iacoma