

# Mapping Data Warehouse to Multiple Processor



# Relational Database Technology for Data Warehouse

- Search for better performance and scalability becomes necessity for a data warehouse.
- The search pursuing two goals:
  - **Speed-up**- ability to execute the same request on the same amount of data in less time
  - **Scale-up** – ability to obtain the same performance even if database size increases.
- The goals of linear performance and scalability can be satisfied by Parallel hardware architectures, Parallel operating systems and parallel databases.
- Parallel hardware architecture are based on Multiprocessor systems designed as a shared memory model (SMPs), shared-disk model, or a distributed-memory model (MPPs) and clusters of uniprocessors



# Types of Parallelism

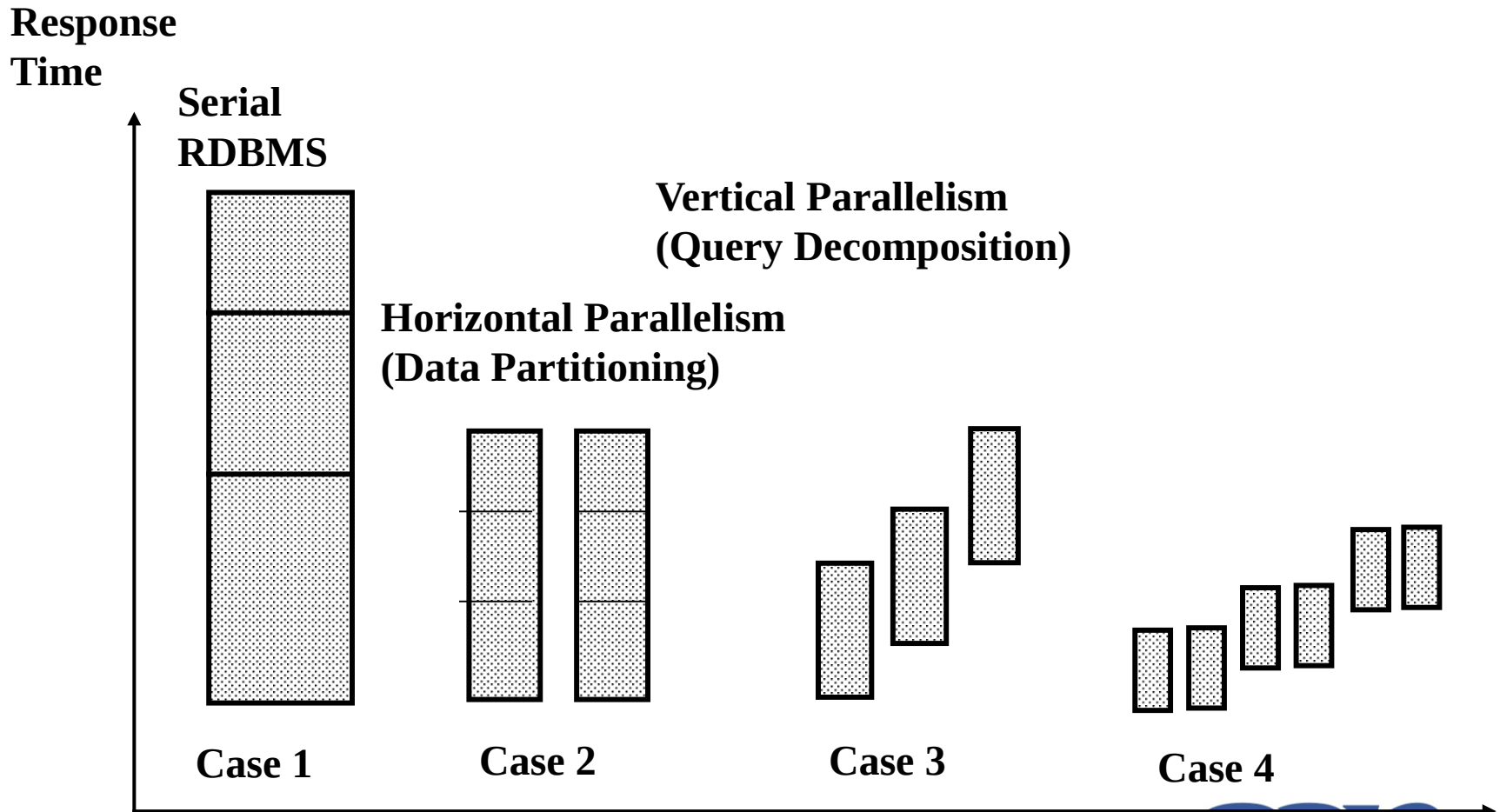
- Parallel hardware architectures are implemented using multi server and multi threaded systems designed to handle large number of client requests.
- **Interquery parallelism :**
  - Different threads (or process) handle multiple requests at the same time.
  - Implemented on the SMP systems.
  - Each query are processed serially such as scan, join and sort
- **Intraquery parallelism –**
  - Decomposes the serial SQL query into lower-level operations such as scan, join, sort, and aggregation operations
  - They are executed concurrently in parallel.
  - Intraquery parallelism can be done in either of two ways:



# Types of Parallelism

- **Horizontal parallelism:** Database is partitioned across multiple disks, and parallel processing occurs within a specific task (table scan) that is performed concurrently on different processors on different sets of data.
- **Vertical parallelism** – All complex query operations are executed in parallel in a pipeline fashion.
  - An output from one task (e.g., scan) becomes an input into another task (e.g., join) as soon as records become available

# Types of DBMS parallelism



# Data Partitioning

- It is a key requirement for effective parallel execution of database operations.
- Spreads data from database tables across multiple disks so that I/O operations such as read and write can be performed in parallel.
- Two types – random and intelligent partitioning.
- **Random partitioning:** It includes random data striping across multiple disks on a single server.
- Another option for random partitioning is **round-robin partitioning** in which each new record is placed on the next disk assigned to the database.
- The effectiveness of round-robin partitioning is reduced depending upon the data distribution and query selectivity



# Data Partitioning

DBMS knows where a specific record is located and does not waste time searching for it across all disks.

- **Hash partitioning:** A hash algorithm is used to calculate the partition number (hash value) based on the value of the partitioning key for each row.
- **Key range partitioning:** Rows are placed and located in the partitions according to the value of the partitioning key (all rows with the key value from A to K are in partition 1, L to T are in partition 2 etc.).



# Data Partitioning

- **Schema partitioning:** An entire table is placed on one disk, another table is placed on a different disk, etc.
- **User-defined partitioning:** This is a partitioning method that allows a table to be partitioned on the basis of a user-defined expression.





# Database Architectures for Parallel Processing

1. Shared-memory architecture- SMP (Symmetric Multiprocessors)
2. Shared-disk architecture
3. Shared-nothing architecture
4. Combined architecture



# Shared-memory architecture - SMP (Symmetric Multiprocessors)

- **Shared memory or Shared everything** style is the traditional approach to implementing an RDBMS or SMP hardware.
- A single RDBMS server can potentially utilize all processors, access all memory and access the entire database thus providing the user with a consistent single system image.
- Multiple database components executing SQL statements communicate with each other by exchanging messages and data via the shared memory.
- Scalability can be achieved through process-based multitasking or thread based multitasking.

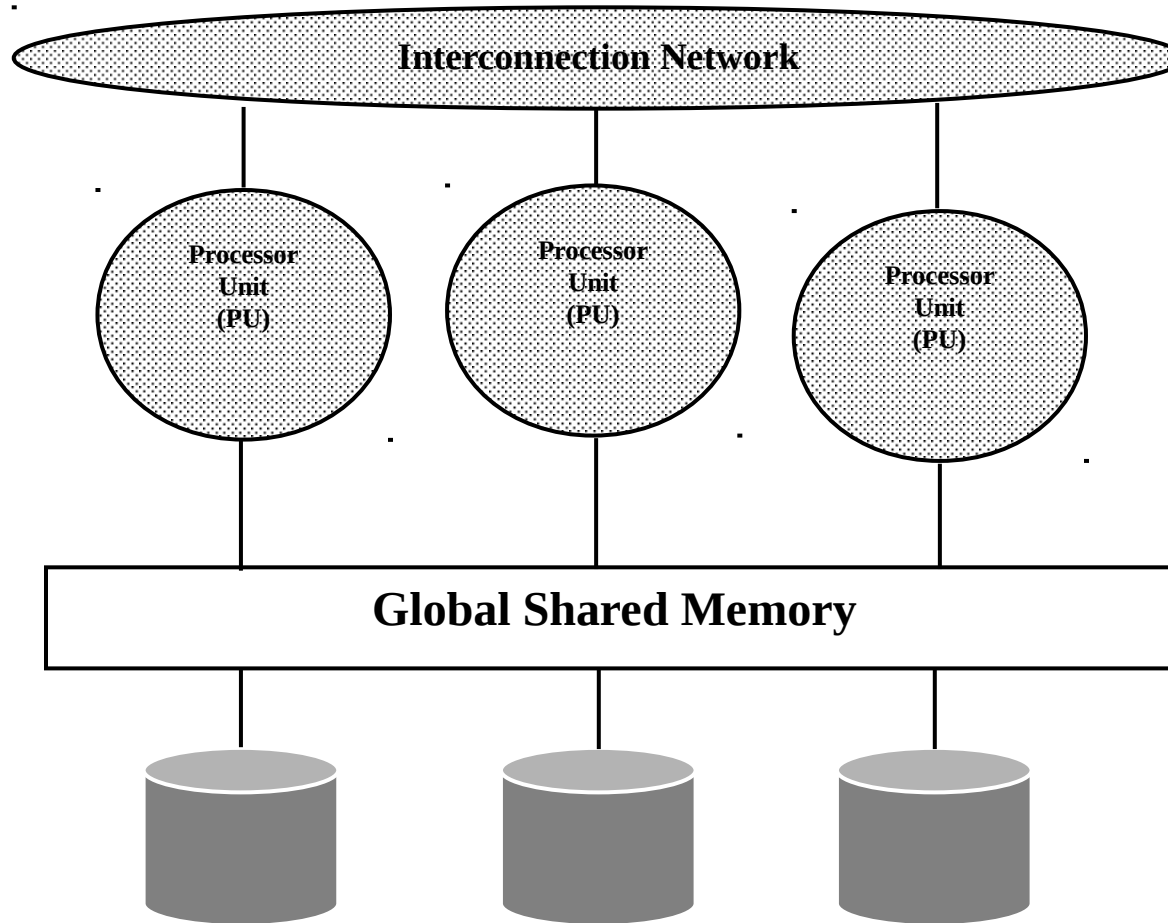


# Shared-memory architecture - SMP (Symmetric Multiprocessors)

- **Advantage:** Threads provide better resource utilization and faster context switching.
- **Disadvantage:** Threads tightly coupled with OS and may limit RDBMS portability



# Shared-memory architecture - SMP (Symmetric Multiprocessors)



# Shared-disk architecture

- Implements the concept of shared ownership of the entire database between RDBMS servers, each of which is running on a node of the distributed memory system.
- Each RDBMS server can read, write, update, and delete records from the same shared database.
- Implemented by using distributed lock manager (DLM).
- Disadvantages: “**Pinging problem**”: All nodes are reading and updating the same data, the RDBMS and its DLM will have to spend a lot of resources synchronizing multiple buffer pools.
- It may have to handle significant message traffic in a highly utilized RDBMS environment.

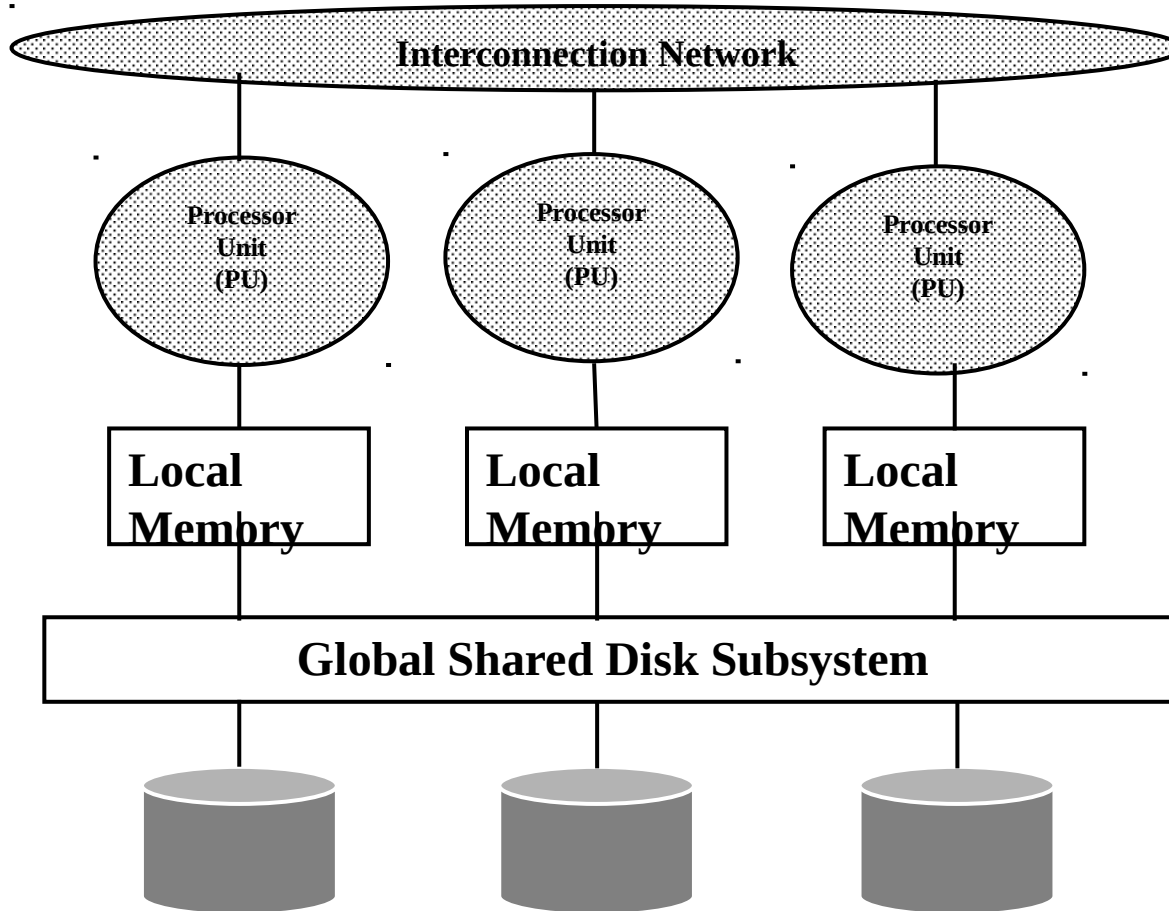


# Shared-disk architecture

- **Advantages:**
- It reduces performance bottlenecks resulting from data skew and can significantly increase system availability.
- It eliminates the memory access bottleneck typical of large SMP systems
- Help to reduce DBMS dependency on data partitioning.



# Shared-disk architecture



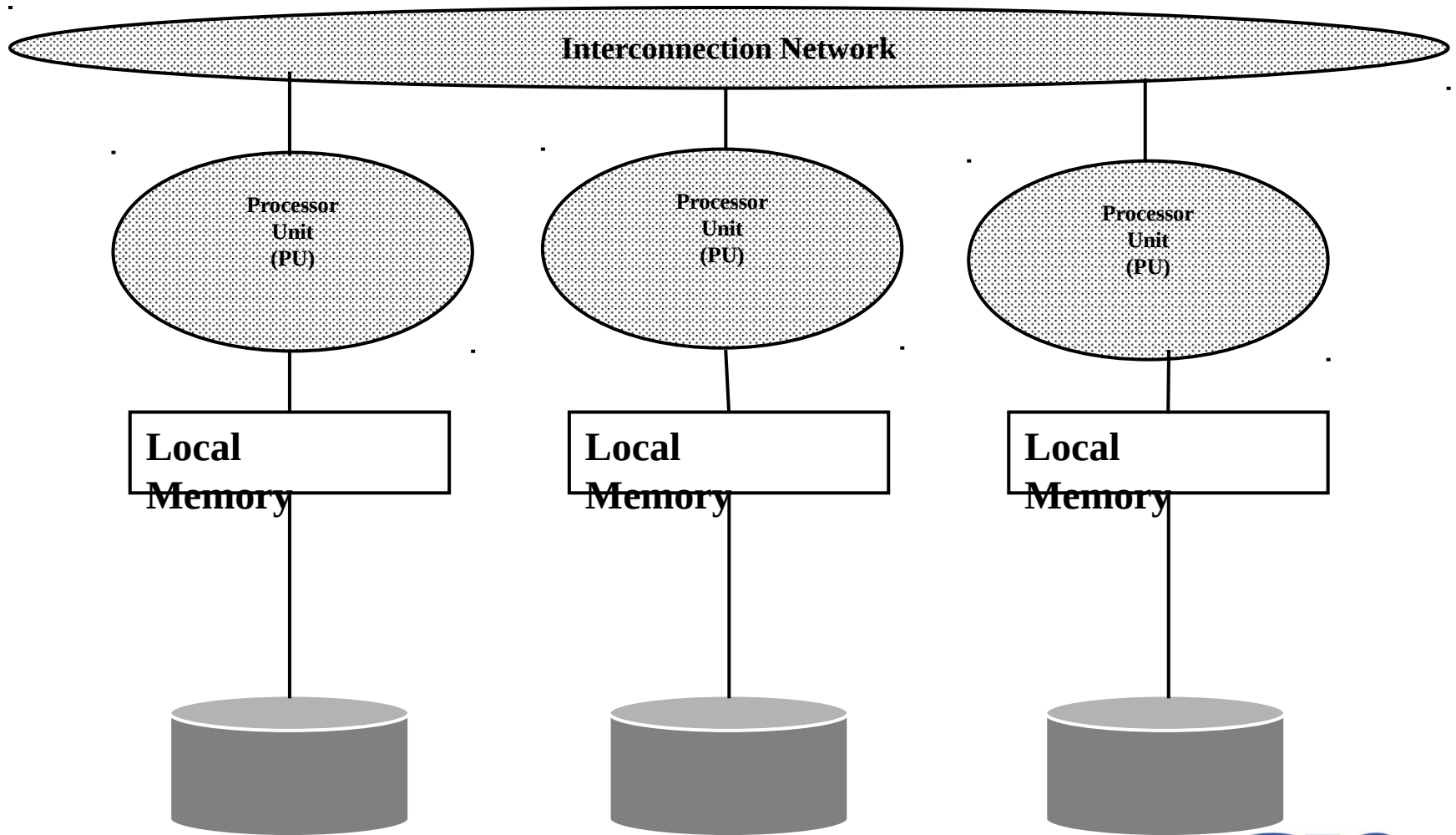
# Shared-nothing architecture

- Data is partitioned across all disks and the DBMS is partitioned across multiple co-servers which resides in individual nodes of the parallel system.
- Each processor has its own memory and disk, and communicates with other processors by exchanging messages and data over the interconnection network.
- Offer near-linear scalability.
- **Disadvantages:**
- It is most difficult to implement.
- It requires a new programming paradigm, new OS, new compilers and new programming languages





# Shared-nothing architecture



# Combined architecture

Combined hardware architecture could be a cluster of SMP nodes.

Combined parallel DBMS architecture should support **intersever parallelism** of distributed memory MPPs and **intrasever parallelism** of SMP nodes.



# Parallel DBMS Vendors

- Oracle
- Informix
- IBM
- Sybase
- Microsoft



Oracle supports parallel database processing with its add-on oracle parallel server option(OPS) and parallel query option(PQO).

## **Architecture**

- Virtual shared-disc capability
- Process-based approach
- Facilitate the inter query parallelism
- PQO supports parallel operations such as index build, database load, backup, and recovery

## **Data partitioning**

- It supports random striping of data across multiple disks.
- Oracle supports dynamic data repartitioning

## **Parallel operations**

- Generates a parallel plan



# Informix

## Architecture

- Support shared-memory, shared-disk, and shared-nothing models.
- It is thread based architecture.

## Data partitioning

- Round-robin, schema, charts, key range, and user-defined partitioning methods .
- Both data and index can be partitioned

## Parallel Operations

- Executes queries in parallel.



# DB2 -PE

Client/Server database product-DB2 parallel Edition

## **Architecture**

- DB2 PE is a shared-nothing architecture in which all data is partitioned across processor nodes.
- Each node is aware of the other nodes and how the data is partitioned

## **Data partitioning**

- Allow a table to span multiple nodes.
- The master system catalog for each database is stored on one node and cached on every other node.

## **Parallel operations**

- All database operations are fully parallelized



# Sybase

Sybase has implemented its parallel DBMS functionality in a parallel product called DYBASE MPP.

## Architecture

- It is a shared-nothing systems that partitions data across multiple SQL servers
- Open server application that operates on top of existing SQL servers.
- All the knowledge about the environment, data partitions, and parallel query execution is maintained by SYBASE MPP software.

SYBASE MPP consists of specialized servers.

- Data server, the smaller executable unit of parallelism that consists of SQL server, split server (performs joins across nodes), and control server (coordination of execution and communication)
- DBA server handles optimization, DDL statements, security and global systems catalog.
- Administrative server, a graphical user interface for managing SYBASE MPP.



# Sysbase

Administrative server, a graphical user interface for managing SYBASE MPP.

## **Data partitioning.**

- Supports hash, key range, and schema partitioning, indexes partitioning.

## **Parallel operations**

- All SQL statements and utilities in parallel across SQL servers

