The Ultimate IT Goal

To Cost Effectively Support User Organizations With:

- Application Function and Information Access
- That Represents Ease Of Use
- Of Seamless and Transparent Systems
Information Access Enablers

1. A shift to a network-based computer model made possible in many ways by the personal computer’s ability to send and receive messages and its ability to act independently.

2. The emergence of standards and technologies that facilitate wide-spread deployment of connectivity potential through the world’s largest network. Which is . . . .

The Internet!
Key to this is Data Management

1. Capture
2. Indexing and defining the data
3. Compression
4. Storage
5. Access and retrieval
6. Transport
7. Presentation
BUSINESS AND INFORMATION TECHNOLOGY DYNAMICS

MULTI-VENDOR/MULTI-PRODUCTS

- INFORMATION SYSTEMS ARCHITECTURE
- OPEN SYSTEMS
- STANDARDS

VENDOR HARDWARE, SOFTWARE PRODUCTS AND SERVICES

USER ORGANIZATIONS
- APPLICATION FUNCTION
- EASE OF USE
- SEAMLESS AND TRANSPARENT
Management and IS Models

• Centralized
• Decentralized
• Distributed through some kind of network
• Distributed through a network based on client-server architecture
Distributed Systems: Multiple processors that share a network.

Peer-to-peer Networking: Technologies that permit arbitrary communication among computers.

Cooperative Processing: Applications are processed on multiple processors and this is transparent to the user.

Client-Server Computing: A network shared by multiple processors involving split processing with specialized purposes.
Internet Distributed Computing

The Internet offers an amazing diversity of services that involve a form of distributed computing:

• Sending email
• Retrieving a file
• Browsing through directories
• Searching a database
• Printing a document
• Transmitting a FAX
• Listening to music
Internet Scheme is Client-Server

Some computers on the Internet offer services that others can access based on three basic facts:

1. Programs communicate based on the fact that both computers use TCP/IP protocol software.

2. TCP/IP does not create or run application programs. The Internet works like the telephone system since it allows one program to call another and it in turn answers the call for communication to be possible.

3. Computers can run multiple programs at the same time.
Basic Points to Remember

• Programs are either clients or servers.

• A server must always run (to be ready to receive requests).

• Users usually run client software when they access a service.

• A client program contacts a server, sends a request and displays the server’s request.
IS Architecture

A blueprint that depicts how future systems will be designed and built. The various systems components are explicitly defined so that multiple people can build systems around the same base line.
Open Systems Environment

1. Operating System
2. Communication Services
3. Database
4. User Interface
5. Software Development Tools
6. Systems Management Services

Software Applications
The Pull for Standards

As long as your systems remain within the confines of your organization you can implement an application any way YOU want to do it.

When your information systems go beyond the boundaries of your organization you will probably have to operate consistent with accepted standards.
Client Server Computing

“The Best of Both Worlds!?”

- Better Performance
- Shared Data
- Easier to Use
- Sense of Ownership
- Good Modularity (expand and contract)
- Applications Implemented Faster
- More Cost Effective
Client-Server Architecture Options

1. File Servers
2. Database Servers
3. Transaction Servers (OLTP)
4. Groupware Servers
5. Object Servers
6. Web Servers
**File Server:** A client passes a request for a file over a network to a server.

**Database Server:** The client sends a SQL request to the database server that processes the request and returns a message for each message received.

**Transaction Server:** The client invokes a remote procedure that resides on the server with a SQL database engine.

**Groupware Server:** Manages text, image, mail, bulletin boards and the flow of work.

**Object Server:** Applications are written as a set of communicating objects. Client objects can communicate with service objects using an object request broker. The results are returned to the client.

**Web Servers:** Functions like one of the above but uses Internet technologies and standards.
Middleware

The cost of implementing an enterprise-wide homogeneous database in which the data is stored in the same format at all sites is prohibitive and does not mirror reality.

Company databases resemble a federation of data that is organized according to relational, object-oriented or hierarchical data models.

Many applications use middleware to access data from multiple platforms and disparate data sources. Middleware promotes adaptability, new technologies, applications and data services that can be added to existing services without limiting the newer models to fit the older ones.
Middleware Evaluation Criteria

Picking the right one will continue to be a challenge.

- Performance (meet the demands of the application)
- Openness (how open is open?)
- Permissiveness (flexibility)
- Reliability (meet your requirements?)
- Transparency (the goal of middleware)
- Stability (some architecture maturity)
- Manageability (exercising practical control)

by Robert Orfali, Dan Harkey, Jeri Edwards


Availability: Usually ships within 24 hours.

Paperback - 800 pages 3rd edition (February 1999)
Amazon.com Sales Rank: 1,912
Average Customer Review: 4.5
Number of Reviews: 13
I have been doing nothing but client/server for the last five years. In spite of that I cannot keep up with all the latest developments. I am a computer consultant -- in my field it is imperative that I have a very broad overview -- I never know what my clients might ask and must have something worthwhile to answer.

This book is an excellent book for understanding the entire client/server space. The level of detail is excellent for such a broad overview. It is not too deep. It is not shallow.

Use this book as a jump-start to understand the client/server landscape.