

3.8. Date's 12 Objectives for Distributed Database Systems

Date identified 12 fundamental principles or objectives for distributed systems. He [Date] has stated that these 12 objectives are not all independent of one another, they are not necessarily exhaustive, nor are they all equally significant. Date's 12 commandments are very useful as a basis for understanding distributed database technology and as a framework for characterizing the functionalities of a specific distributed system. The discussion about distributed system will not be complete until Date's 12 commandments about distributed system are discussed, which are listed in the following.

- i. **Local Autonomy** – The sites in a distributed system should be autonomous. Local autonomy means that all operations at a given site should be managed by the particular site and no site should depend on some other site for its successful operation. Local autonomy also implies that local data is locally owned and managed with local accountability.
- ii. **No Reliance on a Central Site** – This feature implies that no site in the network relies on a central site, that is, there should be no one site in the distributed system without which the system cannot operate. There should be no central servers in a distributed system for services such as transaction management, deadlock detection, query optimization and management of the global system catalog.
- iii. **Continuous Operations** – Ideally, there should never be a requirement for a planned system shutdown for operations such as adding or removing a site from the distributed system, upgrading the DBMS at an existing site to a new release level, or the dynamic creation and deletion of fragments at one or more sites.
- iv. **Location Independence** – The basic idea of location independence is that users should not have any idea regarding the physical storage of data, but users must be able to access data from all sites, no matter where it is physically stored. Location independence is equivalent to location transparency.
- v. **Fragmentation Independence** – Fragmentation independence implies that users should not be aware of the data fragmentation. Users must be able to access all data no matter how it is fragmented.
- vi. **Replication Independence** – Replication independence means that users should be unaware that data has been replicated. Thus, users must not be able to access a particular copy of a data item directly, nor they should specifically update all copies of a data item.
- vii. **Distributed Query Processing** – Distributed systems must be able to process distributed queries that references data from more than one site. Query optimization is performed transparently by the distributed DBMS.
- viii. **Distributed Transaction Processing** – A distributed DBMS must support transaction as the unit of recovery. In a distributed DBMS, both local and global transactions should ensure data consistency, that is, the system must ensure that both global and local transactions conform to the ACID property of transactions, namely, atomicity, consistency, isolation and durability.
- ix. **Hardware Independence** – Hardware independence ensures that it is possible to run the distributed DBMS on a variety of hardware platforms.

- x. Operating System Independence** – This rule ensures that the distributed DBMS runs on a variety of operating systems.
- xi. Network Independence** – Network independence ensures that it is possible to run the distributed DBMS on a variety of disparate communication networks.
- xii. Database Independence** – This rule ensures that it is possible to have a distributed DBMS made up of different local DBMSs, perhaps supporting different underlying data models. In other words, the distributed system must support heterogeneity.

You are currently reading a **PREVIEW** of this book.
Get instant access to over \$1 million worth of books a

