NBS

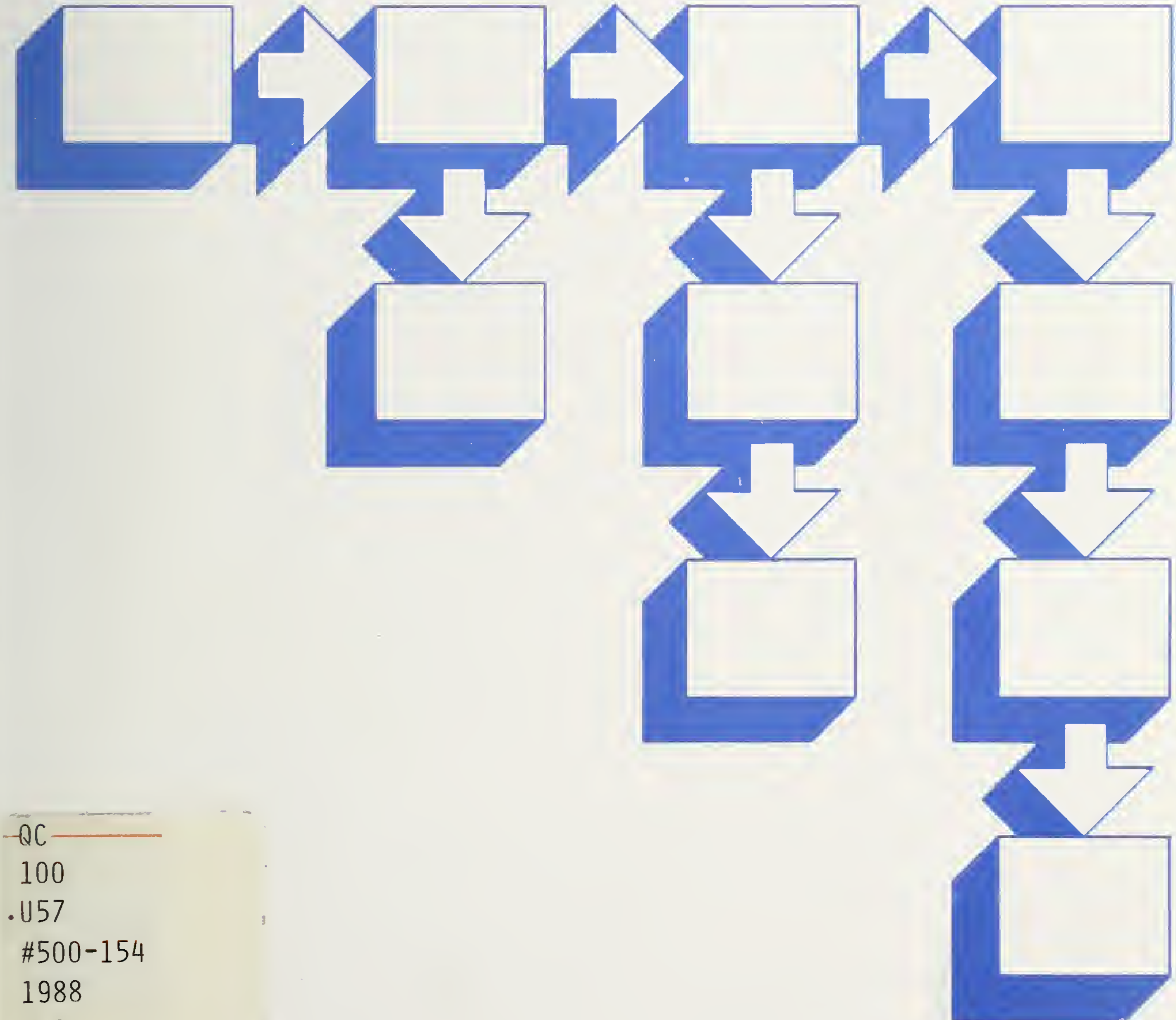
PUBLICATIONS

# Computer Science and Technology

## NBS Special Publication 500-154

# Guide to Distributed Database Management

Elizabeth N. Fong
Bruce K. Rosen



The National Bureau of Standards[1] was established by an act of Congress on March 3, 1901. The Bureau's overall goal is to strengthen and advance the nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research to assure international competitiveness and leadership of U.S. industry, science and technology. NBS work involves development and transfer of measurements, standards and related science and technology, in support of continually improving U.S. productivity, product quality and reliability, innovation and underlying science and engineering. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, the Institute for Computer Sciences and Technology, and the Institute for Materials Science and Engineering.

## *The National Measurement Laboratory*

Provides the national system of physical and chemical measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; provides advisory and research services to other Government agencies; conducts physical and chemical research; develops, produces, and distributes Standard Reference Materials; provides calibration services; and manages the National Standard Reference Data System. The Laboratory consists of the following centers:

- Basic Standards[2]
- Radiation Research
- Chemical Physics
- Analytical Chemistry

## *The National Engineering Laboratory*

Provides technology and technical services to the public and private sectors to address national needs and to solve national problems; conducts research in engineering and applied science in support of these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the ultimate user. The Laboratory consists of the following centers:

- Applied Mathematics
- Electronics and Electrical Engineering[2]
- Manufacturing Engineering
- Building Technology
- Fire Research
- Chemical Engineering[3]

## *The Institute for Computer Sciences and Technology*

Conducts research and provides scientific and technical services to aid Federal agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Institute consists of the following divisions:

- Information Systems Engineering
- Systems and Software Technology
- Computer Security
- System and Network Architecture
- Advanced Systems

## *The Institute for Materials Science and Engineering*

Conducts research and provides measurements, data, standards, reference materials, quantitative understanding and other technical information fundamental to the processing, structure, properties and performance of materials; addresses the scientific basis for new advanced materials technologies; plans research around cross-cutting scientific themes such as nondestructive evaluation and phase diagram development; oversees Bureau-wide technical programs in nuclear reactor radiation research and nondestructive evaluation; and broadly disseminates generic technical information resulting from its programs. The Institute consists of the following Divisions:

- Ceramics
- Fracture and Deformation[3]
- Polymers
- Metallurgy
- Reactor Radiation

[1]Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted; mailing address Gaithersburg, MD 20899.
[2]Some divisions within the center are located at Boulder, CO 80303.
[3]Located at Boulder, CO, with some elements at Gaithersburg, MD

# Computer Science and Technology

## NBS Special Publication 500-154

# Guide to Distributed Database Management

Elizabeth N. Fong and Bruce K. Rosen

Information Systems Engineering Division
Institute for Computer Sciences and Technology
National Bureau of Standards
Gaithersburg, Maryland 20899

April 1988

U.S. DEPARTMENT OF COMMERCE
C. William Verity, Secretary

National Bureau of Standards
Ernest Ambler, Director

## Reports on Computer Science and Technology

The National Bureau of Standards has a special responsibility within the Federal Government for computer science and technology activities. The programs of the NBS Institute for Computer Sciences and Technology are designed to provide ADP standards, guidelines, and technical advisory services to improve the effectiveness of computer utilization in the Federal sector, and to perform appropriate research and development efforts as foundation for such activities and programs. This publication series will report these NBS efforts to the Federal computer community as well as to interested specialists in the academic and private sectors. Those wishing to receive notices of publications in this series should complete and return the form at the end of this publication.

**Library of Congress Catalog Card Number: 88-600516**
**National Bureau of Standards Special Publication 500-154**
**Natl. Bur. Stand. (U.S.), Spec. Publ. 500-154, 33 pages (Apr. 1988)**
**CODEN: XNBSAV**

**U.S. GOVERNMENT PRINTING OFFICE**
**WASHINGTON: 1988**

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington DC 20402

TABLE OF CONTENTS


ABSTRACT . . . . . . . . . . . . . . . . . . . . . . . . . . 1

1. INTRODUCTION . . . . . . . . . . . . . . . . . . . . . . 2
   1.1 Purpose . . . . . . . . . . . . . . . . . . . . . . 2
   1.2 Scope . . . . . . . . . . . . . . . . . . . . . . . 3

2. DISTRIBUTED DATABASE MANAGEMENT SYSTEM . . . . . . . . . 4
   2.1 Characteristics of a DDBMS . . . . . . . . . . . . . 4
      2.1.1 Objects for Distribution . . . . . . . . . . . 4
      2.1.2 Types of Distribution . . . . . . . . . . . . . 5
      2.1.3 Distribution Transparency . . . . . . . . . . . 5
      2.1.4 Replication Transparency . . . . . . . . . . . 6
      2.1.5 Degree of Heterogeneity . . . . . . . . . . . . 6
      2.1.6 Domain of Controls . . . . . . . . . . . . . . 7

3. DISTRIBUTED ARCHITECTURAL ALTERNATIVES . . . . . . . . . 8
   3.1 Centralized Databases . . . . . . . . . . . . . . . 8
   3.2 Logically Centralized, Physically Decentralized Databases . . . . . . . . . . . . . . . . . . . . . 8
   3.3 Logically and Physically Decentralized Databases . 8
   3.4 Multi-database Systems . . . . . . . . . . . . . . 9
   3.5 Uses of the Four Architectures . . . . . . . . . . 9

4. BENEFITS AND PROBLEMS OF A DDBMS . . . . . . . . . . . . 11
   4.1 Benefits of Distributed Environment . . . . . . . . 11
   4.2 Problems of Distributed Environment . . . . . . . . 11
   4.3 Technical Problem Issues . . . . . . . . . . . . . 11
      4.3.1 Consistency of Replicated Copies . . . . . . 11
      4.3.2 Concurrency Control . . . . . . . . . . . . . 12
      4.3.3 Recovery . . . . . . . . . . . . . . . . . . . 12
      4.3.4 Performance Monitoring . . . . . . . . . . . 13
   4.4 Factors to be Considered . . . . . . . . . . . . . 13
   4.5 Applications that could Benefit from the Use of DDBMS . . . . . . . . . . . . . . . . . . . . . . . 14

5. CENTRALIZED VERSUS DECENTRALIZED REQUIREMENTS . . . . . . 15
   5.1 What Should be Centralized or Distributed? . . . . 15
   5.2 Properties Favoring Centralization . . . . . . . . 16
   5.3 Properties Favoring Decentralization . . . . . . . 16
   5.4 Properties Favoring Migration to DDBMS . . . . . . 17

6. PATHS TO DISTRIBUTED DBMS . . . . . . . . . . . . . . . 18
   6.1 Centralized Starting Point . . . . . . . . . . . . 18
   6.2 Decentralized Starting Point . . . . . . . . . . . 19
   6.3 Recommended Tools to be Used . . . . . . . . . . . 20

7. IMPACT OF DA AND DBA . . . . . . . . . . . . . . . . . . . 21
7.1 Centralized Controls . . . . . . . . . . . . . . . . 21
7.2 Distributed Controls . . . . . . . . . . . . . . . . 22

8. RECOMMENDED COURSE OF ACTION . . . . . . . . . . . . . . . 24
8.1 Planning Activities . . . . . . . . . . . . . . . . 24
8.2 DDBMS Development Phases . . . . . . . . . . . . . . 24

REFERENCES . . . . . . . . . . . . . . . . . . . . . . . . 27

# GUIDE TO DISTRIBUTED DATABASE MANAGEMENT

Elizabeth N. Fong
Bruce Rosen

Information Systems Engineering Division
Institute for Computer Sciences and Technology
National Bureau of Standards

ABSTRACT

Distributed Database Management Systems are exciting and potentially very powerful. However, distributed database management systems often have created increased complexity of database management and controls without providing the expected benefit to the organization's operations. Distributed database management systems may not be desirable for every organization. Their benefits can be realized only with careful planning, and evaluation of alternative strategies.

This guide provides an organization's decision makers the appropriate information to make good decisions in evaluating distributed database management technology for their individual environments. Also, this guide aids in planning for an orderly migration path into a distributed database environment.

Key words: Centralized environment, decentralized environment, distributed database management, heterogeneous systems, management controls, technical controls.

# GUIDE TO DISTRIBUTED DATABASE MANAGEMENT

Elizabeth N. Fong
Bruce Rosen

## 1. INTRODUCTION

The hardware and software buying sprees of the early 1980s resulted in an enormous proliferation of dissimilar hardware and software products even within individual organizations. Distributed database management is an approach to linking incompatible mixed-vendor systems and database management systems together so that there appears to a user to be a single system running a single database. The benefits of distributed database management are many, the pitfalls are great. It is essential that managers develop a effective strategy which identifies the architectural choices and resources necessary for distributed database processing.

### 1.1 Purpose

The goal of this guide is to provide answers to the following questions:

- Will the new distributed concepts make obsolete the old centralized approach?

- What should a data manager consider in determining whether a centralized configuration or a distributed configuration, or perhaps some combination of the two, will provide the best solution for the organization?

- What basic steps are involved in the analysis, design, and implementation of a distributed database environment?

- Will it be possible to integrate existing, installed equipment in the new environment (centralized or distributed), or must it all be replaced?

There is no single answer for all organizations. For some, the classic centralized approach will provide the most efficient, economical, and manageable solution. For others, distribution of one form or another offers the only workable approach to resolving the organization's database requirements.

## 1.2 Scope

Chapter 2 of this report explains the key concepts and terminology for distributed database management system.

Chapter 3 describes four possible alternatives for a distributed database management system architecture and discusses their usage.

Chapter 4 identifies the benefits and problems of a distributed database environment. Included is a list of factors to be considered by an organization migrating to a distributed database environment.

Chapter 5 describes the relationship between an organization's information processing requirements and information management resources, and the decision to centralize, decentralize or distribute.

Chapter 6 describes some of the issues involved in planning an organization's transition to a distributed database environment.

Chapter 7 reviews the data administration and database administration functions depending upon the type of distributed environment that the organization will operate.

Chapter 8 concludes the report with a list of actions for planning for the migration toward a distributed environment and a summary of distributed database management system development phases.

## 2. DISTRIBUTED DATABASE MANAGEMENT SYSTEM

A distributed database management system (DDBMS) is a collection of centralized database management systems (DBMSs) connected together via a communications network and integrated together in their operations. Thus, a DDBMS can be defined as having the following properties:

- o The system consists of multiple separate databases, and
- o The system contains automated communications which connect the separate DBMSs.

When an organization implements DDBMS, that organization is said to operate in a distributed database environment, or in short, distributed environment. The distributed environment consists of collections of data at various sites which are in direct support of the DBMS and application programs at the site and are also accessible to applications at other sites.

Each DDBMS with databases at multiple sites can be connected in many different ways. The data and other resources may, or may not be duplicated at many different sites, and the individual DBMSs making up a distributed DBMS might be maintained and controlled by a variety of policies. In an effort to classify the wide range of DDBMS architectures, a set of characteristics is used.

### 2.1 Characteristics of a DDBMS

The following section identifies the characteristics and the various options that make up the different architectures of a distributed environment:

**2.1.1 Objects for Distribution.** In establishing a distributed environment, many components could conceivably be candidates for distribution. Four main objects are identified in this article as feasible elements for distribution. They are hardware, software, data, and control.

Hardware consists of processors, storage, I/O devices, and communications facilities. All of these could be physically distributed or centralized.

The software objects are application programs, operating systems, and, in particular, the various DBMSs. All or any combination of these may be centralized, distributed, or duplicated on different nodes.

It should be noted that when one speaks about distributed DBMS, one generally refers to databases that are physically dispersed. Additionally, these databases may also be geographically dispersed. The data objects that can be distributed include application data that is stored in databases as well as meta data, i.e., data describing the data being stored.

The final object that can be distributed is control. This includes responsibility for planning and developing policies, the ownership and maintenance of data, and the day-to-day coordination and control associated with managing the distributed environment.

**2.1.2 Types of Distribution.** A system is distributed with respect to the various ways in which the objects that make up the system are centralized or distributed. Distributed hardware implies multiple equipment in different physical locations, while distributed software implies multiple copies of the same software which might reside in either the same machine or different machines. In the case of databases, each conceptual database may even be distributed by decomposing the whole database into separate physical fragments that are distributed. There can be many types of fragmentation. An example of data fragmentation as applied to relational databases are:

- o Horizontal fragmentation - The fragmentation rule follows the concept of partitioning the rows of a relational table into subsets.

- o Vertical fragmentation - The fragmentation rule follows the concept of partitioning the columns of a relational table into subsets.

- o Mixed fragmentation - The fragmentation rule may be defined based upon semantic properties of the data. For example, the partitioning of a database into subsets could be done based upon geographical properties, functional properties, or combinations of the above.

Distributed control is discussed in Section 2.1.6.

**2.1.3 Distribution Transparency.** The most important characteristic of a distributed environment is the visibility level of the location distribution to the users of the systems. The options are:

- o Visible distribution - Under this option, the distribution of the objects is highly visible to the users and/or applications. In the case of data distributed across multiple sites, the users, in accessing data must specify the physical location of where to go to fetch the data.

- Invisible distribution - Under this option, the distribution of the objects is invisible to the users and/or applications. In the case of data distribution, this implies that the users are unaware of the physical location of data and can issue single queries that access data from more than one database residing at different sites.

**2.1.4 Replication Transparency.** If the same objects exist at multiple sites, then these objects are said to be replicated. In the case of data, the options are:

- No replication - No data is replicated between separate databases. A system without data replication is called partitioned because each data item appears at one, and only one, node.

- Replication that is visible - Data is replicated between separate databases. The replication is visible to the users, and it is the users' responsibility to ensure data consistency.

- Replication that is invisible - Data is replicated between separate databases. The replication is invisible to the users, and the users may treat a replicated data item as if it were stored as a single data item in a single database. The software, as well as the database administrator, coordinates and controls this replication and ensures data consistency.

**2.1.5 Degree of Heterogeneity.** This characteristic describes the extent to which the separate sites of a DDBMS are similar. The factors which determine this characteristic of a DDBMS are the combinations of hardware, operating systems software and DBMSs being joined together. These various combinations can best be understood by way of a two dimensional matrix, as shown in Table 1. The horizontal axis of the matrix covers the areas of hardware and operating systems software while the vertical axis covers differences in DBMSs. Thus each block of the matrix then describes a different degree of DDBMS heterogeneity based upon the combination of hardware, operating systems software and DBMSs involved. For example "Block A" represents the situation of having a completely homogenous DDBMS where all of the hardware/software and DBMSs are the same for all sites within the DDBMS, while "Block I" represents the heterogeneous DDBMS situation where all the hardware, operating systems and DBMSs are different for all sites within the DDBMS. It must be remembered that even a heterogeneous DDBMS can be made totally invisible to users by providing a unified data manipulation language, or it can be left visible to users by having the user specify the location of data and to access the data via its local data manipulation language.

Table 1. Degree of Heterogeneity

| | same hardware/ software | compatible hardware/ software | incompatible hardware/ software |
|---|---|---|---|
| same DBMSs | Block A | Block B | Block C |
| different DBMS, same data model | Block D | Block E | Block F |
| different DBMS, different data model | Block G | Block H | Block I |

**2.1.6 Domain of Controls.** The other key characteristic of a distributed environment is the coordination and control of the DDBMS. This coordination involves establishing and maintaining the policies for accessing and updating the data, for establishing data availability, setting up responsibility for data integrity, etc. The options are:

- o Central control - There is a central function where all the decisions for managing the DDBMS are made. Typically this central function is accomplished by system-wide data administrators with one or more database administrators.

- o Local autonomy - Each local site manages its own hardware, software and databases. There does not exist a system-wide control. Each local site can exercise their own policy. However, changes and necessary information that other sites need to know are broadcast to every site that participates in the DDBMS.

- o Hybrid control - There are many variations of control that can be set up between the two extremes listed above. A typical domain of control might be that of a hierarchical nature in which there is one master site that coordinates all decisions, but does not have the authority to make unilateral decisions.

## 3. DISTRIBUTED ARCHITECTURAL ALTERNATIVES

Distributed database architecture is the way in which data is integrated with respect to an application. The distributed database architectural alternatives as described in [LARS87] are based upon two dimensions in which databases are structured as collections of information: logical/physical dimension, and centralized/decentralized dimension. Thus, four possible classes of distributed database architectures can be identified:

- logically and physically centralized databases,
- logically centralized and physically decentralized databases,
- logically and physically decentralized databases,
- multi-databases.

These are briefly described below.

### 3.1 Centralized Databases

This architectural alternative is the traditional centralized DBMS environment. Data is retrieved and updated only from the main databases although access requests may be coming from remote sites. There is only one centralized copy of any database. Other redundant copies of databases serve only as shadow copies for recovery purposes. Most commercial DBMSs for mainframes support some variation of this architecture.

### 3.2 Logically Centralized, Physically Decentralized Databases

This architecture includes many distributed DBMSs described in the literature [CERI84]. Data is physically and geographically separate, but is logically integrated via a global schema approach. In such an architecture, users and applications access data described through a single global schema, and the data is accessed via a computer network interconnected across several computer systems. Most often, this architecture provides a unified system, and therefore offers centralized control over the physically decentralized databases.

### 3.3 Logically and Physically Decentralized Databases

This architecture is frequently referred to as "federated" databases [HEIM85] or loosely-coupled DBMS in which each local site may be viewed as an autonomous entity. The local administrators retain control over who can access data in that database and the manner in which the data can be accessed. This is typically achieved by defining three types of schemas: private schema which describes that portion of a database that is local and private to the site; export schema which describes that portion of the database which is shared within the federation; and import schema which specifies that portion of a database that the site desires to access at other sites.

This architecture often emerged in companies in which different database needs resulted in many databases established within different organization groups. Because of organizational and/or political reasons, it is frequently not possible to integrate these databases into a single organization-wide database controlled by one centralized administrator. If the organizational and/or political problems can be solved, then this architecture can constitute the first step in the migration path from several separate databases to the logically centralized physically decentralized database architecture described above.

### 3.4 Multi-database Systems

In this architectural alternative, there are no communication links between DBMSs. Thus, this alternative is not a DDBMS according to our definition. It is described here for completeness. The users or programmers extract portions of individual databases by selection conditions, and then specify how these pieces of data are to be combined to produce the desired report. There is no movement of data from one site to another and there is no update of the databases from remote sites.

Multi-database systems are useful in situations where databases exist commercially such as those for stock exchange information, news, etc. and a user wishes only to retrieve data to perform further analysis, but does not wish to update those databases.

### 3.5 Uses of the Four Architectures

Each of the alternatives described supports a different organizational environment. The needs of the organization's users must be evaluated in order to decide which of the architectural alternatives is best for the organization.

The centralized database approach provides users with a single view of the database along with centralized control policy over the sharing and administration of the database. Further, it is technically easier to support the integrity of the data under a centralized database.

The logically centralized, physically decentralized database approach provides users with all the advantages of a distributed DBMS environment, but requires more complex procedures for enforcing data integrity and controlling the sharing of the databases. One way to limit this complexity is to issue a policy that allows no on-line updating of databases. If users do not have the need for instantly updated data, then this variation of the logically centralized, physically decentralized database approach can be utilized to alleviate update control problems and gain increased performance of the distributed system.

The federated databases approach provides site autonomy while still providing for a controlled sharing of the databases. The administration of this option, however, needs to be carefully specified. Under this alternative, there will be at least two layers of DBAs or DAs (organization-wide and local administrators). Changes to existing schemas, or creation of new schemas at a local site must be properly broadcast and administered if the affected databases are shared within the federation.

In the above three options, the user perceives a single database system. That means the DDBMS provides location transparency, replication transparency, failure transparency, concurrency transparency, and heterogeneity transparency.

Under the multi-database approach there is no attempt made to provide any transparency to the user since there are no links between these DBMSs. The users or programmers must be familiar with all of the DBMS's languages. However, there is also no fear of remote users doing an update that would violate the integrity constraints of the databases.

## 4. BENEFITS AND PROBLEMS OF A DDBMS

The scope of information resource management as practiced by an organization can be centralized, decentralized, or distributed. The requirement for an organization to move toward a distributed environment depends on the specific organization's application and mission. There are many reasons for or against a migration toward a distributed environment. The factors encouraging and inhibiting distribution are discussed below.

### 4.1 Benefits of Distributed Environment

There are many benefits associated with a distributed database environment. One of the most attractive benefits is that it can allow existing applications to evolve into the distributed database environment without undergoing major conversions from one DBMS to another. Another benefit is the ability to increase data sharing within and between geographical sites. A distributed environment also places computing resources closer to end-users thus encouraging end-users to do their own programming tasks.

### 4.2 Problems of Distributed Environment

The one big pitfall of a distributed database environment is that it will have all of the problems associated with the centralized database environment, but at an even greater level. From the management control point of view, a distributed environment may mean loss of overall control of the organization's information assets. Very often, a distributed environment means that the total cost of an organization's systems will be increased. From the technical point of view, there are many problem issues in the application of DDBMS technology that have yet to be solved. Finding solutions to these problems will be the challenge for researchers in database technology over the next few years. Some of these problem issues are described below.

### 4.3 Technical Problem Issues

Included with the problems identified here are some suggestions for possible solutions. Currently most applications do not seek generalized solutions to these problems, but instead they seek case by case methods of avoiding them. At present, it is likely that general solutions to these problems, if they even exist, could not be implemented economically.

**4.3.1 Consistency of Replicated Copies.** When there are multiple copies of the same data, an obligation is placed on the DBMS to keep them in step with each other. Thus there exists the problem of maintaining data consistency between multiple databases. The solution to this problem is to commit all update transactions only after all available copies of the data have been updated. However, the cost of this can be prohibitive for many applications. Another solution to this problem is to keep all updates in a temporary file and then apply the updates from time to time, or on some regular schedule. Use of this solution requires that user applications be capable of tolerating data discrepancies for periods of time. The major trade-off between these solutions revolves around the requirement of data timeliness versus paying for the large overhead needed to provide data consistency by way of synchronous updates.

4.3.2 **Concurrency Control.** A distributed DBMS is required to process many transactions at the same time. Usually these transactions are performing unrelated tasks and they can proceed independently of one another. However there is always the possibility that they may interact through use of the same data resources.

The problems of concurrency are generally known as the double update problem and deadlock problem. The double update problem exists when two programs each are reading and then writing the same data resource. The result then depends on the sequence in which the reads and writes occurred, thus the last write will be accepted and the other change will be lost. This can result in data being changed based not upon the most recent data, but upon data that was obsolete. To prevent this loss, it is possible to lock the data resource so that the second program has to wait until the lock is released. However locking can lead to the deadlock problem which means that two processes are waiting for each other to release the resource and thus neither process can proceed.

There are many possible solutions to the above two problems for centralized database systems. The solutions need to guarantee deadlock detection and transaction serializability. One solution for deadlock detection is to send all locks to one site for any process that is held up. Another is to use a clock to time out transactions, but setting an acceptable time interval is rather difficult. Transaction serializability is the requirement that whenever a series of transactions overlap one another in time, their effect on the database and their environment must be the same as it would have been if they had been executed one after the other in the sequence in which they were initiated. The implementation requirements to achieve deadlock detection and transaction serializability for distributed databases are both difficult and costly.

4.3.3 **Recovery.** A distributed DBMS does not have the failure characteristic of a single system: a single system is either functioning or it is down. A distributed DBMS can experience failure in some nodes independent of others, and it can also fail in the interconnect capabilities even though all nodes continue to function. Replication of data is sometimes used as a solution which permits provision of a service in spite of failures. However, when the failed nodes are restored to the system it is necessary to bring their data into step with the data from the alternate site. This raises the problem of concurrency control since it is possible that new update attempts could be made against a database that is in the middle of a recovery cycle. Current solutions for recovery in a distributed system are very costly and are highly dependent upon organizational requirements.

**4.3.4 Performance Monitoring.** Monitoring and evaluating performance in DDBMS is more complex than a centralized DBMS because data may be replicated, fragmented, or both. Decisions about fragmentation, replication, and migration of data must include a consideration of operational accesses to data. The gathering of the statistics needed to facilitate tuning the DDBMS in order to improve system performance still remains a difficult problem.

### 4.4 Factors to be Considered

The benefits and problems of operating in a distributed database environment must be evaluated against the following factors:

- o Economic - DDBMS processing is sometimes encouraged by the relative costs of storage versus communications. The trade-off is the cost of storage and manpower to augment the existing centralized system versus paying for the communication cost to use a remote site's resources. Based upon the organization's current configuration, economic factors may vary depending upon the targeted distributed environment.

- o End-user Computing - Many end-users become dissatisfied with the service they receive from central data processing organizations. With the availability of user-friendly software, end-users are becoming more knowledgeable about data processing and are more willing to create their own applications without professional programmers. Having a distributed environment definitely promotes end-user computing facilities.

- o Organizational and Geographical Factors - Sometimes it is natural for an organization to migrate into a distributed environment based upon its location and/or functions.

- o Incremental Growth - Distributed systems allow more flexible growth and expansion. Adding another node to a network is much easier than upgrading to a different, more powerful computer family.

- Availability and Reliability - For certain applications where failure of the computer is critical, loss of computing power and unavailability of data could cause a major catastrophe. In this case, redundancy of data as well as hardware is essential. In this circumstance it is definitely wise to consider establishing a distributed environment because it can provide increased reliability.

- Security Requirement - Operating in a distributed environment tends to create a new set of security problems compared to a single site system. In a distributed environment it may be the responsibility of each local site to protect their own data at their site, yet this data, or some portion of it, must also be made available to remote users. However, allowing for remote access also makes the system easier to penetrate for unauthorized individuals. Further it becomes necessary to develop some type of overall system security policy and to ensure that this policy is followed by all elements of the organization. Another aspect of security that must be considered in a distributed environment is the possibility of communications intercept by outside organizations or individuals. The solution to this problem is the use of encryption, which of course increases the cost of establishing and maintaining the distributed system.

- Interconnection of Existing Databases - When an organization needs to integrate several already existing individual databases, it is often economically impractical for the organization to consider going back to square one and develop a distributed DBMS from scratch. The more practical solution, sometimes referred to as the "bottom-up" approach, is to create an application program development environment that operates on the existing databases.

### 4.5 Applications that could Benefit from the Use of DDBMS

The type of applications that lend themselves to a distributed environment are those that usually occur in organizations which have many regional offices. Data is collected and maintained at each regional office, but this same data must be shared organization-wide. Examples of this are multi-plant manufacturing, military command and control, electronic funds transfer in banking networks, airline or hotel reservation systems, etc.

## 5. CENTRALIZED VERSUS DECENTRALIZED REQUIREMENTS

There are many factors involved in the decision whether to centralize, decentralize or distribute. Decentralization differs from distribution and distributed data because there are no links between them [BRAY86]. Some applications are best run on a centralized machine and some data are best stored centrally, while some applications are best run in a decentralized manner at an end-user location and may be best designed at that location. The planning of what portions of an organization's data assets should be centralized, decentralized, or distributed is critical to the overall design of a distributed environment.

The top corporate planners should be asking how distribution of databases, data communications, and distributed systems would affect the way the organization operates. The focus needs to be on the practical aspect of how the organization uses and manages data as an information resource, how the organizational units communicate with each other and with external organizations, and whether or not in-house personnel have the level of sophistication and understanding needed to operate and maintain a distributed environment.

### 5.1 What Should be Centralized or Distributed?

In determining the answers to this question it is necessary to look at the system from both the technical and the management points of view.

From the technical point of view the designers need to ask what system configuration would result in the most effective application of the hardware/software resources to be utilized. Based on this technical assessment, the question can then be answered from the technical side as to what data would be stored centrally and what data would be distributed.

The technical point of view does not, however, take into account the management level questions that must be answered before making any final central versus distributed decision. Management questions could include:

- What level of responsibility should be placed with local managers?
- Which is more important, easy access to data for customer service, or for central decision making?
- What resources are available for implementing a distributed system?
- What impact does management wish the system to have on how the organization is structured and does business?

### 5.2 Properties Favoring Centralization

The following identifies some of the reasons that favor centralization:

- o The organization's applications are already implemented centrally, and the central group is reliable and responsive.
- o There is a need for strong centralized corporate-wide strategic planning and control.
- o The data is frequently used by centralized applications such as corporate-wide payroll.
- o The data is frequently updated and users in all areas need constant access to the same data and need the current up-to-the-minute version.
- o Many queries will require searching major portions of the organization's data as a whole. Searching data which is geographically scattered is extremely time consuming. The software and hardware for efficient data search requires that the data be in one location.

### 5.3 Properties Favoring Decentralization

The following identifies some of the reasons that favor decentralization:

- o Data usage is generated from different locations, and fast response time is important.
- o Data is generated and used at individual sites, and information sharing between sites is rare or closely controlled.
- o The organization operates under the policy that accuracy, privacy and security of data and applications is a local responsibility.
- o Applications are simple and are only used by one or a few users.
- o The update rate is too high for a single centralized DBMS.
- o The end-users at each local site manipulate and maintain their own data operations, which results in a sense of "data ownership." Excessive centralization may

then cause conflict and may result in a loss of responsibility for maintaining accurate data.

## 5.4 Properties Favoring Migration to DDBMS

Properties that favor a migration to a DDBMS environment can exist in organizations that are operating in either a centralized mode or a decentralized mode. Further, a change to a DDBMS environment, if properly structured, can allow an organization to continue to enjoy the benefits of their previous centralized or decentralized mode of operation while gaining the additional benefits made available under a DDBMS environment. Thus the organization can enjoy the best of both worlds. The following are some of the properties that favor migration to a DDBMS environment:

- The organization requires the capability of enforcing centralized standards while still offering a high degree of autonomy to its local sites.
- The characteristics of the organization require a mix of both centralized computing and local applications and databases.
- The structure of the organization is such that centralized computing is resulting in tremendous expenditures for larger centralized computing power, but the central computer is becoming a choke point for information flow within the organization.
- The manpower that already exists in the organization is sufficiently skilled so as to plan a migration into a DDBMS environment as a natural part of normal system upgrades.

## 6. PATHS TO DISTRIBUTED DBMS

If an organization decides to move into the distributed database environment, then the migration from the current mode of data processing to the distributed mode needs to be carefully planned. Organizations can move into a distributed database environment from either a totally centralized or decentralized starting point. This section describes some of the issues involved in planning an organization's transition to a distributed environment.

### 6.1 Centralized Starting Point

The approach that should be utilized to move from an existing centralized application to a distributed environment is through expansion and evolution. This means that the distributed database system is implemented by expanding existing centralized computerized applications. Areas of concern which could potentially affect the migration are:

- o What would be the number of remote sites to be added? The distributed architecture that best fits the organization's mission must be determined. From this an estimate of the number of remote sites can be established.

- o What would be the required hardware, communication links, software (including DBMS), databases and application programs which would have to be installed at each remote site?

- o What application-related functions currently performed by a central host computer can be profitably moved into distributed components?

- o What DDBMS architecture is appropriate? Is it necessary to maintain centralized control?

- o What methods would be used for distributing the centralized database into various locations? If the centralized database needs to be partitioned, a careful analysis of the method of distribution must be conducted. The analysis needs to take into consideration the problems of data security, data integrity, on-line update, ownership and responsibility of data.

- o How will the organization train and involve users and in-house data processing staff in the migration planning for a distributed environment? This also must include establishing policies and controls in preparation for the shift. The establishment of the DA and DBA functions will be addressed in the next section.

An interim approach to migrating from a centralized starting point is to leave the database centralized and distribute only data access functions. Thus the needed remote nodes may be established but they would not maintain any databases. Instead they would access the centralized database. To assist in establishing this type of architecture, a Remote Data Access (RDA) protocol is currently in draft status and is expected to be available as an International Standard [ISO87].

The RDA standard defines the interworking between two program components in different end systems, where one controls a database and the other is required to read and update the data. One use of the RDA would be to support access from a user at a workstation to a database which is physically remote. The first phase of the RDA standard specifies a relational data manipulation language, the Structured Query Language (SQL). The SQL database language has been adopted by the Federal Government as the Federal Information Processing Standards [FIPS87], and it is also an American National Standard [ANSI86].

The remote data access to a centralized database approach offers a simple first option in the migration process. The hardware, communication links, and application programs may be designed and installed at remote sites as an initial step before worrying about the distribution of the database. As distributed functions are identified and as users become skilled, then a fully distributed database environment can be planned for and implemented. Depending upon the nature of the organization, this slow evolution approach may prove to be the most advantageous.

## 6.2 Decentralized Starting Point

A decentralized environment involves multiple computer sites with virtually no automated communications between them. Depending on how the organization functions, these divisions may be performing different or similar functions. If the organization has decided to move into a distributed environment, then one way is to consider each site as essentially an occurrence of the centralized model. Another interim solution for moving from a decentralized database environment to a distributed database environment is the use of "Intelligent Gateways."

"Intelligent Gateways (IG)" is a hardware/software configuration that enables a user at a single terminal to access and retrieve data from dissimilar systems. Different IGs may offer various services. The basic IG provides the user with transparent log-on to the various target hosts. However, once logged on in this manner, the user must know the data manipulation language of each remote DBMS system he wishes to access. Another type of IG is one which supports access in a heterogeneous environment by providing a uniform, integrated interface for retrieving data from heterogeneous databases, but which does not require changes to the pre-existing databases and their DBMSs.

The IG architecture offers an organization an interim solution for establishing a distributed environment without having to start from ground zero. The drawbacks of IG are, that it adds a layer of software thus increasing the response time and creating, in some cases, serious performance problems. Some existing IG projects handle only queries and do not allow on-line updates. Also, some differences between systems cannot be completely represented in a uniform way thus the objective of having a uniform, integrated interface cannot be totally achieved.

### 6.3 Recommended Tools to be Used

One of the key software tools to be used during the migration planning phase is an Information Resource Dictionary System (IRDS) [GOLD88]. An IRDS is a computer software package which provides facilities for recording, storing and processing descriptions of an organization's data and data processing resources. The use of an IRDS will reveal the organization's total data structure and capture information as to where, when, and how the data is being used. An IRDS can also be used as a tool for planning and designing various alternatives of data distribution.

The IRDS is expected to become an American National Standard and efforts are underway for the IRDS to become a Federal Information Processing Standard.

## 7. IMPACT OF DA AND DBA

As the organization migrates toward a distributed database environment, the data administration (DA) and database administration (DBA) functions must change. Depending on the nature of the distributed architecture, the controls can be centralized or distributed.

### 7.1 Centralized Controls

In centralized control, all the functions of DA and DBA will be performed at a primary node. The roles of the DA and DBA vary in emphasis and importance from organization to organization. There is, nevertheless, some general consensus that the DA functions include:

- o The responsibility for administration of organization-wide policies. This includes the establishment of rules for the cooperative processing within the distributed database environment.
- o The management of inter-system standards.
- o The issuance of approval for participation within the distributed database environment.

There is also some general consensus that the DBA functions include:

- o The responsibility for the technical administration of the organization-wide database environment. This includes the maintenance of global meta data, local meta data, and all the local DBMS software.
- o Technical coordination of distributed database design. This includes the administration of inter-system meta data migration. Data that is replicated must be kept consistent.
- o The control of the sharing of databases by means of a global data dictionary and directory and one or more local data dictionaries.
- o The control of security and privacy by establishing access and updating rights.
- o Providing support to application development for every node.
- o Establishing the procedures for global and local recovery and back-up.

## 7.2 Distributed Controls

Distributed controls typically consist of a hierarchical control structure based on a global DA/DBA who has the responsibility for the overall organizational database, and on local DAs/DBAs who have the responsibility for their respective local databases which, when combined, form the overall organizational database. The degree of site autonomy may range between having complete site autonomy, without any centralized DA/DBA controls, to having almost completely centralized control.

An important aspect of the global DA function is intersite coordination. Depending upon the degree of local site autonomy, the intersite coordination functions typically include the following:

- o Setting system-wide policy on the use and operation of the DDBMS.
- o Issuing approval for new site participation within the federated database environment.
- o Resolving any disputes or conflicts among the local sites.
- o Establishing user groups by providing a single contact point for the federated environment.

For a large organization, a communication network administration function may be established to support the communication aspect of the DDBMS. Some of the tasks for this function would include:

- o Configure and install the communication network linking the local DBMSs.
- o Analyze and administer routing within the network.
- o Responsibility for the communications aspects of database synchronization and concurrency controls.

The global DBA functions include those listed under centralized controls, plus these additional tasks:

- o Manage and maintain the global data dictionary and directory.
- o Monitor the overall DDBMS and its communication networks for availability, efficiency, integrity, security, and recovery.

- Supervise local DBAs to ensure global operation.

Each local site has a set of local DAs/DBAs who are responsible for managing the operations of the local site. Their tasks include all those identified for centralized control but with the responsibility solely centered on the local site. Additionally, they have the task of communicating and coordinating with the global DA/DBA.

## 8. RECOMMENDED COURSE OF ACTION

Prior to any attempt to move into a distributed database environment an organization should first perform an objective review of the organization's database requirements. If, after this review, the decision is made to move into a distributed database environment, then the recommended course of action consists of first conducting an organization-wide planning activity followed by step-by-step distributed DBMS development phases.

### 8.1 Planning Activities

The organization's planning activities should include the following tasks:

- o Identify existing, installed facilities that are to be included in the new configuration.
- o Identify the additional functions necessary for accommodating additional applications.
- o Derive and evaluate several possible configurations. These configurations should include varying degrees of centralization and/or distribution.
- o Identify the user population for which the distributed DBMS is intended.
- o Consider communications needs, security needs, data integrity and timeliness needs.
- o Plan for possible future growth of organization and/or user population.
- o Consider various policies for data sharing and control procedures.
- o Integrate current DBMSs and databases with the new and expanded distributed DBMSs and databases. The use of an IRDS software tool is highly recommended here.

### 8.2 DDBMS Development Phases

The step-by-step development of a distributed DBMS environment, as summarized in Table 2, includes the following major phases:

- o Planning
- o Designing

- Installing
- Supporting

The planning phase needs first to consider the starting point of the organization. After assessing the starting point, the key objective of the technical activity of this phase is to determine the distributed architecture. In conjunction with the technical activity, the managerial activity in this phase is to determine and commit the amount of resources (people and equipment) necessary for the development of a DDBMS. The management and technical activities must work in concert since the amount of resources management can make available will have a definite impact on the distributed architecture that is finally selected.

Once the needed resources have been committed, and the distributed architecture determined, the organization then moves into the designing phase. The technical aspect of this phase involves (1) design of interface software with the emphasis on reliable global cooperation between the elements involved in distributed processing and also between the distributed processing elements and application specific software, (2) design of application specific software which specifies the global application function and each local application function of the software, and (3) determination of the distribution method of the databases by designing the global data dictionary along with each local data dictionary and the interface between these dictionaries. The managerial activity in this phase is the establishment of the DA and the DBA functions along with preparing the organizational policy statements that define these positions.

The installing phase consists of the actual implementation and testing of the DDBMS. The technical activities of this phase involve installing hardware, software, and databases for each node. The managerial activity involves finalizing and implementing all the needed controls for database access. Among some of the controls are those for ensuring security and privacy of the data, ensuring consistency of the data, ensuring availability of the data, etc.

The supporting phase consists of operational level activities. From the technical aspect, the activities should concentrate on tuning of the system for better performance. Once in the support phase, managerial activities are oriented towards providing needed resources for training of users and establishing mechanisms for helping users in maintaining an effective distributed database operation.

Table 2. DDBMS Development Phases

| Phase | Technical Activities | Administrative Activities |
|---|---|---|
| Planning | o Determine Distributed architecture | o Management committed to move to DDBMS |
| Designing | o Software design<br>o Application design<br>o Distributed data des. | o Establish DA and DBA functions |
| Installing | o Implement and test for each node | o Set up rules/controls globally and locally |
| Supporting | o Tune for better performance<br>o Provide backup and recovery | o User training<br>o User support and help |

Due to the complex nature of establishing a distributed database architecture, and the tremendous impact it can have on an organization, it is impossible to predict in this guide if a given organization will be successful in such a migration. However, as with any complex task, careful planning will make the transition less painful, and in the long range will prove to be the only way for an organization to utilize the latest advances in technology to achieve efficient data processing.

## REFERENCES


[ANSI86] American National Standard for Information Systems - Database Language SQL, ANSI X3.135-1986, ANSI, Inc. 1430 Broadway, New York, NY 10018.

[BRAY86] Bray, O., "IRM in a Decentralized/Distributed Environment," in Chapter 6 of Fong, E. and Goldfine, A. (Editors), Data Base Directions: Information Resource Management - Making it Work, NBS Special Publication 500-139, National Bureau of Standards, June 1986.

[CERI84] Ceri, S. and Pelagatti, G., Distributed Databases Principles and Systems, McGraw-Hill Book Company, New York, 1984.

[FIPS87] Federal Information Processing Standards Publication, Database Language SQL, FIPS PUB 127, National Bureau of Standards. March 1987.

[GOLD88] Goldfine, A. and Konig P., A Technical Overview of the Information Resource Dictionary System (Second Edition), NBSIR 88-3700, National Bureau of Standards, January 1988.

[HEIM85] Heimbigner, D. and McLeod, D., "A Federated Architecture for Information Management," ACM Transactions on Office Information Systems, Vol. 3, No. 3, July 1985, Pages 253-278.

[ISO 87] ISO/TC97/SC21 N1926, Remote Database Access Project No. 97.21.31, Third Working Draft submitted for DP Registration, July 1987.

[LARS87] Larson, James A., "Alternative Architectures for Distributed DBMSs," Auerbach Information Management Series 24-01-15, Auerbach Publishers Inc. 1987.

NBS-114A (REV. 2-80)

| U.S. DEPT. OF COMM. **BIBLIOGRAPHIC DATA SHEET** *(See instructions)* | 1. PUBLICATION OR REPORT NO. NBS/SP-500/154 | 2. Performing Organ. Report No. | 3. Publication Date April 1988 |
|---|---|---|---|

**4. TITLE AND SUBTITLE**

Computer Science and Technology:
Guide to Distributed Database Management

**5. AUTHOR(S)**

Elizabeth N. Fong and Bruce Rosen

**6. PERFORMING ORGANIZATION** *(If joint or other than NBS, see instructions)*

**NATIONAL BUREAU OF STANDARDS**
**U.S. DEPARTMENT OF COMMERCE**
**GAITHERSBURG, MD 20899**

**7. Contract/Grant No.**

**8. Type of Report & Period Covered**

Final

**9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS** *(Street, City, State, ZIP)*

**10. SUPPLEMENTARY NOTES**

Library of Congress Catalog Card Number 88-600516

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

**11. ABSTRACT** *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

Distributed Database Management Systems are exciting and potentially very powerful. However, distributed database management systems often have created increased complexity of database management and controls without providing the expected benefit to the organization's operations. Distributed database management systems may not be desirable for every organization. Their benefits can be realized only with careful planning, and evaluation of alternative strategies.

This guide provides an organization's decision makers the appropriate information to make good decisions in evaluating distributed database management technology for their individual environments. Also, this guide aids in planning for an orderly migration path into a distributed database environment.

**12. KEY WORDS** *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

Centralized environment, decentralized environment, distributed database management, heterogeneous systems, management controls, technical controls.

**13. AVAILABILITY**

☒ Unlimited

☐ For Official Distribution. Do Not Release to NTIS

☒ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

☐ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

**14. NO. OF PRINTED PAGES**

33

**15. Price**

USCOMM-DC 6043-P80

☆U.S. Government Printing Office : 1988 - 201-597/82547

# NBS Technical Publications

## Periodical

**Journal of Research**—The Journal of Research of the National Bureau of Standards reports NBS research and development in those disciplines of the physical and engineering sciences in which the Bureau is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Bureau's technical and scientific programs. Issued six times a year.

## Nonperiodicals

**Monographs**—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

**Handbooks**—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

**Special Publications**—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

**Applied Mathematics Series**—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

**National Standard Reference Data Series**—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NBS under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW, Washington, DC 20056.

**Building Science Series**—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

**Technical Notes**—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

**Voluntary Product Standards**—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

**Consumer Information Series**—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

*Order the* **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

*Order the* **following** *NBS publications—FIPS and NBSIR's—from the National Technical Information Service, Springfield, VA 22161.*

**Federal Information Processing Standards Publications (FIPS PUB)**—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

**NBS Interagency Reports (NBSIR)**—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

**U.S. Department of Commerce**
National Bureau of Standards
Gaithersburg, MD 20899