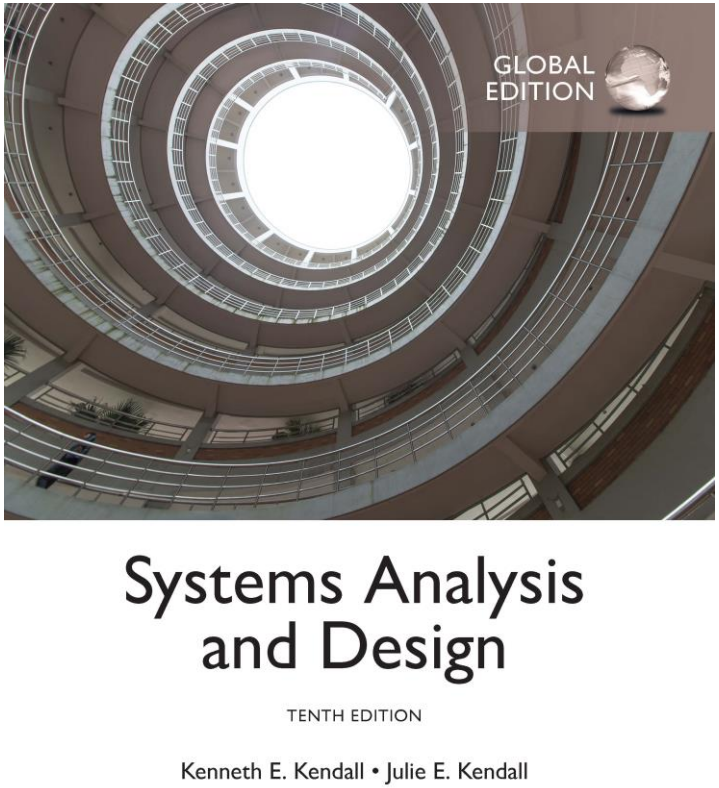
Systems Analysis and Design

Tenth Edition
Global Edition



Chapter 16

and

Quality Assurance and Implementation

Learning Objectives (1 of 2)

16.1 Recognize the importance of users and analysts taking a total quality approach to improve the quality of software design and maintenance.

16.2 Realize the importance of documentation, testing, maintenance, and auditing.

16.3 Understand how service-oriented architecture and cloud computing is changing the nature of information system design.

Learning Objectives (2 of 2)

16.4 Design appropriate training programs for users of the new system.

16.5 Recognize the differences among physical conversion strategies, and be able to recommend an appropriate one to a client.

16.6 Address security, disaster preparedness, and disaster recovery concerns for traditional and Web-based systems.

16.7 Understand the importance of evaluating the new system, and be able to recommend a suitable evaluation technique to a client.

Major Topics (1 of 2)

- Six Sigma
- Quality assurance
- Walkthroughs
- Structure charts
- Modules
- Documentation
- Testing

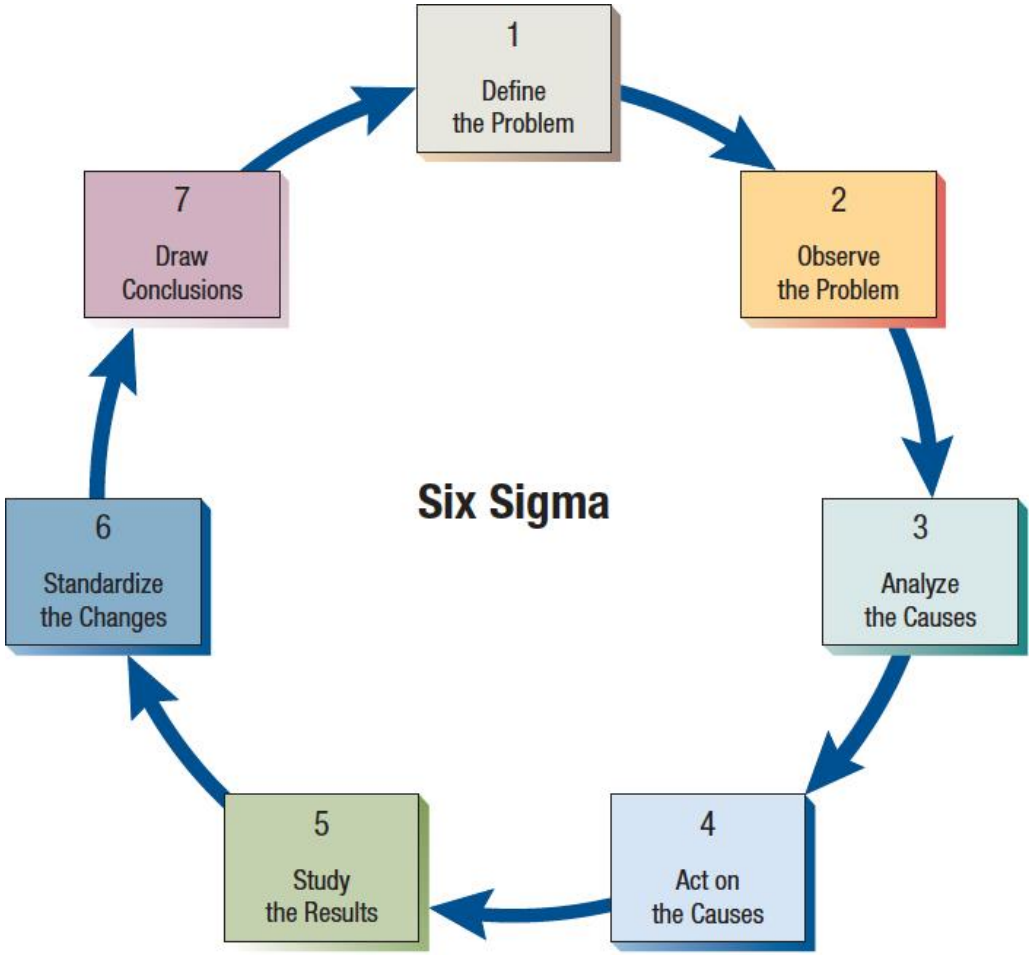
Major Topics (2 of 2)

- Client-server computing
- Network types
- Groupware
- Training
- Security
- Organizational metaphors
- Evaluation

Six Sigma

- A culture built on quality
- Uses a top-down approach
- Project leader is called a Black Belt
- Project members are called Green Belts
- Master Black Belts have worked on many projects and are available as a resource to project teams

Figure 16.1 Every Systems Analyst Should Understand the Methodology and Philosophy of Six Sigma



Responsibility for Total Quality Management

- Full organizational support of management must exist
- Early commitment to quality from the analyst and business users

Structured Walkthroughs

- One of the strongest quality management actions is to do structured walkthroughs routinely
- Use peer reviewers to monitor the system's programming and overall development
- Point out problems
- Allow the programmer or analyst to make suitable changes

Involved in Structured Walkthroughs

- The person responsible for the part of the system being reviewed
- A walkthrough coordinator
- A programmer or analyst peer
- A peer who takes notes about suggestions

The Top-Down Approach

- Top-down design allows the systems analyst to ascertain overall organizational objectives and how they are best met in an overall system
- The system is divided into subsystems and their requirements

Advantages of the Top-Down Approach

- Avoiding the chaos of attempting to design a system all at once
- Enables separate systems analysis teams to work in parallel on different but necessary subsystems
- Prevents losing sight of what the system is supposed to do

Modular Development

- Breaking the programming into logical, manageable portions or modules
- Works well with top-down design
- Each individual module should be functionally cohesive, accomplishing only one function

Advantages of Modular Programming

- Modules are easier to write and debug
- Modules are easier to maintain
- Modules are easier to grasp because they are self-contained subsystems

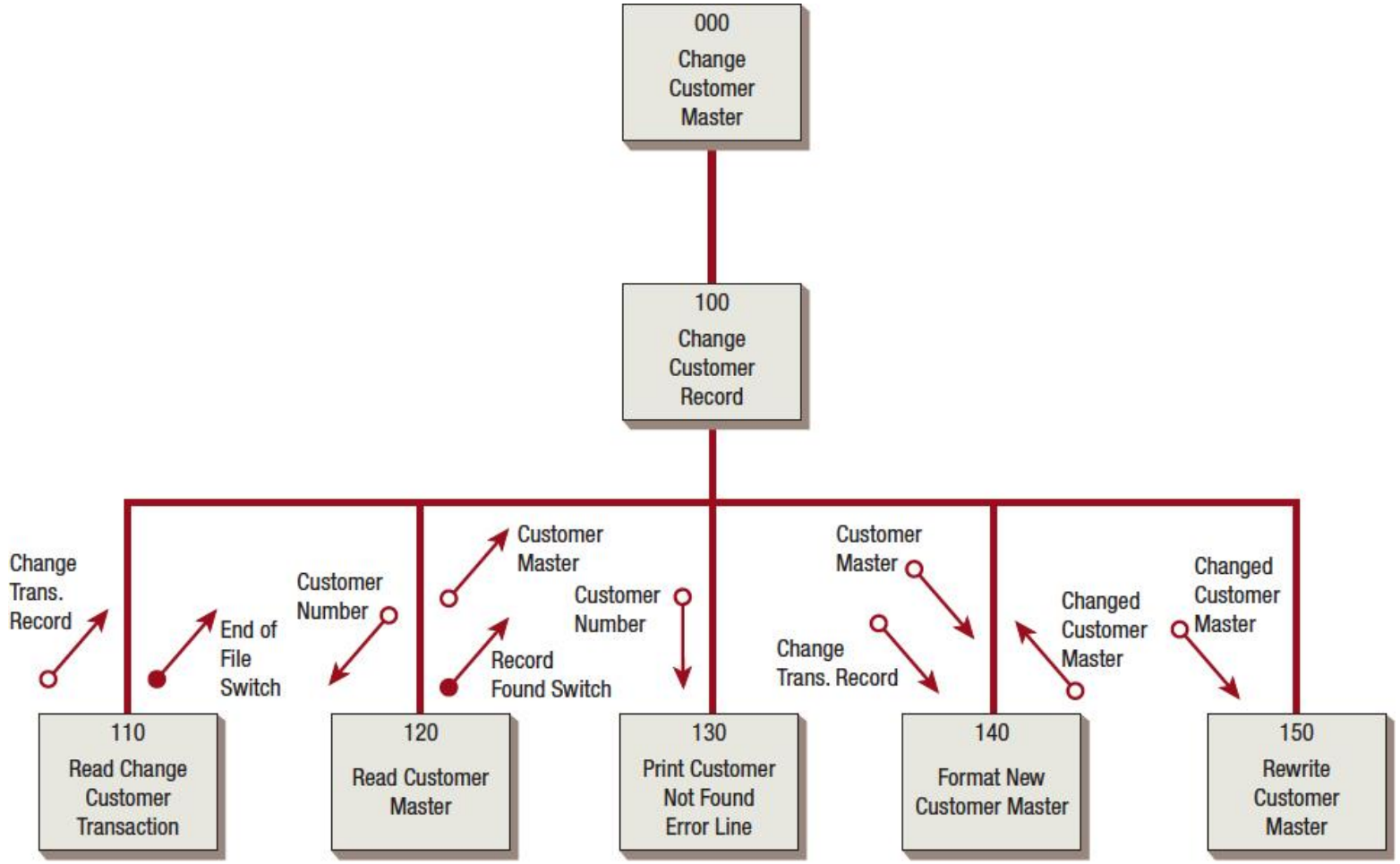
Guidelines for Modular Programming

- Keep each module to a manageable size
- Pay particular attention to the critical interfaces
- Minimize the number of modules the user must modify when making changes
- Maintain the hierarchical relationships set up in the top-down phases

Using Structure Charts to Design Systems

- The recommended tool for designing a modular, top-down system is a structure chart
- A structure chart is simply a diagram consisting of rectangular boxes, representing the modules, and connecting lines
- Hollow circle arrows represent data being passed up and down the structure
- Filled-in circle arrow represents a control switch or flag

Figure 16.3 A Structure Chart Encourages Top-Down Design Using Modules



Service Oriented Architecture (SOA) (1 of 3)

- The SOA approach is to make individual SOA services that are unassociated or only loosely coupled to one another
- Each service executes one action
- Each service can be used in other applications within the organization or even in other organizations

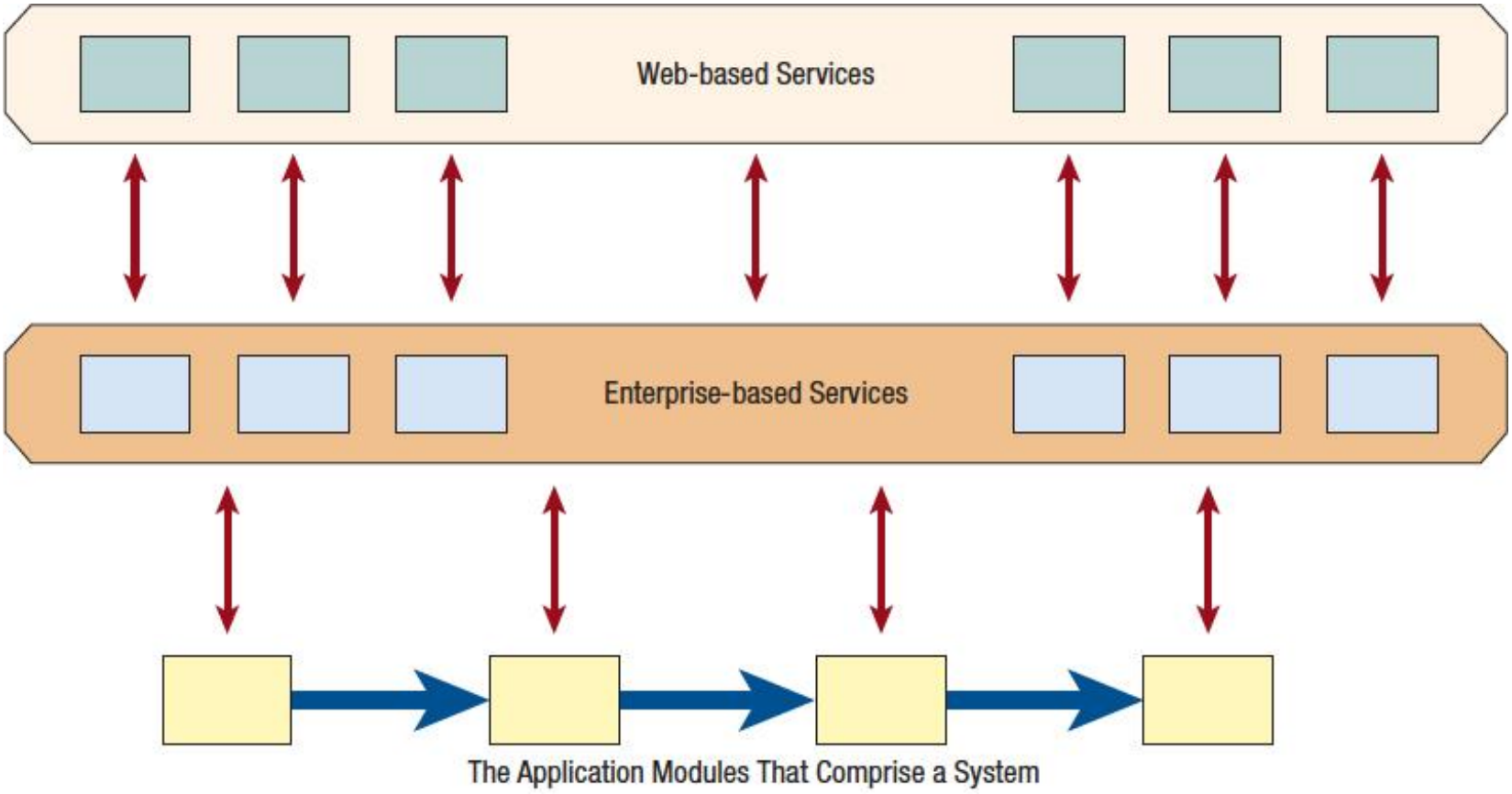
Service Oriented Architecture (SOA) (2 of 3)

- A service can use certain defined protocols so that it can communicate with other services
- The burden of connecting services in a useful fashion, is a process called orchestration
- This can even be accomplished by selecting services from a menu of services and monitoring them by setting up a SOA dashboard

Service Oriented Architecture (SOA) (3 of 3)

- In order to set up a SOA, services must be:
  - Modular
  - Reusable
  - Work together with other modules (interoperability)
  - Able to be categorized and identified
  - Able to be monitored
  - Comply with industry-specific standards

Figure 16.4 Modules in Service-Oriented Architecture Are Independent and Can Be Ubiquitous



System Documentation

- Procedure manuals
- The **Folklore** method

Procedure Manuals (1 of 2)

- The English-language component of documentation
- Key sections:
  - Introduction
  - How to use the software
  - What to do if things go wrong
  - A technical reference section
  - An index
  - Information on how to contact the manufacturer

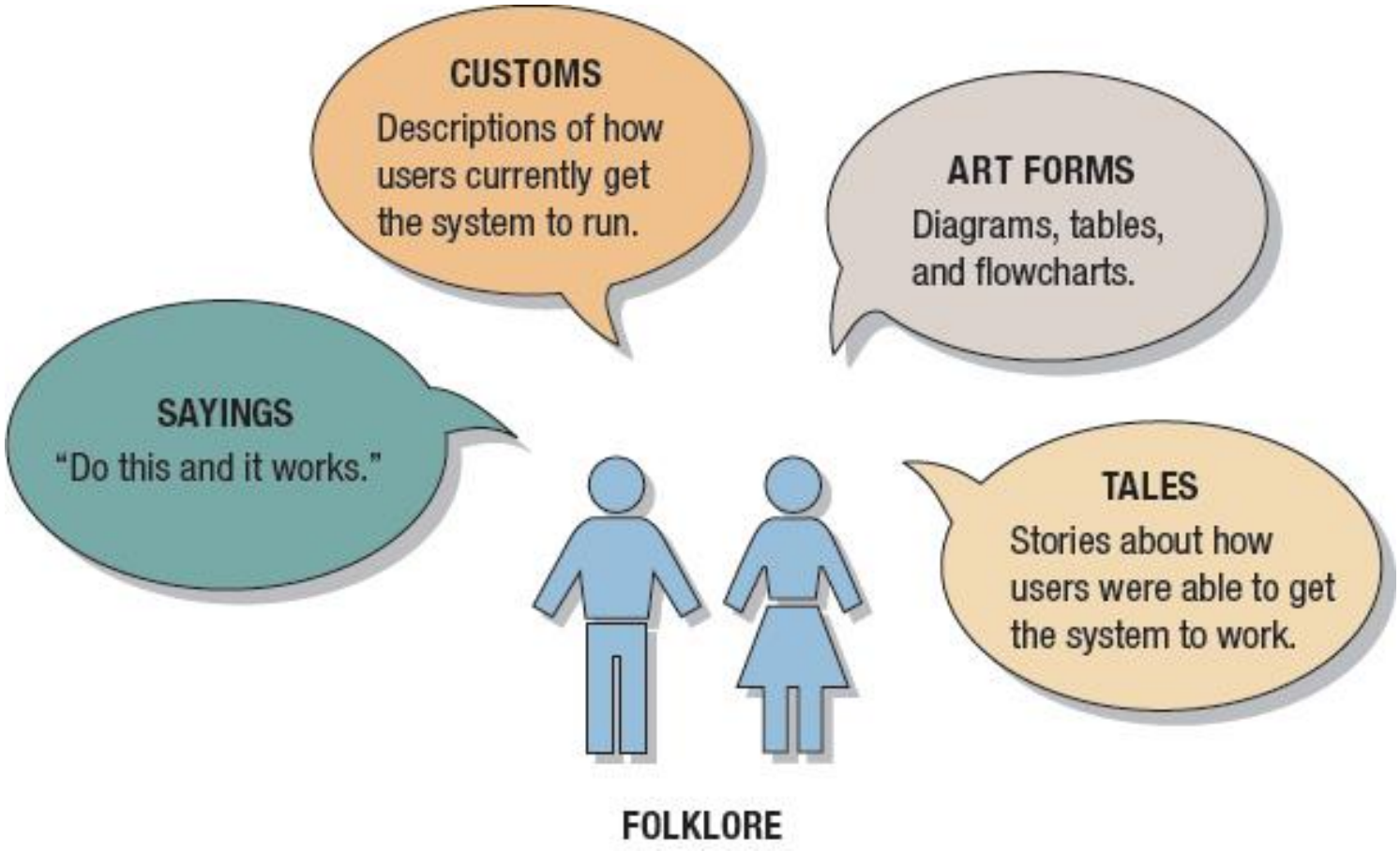
Procedure Manuals (2 of 2)

- Procedure manual complaints:
  - They are poorly organized
  - It is hard to find needed information
  - The specific case in question does not appear in the manual
  - The manual is not written in plain English

The FOLKLORE Method

- Collects information in categories:
  - Customs
  - Tales
  - Sayings
  - Art forms

Figure 16.5 Customs, Tales, Sayings, and Art Forms Used in the FOLKLORE Method of Documentation Apply to Information Systems



Choosing a Design and Documentation Technique (1 of 2)

- Is it compatible with existing documentation?
- Is it understood by others in the organization?
- Does it allow you to return to working on the system after you have been away from it for a period of time?

Choosing a Design and Documentation Technique (2 of 2)

- Is it suitable for the size of the system you are working on?
- Does it allow for a structured design approach if that is considered to be more important than other factors?
- Does it allow for easy modification?

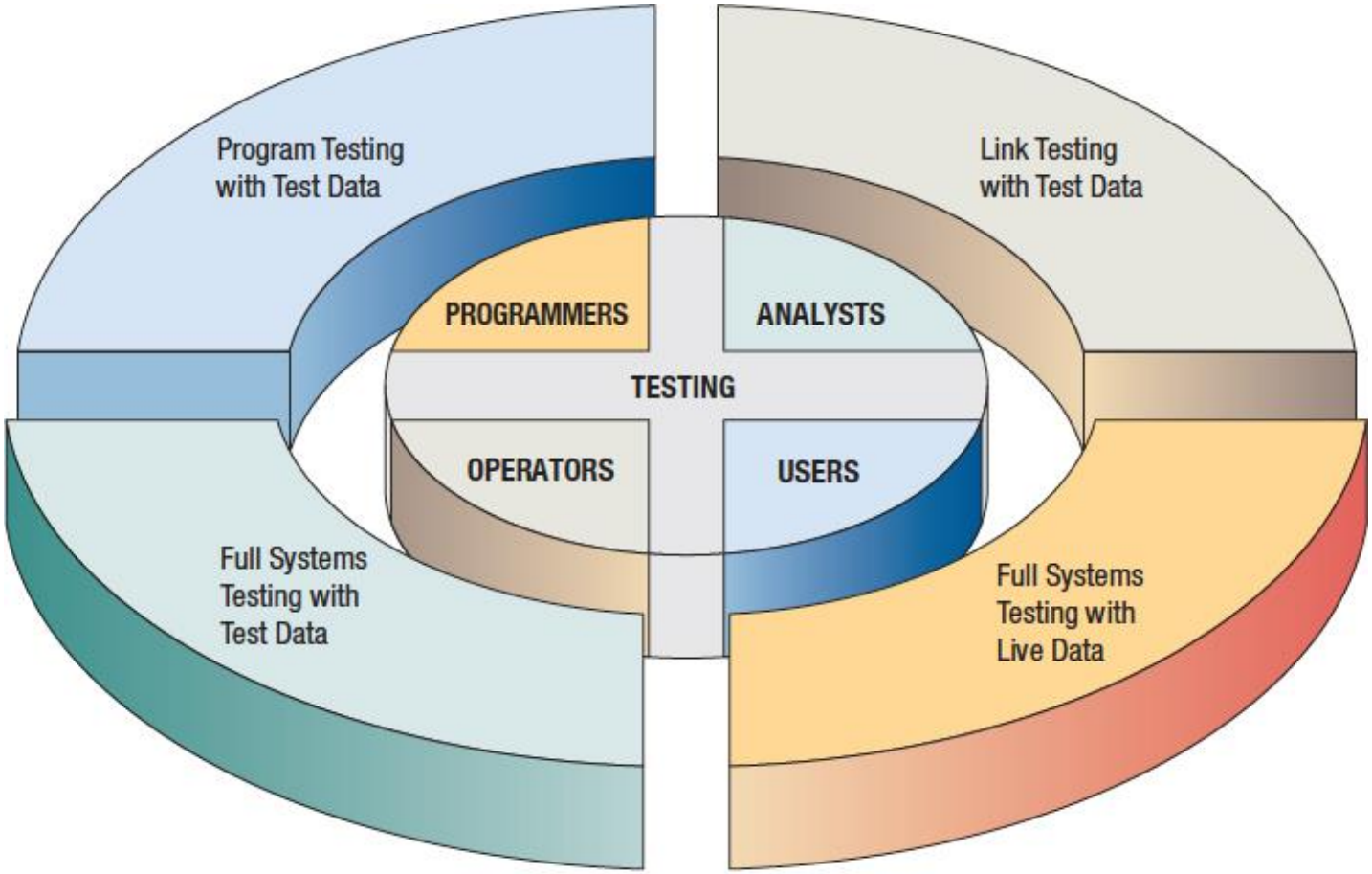
Testing, Maintenance, and Auditing

- The testing process
- Maintenance practices
- Auditing

The Testing Process

- Program testing with test data
- Link testing with test data
- Full system testing with test data
- Full system testing with live data

Figure 16.6 Programmers, Analysts, Operators, and Users All Play Different Roles in Testing Software and Systems



Program Testing with Test Data

- Desk check programs
- Test with both valid and invalid data
- Check output for errors and make any needed corrections

Link Testing with Test Data

- Also referred to as **string testing**
- Checks to see if programs that are interdependent actually work together as planned
- Test for normal transactions
- Test with invalid data

Full System Testing with Test Data

- Adequate documentation in procedure manuals
- Are procedure manuals clear enough?
- Do work flows actually “flow”?
- Is output correct and do users understand this output?

Full System Testing with Live Data

- Comparison of the new system's output with what you know to be correctly processed output
- Only small amounts of live data are used

Maintenance Practices

- Reduce maintenance costs
- Improve the existing software
- Update software in response to the changing organization
- Ensure channels for feedback

Auditing

- Having an expert who is not involved in setting up or using the system examine information in order to ascertain its reliability
- There are internal and external auditors
- Internal auditors study the controls used in the information system to make sure that they are adequate
- External auditors are used when the information system processes data that influences a company's financial statements

Implementing Distributed Systems

- Can be conceived of as an application of telecommunications
  - Includes work stations that can communicate with each other and with data processors
  - May have different hierarchical architectural configurations of data processors that communicate with one another

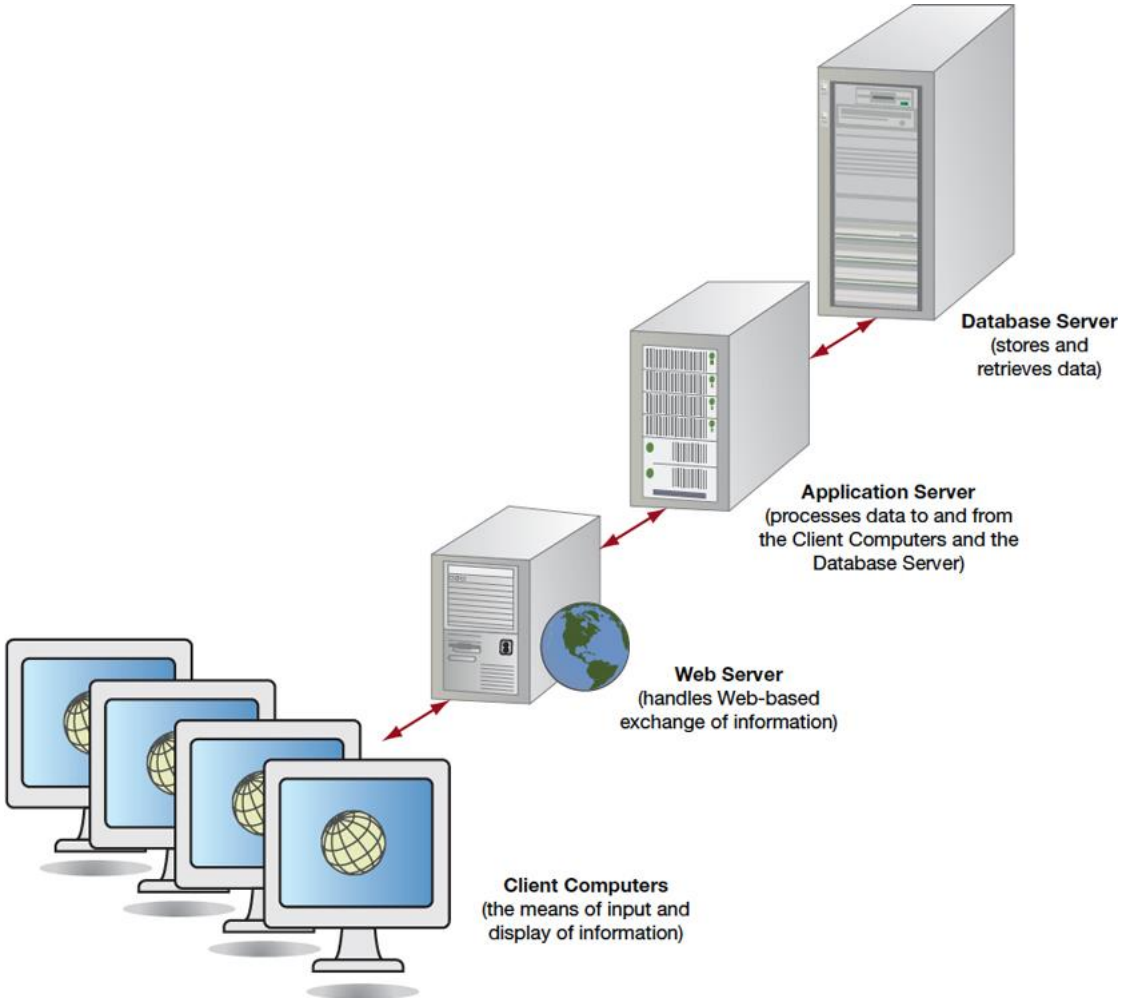
Client-Server Technology

- The client-server model refers to a design model that can be thought of as applications running on a network
- The client is a networked computer that uses small programs to do front-end processing, including communicating with the user

Client-Server Model Advantages, Disadvantages

- Advantage—greater computer power and greater opportunity to customize applications
- Disadvantage—more expensive and applications must be written as two separate software components each running on separate machines

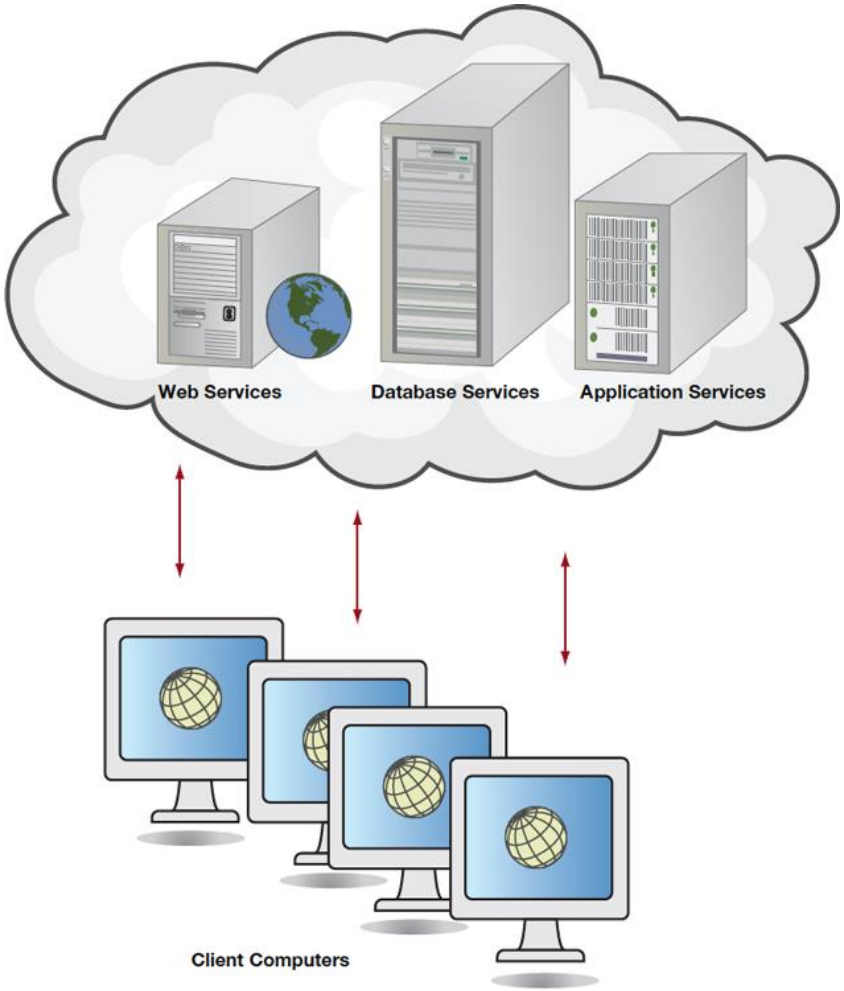
Figure 16.7 A Three-Tiered Client/Server Configuration



Cloud Computing (1 of 2)

- Organizations and individual users can use:
  - Web services
  - Database services
  - Application services over the Internet
- Done without having to invest in corporate or personal hardware, software, or software tools

Figure 16.8 Cloud Computing



Cloud Computing (2 of 2)

- A large collection of corporate users jointly bear the lowered cost
- Realize increases in peak-load capacity
- Companies also hope to improve their ability to perform disaster recovery

Cloud Computing Trade-Offs (1 of 2)

- One major decision is whether to use an external cloud provider (public cloud) or an internal one (private cloud)
- Sometimes the decision to use a public cloud has to do with how critical it is for the company to strategically control data

Cloud Computing Trade-Offs (2 of 2)

- A hybrid approach, in which more critical data linked to strategic aims and corporate intellectual property are held in a private cloud, and less critical data for the same organization makes use of a public cloud, may be a solution

Factors in Choosing a Cloud Provider (1 of 3)

- If a client of the cloud services provider is hacked, how will it adversely affect other businesses the cloud provider is hosting?
- How can your client be sure data are completely deleted from cloud systems when these systems are eventually decommissioned?

Factors in Choosing a Cloud Provider (2 of 3)

- Are the data architectures being used by the cloud provider standard and transparent so if the client wants to switch providers it would be realistic to do so?

Factors in Choosing a Cloud Provider (3 of 3)

- Are there too many nonstandard practices and proprietary programming structures in use, making transferring data out of the question?
- How can your client be certain that no one at the cloud provider or their staff working for other clients has access to your organization's data?

ERP and Cloud Computing

- Many of the issues and challenges associated with implementing ERP systems can be mitigated by ERP offerings that take advantage of cloud computing

Network Modeling

- Draw a network decomposition diagram to provide an overview of the system
- Draw a hub connectivity diagram
- Explode the hub connectivity diagram to show the various workstations and how they are connected

Figure 16.9 Use Special Symbols When Drawing Network Decomposition and Hub Connectivity Diagrams

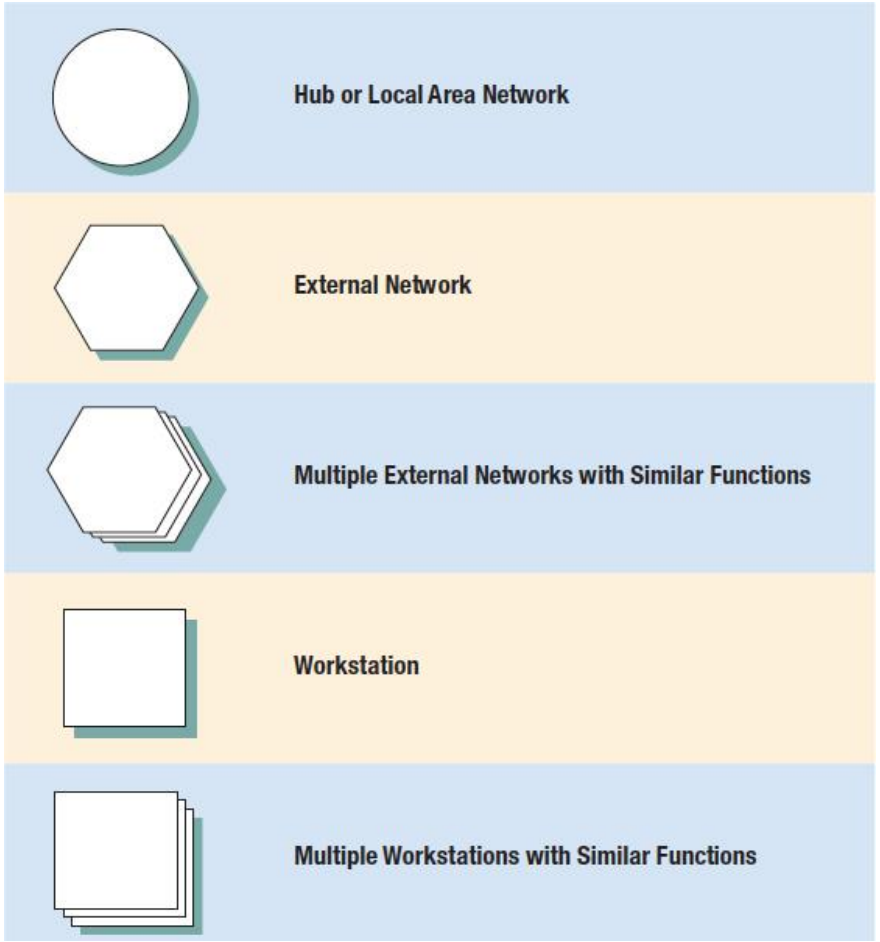


Figure 16.10 A Network Decomposition Diagram for World's Trend

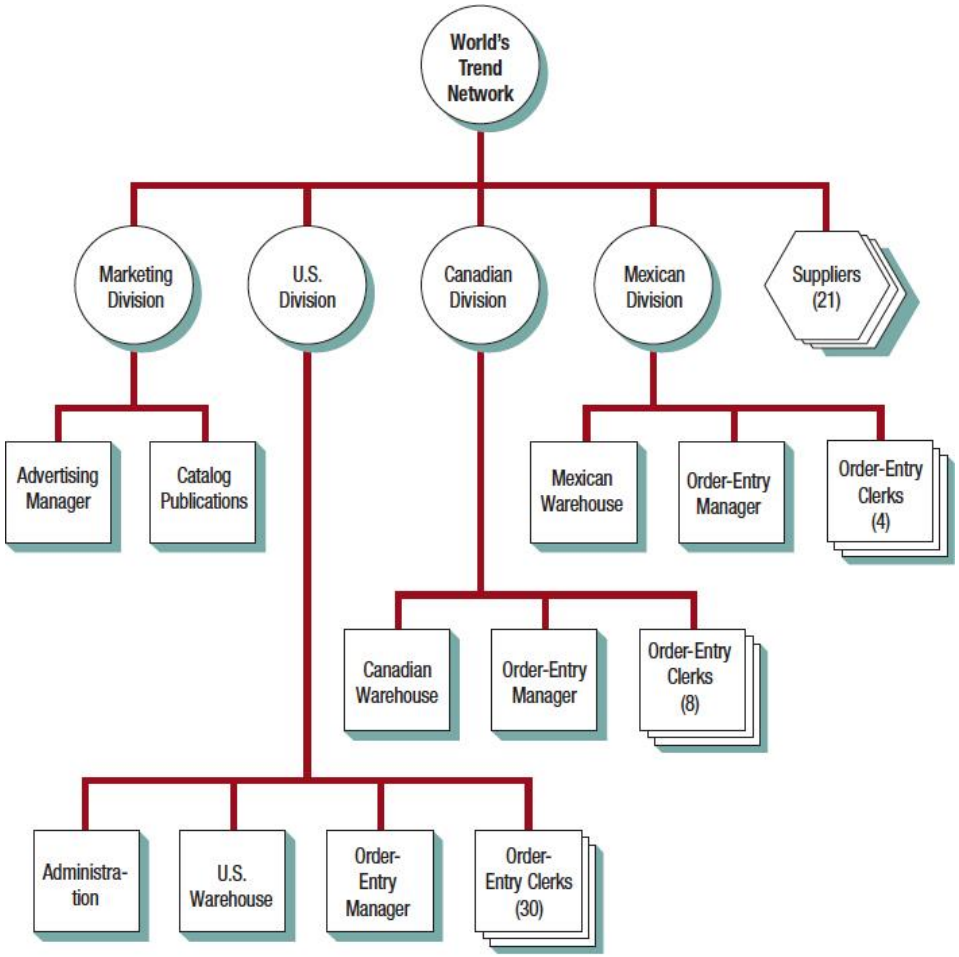


Figure 16.11 A Hub Connectivity Diagram for World's Trend

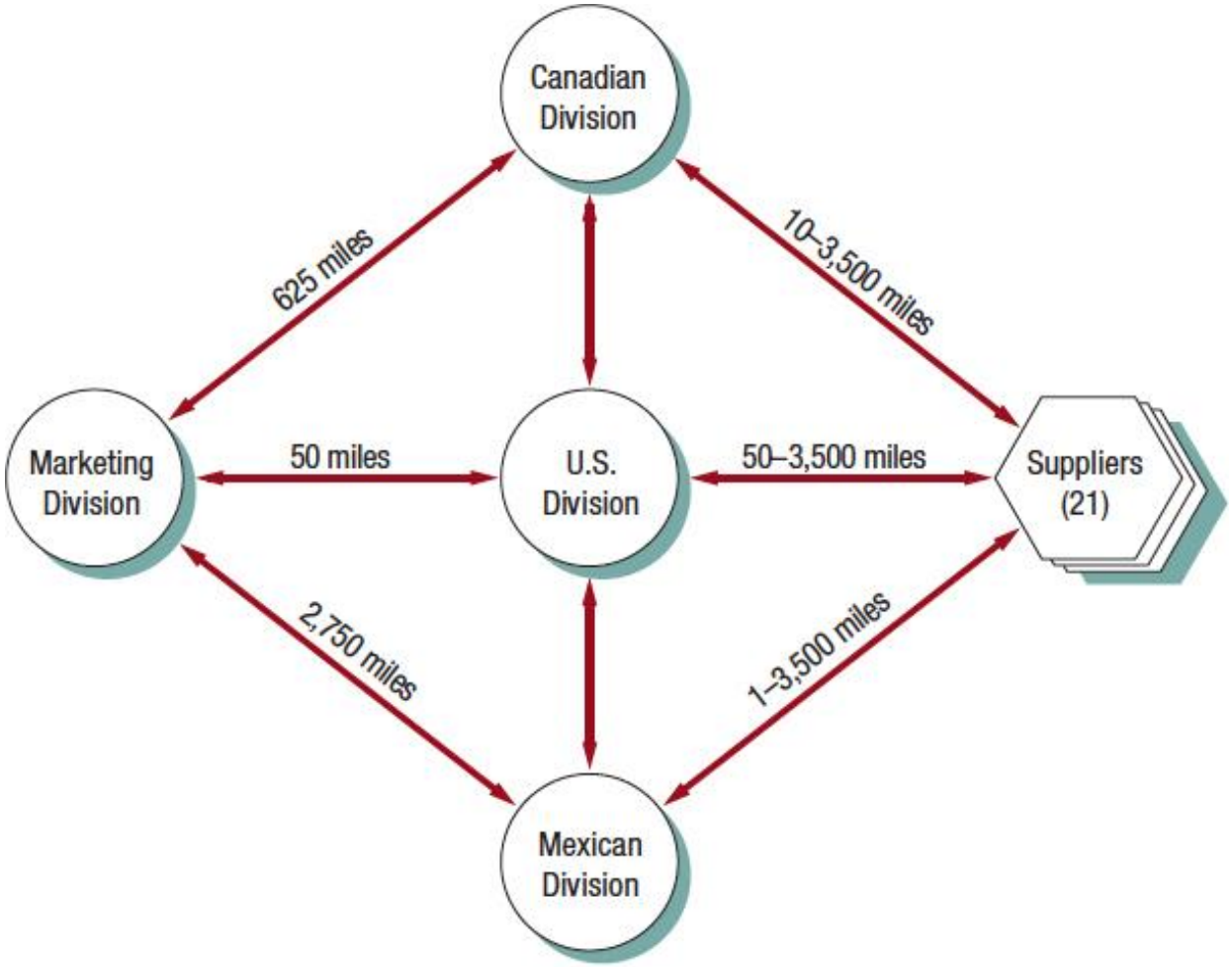
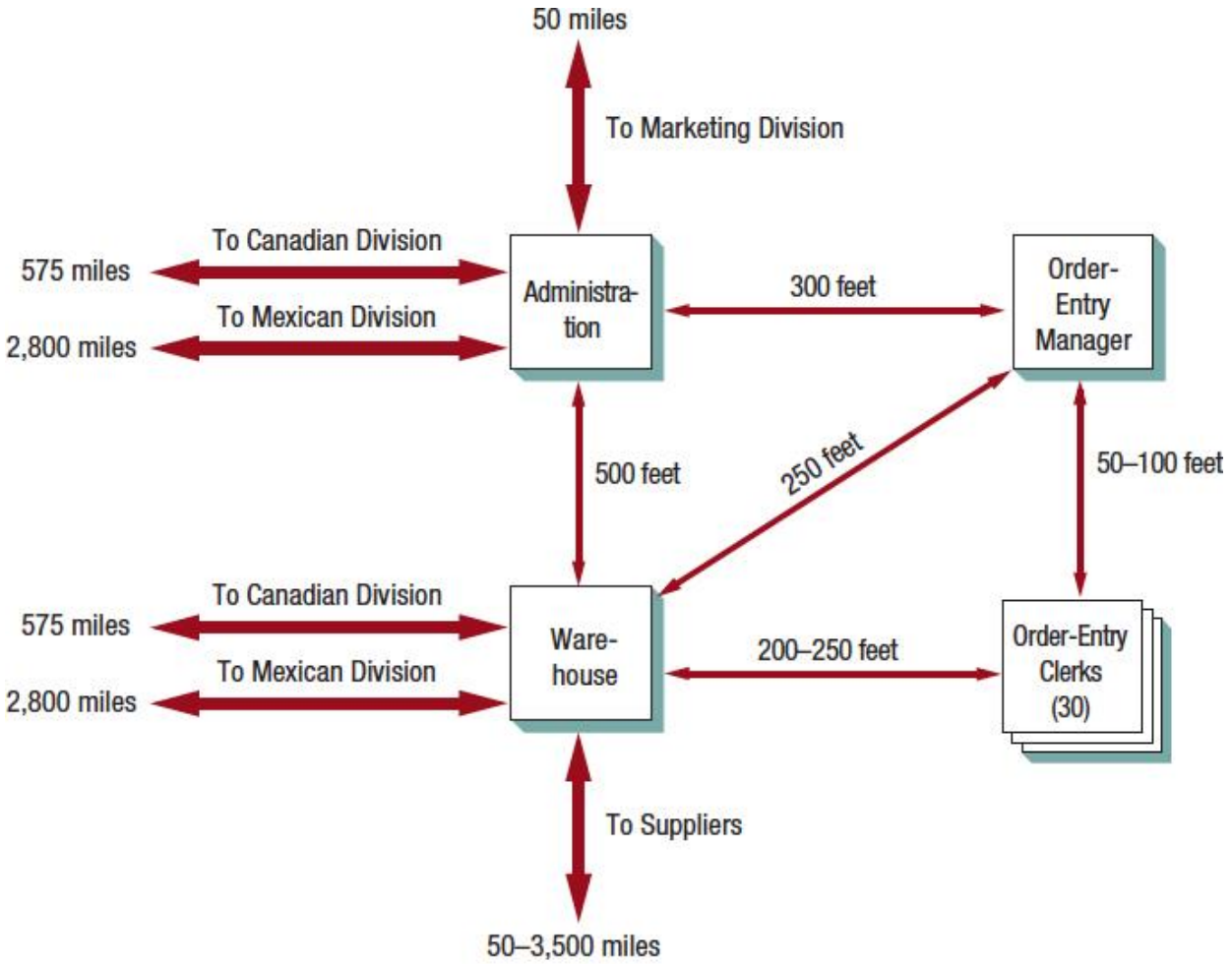


Figure 16.12 A Workstation Connectivity Diagram for World's Trend



Training

- Who to train
- People who train users
- Training objectives
- Training methods
- Training sites
- Training materials

Who to Train

- All people who will have primary or secondary use of the system
- Ensure that users of different skill levels and job interests are separated

People Who Train Users

- Vendors
- Systems analysts
- External paid trainers
- In-house trainers
- Other system users

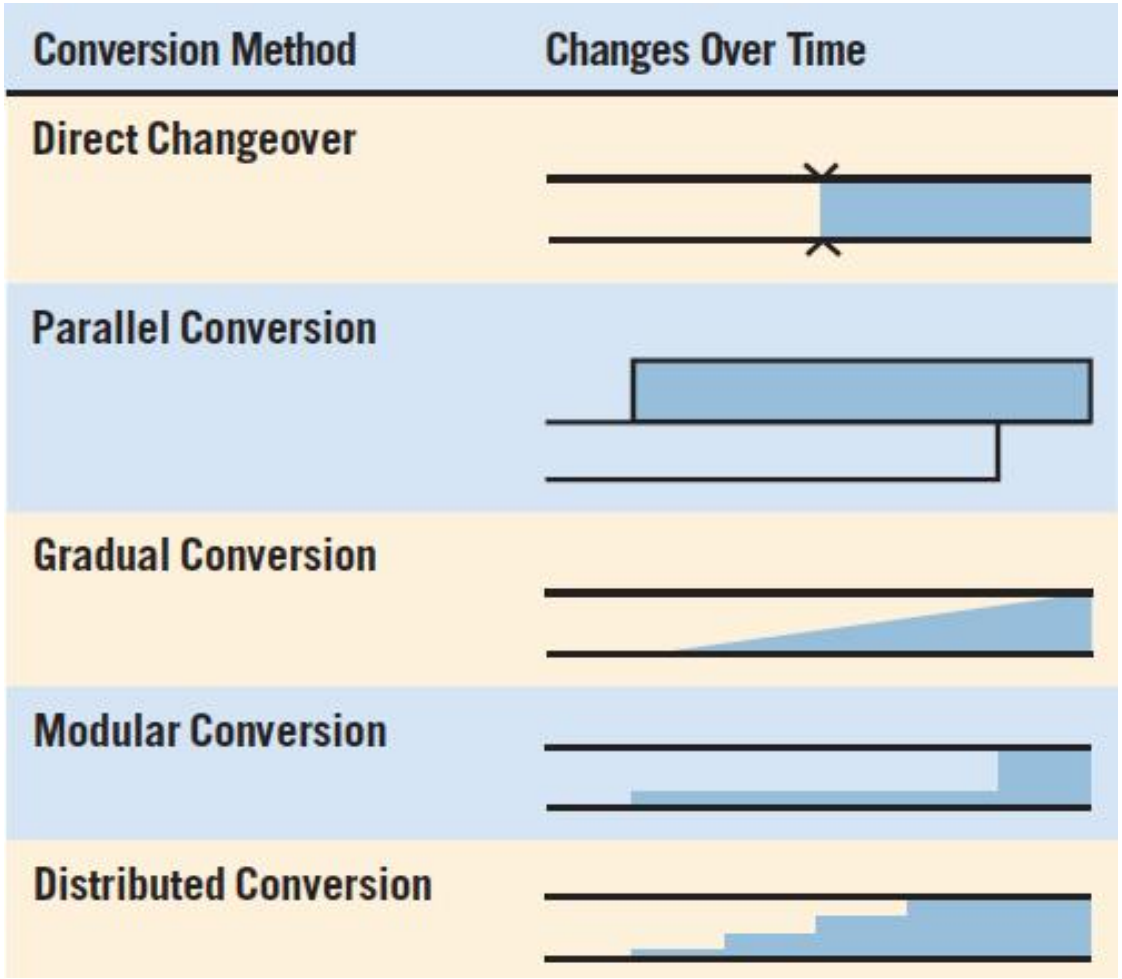
Figure 16.13 Appropriate Training Objectives, Methods, Sites, and Materials Are Contingent on Many Factors

Elements	Relevant Factors
Training Objectives	Depend on requirements of user's job
Training Methods	Depend on user's job, personality, background, and experience; use combination of lecture, demonstration, hands-on, and study
Training Sites	Depend on training objectives, cost, availability; free vendor sites with operable equipment; in-house installation; rented facilities
Training Materials	Depend on user's needs; operating manuals, cases, prototypes of equipments and output; online tutorials

Conversion Strategies

- Direct changeover
- Parallel conversion
- Gradual or phased conversion
- Modular prototype conversion
- Distributed conversion

Figure 16.14 Five Conversion Strategies for Information Systems



Direct Changeover

- Old system stops, new one starts
- Needs extensive testing
- Risky approach to conversion
- Users may resent being forced into using an unfamiliar system without recourse
- No adequate way to compare new results to old

Parallel Conversion

- Run the old and new systems at the same time
- The advantage is that you can check new data against old data
- The disadvantage is doubling employees' workloads

Gradual Conversion

- Combines best features of parallel and direct conversion
- Volume of transactions is gradually increased
- Advantage is that users get involved with the system gradually
- Agile methodologies use this conversion approach

Modular Prototype Conversion

- Each module is tested and put into use
- The advantage is that each module is thoroughly tested before being used
- Users are familiar with each module as it becomes operational
- Object-oriented methodologies often use this approach

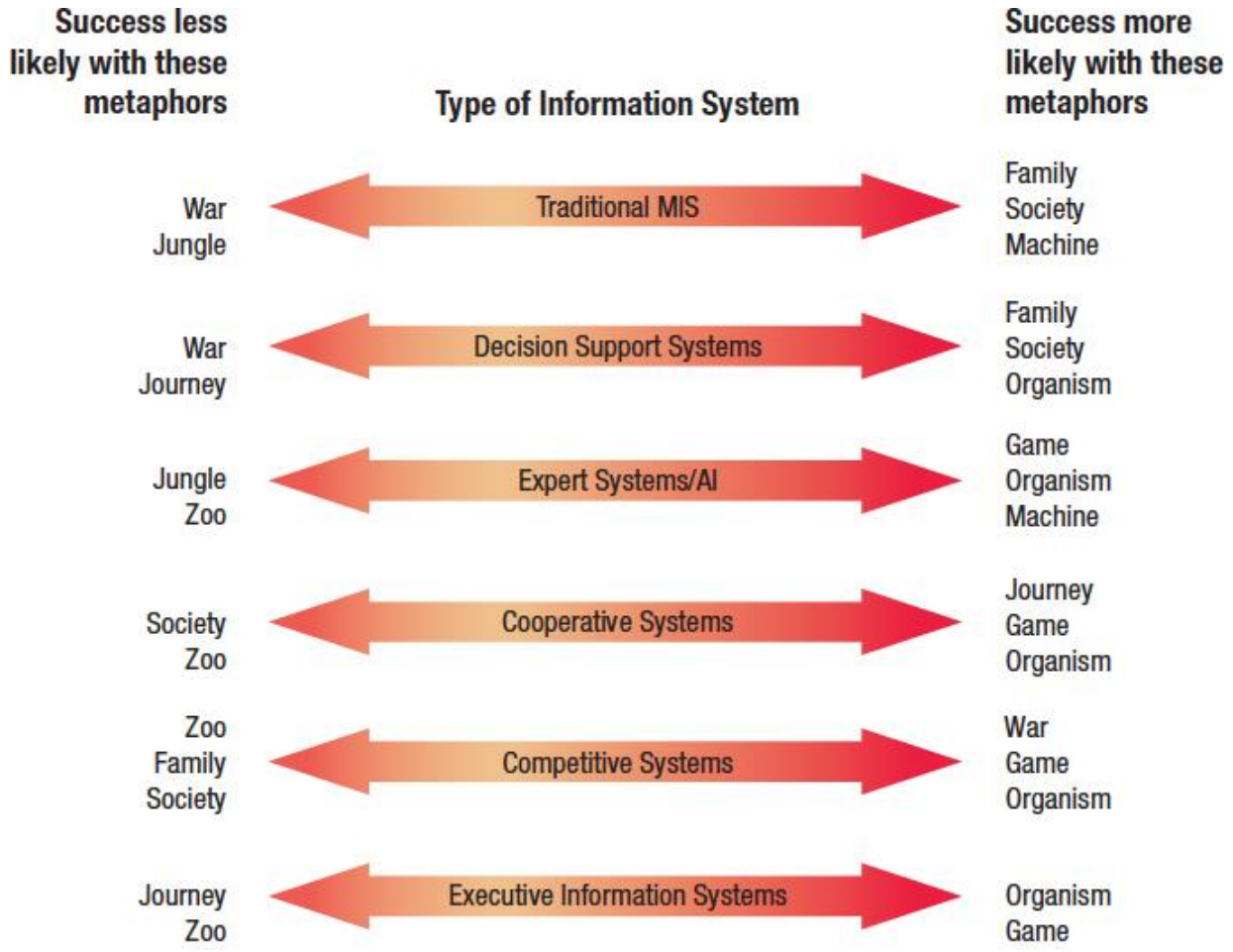
Distributed Conversion

- When there are many installations of the same system, such as at branch offices
- Install software at one office
- Advantage is that problems can be detected and contained
- Disadvantage is that even when one conversion is successful, each site will have its own peculiarities to work through

Other Conversion Considerations

- Ordering equipment
- Ordering any external materials supplied to the information system
- Appointing a manager to supervise the preparation of the installation site
- Planning, scheduling, and supervising programmers and data entry personnel

Figure 16.15 Organizational Metaphors May Contribute to the Success or Failure of an Information System



Security Concerns (1 of 2)

- Physical security
- Logical security
- Behavioral security

Security Concerns (2 of 2)

- Physical security is securing the computer facility, its equipment, and software through physical means
- Logical security refers to logical controls in the software itself
- Behavioral security is building and enforcing procedures to prevent the misuse of computer hardware and software

Special Security Considerations for Ecommerce (1 of 2)

- Virus protection software
- Email filtering products
- URL filtering products
- Firewalls, gateways, and virtual private networks
- Intrusion detection products

Special Security Considerations for Ecommerce (2 of 2)

- Vulnerability management products
- Security technologies such as secure socket layering (SSL) for authentication
- Encryption technologies
- Public key infrastructure (PKI) use and obtaining a digital certificate

Privacy Considerations for Ecommerce (1 of 2)

- Start with a corporate policy on privacy
- Only ask for information required to complete the transaction
- Make it optional for customers to fill out personal information on the Web site

Privacy Considerations for Ecommerce (2 of 2)

- Use sources that allow you to obtain anonymous information about classes of customers
- Be ethical

Disaster Recovery Planning (1 of 2)

- Identify teams responsible for managing a crisis
- Eliminate single points of failure
- Determine data replication technologies that match the organization's timetable
- Create detailed relocation and transportation plans

Disaster Recovery Planning (2 of 2)

- Provide recovery solutions that include an off-site location
- Ensure the physical and psychological well-being of employees and others

Identify Who is Responsible

- Whether business operations will continue
- How to support communications
- Where people will be sent if the business is uninhabitable
- Where personnel will go in an emergency
- Seeing to the personal and psychological needs
- Restoring the main computing and working environments

Single Points of Failure and Data Replication Technologies

- Redundancy of data provides the key for servers running Web applications
- SNAs and data mirroring

Relocation and Transportation Plans

- Send employees home
- Remain on site
- Relocate to a recovery facility

Communication Channels

- Email
- Emergency information Web page
- Emergency hotline
- Emergency response agencies

Recovery Solutions and Support for the Well-Being of Employees

- Recovery involves an off-site location and converting paper documents to digital formats
- Well-being of employees might include providing water or safety kits

Evaluation Techniques

- Cost-benefit analysis
- Revised decision evaluation approach
- User involvement evaluations
- The information system utility approach

Information System Utility Approach (1 of 3)

- Possession
- Form
- Place
- Time
- Actualization
- Goal

Information System Utility Approach (2 of 3)

- Possession utility answers the question of who should receive output
- Goal utility answers the why of information systems by asking whether the output has value in helping the organization achieve its objectives
- Place utility answers the question of where information is distributed

Information System Utility Approach (3 of 3)

- Form utility answers the question of what kind of output is distributed to the decision maker
- Time utility answers the question of when information is delivered
- Actualization utility involves how the information is introduced and used by the decision maker

Website Evaluation (1 of 2)

- Know how often the Web site is visited
- Learn details about specific pages on the site
- Find out more about the Web site's visitors

Website Evaluation (2 of 2)

- Discover if visitors can properly fill out the forms you designed
- Find out who is referring Web site visitors to the client's site
- Determine what browsers visitors are using
- Find out if the client's Web site visitors are interested in advertising

Summary (1 of 5)

- TQM
  - Designing systems and software with a top-down, modular approach
  - Designing and documenting systems and software using systematic methods
  - Testing systems and software so that they can be easily maintained and audited

Summary (2 of 5)

- Six Sigma
  - Define the problem
  - Observe the problem
  - Analyze the causes
  - Act on the causes
  - Study the results
  - Standardize the changes
  - Draw conclusions

Summary (3 of 5)

- Structure charts
- Procedure manuals
- **Folklore**
- Testing
- System maintenance
- Auditing

Summary (4 of 5)

- Implementation
- Distributed systems
  - Client-server
- Training users and personnel
- Conversion
  - Direct changeover
  - Parallel
  - Phased
  - Gradual
  - Modular prototype

Summary (5 of 5)

- Security
  - Physical
  - Logical
  - Behavioral
- Organizational metaphors
- Evaluation