Distributed Database Systems

In a distributed database system, the database is stored in several computers, So a distributed database is a collection of partially independent databases that share a common schema and coordinate the processing of transactions that access non-local data. Each processing facility communicates with the other facilities through the network.

The computers in a distributed system are called sites or nodes. The term site is appropriate because the systems are in geographically different locations.

Reasons and/or Advantages:

1. Sharing:
   The major benefit in building a distributed database system is that users at one site can access the data residing at another site .

2. Autonomy:
   The primary advantage with sharing data in this way is that users of a certain site can retain a certain degree of control over data that are stored locally. The difference with a centralized database system and a distributed database system is that in a centralized system the DBA of the central controls the database whereas in a distributed database system the local DBA controls over the database.

3. Availability:
   Distributed systems ensure very high availability because, if one site is down the operations of other sites will not be affected. Only those transactions which depend on the site currently down only will be affected.

4. Data management constraints:
   If the data available for processing is quite large and unmanageable at a single processing facility then distributed database system can be a requirement. A distributed databases divide a big and unmanageable problem into smaller pieces and solve it efficiently in a coordinated manner.

Disadvantages:

1. Recovery:
   Recovery from a failure in a distributed system is very complex.

Parallel Vs Distributed

Parallel systems have one of the following architecture:-
1. Shared memory(Tightly coupled architecture)
   Multiple processors share primary and secondary storage.
2. Shared disk(Loosely coupled architecture)
   They share only the secondary storage but have their own primary storage.
3. Shared nothing architecture
   They share neither a common disk nor a common memory.

This shared nothing architecture resembles the distributed system in many ways. However each processing facility in a parallel system is homogeneous but in a distributed system each site has its unique architecture and it is a conglomeration of dissimilar architectures. Parallel systems are very costly to setup in comparison to distributed systems. The parallel systems are also difficult to scale-up whereas in a distributed system to scale-up we have to do some modification in the database organization to add another site to the network.

Implementation Issues:

Atomicity of transactions is the most important issue in a distributed system. If a transaction uses multiple sites, it may commit at one site and the commit may fail at another site. To mitigate this problem a two phase commit protocol(2PC)[∂] is used.

A distributed software development cost is significantly higher than single site applications.

Since each site constitutes a part of the whole system, it is difficult to find bugs in the application program and data access algorithms.

The message passing between different sites is an additional computation overhead which is required to achieve inter-site coordination.

Benefits from the management of data:-
- Distribution or network transparency
  this is the freedom of the user from the operational details of the network by locational[L] and naming[N] transparency.
- Replication transparency
  Copies of data may be stored at multiple sites for better availability.
- Fragmentation transparency
  Both Horizontal[H] and Vertical[F] fragmentation is possible.

[∂] Each site will execute the transaction till just before commit and then leave the commit decision on a centralized authority. This authority is called the coordinator site. Just before commit is called the ready state in any site. If the transaction enters ready state in every site then the coordinator decides to commit the transaction. Otherwise it is aborted. If a site is in ready state but fails, then after recovery it should be in a state where it can either abort or commit.
Two phases; Phase1 for ready and Phase2 for commit.

[L] Command in independent from the site location where the data is actually present.

[N] Once the name of an object is specified, there is no need to provide any extra information. The named object would be accessible unambiguously throughout the network.

[H] Distributes a relation into sets of tuples.

[F] Dividing a relation into sub-relations with smaller set of columns.