Database Integration over Hybrid Networks*

Ouri Wolfson+, Naphtali Rishe, Scott Graham, Wei Sun

+Department of Electrical Engineering and Computer Science
University of Illinois at Chicago
851 S. Morgan St., Chicago, IL 60612, USA.
Tel: (312) 996-6770, Fax: (312) 413-0024, E-mail: wolfson@eecs.uid.edu

High Performance Database Research Center
School of Computer Science
Florida International University
11200 SW 8TH ST, Miami, FL 33199 USA
Tel: (305) 348-1706, Fax: (305) 348-1705, E-mail: {rishen, grahams, weisun}@cs.fiu.edu

Abstract: *The use of intelligent network technology and hybrid networks that include intelligent networks has created new challenges for those who wish to integrate heterogeneous databases distributed across these networks. Advances in public network technology enable the implementation of new distributed database systems that require improved query distribution and optimization algorithms. Reducing response time has typically been the key focus of distributed query execution over traditional networks. The new intelligent and hybrid networks allow optimizations to be made to minimize the communication costs needed to execute queries. In this paper we extend our database integration methodology to hybrid networks by simulating ISDN networks over TCP/IP.*

KEYWORDS: Database Integration, Query Optimization, Intelligent Networks, Hybrid Networks

INTRODUCTION

Network technology is continuously developing to meet the requirement of information sharing among customers and business organizations. New versions of these networks are expected to become available to businesses and consumers since users expect to be able to retrieve information across networks at a faster speed and at less cost. Intelligent public networks, such as ISDN for example, have become widely available in North America during the past few years. These networks can be used to provide distributed computing to those who have found it too expensive in the past by providing a cost savings over traditional distribution techniques such as leased lines. ISDN lines are also available to a wider range of customers than newer technologies such as DSL lines. Intelligent networks are significant to the database research community because they render many assumptions and parameters of conventional distributed database systems obsolete. As such, database researchers need to pay more attention to the new challenges brought about by these newly emerging technologies.

* This research was supported in part by NASA (under grants NAG5-9478, NAGW-4080, NAG5-5095, NAS5-97222, and NAG5-6830), NSF (CDA-9711582, IRI-9409661, HRD-9707076, and ANI-9876409), and ONR (N00014-99-1-0952).

There have been many theoretical studies in database integration. [1], [2], [3], and a number of prototype systems such as Information Manifold [4], Multibase [5], MRDSM [6], Pegasus [7], Carnot [8], SIMS [9], TSIMMIS [10,11], and SEMHDB [15,16] each represent a different methodology. This previous work was all performed with traditional distributed database environments, where databases are connected by privately owned networks, in mind. By making use of intelligent network technologies, such as ISDN, distributed databases will be able to communicate through public networks provided by the telecommunication companies. This transition has changed the nature of the networks from static to dynamic. Using these new technologies, network resources can be dynamically allocated and manipulated directly by the database systems to allow them to achieve better network utilization and overall performance. These changes require database integration techniques, including distributed query optimization techniques, to be revisited since the assumptions that were typically made about the networks in previous studies are no longer valid.

In [17], we presented an overview of our approach to integrating databases over intelligent networks, which was first proposed in [12]. This paper addresses our efforts to extend our approach to include hybrid networks. The remainder of this paper is organized as follows. Section 1 presents an overview of our technical approach by discussing the main research objectives. Section 2 gives an overview of our original prototype system that meets the needs described in Section 1. Section 3 presents our techniques for extending this work to hybrid networks. Finally, in Section 4, we give the conclusion of our work.

1. Query Distribution Method

Our Query Distribution Algorithm (QDA) is a heterogeneous distributed database system based on ISDN network. It adopts the concentrated management mode. AMIP, the Application Management Interface Program, is the concentrated manager server of the whole system. LIP, the Local Interface Program, is the client program offering a user interface. The heterogeneous DBMS in each local site communicates with AMIP and LIP through ODBC. The system's physical architecture is depicted in figure 1.1.

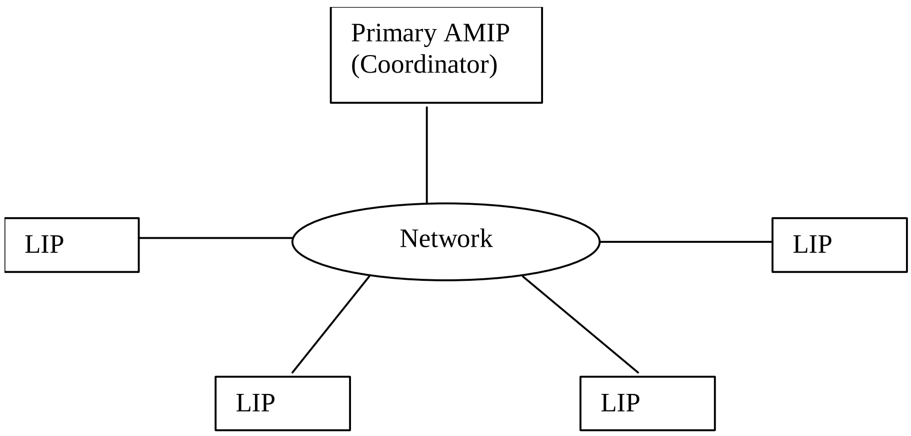

Fig 1.1 - Physical Architecture of QDA

Our system maintains a collection of essential knowledge, including global schemata and mapping information, to resolve the possible heterogeneity [13], determining schematic and data conflicts between the global schema and local schemata [14]. Usually, such conflicts occur when semantically equivalent data is represented in different ways. Our prototype focuses on the four major types of conflicts that are most often cited in the literature. They are:

- *Structural heterogeneity.* Logical data structures (i.e. the number of relations and the foreign key joins between the relations) may vary across databases that contain similar data. This is due to different preferences on how data should be organized and due to variations in database contents.
- *Abstraction heterogeneity.* This class of heterogeneity arises when two databases use different levels of generalization and aggregation and retain different levels of information detail.
- *Naming heterogeneity.* Naming variations can appear in two levels: the relation level and attribute level. Differences in the names of two similar relations or attributes will make them appear different to the database system, thus requiring semantic reconciliation.
- *Domain heterogeneity.* Even for same-named attributes, the underlying domains may be different.

Our system makes use of intelligent network features to optimize for interconnection costs as well as response time. We have identified techniques to take advantages of ISDN's tariff structure. These include the following:

- Intelligent connect/disconnect strategies: For long-distance ISDN connections, the charge is based on usage increments, regardless of whether or not one uses up all the available bandwidth during the entire increment. Thus, the connection must be kept until the current increment expires so that the remaining increment can be used for another possible data transfer without additional cost.
- Dynamic channel allocation: Weighing between query response time and ISDN monetary cost, techniques are developed to determine how many channels should be allocated to achieve a maximum weighted cost.
- Least-cost data transfer paths: To minimize the ISDN monetary cost incurred, the least-cost path must be found for each data transfer. We have modified Dijkstra's shortest path algorithm to take into account both ISDN cost and transfer time. The new algorithm not only determines the path via which the data should be routed, it also determines how many channels should be allocated for each direct edge along the path.

2. Intelligent Network Prototype

We have implemented a prototyping system based on the methodology introduced in Section 1. This section presents a brief introduction to our prototype system.

Figure 2.1 depicts the basic architecture of our loosely coupled multi-database environment. Each site in the figure runs an autonomous DBMS. One site among them is designated as the coordinator whose responsibility is to receive inter-database query requests from the other sites and generate the corresponding global execution plan. There are two different kinds of software components in the system: the Application Manager Interface Program (AMIP), which runs on the coordinator site, and the Local Interface Program (LIP), which runs on the remaining sites. We use ODBC to resolve the potential heterogeneity among the different kinds of DBMSs. The LIP is a specific ODBC driver. An application program submits ODBC compliant SQL queries to an LIP by means of the ODBC manager. If the queries only relate to the local database, the LIP will access the local database by using the ODBC manager to select another ODBC driver that is provided by the local DBMS. If the queries need to access one or more remote databases, the LIP will send the queries to the AMIP. The AMIP parses and decomposes the received queries into a global execution plan that is composed of a set of sub-queries. The AMIP coordinates the processing of each sub-query by brokering connections among the involved local sites. To reduce ISDN costs, with AO/DI, queries and control messages are communicated through D channels while the query results are routed through B channels.

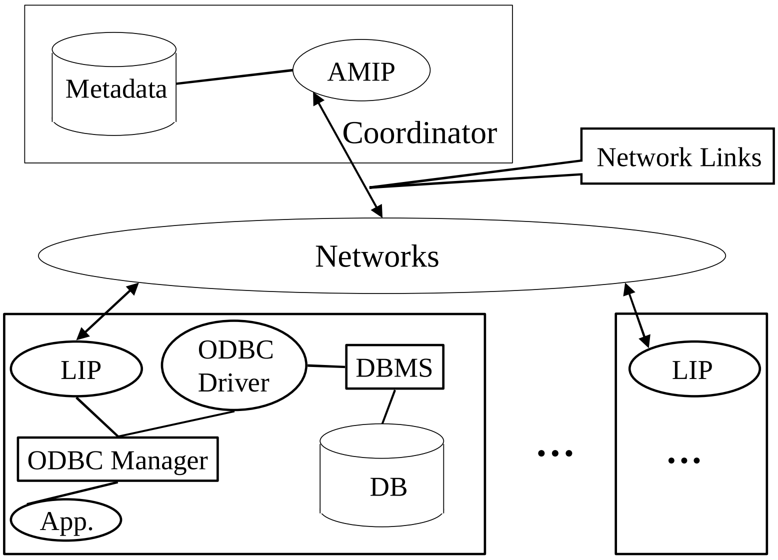

Figure 2.1 - System Architecture of the Prototype System.

3. Extending to Hybrid Networks

A distributed environment might consist of a diverse set of networks that connect a number of computers. For example, within an enterprise, computers may be connected via Local Area Networks (LANs) or Wide Area Networks (WANs), but different enterprises may choose ISDN lines to communicate with each other. It is, in fact, quite common for users to have to face a hybrid network infrastructure that is, for example, a combination of traditional networks and intelligent networks.

We have proposed a query distribution strategy for such hybrid network infrastructures based on the fundamental principles presented in section 1. Our query distribution strategy works well with traditional TCP/IP based networks as well as ISDN.

Figure 3.1. depicts a typical example of a hybrid network infrastructure. Assume that there are three sites, $Site_0$, $Site_1$, and $Site_2$, in the system. $Site_0$ and $Site_1$ are connected via ISDN lines and $Site_0$ and $Site_2$ are connected via a LAN. For this topology, we choose $Site_0$ to be the system coordinator, so the AMIP runs at $Site_0$.

A RAS server is required to run on $Site_0$ to enable the LIP running at $Site_1$ to communicate with the AMIP running at $Site_0$ through ISDN lines via Dial-up Networking. At the same time, another LIP on $Site_2$ communicates with the AMIP at $Site_0$ via the LAN. Technically, to realize this hybrid-networking version of QDA, we have designed an IP-Site map table so that all LIP sites are known to one another. The number of LIP sites dialing in through ISDN network is limited by the capacity of the ISDN NAS.

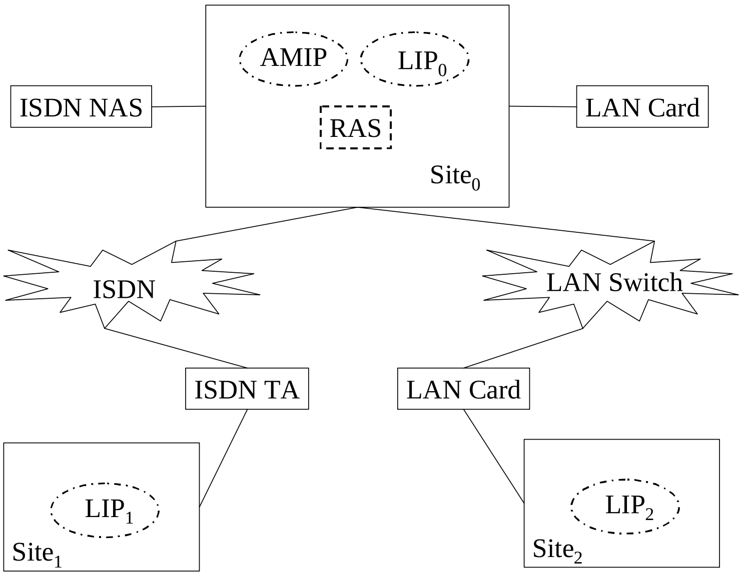

Figure 3.1 - Example Hybrid Network Infrastructure

Simulating ISDN networks with WinSock

In order to add this flexibility to our prototype system, we have implemented another communication module based on WinSock. This module allows the prototype system to retrieve information from databases across a LAN or WAN using TCP/IP protocol, as well as from databases across the ISDN networks.

The WinSock-based module was designed to allow us to simulate ISDN networks via TCP/IP networks. This allows the low-level communication module to be separated from the high level QDA program and LIP program.

Without losing generality, we can place the AMIP and one of the LIPs, for example LIP1, at the same computer in our prototype. In such cases, the communication between the AMIP and LIP1 does not, therefore, use the ISDN network, but instead uses shared system memory. In fact, it uses a kind of inter-process communication, which can be implemented using a File Mapping technique. Using TAPI, we have implemented a module, MONITOR, whose responsibility is to monitor incoming calls, receive messages through the ISDN network, and to put them into the shared memory for the AMIP or the LIP. We have also implemented a dynamic linking library (DLL), Dcomm, for sending messages through the simulated “D-channnel,” which is actually a socket. The WinSock simulation communication module for the computer that runs both the AMIP and a LIP is shown in figure 3.2.

The arrows in the figure depict the transmission of messages. The thin solid arrows represent the transmission of the incoming messages. The MONITOR monitors line status. When an incoming message appears, the MONITOR will receive it, and then put it into the AMIP’s or the LIP’s shared memory as appropriate by referring to the header flag embedded in the message. The thick solid arrows represent the transmission of outgoing messages. The AMIP and the LIPs are able to send a message to the network by calling the *SendMessage* function in the Dcomm library. The dashed arrows represent the transmission of messages between the AMIP and the LIP, which reside on the same computer in this example.

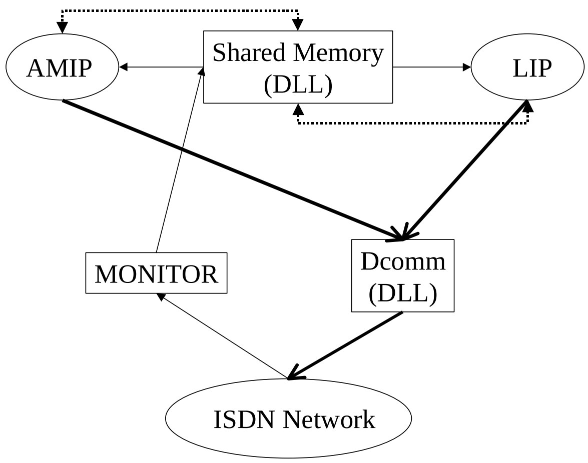

Figure 3.2 - Winsock Communications Module

To better simulate the ISDN communication, a special data structure for table mapping is needed to describe the relationship between bound B-Channel connections and real B-Channel handles or WinSock handles. We can easily migrate the QDA program from an ISDN communication environment to a TCP/IP network environment by changing the appropriate field in this table—simply replace the B-Channel file handle with the WinSock handle .The data structure of the mapping table is depicted in Figure 3.3.

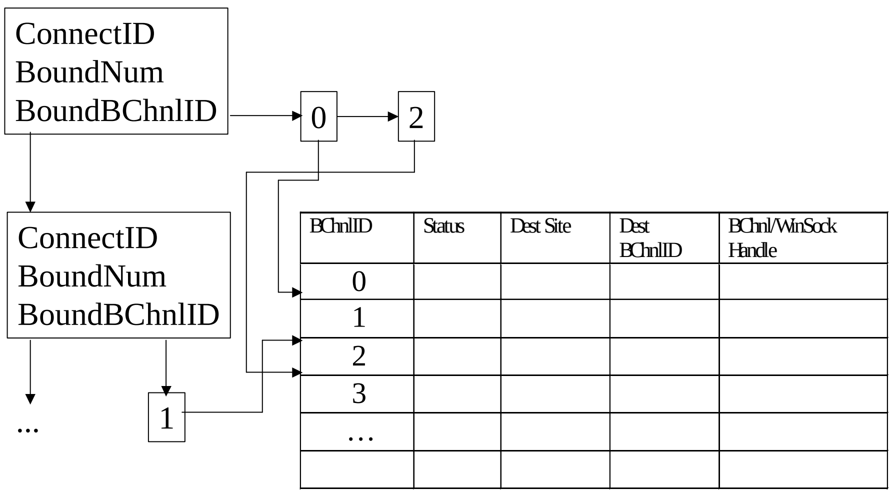

Figure 3.3 - Communication Mapping Table

Additionally, there is another table mapping BchnlID and BChannel phone number at each LIP site. When a B channel connection needs to be set up, the LIPs find the relevant B channel number via this mapping table and then dials the number. In the WinSock version, there is no need for such a mapping table.

4. Conclusions

Our prototype implementation has shown that our methodology is very scalable. Our query distribution methodology has no theoretical limit to the number of the participating sites.

Practically, however, the scale of the system is limited by hardware capacities and, in particular, the speed of the computer at the AMIP site. Since all query requests are first sent to the AMIP site, a large number of concurrent transactions can become a great burden to the AMIP site. Regarding our query distribution strategy, the number of participating sites is not the key criteria for scaling. The key point is how many sites will actually be involved in each particular distributed query. Our experience in database application development has shown us that the number of sites participating in each query is likely to be less than 5. As a result, even BRI service, which has only two B-Channels, can most likely meet the basic requirement for the execution of all distributed queries. Therefore, as long as the hardware components are powerful enough, the scalability of our prototype system is quite large.

We feel that several technical points could benefit from continued research to improve the performance, extensibility, adaptability, consistency, and security of our methodology:

- The hybrid network structure, which combines an ISDN network and a conventional LAN together, could be further investigated with the aim of increasing the extensibility and adaptability of this system.
- A dynamic query optimization strategy, which enables the system to change the generated execution plan dynamically based on some changing factors could be explored. Examples of factors that could be considered include the available memory size, the number of the active processes, etc.
- Encryption techniques for data transfer could be explored in order to increase the system's security against intrusion since the security of information is constantly becoming more and more important.
- The introduction of some type of dual AMIP architecture may increase the reliability and durability of this system. This, and other ways to bring fault tolerance to the system, could be explored.

REFERENCES

[1] B. E. Reddy, P. G. Reddy, and A. Gupta: "A methodology for integration of heterogeneous database." *IEEE Transactions on Knowledge and Data Engineering*, 6 (6): 920-933, December 1994.

[2] A. Sheth and V. Kashyap: "So far (schematically) yet so near (semantically)." *Interoperable Database Systems*, pages 283--312, Elsevier Science Publishers B. V. (North-Holland) 1993.

[3] A. Sheth and J. Larson: "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Database." *ACM Computer Surveys*, 22 (3):183-235, Sept. 1990.

[4] A. Levy, et al: "Querying Heterogeneous Information Sources Using Source Descriptions" In *Proceedings of the 22nd VLDB Conference*, 251-262, 1996, Bombay, India.

[5] W. Litwin and A. Abdellatif: "Multidatabase interoperability." *IEEE Computer*, pages 10-18, December 1986

[6] W. Litwin and A. Abdellatif: "An overview of the multi-database manipulation language MDSL." *Proceedings of the IEEE*, 75(5):621-632, May 1987.

[7] R. Ahmed et al: "The Pegasus heterogeneous multidatabase system." *IEEE Computer*, 24 (12):19-27, December 1991

[8] C. Collet, M. N. Huhns, and W. Shen: "Resource integration using a large knowledge base in Carnot." *IEEE Computer*, pages 55--62, December 1991.

[9] Y. Arens, C. Chee, C. Hsu, and C. Knoblock: “Retrieving and integrating data from multiple information sources.” *International Journal on Intelligent and Cooperative Information Systems*, 2(2), June 1993.
[10] H. Garcia-Molina et al: “The TSIMMIS approach to mediation: data models and languages” In *Next Generation Information Technologies and Systems*, June 1995. Nahria, Israel.
[11] Y. Papakonstantinou, S. Abiteboul, and H. Garcia-Molina: “Object fusion in mediator systems” In *Proceedings of the 22nd VLDB Conference*, 1996, Bombay, India.
[12] N. Rishe: “Managing network resources for efficient, reliable information systems,” Proceedings of the Third International Conference on Parallel and Distributed Information Systems, p.223-6, Austin, TX, USA, Sept. 1994.
[13] W. Kim, et al: “On Resolving Schematic Heterogeneity in Multidatabase System,” Distributed and Parallel Databases, Vol.1 No. 1, p.251-279, 1993.
[14] N. Rishe, R. Athauda, J. Yuan, S. Chen: “Semantic Relations: The key to integrating and query processing in heterogeneous databases,” The 4th World Multiconference on Systemics, Cybernetics and Informatics, SCI 2000, Vol. VII, pp. 717-722.
[15] N. Rishe, et al: “Semantic Access: Semantic Interface for Querying Databases,” Proceedings of the 26th International Conference on Very Large Databases, 2000, pp. 591-594
[16] N. Rishe, et al: “SemWrap: A Semantic Wrapper over Relational Databases with Substantial Size Reduction of User’s SQL Queries,” EDBT 2000: Seventh International Conference on Extending Database Technology, pp. 13-14.
[17] N. Rishe, et al: “On Database Integration over Intelligent Networks,” Proceedings of the ISCA 2nd International Conference on Information Reuse and Integration (IRI-2000), pp. 70-75.