

SwingDB: An Embedded In-memory DBMS Enabling Instant Snapshot Sharing

Qingzhong Meng

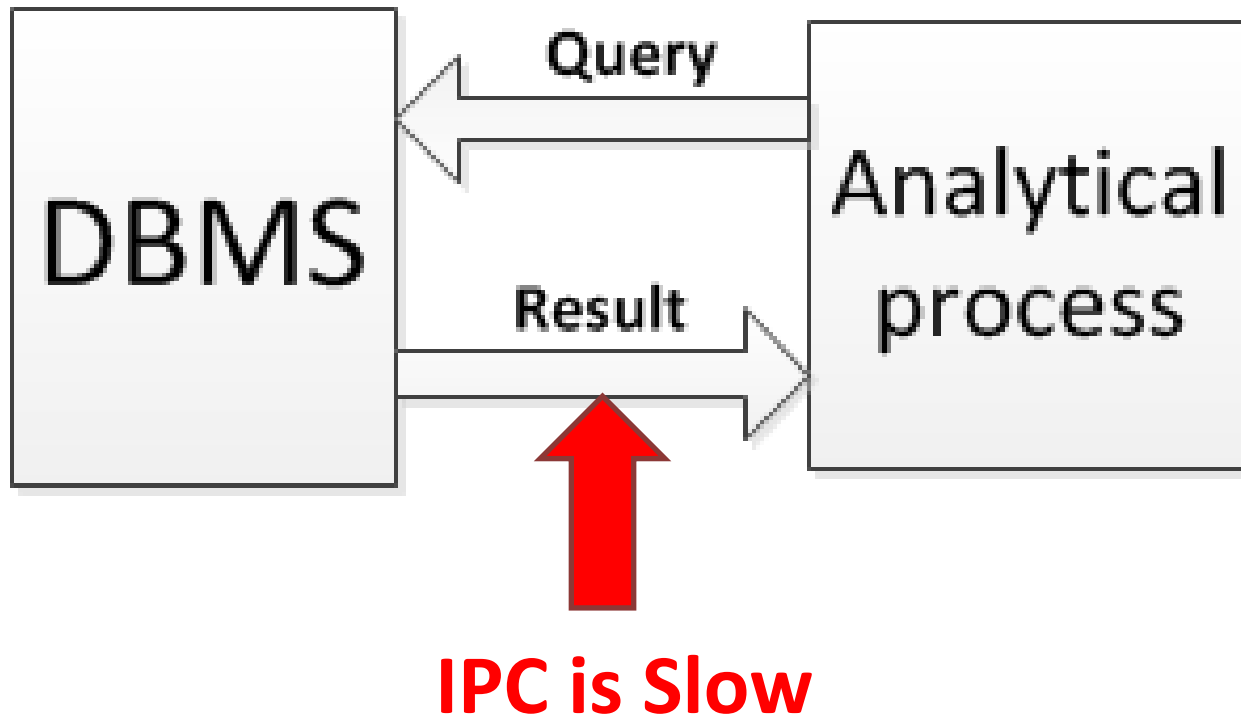
Xuan Zhou

Shiping Chen

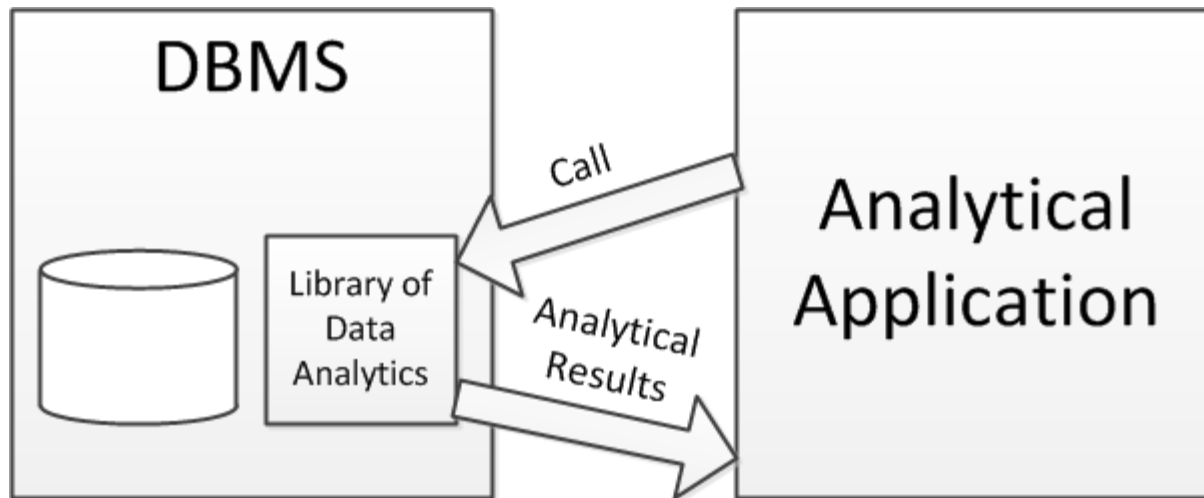
Shan Wang



Typical Case of Data Analysis

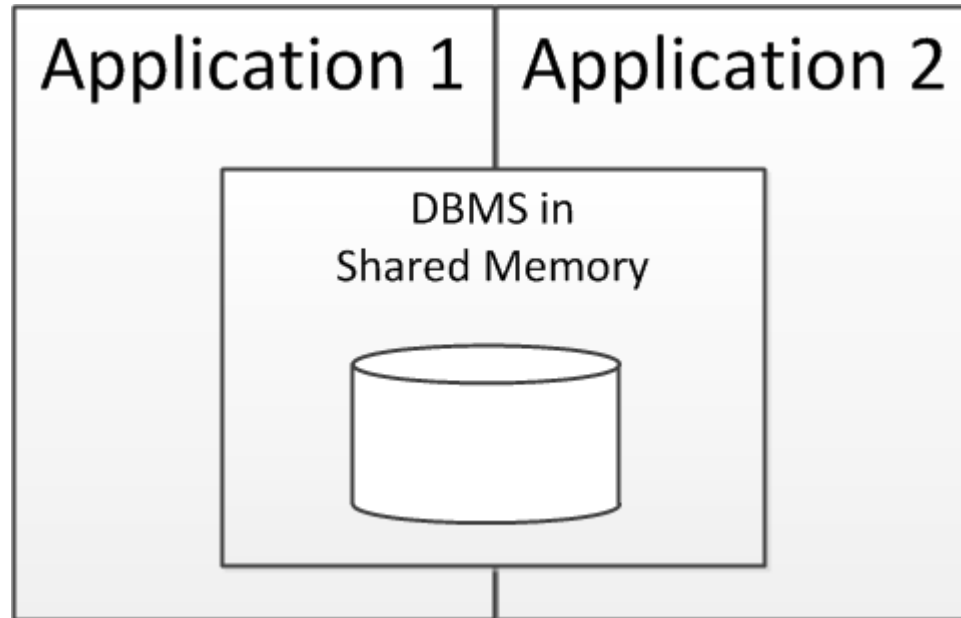


Workaround 1



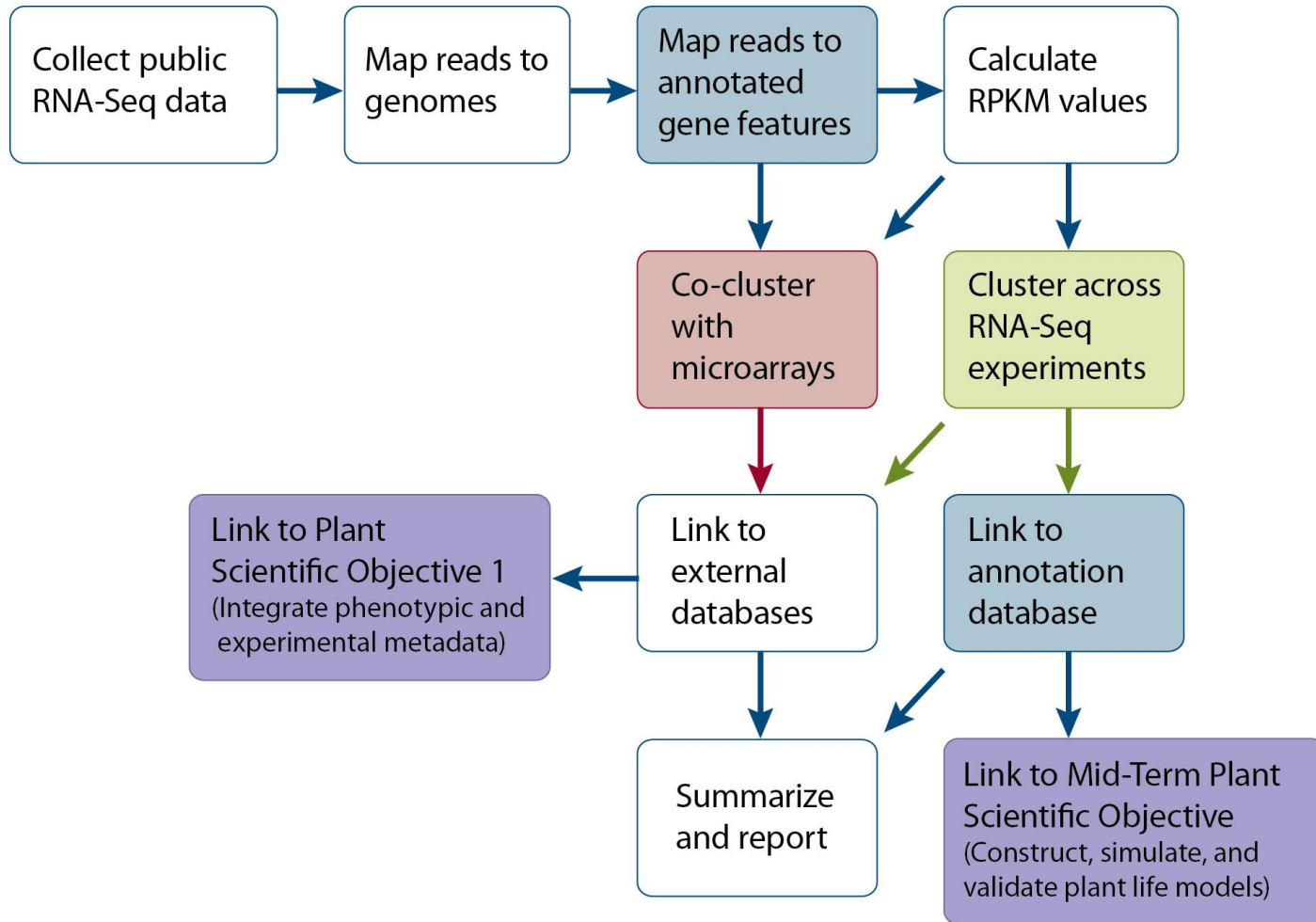
Impossible for DBMS to include all possible data analytical libraries, as it has no a-priori knowledge about the requirements of applications.

Workaround 2

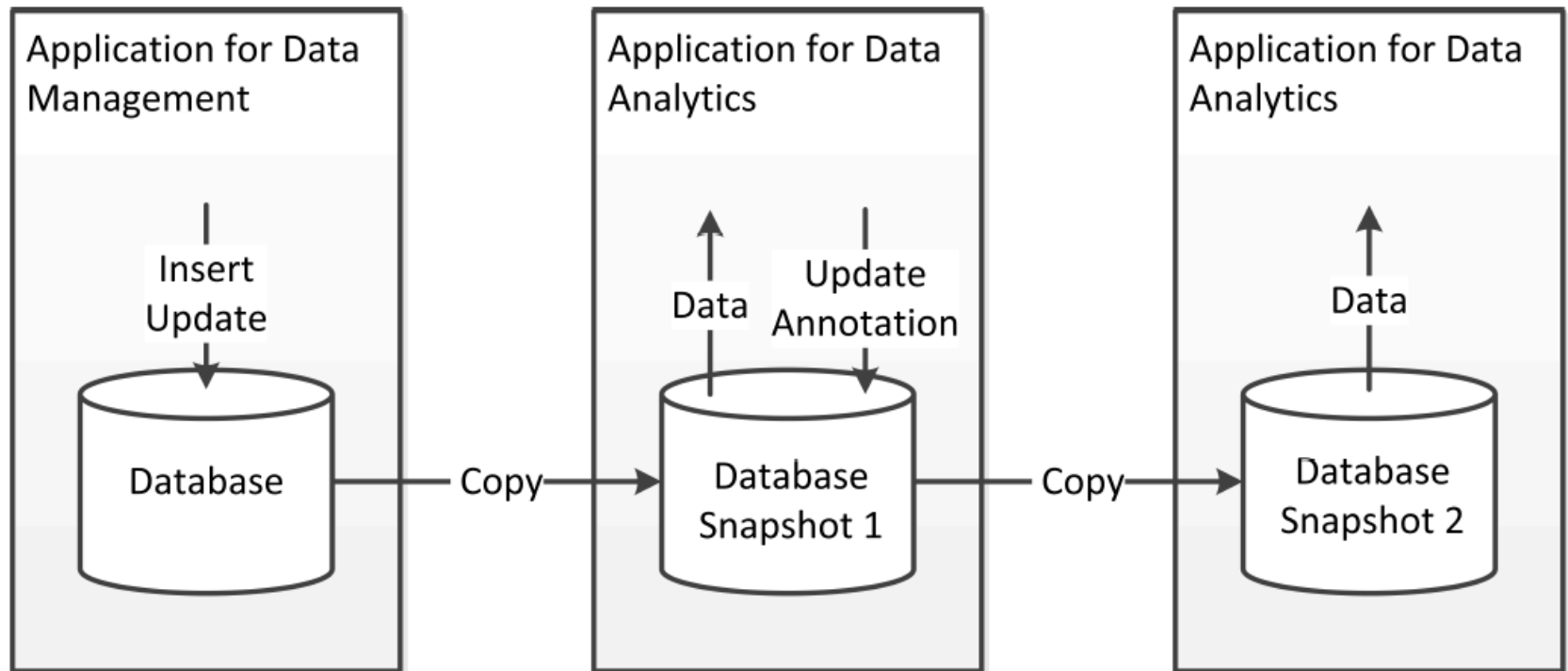


DBMS in shared memory requires synchronization. (The analytical process, who wants to access the data directly, needs to perform additional sync steps to make sure it reads clean data.)

An Example of Multi-Stage Data Analysis for Scientific Study



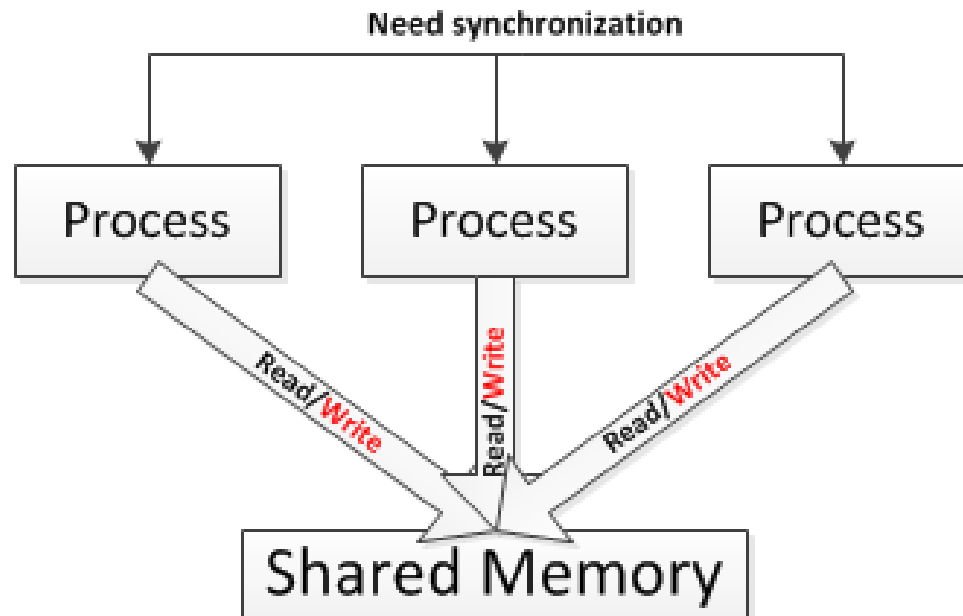
The Approach of SwingDB



Some IPC methods

- Fast
 - Shared Memory
- Slow
 - FIFO
 - Socket

Shared memory

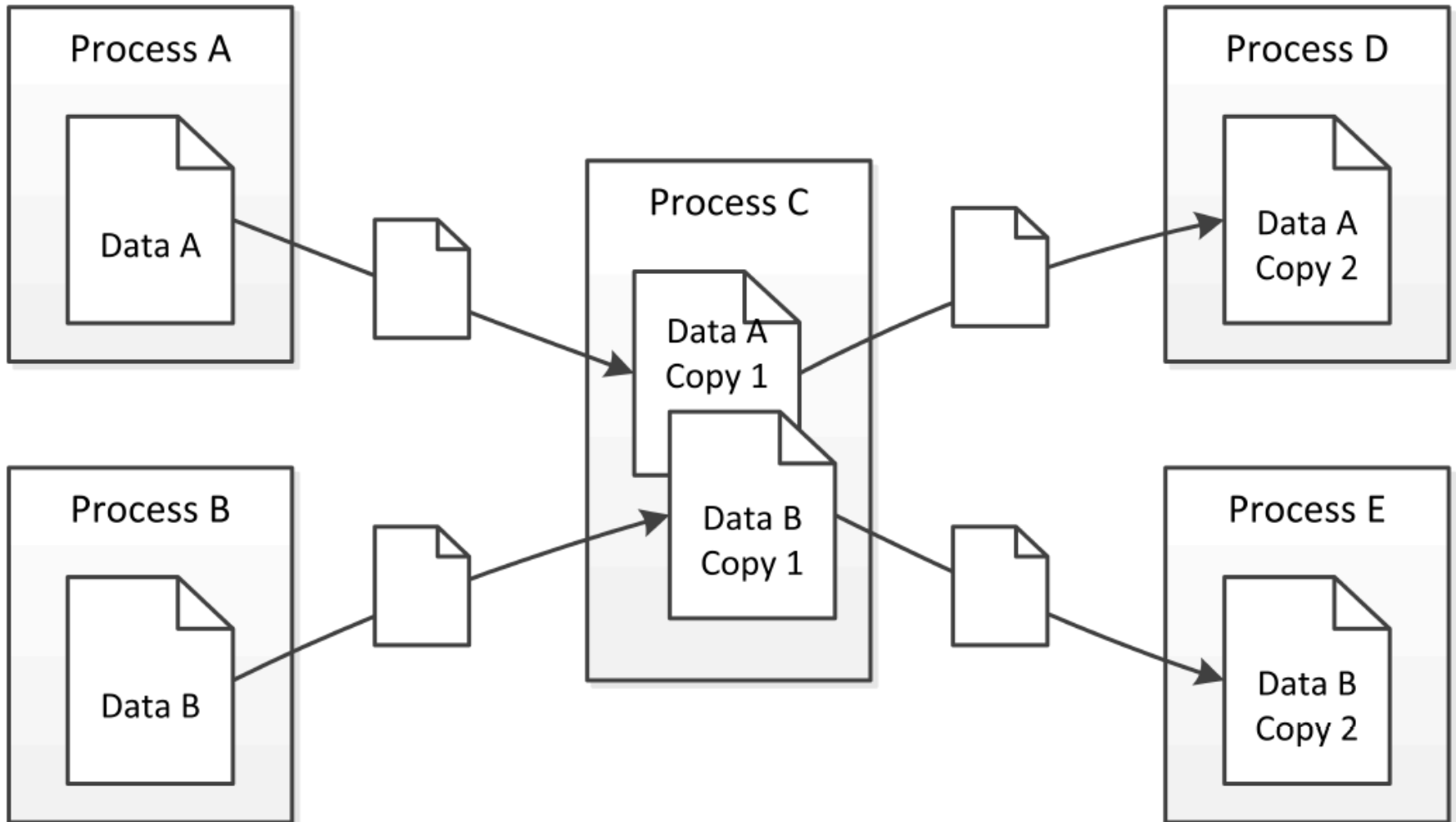


* MAP_PRIVATE can not solve this synchronization problem

Swing

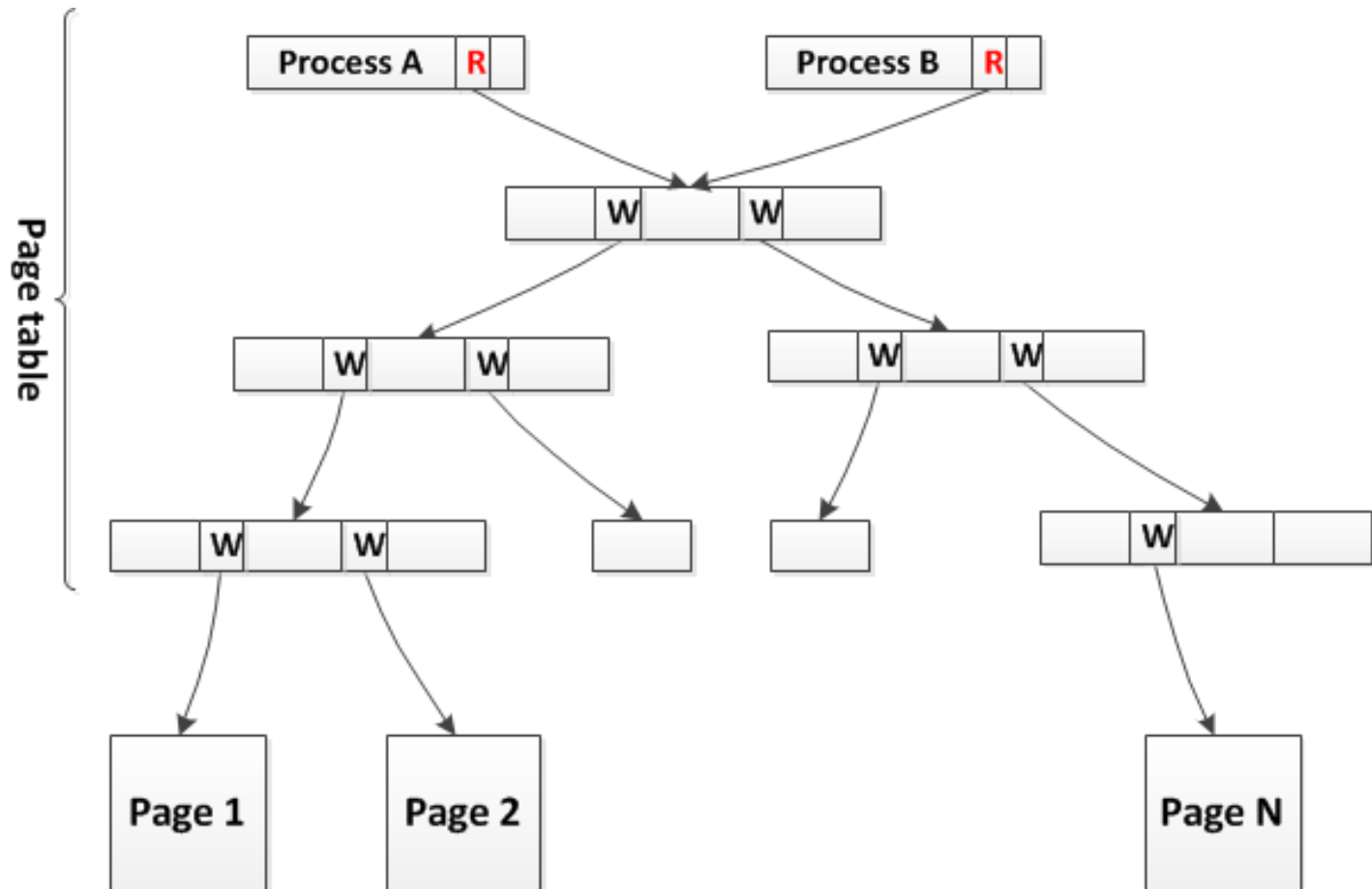
- A new IPC method
- Share physical memory among processes
- Copy on write
 - Processes can only read this part of memory at first
 - Once a process write to somewhere, allocate new physical memory pages
- Can used to many applications independently

Data Sharing Model of Swing

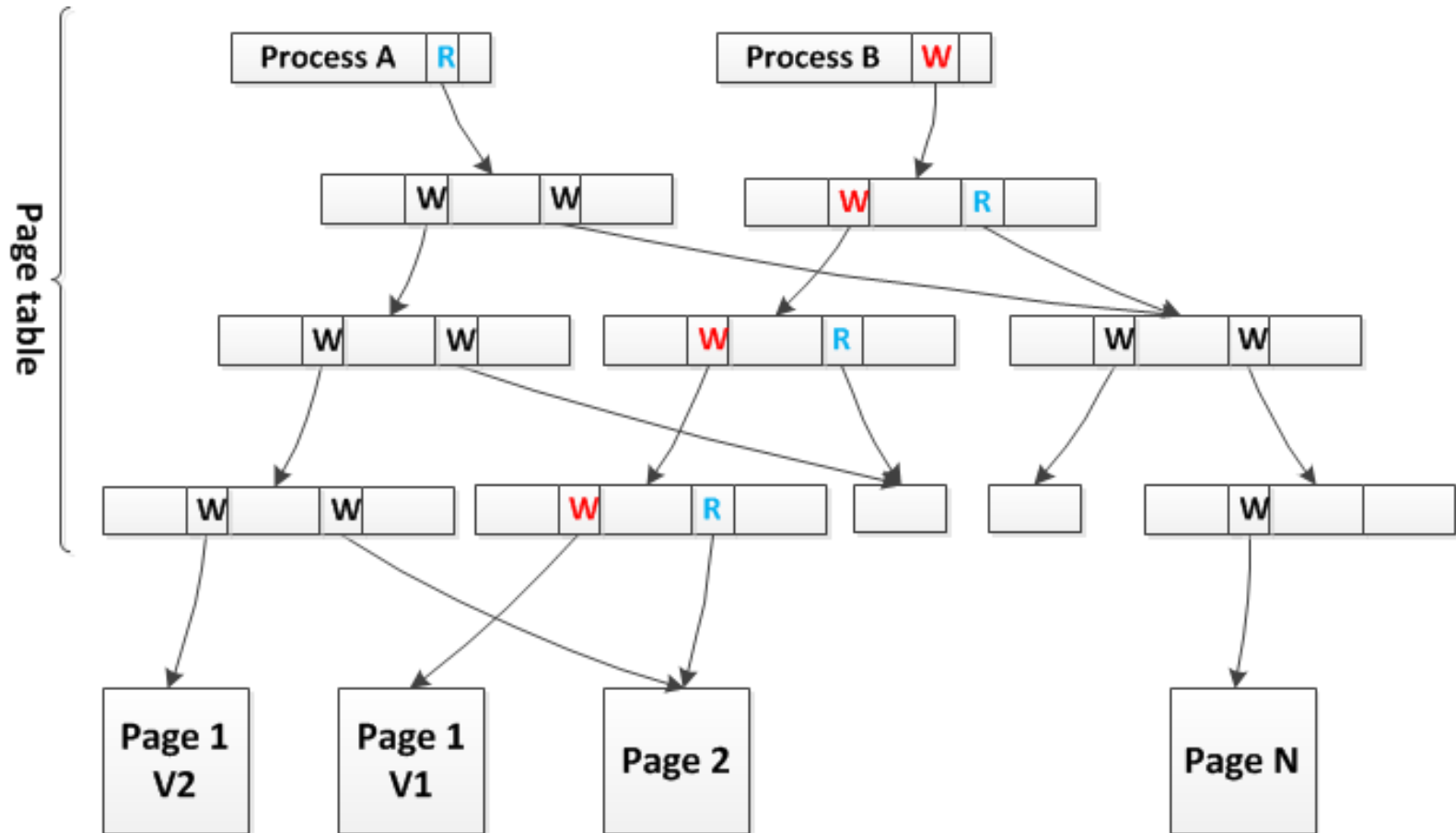


Implementation of Swing

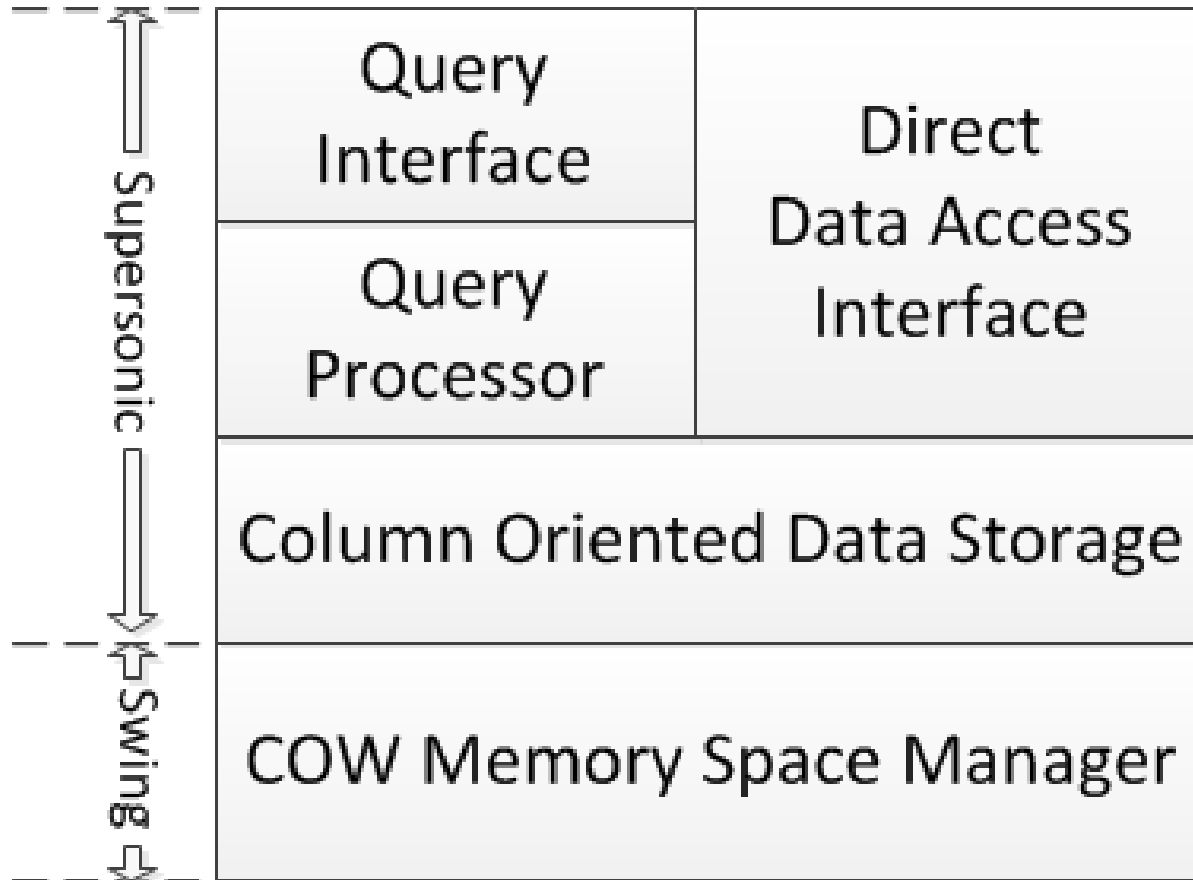
- Share page table



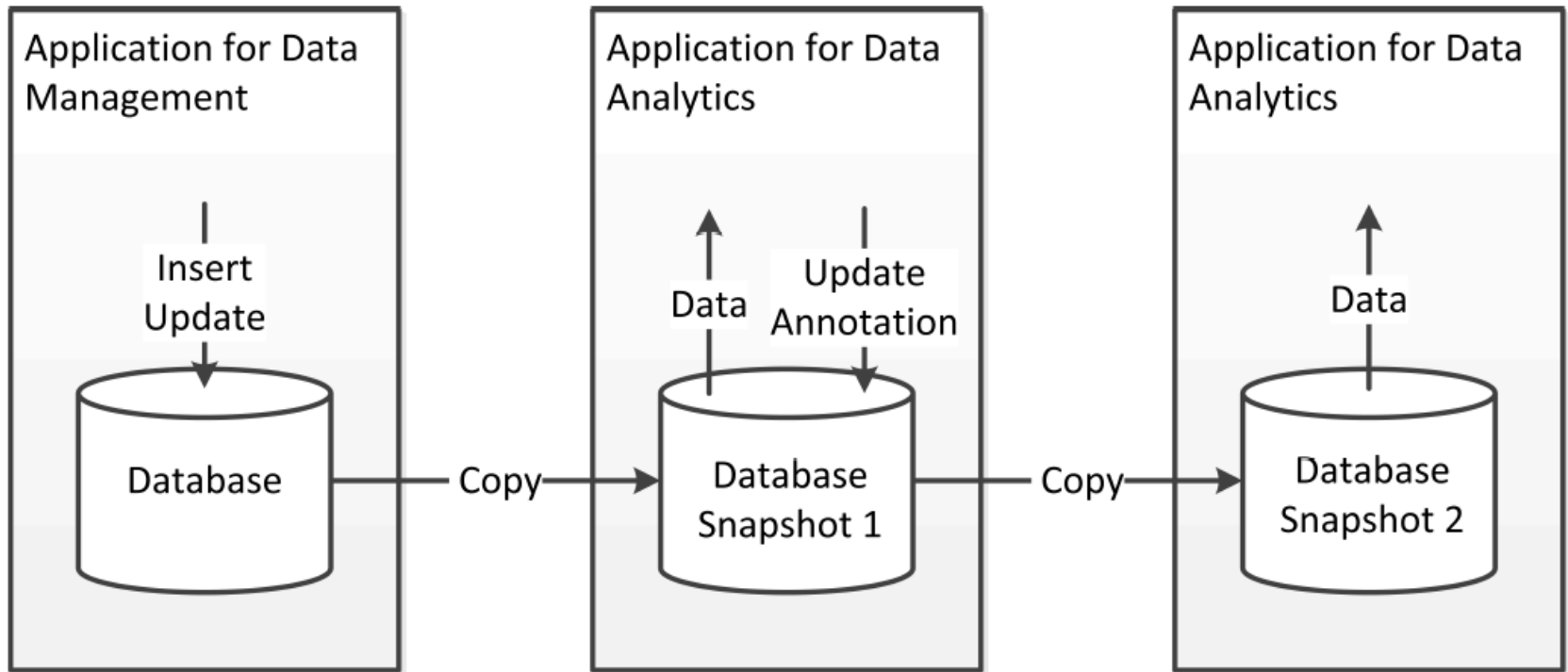
Implementation of Swing



Architecture of SwingDB



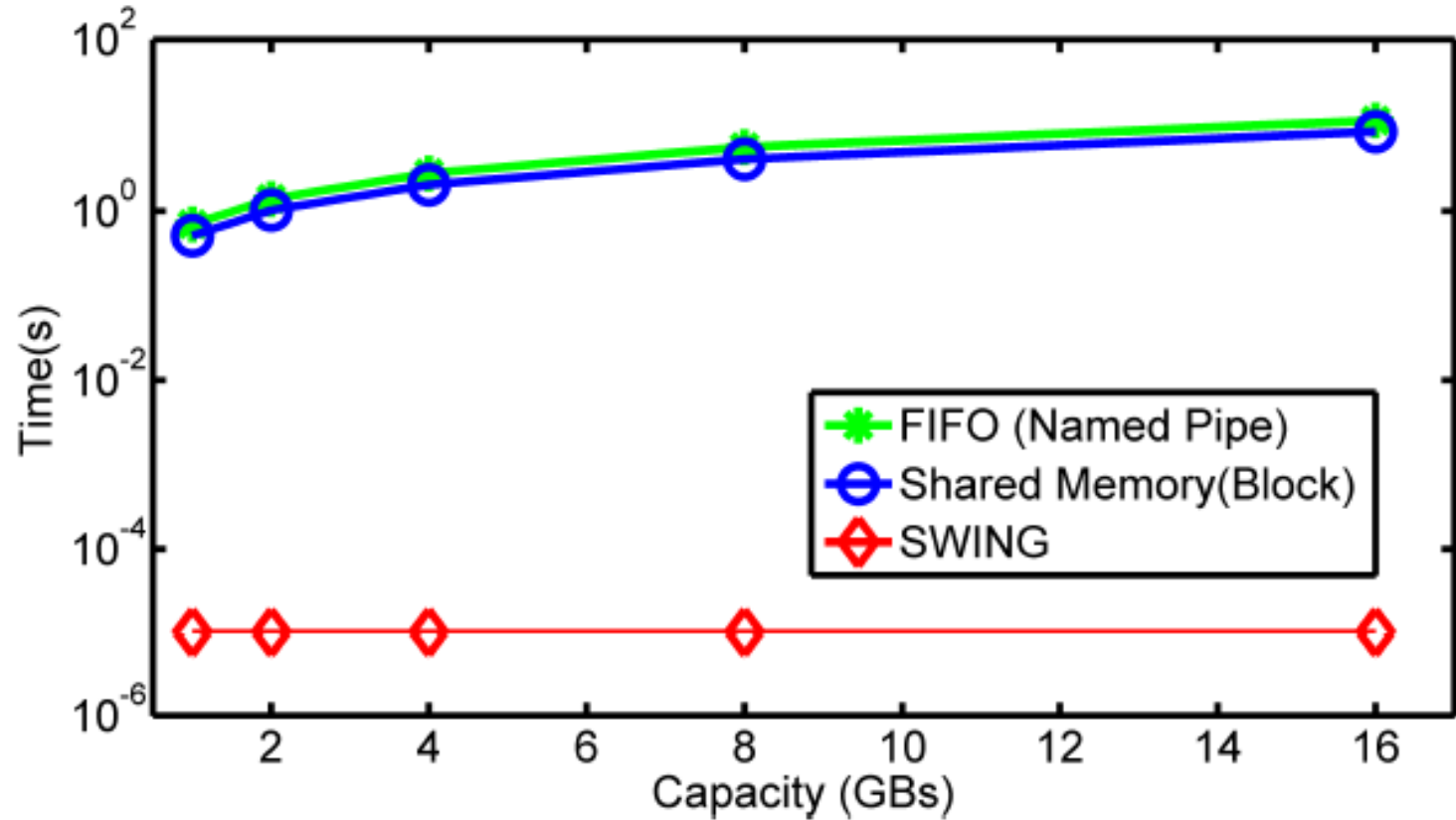
Functionality of SwingDB



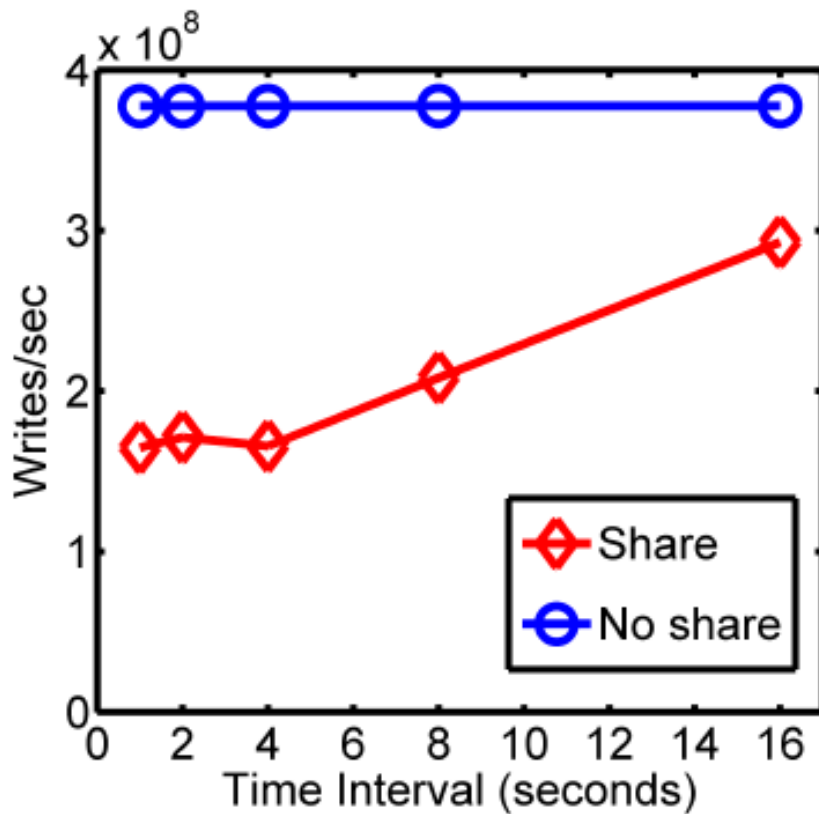
Characters of SwingDB

- Share data fast
- API is simple
- Loose Coupling

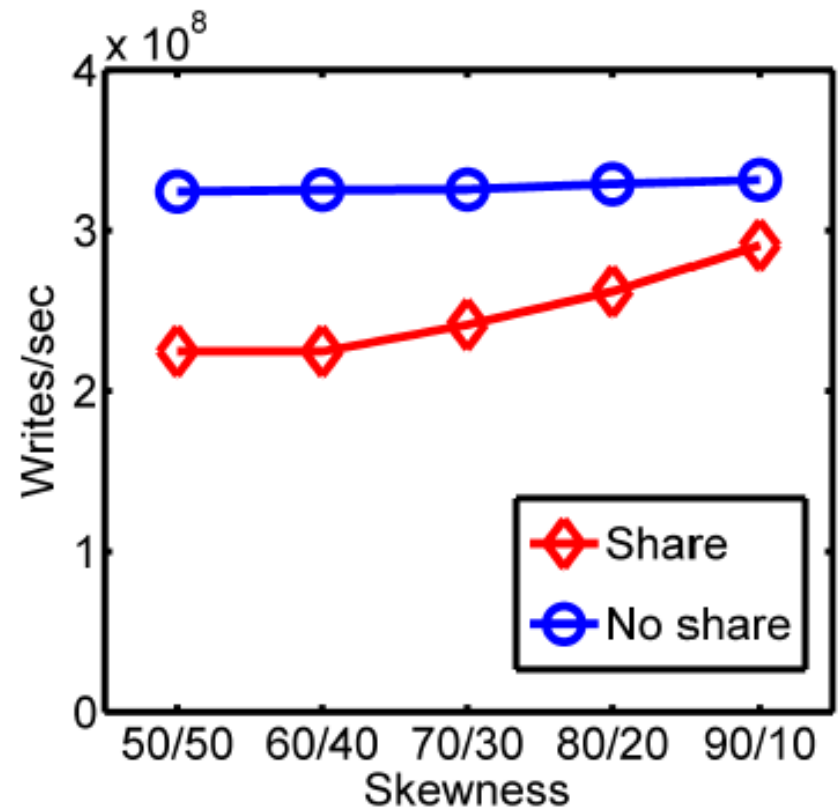
Overhead of Data Transmission



Overhead of Copy on Write

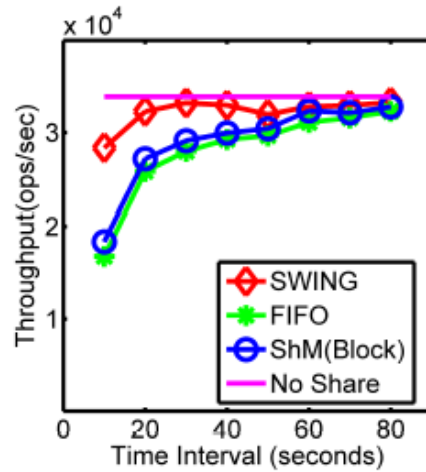


(a) sensitivity to frequency

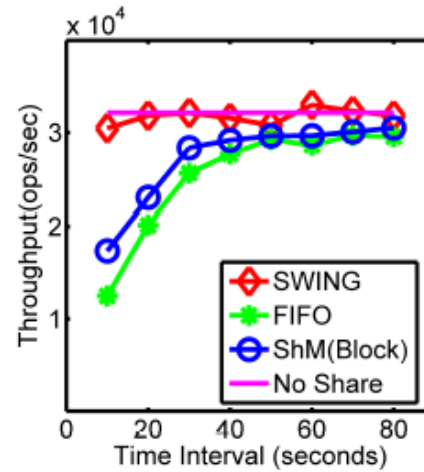


(b) sensitivity to locality

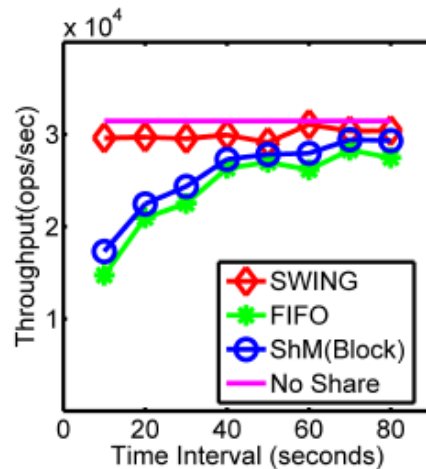
Performance on YCSB



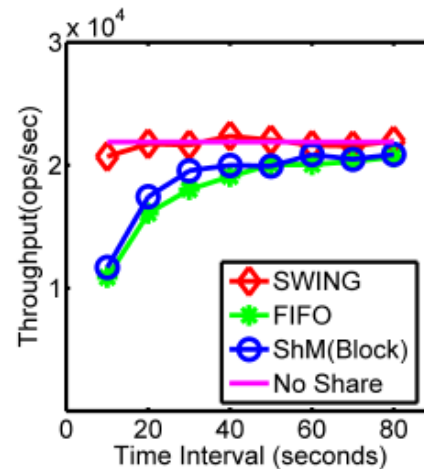
Workload A (w/r = 50/50)



Workload B (w/r = 5/95)

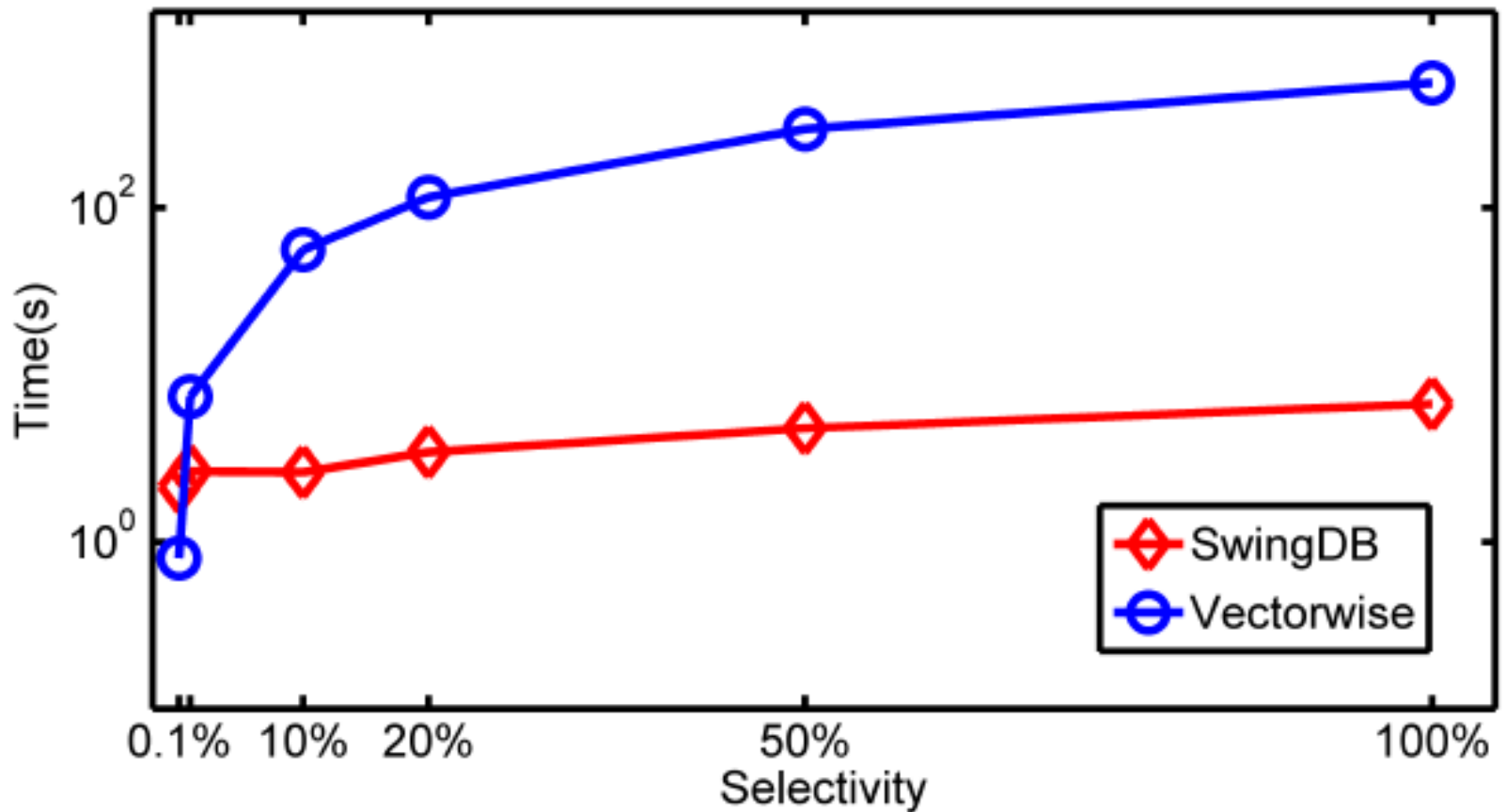


Workload D (w/r = 5/95)

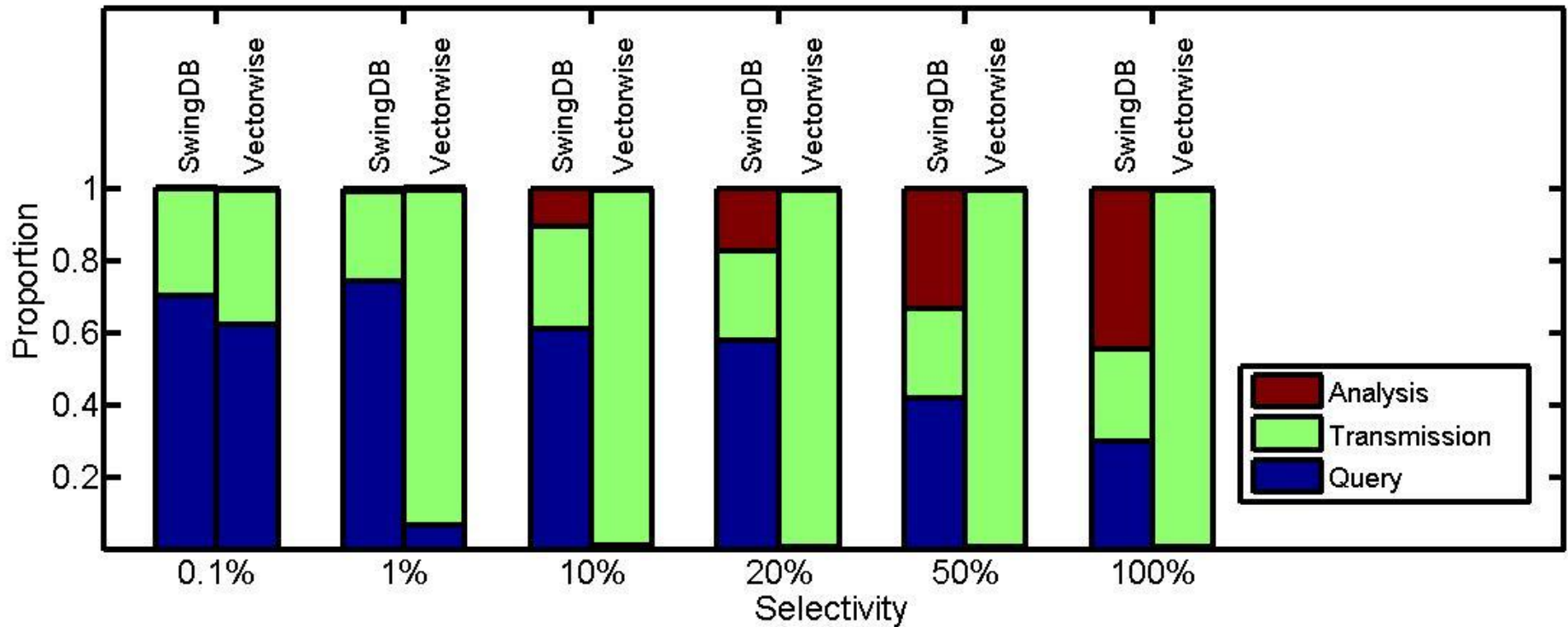


Workload F (w/r = 50/50)

Execution Time of Data Analysis



Breakdown of the Execution Time



Thank you!

Questions?