Database System Architecture — A Walk through Time: From Centralized Platform to Mobile Computing Keynote Address

A.R. Hurson and Y. Jiao

Department of Computer Science and Engineering

The Pennsylvania State University

University Park, PA 16802

USA

hurson, yjiao@cse.psu.edu

Abstract.

Classical distributed database systems monolithically offer distribution transparency and higher performance. This is made possible by making data available and closer to the application domain(s) that uses it by means such as the data distribution, duplication, and fragmentation. However, with the advances in technologies this monolithic and top down approach becomes insufficient. In the new networked computational environment, the data distribution issue has evolved into data integration from several heterogeneous and autonomous data sources. Heterogeneous distributed databases are designed to deal with issue of data integration and interoperability. They are developed to allow timely and reliable access to heterogeneous and autonomous data sources in an environment that is characterized as "sometime, somewhere."

The concept of mobility, where users access information through a remote connection with portable devices, has introduced additional complexities and restrictions to the heterogeneous distributed database system. This keynote address first introduces a three dimensional space to classify and identify the evolution of different classes of database systems. It also extensively discusses Heterogeneous Distributed Database Systems (HDDBS) and Mobile Data Access Systems (MDAS). Finally, it will address several research issues and their potential solutions.

1 Information Everywhere, Computers Everywhere

Advances in computation and communication technologies have enabled users to access information at anytime, from anywhere. However, this flexibility comes at the expenses of new challenges. These challenges stem from the diversity in the range and the exponential growth of information that is available to a user at any given time. The spread of computer networks, the wide breadth of access devices with different physical characteristics, and the extensive need for information sharing have created a demand for cooperation among pre-existing, distributed, heterogeneous, and autonomous information sources in an infrastructure that is characterized by:

- Low bandwidth,
- Frequent disconnection,
- High error rates,
- Limited processing resources, and
- Limited power sources.

Binary data was the main data format in 1970s and it has evolved into text, images, multi media, and sensor data during the past decades. In mid 1980s, it was estimated that the U.S. Patent Office and Trademark has a database of size 25 terabytes (1 terabytes = 10^{12} bytes) subject to search and update. In 1990s, it was estimated that the NASA's Earth Observing Project would generate more than 11,000 terabytes of data during the 15-year time-period. Recently, it was estimated that the amount of new information generated in 2002 was about 5 exabytes (1 exabyte = 10^{18} bytes).

The diversity in representation, the growth in size, and the increased availability of information also introduced new challenges in areas such as security and resource management (e.g., power and network bandwidth). The cost associated with viruses, unsolicited emails, and other attacks has grown exponentially. For example, it is estimated that 7000 new computer viruses were discovered in 2003 and the FBI approximated that computer viruses cost businesses $27 million during that time. As another example, in 2003, unsolicited email cost businesses $20 billion worldwide due to lost productivity, system overhead, user support, and anti-spam software.

The availability of heterogeneous, autonomous, and partially unreliable information in various forms and shapes brings out the following challenges:

- How to locate information intelligently, efficiently, and transparently?
- How to extract, process, and integrate relevant information efficiently?
- How to interpret information intelligently?
- How to provide uniform global access methods?
- How to support user and data source mobility?

2 Database Systems Taxonomy

Different parameters can be used to classify the architecture of data base systems. We classify data base systems along the following three parameters:

- **Physical infrastructure**: This dimension refers to the underlying platform composed of homogeneous/heterogeneous processing devices interconnected through different communication medium. The processing devices ranging from powerful parallel machines to portable units communicating with each other via a wide variety of communication medium ranging from land-based connection to wireless medium.
- **Services**: Along this dimension one can distinguish two classes in which either there is no distinction between services, or there is a distinction between user processes and data processes.
- **Distribution**: This dimension distinguishes distribution of processing, data, and control (also known as autonomy). Note that data distribution also includes data fragmentation and data replication.

2.1 Centralized Databases

A centralized database system is the one that runs on a single computer platform and does not interact with other computer systems. Based on our taxonomy, a centralized database is characterized by: its single processing unit, without distinction between its services, and without any notion of distribution. The centralized database systems can be further classified as, single-user configuration and multi-user configuration. Naturally, database systems designed for single-user configuration do not provide many facilities needed for a multi-user system e.g., concurrency control, and security.

2.2 Parallel Databases

Let us inject processing distribution into the scope of the centralized database system. This brings out the so called parallel database systems, in which several processing resources in cooperation with each other are intended to resolve users' requests. Note that in this environment there exist no notion of data and control distribution. In addition, there is no distinction between services that are provided by the database management system. Parallel configurations are aimed at improving the performance and throughput by distribution of a task at different granularity (fine, medium, or coarse) granularity among several processing units. The literature has introduced four classes of parallel systems:

- Shared Memory (tightly coupled) - All processors share a common global memory.
- Shared Disk (loosely coupled) - All processors share a common set of disks.
- Share Nothing - The processors share neither a common memory nor common disks.
- Hierarchical - A hybrid of the other models.

As noted in the literature; the shared memory configuration is not scalable and the communication network is the system bottleneck. Raid technology can be used to improve performance and reliability of the disk subsystem. The shared nothing configuration offers scalability at the expenses of high inter-processor communication cost. Finally, in the hierarchical topology, at the higher level, system acts as a shared nothing organization and at the lower level, each node could be a shared memory and/or a shared disk system.

2.3 Client/Server Topology

In sections 2.1 and 2.2 we looked at two configurations that did not make any distinction between the services provided by the database management system. Along the *services*, we distinguish two classes of functions:

- The data functions (the back end processes) - query processing, query optimization, concurrency control, and recovery.
- The user functions (the front end processes) - report writer and Graphical User Interface facilities.

This brings out the so called client/server topology that has functionality split between a server and multiple clients. The client/server topology can be grouped into multiple client/single server and multiple client/multiple server configurations. Functionality and processing capability of the client processors and communication speed between the client and server also distinguishes two classes of the client/server topology, namely, transaction server (thin client) and data server (fat client). The client/server topology is one step towards distributed processing. It offers a user-friendly environment, simplicity of implementation, and high degree of hardware utilization at the server side.

2.4 Peer-to-Peer Topology

This topology is a direct evolution of the client/server topology. Note that in the client/server topology functionality is split into user processes and data processes, in which user processes handle interaction with the user and data processes handle interaction with data. In a Peer-to-Peer topology, one should expect to find both classes of processes placed on every machine. From a data logical perspective, client/server topology and Peer-to-Peer topology provide the same view of data - data distribution transparency. The distinction lies in the architectural paradigm that is used to realize this level of transparency.

2.5 Distributed Databases

Distributed databases are based on data distribution. It brings the advantages of distributed computing to the database management domain. A distributed system is a collection of processors, not necessarily homogeneous, interconnected by a computer network. Data distribution is an effort to improve performance by reducing communication costs and to improve quality of service in case of network failure. Based on our taxonomy, a distributed database system has the following characteristics: data is distributed (possibly replicated and/or fragmented) stored in locations close to the application domain(s) that uses it (e.g. increased availability), processors do not share resources (i.e., disks and memory) and processes are more distinct, and the underlying platform is possibly parallel. In comparison to parallel systems in which processors are tightly coupled and constitute a single database system, a distributed data base system is a collection of loosely coupled systems that share no physical components.

How to distribute data in order to improve performance, reliability, and accessibility, and how to provide transparency are the key issues in the design of distributed databases. Table 1 enumerates some important issues that one needs to consider in the design of distributed database systems.

Table 1. Issues in distributed database systems

ISSUES	REMARKS
Data Distribution	How data should be distributed/replicated/fragmented in order to improve performance/reliability/accessibility
Distribution Transparency	Distribution transparency includes network (location), replication, and fragmentation transparency
Keeping Track of Data	Keeping track of the data distribution, fragmentation, and replication
Replicated Data Management	Which replica to access, how to maintain consistency, and how to control number of replica
Database Recovery	Recover from individual site crashes and system failures
Query Processing and Query Optimization	Query resolution, generation of sub-queries, and aggregation of partial results
Transaction Management	Concurrency control protocols
Security	Authenticate users, enforce authorization and access control, and auditing

2.6 Multidatabases

Adding control distribution to the definition of distributed databases as discussed in section 2.5 results in an environment with the following characteristics:

- Data is distributed and stored in several locations,
- Processes are more distinguished,
- Underlying platforms could be parallel, and
- Processing nodes are autonomous.

This brings out a new computational paradigm that is referred to as multidatabase or heterogeneous distributed database. Due to the autonomy, local databases can join or leave the global information infrastructure at will. As noted in the literature, the autonomy comes in the form of design autonomy, communication autonomy, execution autonomy, and association autonomy. Reading between the lines, autonomy implies heterogeneity. Therefore, autonomy and heterogeneity are the major features that distinguish a multidatabase system from a traditional distributed database system.

To conclude, multidatabases are more dynamic and robust than distributed databases - i.e., system expands and shrinks more rapidly. The design of multidatabase is a bottom up approach - i.e., integration and interoperability of pre-existing data bases. Consequently, while data distribution is a major concern in the design of distributed databases, data integration is the major concern in the design of multidatabases. Research issues in multidatabases (see table 2) are similar to the ones in distributed databases, however, their implementations are becoming more complicated because of the autonomy and heterogeneity of local databases.

2.7 Mobile Computing

Remote access to various types of data is not a new concept. Traditional distributed databases and multidatabases relying on fixed network connectivity have addressed many of the issues involved in accessing remote data. However, the concept of mobility, where a user accesses data through a remote connection with a portable device, has introduced several disadvantages for traditional database management systems. These include:

Table 2. Issues in multidatabases

ISSUES	REMARKS
Site Autonomy	Local databases have complete control over local data and local processing
Heterogeneity	Support of providing local translation capability from the local model to the common global model
Differences in Data Representation	This includes issues such as name differences, formal differences, structural differences, abstraction differences, and missing or conflicting data
Global Constraints	Need for some method for specifying and enforcing integrity constraints on inter-database dependencies and relationships
Query processing and Query Optimization	Query decomposition and optimization in face of the component databases which could be unavailable, unwilling, and uncooperative
Transaction Management	Concurrency control protocols in the face of global/global and global/local conflicts among transactions
Security	Potential conflicts between global security protocols and local security protocols
Local Node Requirements	Computational and storage capabilities of local nodes to fulfill some global functions

- Reduced network connectivity,
- Processing and resource restrictions, and
- Effectively locating and accessing information from a multitude of sources.

The mobile computing environment is based on wireless communication that allows the user to access information anywhere at anytime without direct physical link to the network. The wireless network (Figure 1) is mainly composed of the following:

- A number of network servers enhanced by wireless transceivers, called mobile support stations (MSS), scattered along a geographical area and
- A varying number of mobile hosts (MHs) free to move at will.

The MSS provides a link between the wireless network and the wired network. The link between a MSS and the wired network could be either wireless (shown as dashed line) or wire based. The area covered by the individual transceiver is referred to as a cell. The size of the area covered by each cell varies widely, depending on the technology being used. The MH is relatively small, light weight, and portable. It is designed to preserve space and energy, and it usually has limited amount of resources (memory, processing power, etc.). Most of the time, the MH relies on temporary power sources such as batteries as its main power source. To save energy, the MH is design to operate in different operational modes (i.e. active doze, sleep, nap) that consume different amount of power.

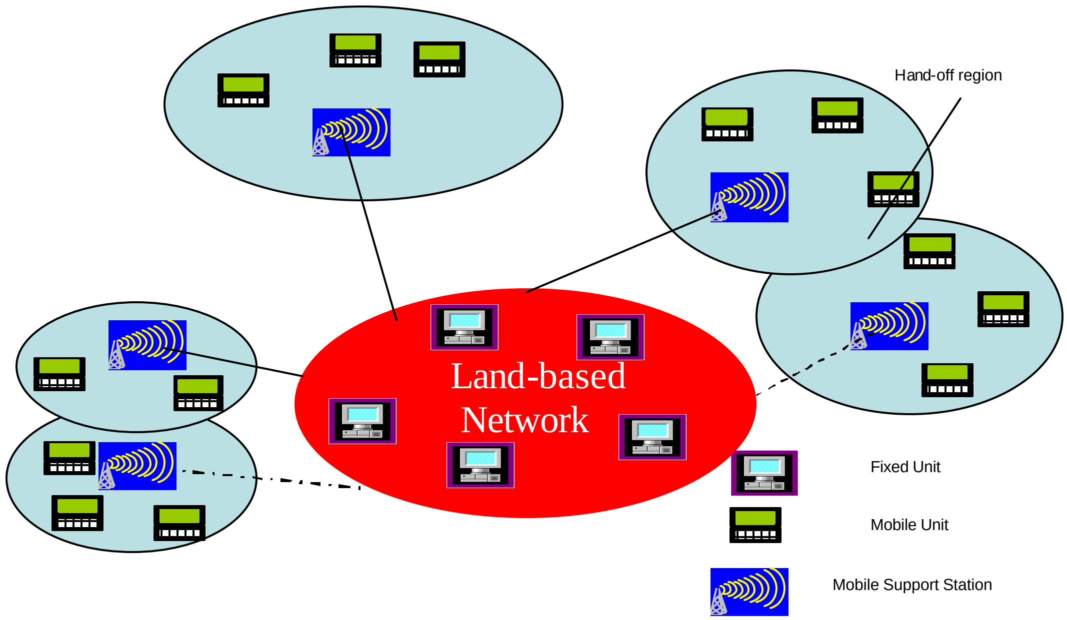

Fig. 1. Architecture of the mobile computing environment

Within the scope of this infrastructure, two types of services are available to the users:

- On-demand-based services, and
- Broadcast-based services.

On-demand-based services. In this configuration, users obtain answers to requests through a two-way communication with the database server (s) - the user request is pushed, data sources are accessed, query operations are performed, partial results are collected and integrated, and generated information is communicated back to the user. In order to overcome the shortcomings of a mobile computing environment effectively, a suitable solution must address issues such as; Security and access control, isolation and autonomy, data heterogeneity, data integration, distribution transparency, location transparency, browsing, query processing and query optimization, transaction processing and concurrency control, limited resources, and limited power sources. Table 3 enumerates issues of importance in this environment.

Broadcast-based services. Many applications are directed towards public information that are characterized by i) the massive number of users and ii) the similarity and simplicity in the requests solicited by the users. The reduced bandwidth attributed to the wireless environment places limitations on the rate and amount of communication. Broadcasting is a potential solution to this limitation. In broadcasting, information is generated and broadcast to all users of the air channels. Mobile users are capable of searching the air channels and pulling their required data. The main advantage of broadcasting is the fact that it scales up as the number of users increases, eliminating the need to multiplex the bandwidth among users accessing the air channel. Furthermore, broadcasting can be considered as an additional storage available over the air for the mobile clients. This is an attractive solution because of the limited storage capability of the mobile unit. Within the scope of broadcasting one needs to address three issues:

- Data selection,
- Effective data organization on the broadcast channel, and
- Efficient data retrieval from the broadcast channel.

Table 3. Issues in on-demand-based environment

ISSUES	REMARKS
Site Autonomy	Local control over resources and data. The degree of autonomy required depends upon the degree of mobile support offered by the system
Heterogeneous Interoperability	Hardware and software heterogeneity
Disconnect and Weak Connection Support	A mobile system should provide a means to provide access to data while faced with a disconnection or weak connection
Support for Resource Scarce Systems	A mobile system should address the inherent limitations of various resource scarce access devices. These include processing, storage, power, and display limitations.
Transaction Management and Concurrency Control	Correct transaction management should satisfy the ACID properties (Atomicity, Consistency, Isolation, and Durability)
Distribution Transparency	Distribution of data is transparent to the user
Location Transparency	The location of the data is transparent to the user
Location Dependency	The content of the data is physically dependent upon the location of the user
System Transparency	The user should be able to access the desired data irrespective of the system
Representation Transparency	Representation transparency includes naming differences, format differences, structural differences, and missing or conflicting data
Intelligent Search and Browsing of Data	The system should provide a means for the user to efficiently search and browse the data
Intelligent Query Resolution	The system should be able to efficiently process and optimize a query submitted by the user

3 Our Research

Our research is intended to develop an information infrastructure that allows anytime, anywhere, transparent, intelligent, secure, timely, reliable, and cost effective access to global information regardless of:

- Heterogeneity of access devices,
- Heterogeneity of communication medium, and
- Heterogeneity and autonomy of data sources.

As part of our research we developed an Intelligent Search Engine - Summary Schemas Model, which can automatically identify the semantic similarity among different access terms. It uses linguistic relationships between schema terms to build a hierarchical global metadata, which describes the information available in all local databases in an increasingly abstract form. The hierarchical metadata is a collection of summary schemas. A summary schema is a concise, abstract description of the semantic contents of a group of input schemas. Through extensive simulation study it has been shown:

- The hierarchical metadata of the summary schemas model is by orders of magnitude smaller than the metadata generated by the global-schema approach.
- The summary schemas model preserves local autonomy.

- The summary schemas model provides good system scalability.
- The summary schemas model reduces average search time.
- The summary schemas model resolves imprecise queries.

The SSM has also been prototyped based on the client/server and mobile agent paradigms. In addition, the scope of the SSM was extended by allowing mobility both at the client side and the database side. Within the scope of this mobile computing environment several issues as numerated in table 4 have been researched.

Table 4. Our research efforts

UNDERLYING ENVIRONMENT	RESEARCH ISSUES
On-Demand-Based Services	Transaction processing Query processing Data caching and data duplication Power management Security
Broadcast-Based Services	Access latency and power management Application of index and organization of objects on single broadcast channel Application of index and organization of objects on parallel broadcast channels Conflict resolution and scheduling of data retrieval from parallel broadcast channels

4 Conclusion

The following is a partial list of challenges that deserve attention of researchers:

- Intelligent energy management,
- Interoperability across platforms,
- Portability,
- Security,
- Location dependent services, and
- Pervasiveness

References

[1] Bright, M.W., Hurson, A.R. and Pakzad, S.H., "Issues in Multidatabase Systems," IEEE Computers, 25(3):50-60, 1992.

[2] Bright, M.W., Hurson, A.R. and Pakzad, S. "Automated Resolution of Semantic Heterogeneity in Multidatabases," ACM Transactions on Database Systems, 19(2):212-253, 1994.

[3] Haridas, H., Hurson, A.R., and Jiao, Y., "Security Aspects of Wireless Heterogeneous Databases - Protocol, Performance, and Energy analysis", IEEE WiSPr 2003, pp. 417 -424.

[4] Hurson, A.R., Ploskonka, J., Jiao, Y., and Haridas, H., "Security issues and Solutions in Distributed heterogeneous Mobile Database Systems," Vol., 61, Advances in Computers, 2004, pp. 107-198.

[5] Hurson, A.R. and Jiao, Y., "Data Broadcasting in a Mobile Environment", Wireless Information Highway, 2004.

[6] Jiao, Y. and Hurson, A.R., "Application of Mobile Agents in Mobile Data Access Systems - A prototype", Journal of Database Management, 15(4):1-24, 2004.

[7] Lim, J.B., Hurson, A.R., "Heterogeneous Data Access in a Mobile Environment — Issues and Solutions," Advances in Computers, Vol. 48, 1999, pp. 119-178.

[8] Lim J.B., and Hurson A.R., "Transaction Processing in Mobile, Heterogeneous Database Systems," IEEE Transactions on Knowledge and Data Engineering, 14(6):1330-1346, 2002.

[9] Ngamsuriyaroj, S., Hurson, A.R., and Keefe, T.F., "Authorization Model for Summary Schemes Model", *Proceedings of the International Database Engineering and Applications Symposium*, IDEAS 2002, pp. 182-191.

[10] Orchowski, N. and Hurson, A.R., "Energy-Aware Object Retrieval from Parallel Broadcast Channels", *Proceedings of the International Database Engineering and Applications Symposium*, IDEAS 2004, pp. 37-46.

[11] Segun, K., Hurson, A.R., Desai, V., Spink, A., and Miller, L.L., "Transaction Management in a Mobile Data Access System," Annual Review of Scalable Computing, Vol. 3, 2001, pp.85-147.

[12] Sun, B., Hurson, A.R., and Hannan J., "Energy-Efficient Scheduling Algorithms of Object Retrieval on Indexed Parallel Broadcast Channels", International Conference on Parallel Processing, 2004, pp. 440-447.