DISTRIBUTED MULTIMEDIA DATABASE MANAGEMENT SYSTEMS
Bhavani Thuraisingham and Beth Lavender
The MITRE Corporation

ABSTRACT

A multimedia database management system (M-DBMS) should provide for the efficient storage and manipulation of data represented as text, images, voice, graphics, and video. A uniform representation of the multimedia database is needed so that users can query and update it in an efficient manner through the M-DBMS.

Recently, much research has been carried out on designing and developing M-DBMSs, and as a result, prototypes and commercial products are now available. Many of these M-DBMSs have extended object-oriented database management systems to support different data types as well as multimedia data types. The different types of data such as voice and video are stored in objects. Synchronization between these objects as well as accessing these objects based on content and context needs more research. In addition, appropriate query languages, data modeling techniques, mappings to the storage objects, and metadata management techniques need to be developed. Finally, concepts in transaction processing for multimedia data types have to be revisited and the synchronization mechanisms have to be integrated into the transaction processing techniques.

Many of the C4I applications which require the support of multimedia data would typically be distributed in nature. For example, in the case of intelligence applications, the databases would probably be at multiple locations. Massive databases may be fragmented and stored as smaller chunks. Data may also be replicated for availability. The data from the multiple databases have to be retrieved, merged, and synchronized for presentation to the user. In addition, the replicated copies have to be kept consistent. Handling data distribution issues in multimedia data management has received little attention. In addition to data distribution, it may also be possible for the data to be represented using different schemes. For example, a video database of "Agency One" may use one representation scheme while that of "Agency Two" may use another representation scheme. There could also be heterogeneity with respect to different data models, retrieval algorithms, query languages, and transaction management techniques.

One of the initial steps to developing distributed multimedia database systems (DM-DBMS) includes the development of an appropriate data model for representing the distributed multimedia database. As stated earlier, much of the earlier work on M-DBMSs focussed on using an object-oriented data model for supporting different data types. However, several features such as synchronization mechanisms, relationships between the objects, decomposition and recombination of objects need to be considered. In addition to developing a conceptual model for the representation of the objects, an appropriate specification language to specify the relationships and constraints needs to be developed. Constraint specification has to be integrated with the object specification so that impedance mismatch can be minimized. Finally, the operations on the data model for retrievals and updates need to be specified.

Other steps include defining the quality of service that has to be provided to the users, developing an execution model, system and functional architectures, storage structures and mappings, and algorithms for retrieval and update. Quality of service primitives would define the type of service that is to be provided for various scenarios. For example, in certain cases it would be sufficient to obtain incomplete answers while in other cases it is essential that complete and accurate answers be provided for queries. While the data model describes the conceptual representation of the distributed database, the storage structures describe the physical representation. The mappings would define the transformations between the conceptual and physical representations. Various types of architectures can be defined for a DM-DBMS. For example, one could assume that each database has its own data manager and these data managers are interconnected using some distributed processing module. In some cases, the local data managers may not exist and therefore, the DM-DBMS will have to perform the functions of the local data managers. Schema architectures, such as the five-schema architecture, have been proposed for distributed database systems. The suitability of such architectures needs to be determined. Various data distribution schemes also need to be examined. For example, one scheme may not support the fragmentation of an object while another scheme may support it. The retrieval algorithms for multimedia data types need to support content and context-based retrieval. In addition, query transformation and optimization schemes have to be developed for each operation on the data model. Update algorithms for multimedia databases is still a major research issue.

In summary, an M-DBMS which supports distributed applications should integrate distributed/heterogeneous database technology and multimedia data management technology. Integrating the two technologies is one of the greatest challenges faced by the information systems researchers and developers.