1. **(a) Define distributed database system. What is the processing element in distributed computing system?**

**Answer:**

A distributed database System is a collection of databases which are distributed over different computers of a computer network.

Each site has autonomous processing capability and can perform local applications.

Each site also participates in the execution of at least one global application which requires accessing data at several sites.



**Distributed computing system** states that it is a number of autonomous processing elements (not necessarily homogeneous) that are interconnected by a computer network and that cooperate in performing their assigned tasks. The “processing element” referred to in this definition is a computing device that can execute a program on its own.

A processing element in a distributed computing system is a device or system that has its own processing capabilities and can execute tasks and communicate with other devices or systems in the network. A processing element can be a computer, a server, a workstation, a mobile device, or any other device that can perform computations and data processing. A processing element may also store and manage its own data, or access shared data from other devices or systems. A processing element is also known as a node in a distributed system.

**(b) Define each of the elements which is being distributed in distributed data processing environment.**

**Answer:**

* **Processing logic**: Processing logic in distributed data processing environment is the way of distributing the computation or processing of data across multiple nodes or machines that are connected by a network. This allows for faster and more efficient data processing, as well as scalability, availability, and fault-tolerance.
* **Function**: Various functions of a computer system could be delegated to various pieces of hardware or software.
* **Data**: Data used by a number of applications may be distributed to a number of processing sites.
* **Control**: The control of the execution of various tasks might be distributed instead of being performed by one computer system

**(c) Describe how improved performance can be achieved in distributed database system.**

**Answer:**

Improved performance can be achieved in distributed database system by using various techniques and strategies, such as:

1. A DDBMS fragments the conceptual database, enabling data to be stored in close proximity to its points of use (data localization).
   * Since each site handles only a portion of the database, contention for CPU and I/O services is not as severe as for centralized databases
   * Localization reduces remote access
2. The inherent parallelism of distributed systems may be exploited for inter-query and intra-query parallelism.
   * Inter-query parallelism results from the ability to execute multiple queries at the same time.
   * On the other hand, intra-query parallelism is achieved by breaking up a single query into a number of subqueries each of which is executed at a different site, accessing a different part of the distributed database.

(d) **Explain the alternative approaches to place the database and applications across different sites for distributed database design.**

**Answer:**

* To place the database and applications across different sites, there are two alternatives:

1. **partitioned** (or non-replicated) and
2. **replicated**.
   * **Partitioned scheme**: Database is divided into a number of disjoint partitions each of which is placed at a different site.
   * **Replicated scheme**: It can be fully replicated where the entire database is stored at each site, or partially replicated where each partition of the database is stored at more than one site but not at all the sites.

* Two fundamental design issues are **fragmentation**- the separation of the database into partitions called **fragments**, and **distribution**- the optimum distribution of the database.

1. **(a) Define the terms i) autonomy ii) distribution**

**Answer:**

i) **Autonomy**: It refers to the distribution of control, not of data. It indicates the degree to which individual DBMSs can operate independently. It is a function of a number of factors:

* whether the component systems exchange information
* whether they can independently execute transactions
* whether one is allowed to modify them.

Dimensions of autonomy:

* **Design autonomy**: Individual DBMSs are free to use the data models and transaction management techniques that they prefer.
* **Communication autonomy**: Each of the individual DBMSs is free to make its own decision as to what type of information it wants to provide to the other DBMSs or to the software that controls their global execution.
* **Execution autonomy**: Each DBMS can execute the transactions that are submitted to it in any way that it wants to.

The distribution dimension of the taxonomy deals with data. We consider the physical distribution of data over multiple sites, the user sees the data as one logical pool.

We consider two classes of DBMSs distribution:

1. client/server distribution
2. peer-to-peer distribution (full distribution).

The **client/server distribution** concentrates data management duties at servers while the clients focus on providing the application environment including the user interface.

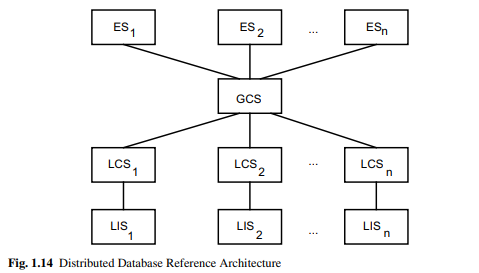
In **peer-to-peer distribution**, there is no distinction of client machines versus servers. Each machine has full DBMS functionality and can communicate with other machines to execute queries and transactions.

[N.B. -> A (**autonomy**), D (**distribution**), and H (**heterogeneity**). The alternatives along each dimension are identified by numbers 0, 1 or 2. Along the autonomy dimension, 0 represents tight integration, 1 represents semiautonomous systems and 2 represents total isolation. Along distribution, 0 is for no distribution, 1 is for client/server systems, and 2 is for peer-to-peer distribution. Along the heterogeneous dimension, 0 identifies homogeneous systems while 1 stands for heterogeneous systems]

(b) What is the significance of (A0, D2, H1)

**(c) Describe distributed database reference architecture for peer-to-peer distributed database system.**

**Answer:**

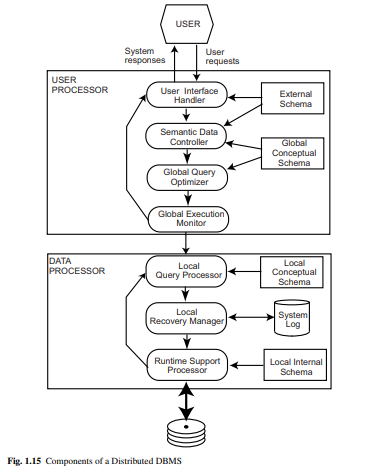


The physical data organization on each machine is different. There needs to be an individual internal schema definition at each site (local internal schema (LIS)), The enterprise view of the data is described by the global conceptual schema (GCS)- it describes the logical structure of the data at all the sites.

To handle fragmentation and replication, the logical organization of data at each site needs to be described. Therefore, there needs to be a third layer in the architecture, the local conceptual schema (LCS). The GCS is the union of the LCSs. User applications and user access to the database is supported by external schemas (ESs).

**(d) Show the components of the distributed DBMS using a diagram.**

**Answer:**



1. **(a) Describe top-down design strategies for designing distributed database.**

**Answer:**

The activity begins with a requirements analysis that defines the environment of the system and elicits both the data and processing needs of all potential database users.

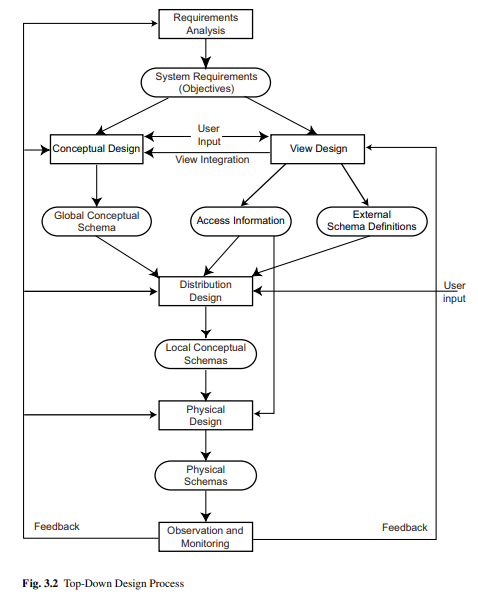
The requirements study also specifies where the final system is expected to stand with respect to the objectives of a distributed DBMS.

The requirements document is input to two parallel activities: view design (interfaces) and conceptual design (entity types and relationships).

The GCS and access pattern information collected by view design are inputs to the distribution design step. This step is used to design the local conceptual schemas by distributing the entities over the sites.

Rather than distributing relations, they are divided into sub-relations, called fragments, which are then distributed.

The last step in the design process is the physical design, which maps the local conceptual schemas to the physical storage devices available at the corresponding sites. The inputs to this process are the local conceptual schema and access pattern information about the fragments in these.



**(b) Write down the advantages of fragmentation.**

**Answer:**

Fragmentation in distributed database is the process of dividing a database into smaller parts or fragments that are stored on different computers or nodes within a network. Some of the advantages of fragmentation are:

* It enhances performance by allowing parallel processing of queries on different fragments.
* It improves data availability by providing multiple copies of data at different sites.
* It distributes the processing load by reducing the network traffic and the contention for resources.
* It allows local query optimization methods for some queries as the data is available locally.
* It maintains the security and privacy of the database system by restricting access to certain fragments.

**(c) Explain the use of reconstruction rule to ensure the correctness of fragmentation.**

**Answer:**

The following three rules together ensure that the database does not undergo semantic change during fragmentation:

1. ***Completeness***: If a relation instance **R** is decomposed into fragments R1, R2, …, Rn, each data item (HF – item refers tuple, VF – item refers attribute) that can be found in **R** can also be found in one or more of **Ri**’s (similar to lossless decomposition).
2. ***Reconstruction***: If a relation **R** is decomposed into fragments R1, R2, …, Rn, it should be possible to define a relational operator ∇ such that R = ∇Ri, ∀Ri ∈ FR. The reconstructability of the relation from its fragments ensures that constraints defined on the data in the form of dependencies are preserved.
3. ***Disjointness***: If a relation **R** is horizontally decomposed into fragments R1, R2, …, Rn and data item di is in Rj, it is not in any other fragment Rk (k ≠ j). In case of vertical partitioning, disjointness is defined only on the **nonprimary key** attributes of a relation.

**(d) Describe how primary horizontal fragmentation can be obtained using the PHORIZONTAL algorithm.**

**Answer:**

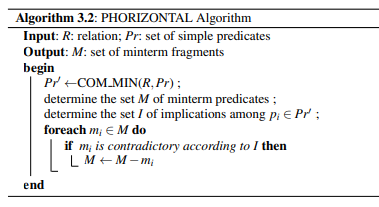
A primary horizontal fragmentation is defined by a selection operation on the owner relations of a database schema. For a given relation R, its horizontal fragments are given by

Ri = σ Fi (R), 1 ≤ I ≤ w

where Fi is the selection formula used to obtain fragment Ri. If Fi is in conjunctive normal form, it is a minterm predicate (mi).

Primary horizontal fragmentation is a technique of dividing a table into smaller sub-tables based on some simple conditions or predicates. The PHORIZONTAL algorithm is a method of obtaining primary horizontal fragmentation for a distributed database. The algorithm works as follows:

* **Input**: A relation R with attributes [A1, A2, …, An] and a set of simple predicates P = {P1, P2, …, Pm} that define the conditions for fragmentation.
* **Output**: A set of fragments F = {F1, F2, …, Fk} that are disjoint and complete, meaning that each tuple of R belongs to exactly one fragment and the union of all fragments is equal to R.



1. **(a) Explain the heuristic approaches for the vertical fragmentation of global relations.**

**Answer:**

(b) What is semantic integrity control? Define semantic integrity constraints.

(c) Explain how views can provide data security. Create a view to see the employee details of a particular Job Title.

(d) Define different aspects of data security.

1. (a) What is integrity constraint? Specify the following constraints for the given database where the primary keys are underlined:

*Emp*(**eno**, ename, title), *proj*(**pno**, pname, budget), *asg*(**eno, pno**, resp, dur)

1. The employee number functionally determines the employee name.
2. Only the tuples whose budget is 0 may be deleted.
3. The project no *pno* in relation *asg* is a foreign key matching the primary key *pno* of relation *proj*.

(b) What do you mean by integrity enforcement? Describe the ENFORCE algorithm.

1. (a) What is a query processor? Write down the characteristics of a query processor.

(b) What are the main layers of query processing? Draw the layering scheme for distributed query processing.

(c) Write down the steps of query decomposition.

(d) Explain how local query optimization is achieved in distributed query processing.

1. (a) Define normalization. Explain how an SQL query can be converted to normal forms with example.

(b) Describe how the rewriting step can be applied to a given query.

(c) What is multimedia database? Describe the organization of the multimedia data based on the principle of uniformity.

1. (a) Define query optimization. What are the components of a query optimizer?

(b) Define cost function with respect to total time and response time.

(c) Define the terms for INGRES algorithm: i) substitution ii) detachment

(d) Define data mining for distributed database environment.