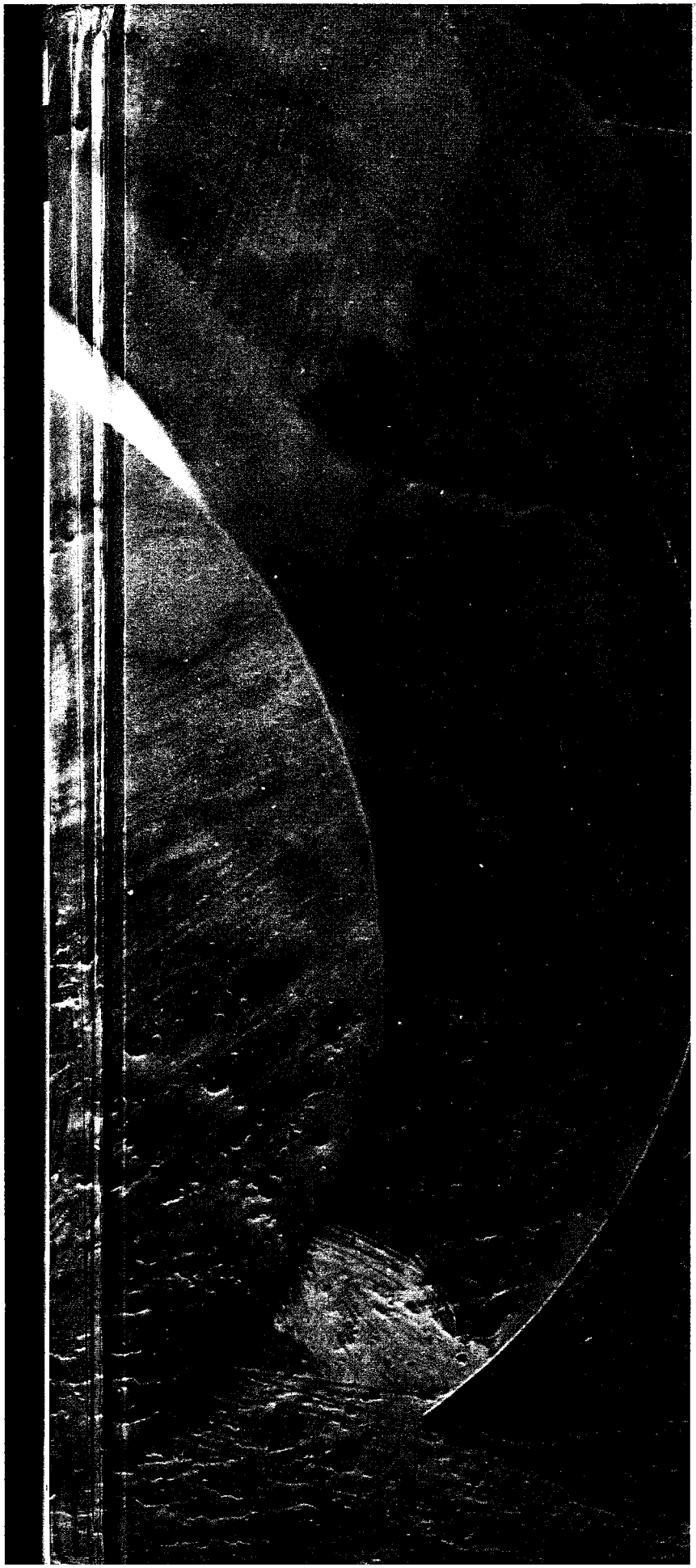

Modern Database Management

Fifth Edition

Fred R. McFadden

Jeffrey A. Hoffer

Mary B. Prescott

■ In memory of my valued colleague Daniel Couger. — *F. R. M.*

■ To Patty, for her sacrifices, encouragement, and support.
To my students, for being receptive and critical, and challenging me to be a better teacher. — *J. A. H.*

■ To Larry, Mike, and Ivan. Their love and support provide a foundation to my life which makes efforts such as writing this book possible. And to Jeff and Fred, who gave me the opportunity to write with them and patiently provided invaluable guidance along the way. — *M. B. P.*

Executive Editor: Michael Roche
Developmental Editor: Maureen Allaire Spada
Assistant Editor: Ruth Berry
Editorial Assistant: Adam Hamel
Senior Marketing Manager: Tom Ziolkowski
Senior Marketing Coordinator: Deanna Storey
Senior Production Supervisor: Patty Mahtani
Manufacturing Buyer: Sheila Spinney
Design Director: Regina Hagen
Cover Designer: Linda Manly Wade
Text Designer: Sandra Rigney
Composition and Project Coordination: Elm Street Publishing Services, Inc.
Cover Image: © **Mick Tarel for Artville**

Library of Congress Cataloging-in-Publication Data

McFadden, Fred R., 1933–
Modern database management / Fred R. McFadden, Jeffrey A. Hoffer, Mary B. Prescott. — 5th ed.
p. cm.
Includes bibliographical references and index.
ISBN 0-8053-6054-9
1. Database management. I. Hoffer, Jeffrey A. II. Prescott, Mary B. III. Title.
QA76.9.D3M395 1999
005.74—dc21 98-4120
CIP

Reprinted with corrections, May 1999

5 6 7 8 9 10 WCT 02 01 00

CHAPTER

11

Distributed Databases

LEARNING OBJECTIVES

After studying this chapter, you should be able to:

- Define the following key terms: **distributed database, decentralized database, location transparency, synchronous distributed database, asynchronous distributed database, replication transparency, failure transparency, commit protocol, two-phase commit, concurrency transparency,** and **semijoin.**
- Explain the business conditions that are drivers for the use of distributed databases in organizations.
- Describe the salient characteristics of the variety of distributed database environments.
- Explain the potential advantages and risks associated with distributed databases.
- Explain four strategies for the design of distributed databases, options within each strategy, and the factors to consider in selection among these strategies.
- State the relative advantages of synchronous and asynchronous data replication and partitioning as three major approaches for distributed database design.
- Outline the steps involved in processing a query in a distributed database and several approaches used to optimize distributed query processing.
- Explain the salient features of several distributed database management systems.

INTRODUCTION

When an organization is geographically dispersed, it may choose to store its databases on a central computer or to distribute them to local computers (or a combination of both). A **distributed database** is a single logical database that is spread

Distributed database: A single logical database that is spread physically across computers in multiple locations that are connected by a data communications link.

physically across computers in multiple locations that are connected by a data communications network. We emphasize that a distributed database is truly a database, not a loose collection of files. The distributed database is still centrally administered as a corporate resource while providing local flexibility and customization. The network must allow the users to share the data; thus a user (or program) at location A must be able to access (and perhaps update) data at location B. The sites of a distributed system may be spread over a large area (such as the United States or the world), or over a small area (such as a building or campus). The computers may range from microcomputers to large-scale computers or even supercomputers.

A distributed database requires multiple database management systems, running at each remote site. The degree to which these different DBMSs cooperate, or work in partnership, and whether there is a master site that coordinates requests involving data from multiple sites distinguish different types of distributed database environments.

Decentralized database: A database that is stored on computers at multiple locations; however, the computers are not interconnected by a network, so that users at the various sites cannot share data.

It is important to distinguish between distributed and decentralized databases. A **decentralized database** is also stored on computers at multiple locations; however, the computers are not interconnected by a network, so that users at the various sites cannot share data. Thus a decentralized database is best regarded as a collection of independent databases, rather than having the geographical distribution of a single database.

Various business conditions encourage the use of distributed databases:

- *Distribution and autonomy of business units* Divisions, departments, and facilities in modern organizations are often geographically (and possibly internationally) distributed. Often each unit has the authority to create its own information systems, and often these units want local data over which they can have controls.
- *Data sharing* Even moderately complex business decisions require sharing data across business units, so it must be convenient to consolidate data across local databases on demand.
- *Data communications costs and reliability* The cost to ship large quantities of data across a communications network or to handle a large volume of transactions from remote sources can be high. It is often more economical to locate data and applications close to where they are needed. Also, dependence on data communications can be risky, so keeping local copies or fragments of data can be a reliable way to support the need for rapid access to data across the organization.

The ability to create a distributed database has existed for over a decade. As you might expect, a variety of distributed database options exist (Bell and Grimson, 1992). Figure 11-1 outlines the range of distributed database environments. These environments are briefly explained by the following:

I. *Homogeneous* The same DBMS is used at each node.

   A. *Autonomous* Each DBMS works independently, passing messages back and forth to share data updates.

   B. *Non-autonomous* A central, or master, DBMS coordinates database access and update across the nodes.

II. *Heterogeneous* Potentially different DBMSs are used at each node.

   A. *Systems* Supports some or all of the functionality of one logical database.

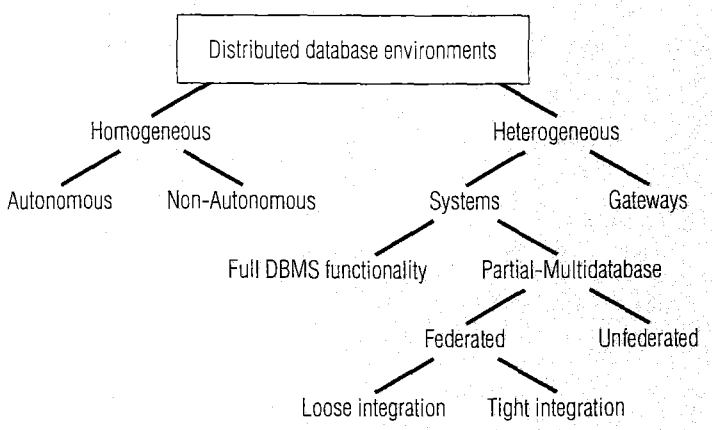

Figure 11-1
Distributed database environments (adapted from Bell and Grimson, 1992)

1. *Full DBMS Functionality* Supports all of the functionality of a distributed database, as discussed in the remainder of this chapter.
2. *Partial-Multidatabase* Supports some features of a distributed database, as discussed in the remainder of this chapter.
   a. *Federated* Supports local databases for unique data requests.
      i. *Loose Integration* Many schemas exist, for each local database, and each local DBMS must communicate with all local schemas.
      ii. *Tight Integration* One global schema exists that defines all the data across all local databases.
   b. *Unfederated* Requires all access to go through a central coordinating module.

B. *Gateways* Simple paths are created to other databases, without the benefits of one logical database.

A homogeneous distributed database environment is depicted in Figure 11-2. This environment is typically defined by the following characteristics (related to the non-autonomous category described above):

- Data are distributed across all the nodes
- The same DBMS is used at each location
- All data are managed by the distributed DBMS (so there are no exclusively local data)
- All users access the database through one global schema or database definition
- The global schema is simply the union of all the local database schemas

It is difficult in most organizations to force a homogeneous environment, yet heterogeneous environments are much more difficult to manage.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.