

Distributive / Centralized Data Models - Benefits and Risks

Michael S. Pallos

Senior Consultant, Network Management

1. Introduction

As our dynamic technological world evolves, new decisions and alternatives are constantly presented. One such evolution is the architectural framework responsible for information management. This article focuses on the advantages and disadvantages of distributed and centralized information management. Although each customer site has unique information requirements, some common threads usually exist. The intent of this article is to offer alternatives blended with actual experience to help with the key information design decisions many of us face, distributive or centralized.

2. Centralized Data Model

Within a centralized data model all of the data is contained within one physical server. Although multiple disk arrays, DASD units, or juke boxes may be used, the data is in one location and is one, usually very large, database.

2.1. Centralized Data Advantages

The major advantage of a centralized information management model is its relative simplicity. Just as a client/server model for systems design is generally more complicated than a host centric system, a distributed data model is more complex than a centralized one, and for much the same reasons. In most categories, from programming to operations, it is easier to support centralized data. Depending on the architectural design, the centralized data model *may* also offer advantages in cost comparison (depending on the DB), response time, and data integrity.

2.2. Centralized Data Disadvantages

The Centralized data model has several disadvantages. First, being in a single geographical location, it is prone to "single point failure" and natural disasters. Even when expensive fault-tolerant hardware is used, the possibility of all data being unavailable for an extended period of time is quite real. This is generally due to site access problems even when the data is stored redundantly. A number of strategies have evolved over the years to compensate for this problem. The strategies vary from stand-by "buddy" data server sites to off-site high availability data schemes.

The second disadvantage of a centralized data model is its significant network bandwidth requirements where there is a significant number of geographically separated user sites with significant data demands. Since every remote site has to rely on the central site for its data services, the only strategy available to provide a required response time is to provide more network bandwidth. While most enterprises have been steadily increasing their network bandwidth from 9,600 baud to T1 to DS3 and higher, the demand of new applications has been outpacing the increase in network capacities. Additionally, the cost of bandwidth has not been decreasing as rapidly as cost of memory, processors, and disk storage.

The third disadvantage of centralized data is that it is conceptually designed to deal with one data manager (database management system in case of DBMS), which hampers the model's extensibility. When new technologies, like knowledge-banks become available, it may be difficult to integrate is harder to make a centralized data manager to collaborate with new data managers. This relates to the fact that centralized data managers generally have no integrative properties of their own. One can obviously wait until the vendor integrates new technology into an existing data manager.

3. Distributive Data Model

Within the distributive data model the data is housed in multiple physical locations. Distributive databases make it easy to write distributive applications without having to first decompose the application. With true distributive databases the location of the data is completely transparent to the application.

3.1. *Distributed Data Advantages.*

The chief advantage of distributed data models is that it allows access to remote data transparently while keeping most of the data local to the applications that actually use it (i.e., the data is physically closer, therefore reducing network latency time). Another, major advantage of distributed data models is that they are designed to deal with multiple and sometimes heterogeneous nodes, and have built-in mechanisms for collaboration. As result, distributed data models are extendible and provide better synergy in interacting with various data managers, from relational DBMSs to object and multimedia databases. It is becoming increasingly more common for a distributed object database to be fully interoperable with a distributed relational database, and support both object and SQL-based applications.

The second advantage of distributed data models is that they align better with organizational structures and locations. As a result, they generally offer a higher degree of scalability and flexibility. When properly designed, distributed data implementations offer better response time and require less network bandwidth. This occurs due to local nature of a significant number of information requests, providing a well-designed distributed data implementation.

Thirdly, distributed data models structurally provide better resilience in case of a natural disaster. When a disaster strikes, only a near-by node is affected while the other continues to operate independently. Additionally, the remaining data nodes can be designed to take over the effected site data since data is normally replicated in these designs.

The fourth advantage of distributed data models is that they require less powerful hardware and can grow to a substantial size without the penalty of specialized hardware, such as massive parallel database machines. The distributed data models can accommodate virtually unlimited capacities, both in storage and processing power, while using less expensive general-purpose hardware. The new "paralleling database" versions of the common commercial relations databases amplify this advantage.

