ANALISA dan RANCANGAN
SISTEM INFORMASI

disusun
oleh
Ir KI PRAMONO HA, MMSi

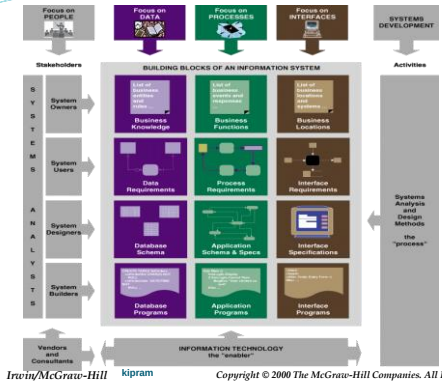
JAKARTA
2004/2007/2010/2014

DAFTAR REFERENSI

1. Hawryzkiewycz. I.T ,*Introduction to System Analysis and Design*, Kedua Prentice Hall Singapore, 1992
2. Kowal, J.A , *Analyzing System* , Prentice Hall New Jersey, 1988
3. Whitten,J.L dan Bentley,L.D, dan Barlow, VM *Analisis dan Desaign Berorientasi Objek*, Kedua, Richard D.Irwin.Inc, 2000
4. Martin J, *Information Engineering, book* II , Prentice Hall Singapore 1990
5. Kendall,K.E dan Kendall,J.E ,*System Analysis and Design*, Kedua Prentice Hall New Jersey, 1992
6. Hoffer,J.A dan George J.F dan Valacich,J.S ,*Modern System Analysis and Design* , The Benjamin / Cummings Publ. Company,Menlo Park, 2014
7. Senn,J.A , *Analysis and Design of Information System* , Kedua Mc Graw hill, Singapore ,1989
8. Ebert, R.E , *User Interface Design* , Prentice Hall New Jersey, 1994
9. Satzinger, Jackson & Burd, "Systems Analysis and Design in a Changing World 7th Ed ,2015

BAB I
PENDAHULUAN

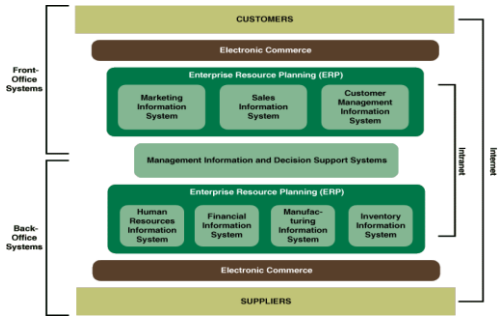
A. Information System Building Blocks



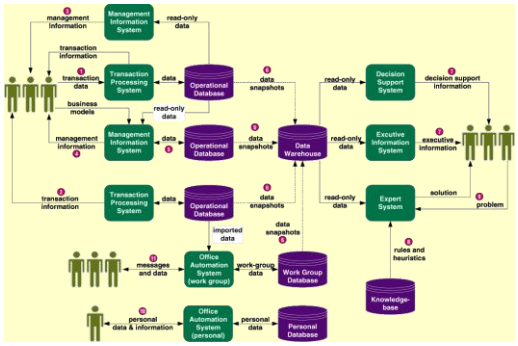
Front- and Back-Office Information Systems

- **Front-office information systems** support business functions that reach out to customers (or constituents).
  - Marketing
  - Sales
  - Customer management
- **Back-office information systems** support internal business operations and interact with suppliers (of materials, equipment, supplies, and services).
  - Human resources
  - Financial management
  - Manufacturing
  - Inventory control

A Federation of Information Systems



Information Systems Applications



Types of Information Systems and Systems Development

- **Transaction Processing Systems (TPS)**
  - Automate handling of data about business activities (transactions)
- **Management Information Systems (MIS)**
  - Converts raw data from transaction processing system into meaningful form
- **Decision Support Systems (DSS)**
  - Composed of database designed to help decision makers
  - Provides interactive environment for decision makers to manipulate data and models
- **Expert Systems (ES)**
  - Codifies and manipulate knowledge instead of information
  - Users communicate with an ES through interactive dialogue

B. Pengembangan dalam analysis dan design :

Software Engineering Process

A process used to create an information system

Consists of:

- ***Methodologies***
  - **A sequence of step-by-step approaches that help develop the information system**
- ***Techniques***
  - **Processes that the analyst follows to ensure thorough, complete and comprehensive analysis and design**
- ***Tools***
  - **Computer programs that aid in applying techniques**

Data and Processes

- **Three key components of an information system**
  - **Data**
  - **Data Flows**
  - **Processing Logic**
- **Data** vs. **Information**
  - **Data**
    - Raw facts about people, objects, and events in an organization such as customer's account number
  - **Information**
    - Data that have been processed and presented in a form that humans can understand

Data and Processes

- **Data**
  - Understanding the source and kind of data a system uses is key to good system design
  - Various techniques are used to describe data and the relationship among data
- **Data Flow**
  - Groups of data that move and flow through the system from one place to another
  - Include description of *sources* and *destination* for each data flow
- **Processing Logic**
  - Describe steps in the transformation of data and events that trigger these steps

Approaches to Systems Development

- **Process-Oriented Approach**
  - Focus is on **how** and **when** data are moved and **transformation** of data in an information system
  - Involves creating graphical representations such as **data flow diagrams** and **charts**
  - Data are tracked from sources, through intermediate steps and to final destinations
  - Natural structure of data is not specified
  - Disadvantage: existence of several data files each locked within different applications.
  - To change a single data element all files has to be updated

Approaches to Systems Development

- **Data-Oriented Approach**
  - Depicts **ideal organization** of data, independent of where and how data are used
  - Data model describes kinds of data and business relationships among the data
  - Business rules depict how organization captures and processes the data

Databases and Application Independence

- **Database**
  - **Shared collection of logically related data**
  - **Organized to facilitate capture, storage and retrieval by multiple users in an organization**
  - **Centrally managed**
  - **Designed around subjects**
    - **Customers**
    - **Suppliers**
- **Application Independence**
  - **Separation of data and definition of data from applications that use these data**

C. Organizational Responsibilities in Systems Development

- Systems Analysts work in teams
  - Project Based
  - Includes
    - IS Manager
    - Programmers
    - Users
    - Other specialists
  - Characteristics of Successful Teams
    - Diversity of backgrounds
    - Tolerance of diversity
    - Clear and complete communication
    - Trust
    - Mutual Respect
    - Reward structure that promotes shared responsibility

- IS Manager
  - May have a direct role in systems development if the project is small
  - Typically involved in allocating resources to and overseeing system development projects.
- Systems Analyst
  - Key individuals in the systems development process

D. Skills of a Successful Systems Analyst

- Analytical
  - Understanding of organizations
  - Problem solving skills
  - System thinking
    - Ability to see organizations and information systems as systems
- Technical
  - Understanding of potential and limitations of technology
- Management
  - Ability to manage projects, resources, risk and change
- Interpersonal
  - Effective written and oral communication skills

- Programmers
  - Convert specifications into instructions that the computer understands
  - Write documentation and testing programs
- Business Managers
  - Have power to fund projects and allocate resources
  - Set general requirements and constraints for projects

- Other IS Managers/Technicians
  - Database Administrator
    - Involved in design, development and maintenance of databases
  - Network and telecommunications experts
    - Develop systems involving data and/or voice communications
  - Human Factors Specialists
    - Involved in training users and writing documentation
  - Internal Auditors
    - Ensure that required controls are built into the system

E. Principles of System Development

- Get the system users involved.
- Use a problem-solving approach.
- Establish phases and activities.
- Document through development.
- Establish standards.
- Manage the process and projects
- Justify systems as capital investments.
- Don't be afraid to cancel or revise scope.
- Divide and conquer.
- Design systems for growth and change.

F. Representative System Development Methodologies

- Architected Rapid Application Development (Architected RAD)
- Dynamic Systems Development Methodology (DSDM)
- *Joint Application Development (JAD)*
- Information Engineering (IE)
- *Rapid Application Development (RAD)*
- *Rational Unified Process (RUP)*
- *Structured Analysis and Design*
- eXtreme Programming (XP)

The PIECES Problem-Solving Framework

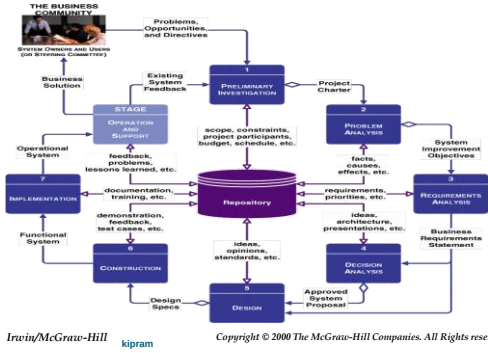
P	the need to improve performance
I	the need to improve information (and data)
E	the need to improve economics, control costs, or increase profits
C	the need to improve control or security
E	the need to improve efficiency of people and processes
S	the need to improve service to customers, suppliers, partners, employees, etc.

Cross Life Cycle Activities

Cross life cycle activities are activities that overlap many or all phases of the methodology.

- Fact-finding
- Documentation and presentation
- Feasibility analysis
- Process and project management

Sharing Knowledge via a Repository



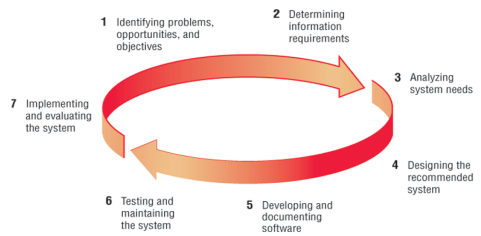
Approaches to Development

- Prototyping
  - Building a scaled-down working version of the system
  - Advantages:
    - Users are involved in design
    - Captures requirements in concrete form
- Rapid Application Development (RAD)
  - Utilizes prototyping to delay producing system design until after user requirements are clear

- Joint Application Design (JAD)
  - Users, Managers and Analysts work together for several days
  - System requirements are reviewed
  - Structured meetings
- Computer-aided software engineering (CASE) tools
  - Facilitate creation of a central repository for system descriptions and specifications

G. SIKLUS PENGEMBANGAN SYSTEM INFORMASI (SDLC)

Figure 1.3 The seven phases of the systems development life cycle.



Irwin/McGraw-Hill

H. Systems Analysis

4 Yang harus dimilikinya :

- Systems Thinking
- Organizational Knowledge
- Problem Identification
- Problem Analyzing and Solving

1.Systems Thinking

- System
  - Bagaimana prosedure hubungan business nya yg digunakan dalam bekerja setiap unit atau keseluruhannya
  - Mempunyai 9 characteristics
  - system yang berjalan termasuk environment
  - A boundary separates a system from its environment

- Characteristics of a System
  - Components
  - Interrelated Components
  - Boundary
  - Purpose
  - Environment
  - Interfaces
  - Input
  - Output
  - Constraints

- Important System Concepts
  - Decomposition
    - The process of breaking down a system into smaller components
    - Allows the systems analyst to:
      - Break a system into small, manageable subsystems
      - Focus on one area at a time
      - Concentrate on component pertinent to one group of users
      - Build different components at independent times

- Important System Concepts (Continued)
  - Modularity
    - Process of dividing a system into modules of a relatively uniform size
    - Modules simplify system design
  - Coupling
    - Subsystems that are dependent upon each other are coupled
  - Cohesion
    - Extent to which a subsystem performs a single function

- Important System Concepts (Continued)
  - Logical System Description
    - Portrays the purpose and function of the system
    - Does not tie the description to a specific physical implementation
  - Physical System Description
    - Focuses on how the system will be materially constructed

- Benefits
  - Identification of a system leads to abstraction
  - From abstraction you can think about essential characteristics of specific system
  - Abstraction allows analyst to gain insights into specific system, to question assumptions, provide documentation and manipulate the system without disrupting the real situation

- Applying Systems Thinking to Information Systems
  - Information systems are subsystems in larger organizational systems
  - Data flow diagrams represent information systems as systems
    - Inputs
    - Outputs
    - System boundaries
    - Environment
    - Subsystems
    - Interrelationships

2.Organizational Knowledge

- Understanding of how organizations work
- Knowledge of specific functions and procedures of organization and department
- How work officially gets done
- Internal policies
- Competitive and Regulatory Environment
- Organizational Strategies and Tactics

3. Problem Identification

- Problem: Difference between an existing situation and a desired situation
- Identification is process of defining differences
- Differences are defined by comparing the current situation to the output of a model that predicts what the output should be

4. Problem Analyzing and Solving

- Four Phases
  - Intelligence
    - All relevant information is collected
  - Design
    - Alternatives are formulated
  - Choice
    - Best alternative solution is chosen
  - Implementation
    - Solution is put into practice

Technical Skills for Systems Analysis

- Constant re-education is necessary as technology changes rapidly
- Activities to keep skills up-to-date
  - Trade publications
  - Professional societies
  - Attend classes or teach at a local college
  - Attend courses sponsored by organization
  - Conferences and trade shows
  - Browse Websites
  - Participate in new groups and conferences

Technical Skills for Systems Analysis

- Understanding of a wide variety of technologies is required
  - Microcomputers, workstations, minicomputers and mainframe computers
  - Programming languages
  - Operating systems
  - Database and file management systems
  - Data communication standards
  - Systems development tools and environments
  - Web development languages and tools
  - Decision support system generators

Management Skills for Systems Analysis

- Four categories
  - Resource Management
  - Project Management
  - Risk Management
  - Change Management

Resource Management

- Systems analyst needs to know how to get the most out of the resources of an organization, including team members
- Includes the following capabilities
  - Predicting resource usage
  - Tracking resource consumption
  - Effective use of resources
  - Evaluation of resource quality
  - Securing resources from abusive use
  - Relinquishing resources when no longer needed

Project Management

- Two Goals
  - Prevent projects from coming in late
  - Prevent projects from going over budget
- Assists management in keeping track of project's progress
- Consists of several steps
  - Decomposing project into independent tasks
  - Determining relationships between tasks
  - Assigning resources and personnel to tasks

Risk Management

- Ability to anticipate what might go wrong in a project
- Minimize risk and/or minimize damage that might result
- Placement of resources
- Prioritization of activities to achieve greatest gain

Change Management

- Ability to assist people in making transition to new system
- Ability to deal with technical issues related to change
  - Obsolescence
  - Reusability

Interpersonal Skills for Systems Analysis

- Mastery of interpersonal skills is paramount to success as a Systems Analyst
- Four types of skills:
  - Communication skills
  - Working alone and with a team
  - Facilitating groups
  - Managing expectations

Communication Skills

- Effective communication helps to establish and maintain good working relationships with clients and colleagues
- Skills improve with experience
- Three types used by Systems Analyst
  - Interviewing and Listening
  - Questionnaires
  - Written and Oral Presentations

Interviewing and Listening

- Means to gather information about a project
- Listening to answers is just as important as asking questions
- Effective listening leads to understanding of problem and generates additional questions

Questionnaires

- Advantages:
  - Less costly than interviews
  - Results are less biased due to standardization
- Disadvantages
  - Less effective than interviews due to lack of follow-up

Written and Oral Presentations

- Used to document progress of project and communicate this to others
- Communication takes several forms:
  - Meeting agenda
  - Meeting minutes
  - Interview summaries
  - Project schedules and descriptions
  - Memoranda requesting information
  - Requests for proposals from vendors and contractors
  - Oral presentations

Steps to Improving Communication Skills

- Practice
  - Conduct a training class
  - Volunteer to speak
- Videotape presentation and do a self-appraisal of your skills
- Make use of college writing centers
- Take classes on business and technical writing

Working Alone and with a Team

- Working alone on aspects of project involves managing:
  - Time
  - Commitments
  - Deadlines
- Team work involves establishing standards of cooperation and coordination
- characteristics of a high-performance team

Facilitating Groups

- Involves guiding a group without being a part of the group
- Useful skill for sessions such as Joint Application Development (JAD)
- lists guidelines for running a successful meeting

2.53

Managing Expectations

- Managing expectations is directly related to successful system implementation
- Skills for successful expectation management
  - Understanding of technology and workflows
  - Ability to communicate a realistic picture of new system to users
  - Effective education of management and users throughout systems development life cycle

Systems Analysis as a Profession

- Standards have been established for education, training, certification and practice
- Several aspects:
  - Standards of Practice
  - Ethics
  - Career Paths

Standards of Practice

- Endorsed Development Methodology
  - Specific procedures and techniques to be used during development process
  - Promote consistency and reliability across all of an organization's development projects
- Approved Development Platforms
  - Organizations standardize around a specific platform, sometimes tied to development methodology

Standards of Practice

- Standardization of Roles
  - Roles are becoming better defined across organizations
- Development of a Common Language
  - Common programming languages
  - Common modeling languages, such as Unified Modeling Language (UML)

Ethics

- Professional Ethics
  - ACM Code of Ethics
- Business Ethics
  - Stockholder approach
    - Any action taken by a business is acceptable as long as it is legal and maximizes stockholder profit
  - Stakeholder approach
    - Any action that violates rights of stakeholder must be rejected
  - Social Contract approach
    - Any action that is deceptive, can dehumanize employees or that could discriminate is rejected

Career Paths

- Consulting
- Information Systems within a large corporation
- Software vendors
- Other opportunities outside of systems analysis

Modern Systems Analysis and Design

Seventh Edition

Jeffrey A. Hoffer
Joey F. George
Joseph S. Valacich

Chapter 4
Identifying and Selecting Systems Development Projects

Learning Objectives

- ✓ Describe the project identification and selection process.
- ✓ Describe corporate strategic planning and information systems planning process.
- ✓ Explain the relationship between corporate strategic planning and information systems planning.
- ✓ Describe how information systems planning can be used to assist in identifying and selecting systems development projects.

Learning Objectives (Cont.)

- ✓ Analyze information systems planning matrices to determine affinity between information systems and IS projects and to forecast the impact of IS projects on business objectives.
- ✓ Describe the three classes of Internet electronic commerce applications: business-to-consumer, business-to-employee, and business-to-business.

Identifying and Selecting Systems Development Projects

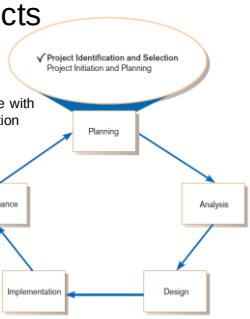


FIGURE 4-1
Systems development life cycle with project identification and selection highlighted

Three main steps:
1. Identifying potential development projects
2. Classifying and ranking IS development projects
3. Selecting IS development projects

The Process of Identifying and Selecting IS Development Projects

1. *Identifying potential development projects*
   - Identification from a stakeholder group
     - Each stakeholder group brings their own perspective and motivation to the IS decision.

The Process of Identifying and Selecting IS Development Projects (Cont.)

- *Top-down source* are projects identified by top management or by a diverse steering committee.
- *Bottom-up source* are project initiatives stemming from managers, business units, or the development group.
- The process varies substantially across organizations.

The Process of Identifying and Selecting IS Development Projects (Cont.)

TABLE 4-1 Characteristics of Alternative Methods for Making Information Systems Identification and Selection Decisions

Selection Method	Characteristics
Top Management	Greater strategic focus Largest project size Longest project duration Enterprise-wide consideration
Steering Committee	Cross-functional focus Greater organizational change Formal cost-benefit analysis Larger and riskier projects
Functional Area	Narrow, nonstrategic focus Faster development Fewer users, management layers, and business functions involved
Development Group	Integration with existing systems focus Fewer development delays Less concern with cost-benefit analysis

(*Source*: Based on McKeen, Guimaraes, and Wetherbe, 1994; GAO, 2000.)

The Process of Identifying and Selecting IS Development Projects (Cont.)

2. *Classifying and ranking IS development projects*
   - Using value chain analysis or other evaluation criteria
     - **Value chain analysis**: Analyzing an organization's activities to determine where value is added to products and/or services and the costs incurred for doing so; usually also includes a comparison with the activities, added value, and costs of other organizations for the purpose of making improvements in the organization's operations and performance

The Process of Identifying and Selecting IS Development Projects (Cont.)

FIGURE 4-2
Organizations can be thought of as a value chain, transforming raw materials into products for customers.

The Process of Identifying and Selecting IS Development Projects (Cont.)

TABLE 4-2 Possible Evaluation Criteria When Classifying and Ranking Projects

Evaluation Criteria	Description
Value Chain Analysis	Extent to which activities add value and costs when developing products and/or services
Strategic Alignment	Extent to which the project is viewed as helping the organization achieve its strategic objectives and long-term goals
Potential Benefits	Extent to which the project is viewed as improving profits, customer service, and so forth, and the duration of these benefits
Resource Availability	Amount and type of resources the project requires and their availability
Project Size/Duration	Number of individuals and the length of time needed to complete the project
Technical Difficulty/Risks	Level of technical difficulty to successfully complete the project within given time and resource constraints

The Process of Identifying and Selecting IS Development Projects (Cont.)

3. *Selecting IS development projects*
   - Based on various factors
   - Both short- and long-term projects considered
   - Most likely to achieve business objectives selected
   - A very important and ongoing activity

The Process of Identifying and Selecting IS Development Projects (Cont.)

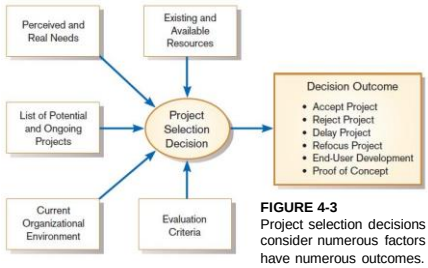


FIGURE 4-3
Project selection decisions must consider numerous factors and can have numerous outcomes.

The Process of Identifying and Selecting IS Development Projects (Cont.)

- One method for deciding among different projects or alternative designs:
  - For each requirement or constraint: Score = weight X rating
  - Each alternative: sum scores across requirements/constraints
  - Alternative with highest score wins

The Process of Identifying and Selecting IS Development Projects (Cont.)

Criteria	Weight	Alternative A		Alternative B		Alternative C	
		Rating	Score	Rating	Score	Rating	Score
Requirements							
Real-time data entry	18	5	90	5	90	5	90
Automatic reorder	18	1	18	5	90	5	90
Real-time data query	14	1	14	5	70	5	70
	50		122		250		250
Constraints							
Developer costs	15	4	60	5	75	3	45
Hardware costs	15	4	60	4	60	3	45
Operating costs	15	5	75	1	15	5	75
Ease of training	5	5	25	3	15	3	15
	50		220		165		180
Total	100		342		415		430

FIGURE 4-4 Alternative projects and system design decisions can be assisted using weighted multicriteria analysis.

Deliverables and Outcomes

- Primary deliverable from the first part of the planning phase is a schedule of specific IS development projects.
- Outcome of the next part of the planning phase—project initiation and planning—is the assurance that careful consideration was given to project selection and each project can help the organization reach its goals.

Deliverables and Outcomes (Cont.)

- **Incremental commitment**: a strategy in systems analysis and design in which the project is reviewed after each phase and continuation of the project is rejustified

Deliverables and Outcomes (Cont.)

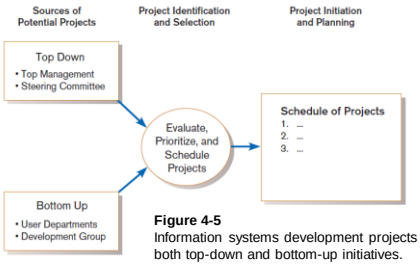


Figure 4-5
Information systems development projects come from both top-down and bottom-up initiatives.

Corporate and Information Systems Planning

- To benefit from a planning-based approach for identifying and selecting projects, an organization must:
  - Analyze its information needs thoroughly.
  - Plan its projects carefully.

Reasons for Importance of Improved Planning

- Increasing cost of information systems (40% of organizational expense)
- Lack of cross-organizational applications and systems
- Systems don't address critical strategic problems
- Too much data redundancy, lack of data quality
- High system maintenance costs
- Long application backlogs

Corporate Strategic Planning

- Ongoing process that defines mission, objectives, and strategies of an organization
- Corporate strategy involves:
  - Mission statement
  - Objective statements
  - Description of competitive strategy

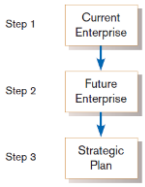


Figure 4-6
Corporate strategic planning is a three step Process.

Corporate Strategic Planning (Cont.)

- **Mission statement**: a statement that makes it clear what business a company is in

Pine Valley Furniture
Corporate Mission Statement

We are in the business of designing, fabricating, and selling to retail stores high-quality wood furniture for household, office, and institutional use. We value quality in our products and in our relationships with customers and suppliers. We consider our employees our most critical resource.

Figure 4-7
Mission statement (Pine Valley Furniture)

Corporate Strategic Planning (Cont.)

- **Objective statement**: a series of statements that express an organization's qualitative and quantitative goals for reaching a desired future position

Corporate Strategic Planning (Cont.)

Pine Valley Furniture
Statement of Objectives

1. PVF will strive to increase market share and profitability (prime objective).
2. PVF will be considered a market leader in customer service.
3. PVF will be innovative in the use of technology to help bring new products to market faster than our competition.
4. PVF will employ the fewest number of the highest-quality people necessary to accomplish our prime objective.
5. PVF will create an environment that values diversity in gender, race, values, and culture among employees, suppliers, and customers.

FIGURE 4-8 Statement of Corporate Objectives (Pine Valley Furniture)

Corporate Strategic Planning (Cont.)

- **Competitive strategy**: the method by which an organization attempts to achieve its mission and objectives
- Main types:
  - Low-cost producer
  - Product differentiation
  - Product focus or niche

Corporate Strategic Planning (Cont.)

TABLE 4-3 Generic Competitive Strategies

Strategy	Description
Low-Cost Producer	This strategy reflects competing in an industry on the basis of product or service cost to the consumer. For example, in the automobile industry, the South Korean–produced Hyundai is a product line that competes on the basis of low cost.
Product Differentiation	This competitive strategy reflects capitalizing on a key product criterion requested by the market (for example, high quality, style, performance, roominess). In the automobile industry, many manufacturers are trying to differentiate their products on the basis of quality (e.g., "At Ford, quality is job one.").
Product Focus or Niche	This strategy is similar to both the low-cost and differentiation strategies but with a much narrower market focus. For example, a niche market in the automobile industry is the convertible sports car market. Within this market, some manufacturers may employ a low-cost strategy and others may employ a differentiation strategy based on performance or style.

(*Source:* Based on The Free Press, a Division of Simon & Schuster Adult Publishing Group, from Porter, 1980. Copyright © 1980, 1998 by The Free Press. All rights reserved.)

Information Systems Planning (ISP)

- An orderly means of assessing the information needs of an organization and defining the systems, databases, and technologies that will best meet those needs
- ISP must be done in accordance with the organization's mission, objectives, and competitive strategy.

Information Systems Planning (Cont.)

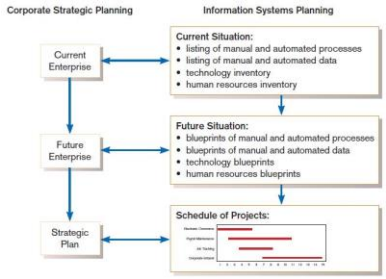


FIGURE 4-10
Parallel activities of corporate strategic planning and information systems planning

Information Systems Planning (Cont.)

- **Top-down planning** attempts to gain a broad understanding of information system needs of the entire organization and offers:
  - Broader perspective.
  - Improved integration.
  - Improved management support.
  - Better understanding.

Information Systems Planning (Cont.)

- **Bottom-up planning** identifies IS development projects based on solving specific operational business problems or taking advantage of specific opportunities.
  - Can be faster and less costly, so may be beneficial in certain circumstances.

Information Systems Planning (Cont.)

FIGURE 4-11
Information systems planning information (Pine Valley Furniture)

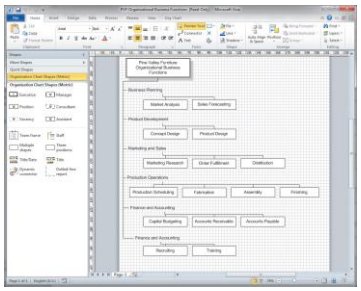
Information Systems Planning (Cont.)

- **Functional Decomposition**: breaking high-level abstract information into smaller units for more detailed planning

Information Systems Planning (Cont.)

FIGURE 4-12
Functional decomposition of information systems planning information (Pine Valley Furniture)

(*Source:* Microsoft Corporation.)



Information Systems Planning (Cont.)

- IS planning matrices describe relationships between pairs of organizational elements (location, function, business unit, objective, process, data, information system).

Types of Planning Matrices

- Location-to-Function
- Location-to-Unit
- Unit-to-Function
- Function-to-Objective
- Function-to-Process
- Function-to-Data Entity
- Process-to-Data Entity
- Process-to-Information System
- Data Entity-to-Information System
- Information System-to-Objective

Information Systems Planning (Cont.)

	Customer	Product	Vendor	Raw Material	Order	Work Center	Equipment	Employees	Invoice	Work Order	...
Marketing and Sales											
Marketing Research	✗	✗									
Order Fulfillment	✗	✗			✗				✗		
Distribution	✗	✗									
Production Operation											
Production Scheduling						✗	✗	✗		✗	
Fabrication						✗	✗	✗		✗	
Assembly						✗	✗	✗		✗	
Finishing						✗	✗	✗		✗	
Finance and Accounting											
Capital Budgeting					✗	✗	✗				
Accounts Receivable	✗	✗	✗	✗	✗				✗		
Accounts Payable											
...											

FIGURE 4-13
Data Entity-to-Function matrix (Pine Valley Furniture)

IS Plan Components

- Organizational Mission, Objectives, and Strategy
  - Brief description of mission, objectives, and strategy of the organization
- Information Inventory
  - Summary of processes, functions, data entities, and information needs of the enterprise

IS Plan Components (Cont.)

- Mission and Objectives of IS
  - Primary role IS will play in the organization to transform enterprise from current to future state
- Constraints on IS Development
  - Limitations imposed by technology and current levels of financial, technical, and personnel resources

IS Plan Components (Cont.)

- Systems Needs and IS Strategy
  - Summarize overall information systems needs in the company and set long-term (2-5 year) strategies for filling the needs
- Short Term Plan
  - Detailed inventory of present projects and systems and detailed plan for the current year

IS Plan Components (Cont.)

- Conclusions
  - Unknown but likely events that can affect the plan, presently known business change elements and their impact on the plan

Information Systems (IS) Plan

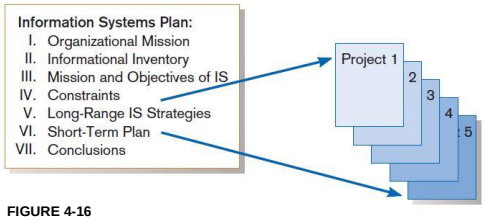


FIGURE 4-16
Systems development projects flow from the information systems plan.

Electronic Commerce Applications and Internet Basics

- **Internet**: a large worldwide network of networks that use a common protocol to communicate with each other
- **Electronic Commerce** (**EC**): Internet-based communication to support day-to-day business activities
- **Business-to-consumer (B2C)**: electronic commerce between businesses and consumers

Electronic Commerce Applications and Internet Basics (Cont.)

- **Business-to-business (B2B)**: electronic commerce between business partners, such as suppliers and intermediaries
- **Business-to-employee (B2E)**: electronic commerce between businesses and their employees
- **Electronic data interchange (EDI)**: the use of telecommunications technologies to directly transfer business documents between organizations

Electronic Commerce Applications and Internet Basics (Cont.)

TABLE 4-5 Unknowns That Must Be Dealt with When Designing and Building Internet Applications

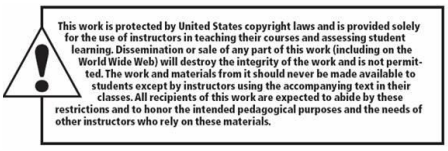
User	• Concern: Who is the user? • Example: Where is the user located? What is the user's expertise, education, or expectations?
Connection Speed	• Concern: What is the speed of the connection and what information can be effectively displayed? • Example: Modem, Cable Modem, DSL, Satellite, Broadband, Cellular
Access Method	• Concern: What is the method of accessing the net? • Example: Web browser, Personal Digital Assistant (PDA), Web-enabled Cellular Phone, Tablet, Web-enabled Television

Summary

- In this chapter you learned how to:
  - ✓ Describe the project identification and selection process.
  - ✓ Describe corporate strategic planning and information systems planning.
  - ✓ Explain the relationship between corporate strategic planning and IS planning.

Summary (Cont.)

- ✓ Describe how IS planning can assist in system development project identification and selection.
- ✓ Analyze IS planning matrices.
- ✓ Describe three classes of E-Commerce applications.



Modern Systems Analysis and Design

Seventh Edition

Jeffrey A. Hoffer
Joey F. George
Joseph S. Valacich

Chapter 6
Determining System Requirements

Learning Objectives

- ✓ Describe options for designing and conducting interviews and develop a plan for conducting an interview to determine system requirements.
- ✓ Explain the advantages and pitfalls of observing workers and analyzing business documents to determine system requirements.
- ✓ Explain how computing can provide support for requirements determination.
- ✓ Participate in and help plan a Joint Application Design session.

Learning Objectives (Cont.)

- ✓ Use prototyping during requirements determination.
- ✓ Describe contemporary approaches to requirements determination.
- ✓ Understand how requirements determination techniques apply to the development of electronic commerce applications.

Performing Requirements Determination

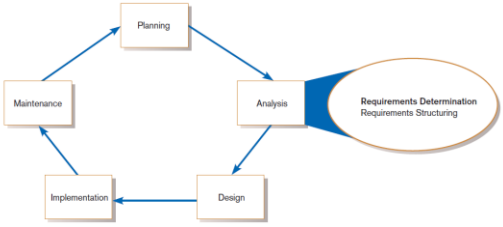


FIGURE 6-1
Systems development life cycle with analysis phase highlighted

The Process of Determining Requirements

- Good Systems Analyst Characteristics:
  - Impertinence—question everything
  - Impartiality—consider all issues to find the best organizational solution
  - Relax constraints—assume anything is possible
  - Attention to details—every fact must fit
  - Reframing—challenge yourself to new ways

Deliverables and Outcomes

- Deliverables for Requirements Determination:
  - From interviews and observations
    - interview transcripts, observation notes, meeting minutes
  - From existing written documents
    - mission and strategy statements, business forms, procedure manuals, job descriptions, training manuals, system documentation, flowcharts
  - From computerized sources
    - Joint Application Design session results, CASE repositories, reports from existing systems, displays and reports from system prototype

Traditional Methods for Determining Requirements

- Interviewing individuals
- Interviewing groups
- Observing workers
- Studying business documents

Interviewing and Listening

- One of the primary ways analysts gather information about an information systems project
- An **interview guide** is a document for developing, planning and conducting an interview.

Guidelines for Effective Interviewing

- Plan the interview.
  - Prepare interviewee: appointment, priming questions.
  - Prepare agenda, checklist, questions.
- Listen carefully and take notes (tape record if permitted).
- Review notes within 48 hours.
- Be neutral.
- Seek diverse views.

Interviewing and Listening (Cont.)

Interview Outline	
Interviewee: *Name of person being interviewed*	Interviewer: *Name of person leading interview*
Location/Medium: *Office, conference room, or phone number*	Appointment Date: Start Time: End Time:
Objectives: *What data to collect* *On what to gain agreement* *What areas to explore*	Reminders: *Background/experience of interviewee* *Known opinions of interviewee*
Agenda: Introduction Background on Project Overview of Interview Topics to Be Covered Permission to Record Topic 1 Questions Topic 2 Questions ... Summary of Major Points Questions from Interviewee Closing	Approximate Time: 1 minute 2 minutes 1 minute 5 minutes 7 minutes ... 2 minutes 5 minutes 1 minute

FIGURE 6-2 Typical interview guide

Interviewing and Listening (Cont.)

General Observations: *Interviewee seemed busy probably need to call in a few days for follow-up questions because he gave only short answers. PC was turned off–probably not a regular PC user.*	
Unresolved Issues, Topics Not Covered: *He needs to look up sales figures from 1999. He raised the issue of how to handle returned goods, but we did not have time to discuss.*	
Interviewee:	Date:
Questions:	Notes:
When to ask question, if conditional *Question: 1* *Have you used the current sales tracking system? If so, how often?*	*Answer* *Yes, I ask for a report on my product line weekly.* Observations *Seemed anxious–may be overestimating usage frequency.*
If yes, go to Question 2	
Question: 2 *What do you like least about the system?*	*Answer* *Sales are shown in units, not dollars.* Observations *System can show sales in dollars, but user does not know this.*

FIGURE 6-2 Typical interview guide (cont.)

Choosing Interview Questions

- Each question in an interview guide can include both verbal and non-verbal information.
  - **Open-ended questions**: questions that have no prespecified answers
  - **Closed-ended questions**: questions that ask those responding to choose from among a set of specified responses

Interviewing Guidelines

- Don't phrase a question in a way that implies a right or wrong answer.
- Listen very carefully.
- Type interview notes within 48 hours after the interview.
- Don't set expectations about the new system unless you know these will be deliverables.
- Seek a variety of perspectives from the interviews.

Interviewing Groups

- Drawbacks to individual interviews:
  - Contradictions and inconsistencies between interviewees
  - Follow-up discussions are time consuming
  - New interviews may reveal new questions that require additional interviews with those interviewed earlier

Interviewing Groups (Cont.)

- Interviewing several key people together
  - Advantages
    - More effective use of time
    - Can hear agreements and disagreements at once
    - Opportunity for synergies
  - Disadvantages
    - More difficult to schedule than individual interviews

Nominal Group Technique (NGT)

- A facilitated process that supports idea generation by groups
- Process
  - Members come together as a group, but initially work separately.
  - Each person writes ideas.
  - Facilitator reads ideas out loud, and they are written on a blackboard or flipchart.
  - Group openly discusses the ideas for clarification.
  - Ideas are prioritized, combined, selected, reduced.
- Used to complement group meetings or as part of JAD effort

Directly Observing Users

- **Direct Observation**
  - Watching users do their jobs
  - Used to obtain more firsthand and objective measures of employee interaction with information systems
  - Can cause people to change their normal operating behavior
  - Time-consuming and limited time to observe

Analyzing Procedures and Other Documents

- **Document Analysis**
  - Review of existing business documents
  - Can give a historical and "formal" view of system requirements

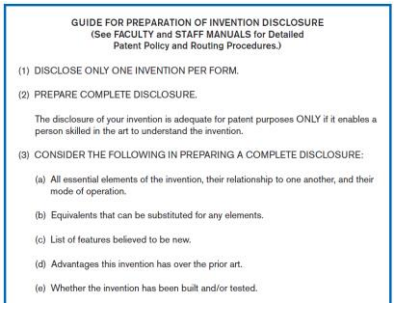
Analyzing Procedures and Other Documents (Cont.)

- Types of information to be discovered:
  - Problems with existing system
  - Opportunity to meet new need
  - Organizational direction
  - Names of key individuals
  - Values of organization
  - Special information processing circumstances
  - Reasons for current system design
  - Rules for processing data

Analyzing Procedures and Other Documents (Cont.)

- **Useful document: Written work procedure**
  - For an individual or work group
  - Describes how a particular job or task is performed
  - Includes data and information used and created in the process

Analyzing Procedures (Cont.)



GUIDE FOR PREPARATION OF INVENTION DISCLOSURE
(See FACULTY and STAFF MANUALS for Detailed Patent Policy and Routing Procedures.)

(1) DISCLOSE ONLY ONE INVENTION PER FORM.

(2) PREPARE COMPLETE DISCLOSURE.

The disclosure of your invention is adequate for patent purposes ONLY if it enables a person skilled in the art to understand the invention.

(3) CONSIDER THE FOLLOWING IN PREPARING A COMPLETE DISCLOSURE:

(a) All essential elements of the invention, their relationship to one another, and their mode of operation.

(b) Equivalents that can be substituted for any elements.

(c) List of features believed to be new.

(d) Advantages this invention has over the prior art.

(e) Whether the invention has been built and/or tested.

FIGURE 6-3 Example of a procedure

Analyzing Procedures (Cont.)

(4) PROVIDE APPROPRIATE ADDITIONAL MATERIAL.

Drawings and descriptive material should be provided as needed to clarify the disclosure. Each page of this material must be signed and dated by each inventor and properly witnessed. A copy of any current and/or planned publication relating to the invention should be included.

(5) INDICATE PRIOR KNOWLEDGE AND INFORMATION.

Pertinent publications, patents or previous devices, and related research or engineering activities should be identified.

(6) HAVE DISCLOSURE WITNESSED.

Persons other than coinventors should serve as witnesses and should sign each sheet of the disclosure only after reading and understanding the disclosure.

(7) FORWARD ORIGINAL PLUS ONE COPY (two copies if supported by grant/contract) TO VICE PRESIDENT FOR RESEARCH VIA DEPARTMENT HEAD AND DEAN.

FIGURE 6-3 Example of a procedure (cont.)

Analyzing Procedures and Other Documents (Cont.)

- Potential Problems with Procedure Documents:
  - May involve duplication of effort
  - May have missing procedures
  - May be out of date
  - May contradict information obtained through interviews

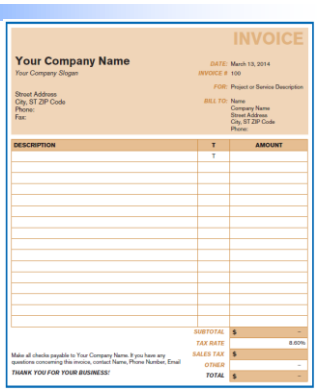
Analyzing Procedures and Other Documents (Cont.)

- **Formal Systems**: the official way a system works as described in organizational documentation (i.e. work procedure)
- **Informal Systems**: the way a system actually works (i.e. interviews, observations)

Analyzing Procedures and Other Documents (Cont.)

- **Useful document: Business form**
  - Used for all types of business functions
  - Explicitly indicates what data flow in and out of a system and data necessary for the system to function
  - Gives crucial information about the nature of the organization

Analyzing Procedures and Other Documents (Cont.)


INVOICE

Your Company Name

Your Company Slogan

DATE: March 13, 2014
INVOICE #: 100
FOR: Project or Service Description

Street Address
City, ST ZIP Code
Phone:
Fax:

BILL TO: Name
Company Name
Street Address
City, ST ZIP Code
Phone:

DESCRIPTION	T	AMOUNT
	T	

SUBTOTAL $ -
TAX RATE 8.60%
SALES TAX $ -
OTHER -
TOTAL $ -

Make all checks payable to Your Company Name. If you have any questions concerning this invoice, contact Name, Phone Number, Email

THANK YOU FOR YOUR BUSINESS!

FIGURE 6-4
An invoice form from Microsoft Excel

Analyzing Procedures and Other Documents (Cont.)

- **Useful document: Report**
  - Primary output of current system
  - Enables you to work backwards from the report to the data needed to generate it
- **Useful document: Description of current information system**

Analyzing Procedures and Other Documents (Cont.)

TABLE 6-4 Comparison of Observation and Document Analysis

Characteristic	Observation	Document Analysis
Information Richness	High (many channels)	Low (passive) and old
Time Required	Can be extensive	Low to moderate
Expense	Can be high	Low to moderate
Chance for Follow-Up and Probing	Good: probing and clarification questions can be asked during or after observation	Limited: probing possible only if original author is available
Confidentiality	Observee is known to interviewer; observee may change behavior when observed	Depends on nature of document; does not change simply by being read
Involvement of Subject	Interviewees may or may not be involved and committed depending on whether they know if they are being observed	None, no clear commitment
Potential Audience	Limited numbers and limited time (snapshot) of each	Potentially biased by which documents were kept or because document was not created for this purpose

Contemporary Methods for Determining System Requirements

- **Joint Application Design (JAD)**
  - Brings together key users, managers, and systems analysts
  - Purpose: collect system requirements simultaneously from key people
  - Conducted off-site
- **Group Support Systems**
  - Facilitate sharing of ideas and voicing of opinions about system requirements

Contemporary Methods for Determining System Requirements (Cont.)

- **CASE tools**
  - Used to analyze existing systems
  - Help discover requirements to meet changing business conditions
- **System prototypes**
  - Iterative development process
  - Rudimentary working version of system is built
  - Refine understanding of system requirements in concrete terms

Joint Application Design (JAD)

- Intensive group-oriented requirements determination technique
- Team members meet in isolation for an extended period of time
- Highly focused
- Resource intensive
- Started by IBM in 1970s

JAD (Cont.)

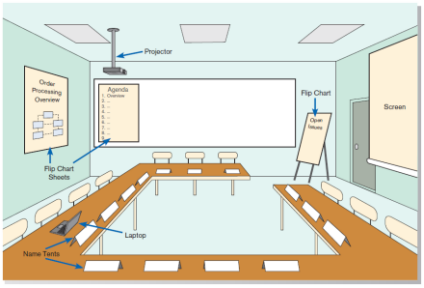


FIGURE 6-6 Illustration of the typical room layout for a JAD
Source: Based on Wood and Silver, 1995.

JAD (Cont.)

- JAD Participants:
  - **Session Leader**: facilitates group process
  - **Users:** active, speaking participants
  - **Managers**: active, speaking participants
  - **Sponsor**: high-level champion, limited participation
  - **Systems Analysts**: should mostly listen
  - **Scribe**: record session activities
  - **IS Staff**: should mostly listen

JAD (Cont.)

- End Result
  - Documentation detailing existing system
  - Features of proposed system

CASE Tools During JAD

- Upper CASE tools are used
- Enables analysts to enter system models directly into CASE during the JAD session
- Screen designs and prototyping can be done during JAD and shown to users

Using Prototyping During Requirements Determination

- Quickly converts requirements to working version of system
- Once the user sees requirements converted to system, will ask for modifications or will generate additional requests

Using Prototyping During Requirements Determination (Cont.)

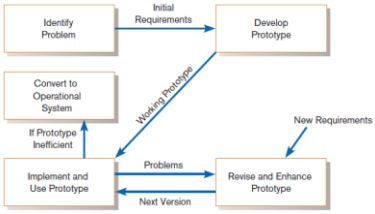


Figure 6-7
The prototyping methodology (*Source:* Based on "Prototyping: The New Paradigm for Systems Development," by J. D. Naumann and A. M. Jenkins, *MIS Quarterly* 6(3): 29–44.)

Using Prototyping During Requirements Determination (Cont.)

- Most useful when:
  - User requests are not clear.
  - Few users are involved in the system.
  - Designs are complex and require concrete form.
  - There is a history of communication problems between analysts and users.
  - Tools are readily available to build prototype.

Using Prototyping During Requirements Determination (Cont.)

- Drawbacks
  - Tendency to avoid formal documentation
  - Difficult to adapt to more general user audience
  - Sharing data with other systems is often not considered
  - Systems Development Life Cycle (SDLC) checks are often bypassed

Radical Methods for Determining System Requirements

- **Business Process Reengineering (BPR)**: search for and implementation of radical change in business processes to achieve breakthrough improvements in products and services

Radical Methods for Determining System Requirements (Cont.)

- Goals
  - Reorganize complete flow of data in major sections of an organization.
  - Eliminate unnecessary steps.
  - Combine steps.
  - Become more responsive to future change.

Identifying Processes to Reengineer

- Key business processes
  - Structured, measured set of activities designed to produce specific output for a particular customer or market
  - Focused on customers and outcome
  - Same techniques as requirements determination are used

Disruptive Technologies

- Information technologies must be applied to radically improve business processes.
- **Disruptive technologies** are technologies that enable the breaking of long-held business rules that inhibit organizations from making radical business changes.

Disruptive Technologies (Cont.)

TABLE 6-6 Long-Held Organizational Rules That Are Being Eliminated through Disruptive Technologies

Rule	Disruptive Technology
Information can appear in only one place at a time.	Distributed databases allow the sharing of information.
Businesses must choose between centralization and decentralization.	Advanced telecommunications networks can support dynamic organizational structures.
Managers must make all decisions.	Decision-support tools can aid nonmanagers.
Field personnel need offices where they can receive, store, retrieve, and transmit information.	Wireless data communication and portable computers provide a "virtual" office for workers.
The best contact with a potential buyer is personal contact.	Interactive communication technologies allow complex messaging capabilities.
You have to find out where things are.	Automatic identification and tracking technology knows where things are.
Plans get revised periodically.	High-performance computing can provide real-time updating.

Requirements Determination using Agile Methodologies

- **Continual user involvement**
  - Replace traditional SDLC waterfall with iterative analyze–design–code–test cycle
- **Agile usage-centered design**
  - Focuses on user goals, roles, and tasks
- **The Planning Game**
  - Based on eXtreme programming
  - Exploration, steering, commitment

Continual User Involvement

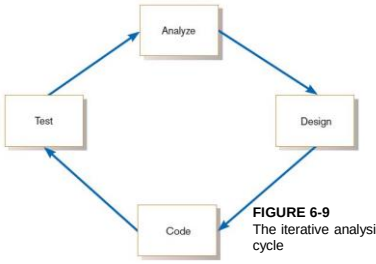


FIGURE 6-9
The iterative analysis–design–code–test cycle

Agile Usage-Centered Design Steps

- Gather group of programmers, analysts, users, testers, facilitator.
- Document complaints of current system.
- Determine important user roles.
- Determine, prioritize, and describe tasks for each user role.
- Group similar tasks into interaction contexts.
- Associate each interaction context with a user interface for the system, and prototype the interaction context.
- Step through and modify the prototype.

The Planning Game from eXtreme Programming

FIGURE 6-10
eXtreme Programming's Planning Game

Electronic Commerce Applications: Determining System Requirements

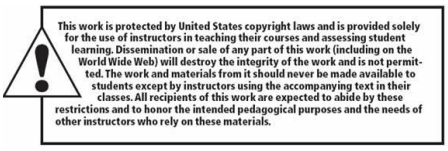
- Determining system requirements for Pine Valley furniture's WebStore
  - System layout and navigation characteristics
  - WebStore and site management system capabilities
  - Customer and inventory information
  - System prototype evolution

Summary

- In this chapter you learned how to:
  - ✓ Describe interviewing options and develop interview plan.
  - ✓ Explain advantages and pitfalls of worker observation and document analysis.
  - ✓ Explain how computing can support requirements determination.
  - ✓ Participate in and help plan Joint Application Design sessions.

Summary (Cont.)

- ✓ Use prototyping during requirements determination.
- ✓ Describe contemporary approaches to requirements determination.
- ✓ Understand how requirements determination techniques apply to the development of electronic commerce applications.



Modern Systems Analysis and Design

Seventh Edition

Jeffrey A. Hoffer
Joey F. George
Joseph S. Valacich

Chapter 7
Structuring System Process Requirements

Learning Objectives

- ✓ Understand the logical modeling of processes by studying examples of data flow diagrams (DFDs).
- ✓ Draw data flow diagrams following specific rules and guidelines that lead to accurate and well-structured process models.
- ✓ Decompose data flow diagrams into lower-level diagrams.
- ✓ Balance higher-level and lower-level data flow diagrams.

Learning Objectives (Cont.)

- ✓ Use data flow diagrams as a tool to support the analysis of information systems.
- ✓ Discuss process modeling for electronic commerce applications.
- ✓ Use decision tables to represent the logic of choice in conditional statements.

Process Modeling

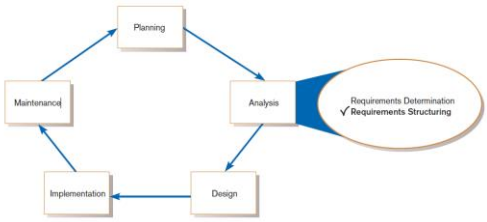


FIGURE 7-1
Systems development life cycle with the analysis phase highlighted

Process Modeling (Cont.)

- Graphically represent the processes that capture, manipulate, store, and distribute data between a system and its environment and among system components.
- Utilize information gathered during requirements determination.
- Model processes and data structures.

Deliverables and Outcomes

- Context data flow diagram (DFD)
  - Scope of system
- DFDs of current physical system
  - Adequate detail only
- DFDs of current logical system
  - Enables analysts to understand current system

Deliverables and Outcomes (Cont.)

- DFDs of new logical system
  - Technology independent
  - Show data flows, structure, and functional requirements of new system
- Thorough description of each DFD component

Data Flow Diagramming Mechanics

- Represent both physical and logical information systems
- Only four symbols are used
- Useful for depicting purely logical information flows
- DFDs that detail physical systems differ from system flowcharts which depict details of physical computing equipment.

Definitions and Symbols

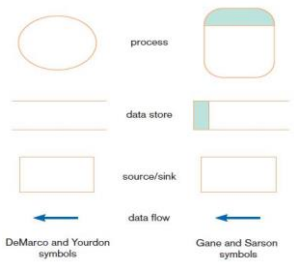


FIGURE 7-2
Comparison of DeMarco and Yourdon and Gane and Sarson DFD symbol sets

Definitions and Symbols (Cont.)

- **Process**: work or actions performed on data (inside the system)
- **Data store**: data at rest (inside the system)

Definitions and Symbols (Cont.)

- **Source/sink**: external entity that is the origin or destination of data (outside the system)
- **Data flow**: arrows depicting movement of data

Developing DFDs

- **Context diagram** is an overview of an organizational system that shows:
  - the system boundaries.
  - external entities that interact with the system.
  - major information flows between the entities and the system.
- Note: only one process symbol, and no data stores shown

Context Diagram

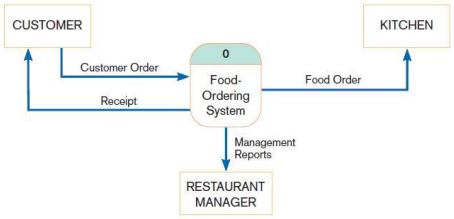


FIGURE 7-4
Context diagram of Hoosier Burger's food-ordering system

Developing DFDs (Cont.)

- **Level-0 diagram** is a data flow diagram that represents a system's major processes, data flows, and data stores at a high level of detail.
  - Processes are labeled 1.0, 2.0, etc. These will be decomposed into more primitive (lower-level) DFDs.

Level-0 Diagram

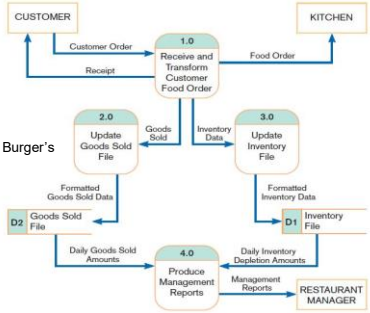


FIGURE 7-5
Level-0 DFD of Hoosier Burger's food-ordering system

Data Flow Diagramming Rules

- There are two DFD guidelines that apply:
  - *The inputs to a process are different from the outputs of that process.*
    - Processes purpose is to transform inputs into outputs.
  - *Objects on a DFD have unique names.*
    - Every process has a unique name.

Data Flow Diagramming Rules (Cont.)

TABLE 7-2 Rules Governing Data Flow Diagramming

Process:

A. No process can have only outputs. It is making data from nothing (a miracle). If an object has only outputs, then it must be a source.
B. No process can have only inputs (a black hole). If an object has only inputs, then it must be a sink.
C. A process has a verb phrase label.

Data Store:

D. Data cannot move directly from one data store to another data store. Data must be moved by a process.
E. Data cannot move directly from an outside source to a data store. Data must be moved by a process that receives data from the source and places the data into the data store.
F. Data cannot move directly to an outside sink from a data store. Data must be moved by a process.
G. A data store has a noun phrase label.

Source/Sink:

H. Data cannot move directly from a source to a sink. It must be moved by a process if the data are of any concern to our system. Otherwise, the data flow is not shown on the DFD.
I. A source/sink has a noun phrase label.

Data Flow Diagramming Rules (Cont.)

TABLE 7-2 Rules Governing Data Flow Diagramming (cont.)

Data Flow:

J. A data flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The latter is usually indicated, however, by two separate arrows because these happen at different times.
K. A fork in a data flow means that exactly the same data goes from a common location to two or more different processes, data stores, or sources/sinks (this usually indicates different copies of the same data going to different locations).
L. A join in a data flow means that exactly the same data come from any of two or more different processes, data stores, or sources/sinks to a common location.
M. A data flow cannot go directly back to the same process it leaves. There must be at least one other process that handles the data flow, produces some other data flow, and returns the original data flow to the beginning process.
N. A data flow to a data store means update (delete or change).
O. A data flow from a data store means retrieve or use.
P. A data flow has a noun phrase label. More than one data flow noun phrase can appear on a single arrow as long as all of the flows on the same arrow move together as one package.

(Source: Adapted from Celko, 1987.)

Decomposition of DFDs

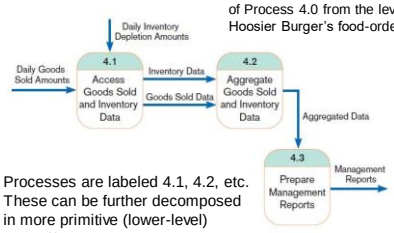
- **Functional decomposition** is an iterative process of breaking a system description down into finer and finer detail.
  - Creates a set of charts in which one process on a given chart is explained in greater detail on another chart.
  - Continues until no subprocess can logically be broken down any further.

Decomposition of DFDs (Cont.)

- *Primitive DFD* is the lowest level of a DFD.
- **Level-1 diagram** results from decomposition of Level-0 diagram.
- **Level-n diagram** is a DFD diagram that is the result of ***n*** nested decompositions from a process on a level-0 diagram.

Level-1 DFD

FIGURE 7-8
Level-1 diagram showing the decomposition of Process 4.0 from the level-0 diagram for Hoosier Burger's food-ordering system



Level-1 DFD shows the sub-processes of one of the processes in the Level-0 DFD.

This is a Level-1 DFD for Process 4.0.

Processes are labeled 4.1, 4.2, etc. These can be further decomposed in more primitive (lower-level) DFDs if necessary.

Level-*n* DFD

FIGURE 7-9
Level-2 diagram showing the decomposition of Process 4.3 from the level-1 diagram for Process 4.0 for Hoosier Burger's food-ordering system

Level-*n* DFD shows the sub-processes of one of the processes in the Level *n-1* DFD.

This is a Level-2 DFD for Process 4.3.

Processes are labeled 4.3.1, 4.3.2, etc. If this is the lowest level of the hierarchy, it is called a *primitive DFD.*

Balancing DFDs

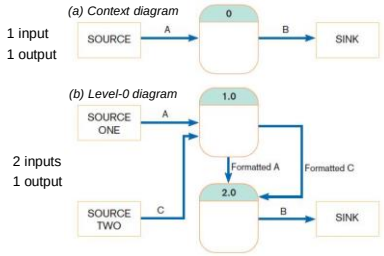
- **Conservation Principle**: conserve inputs and outputs to a process at the next level of decomposition
- **Balancing:** conservation of inputs and outputs to a data flow diagram process when that process is decomposed to a lower level

Balancing DFDs (Cont.)

- Balanced means:
  - Number of inputs to lower level DFD equals number of inputs to associated process of higher-level DFD
  - Number of outputs to lower level DFD equals number of outputs to associated process of higher-level DFD

Balancing DFDs (Cont.)

FIGURE 7-10 An unbalanced set of data flow diagrams



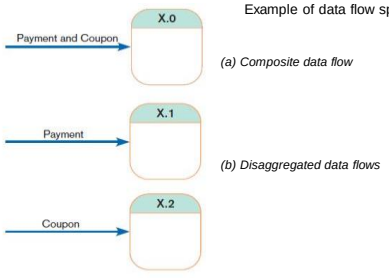
This is unbalanced because the process of the context diagram has only one input but the Level-0 diagram has two inputs.

Balancing DFDs (Cont.)

- **Data flow splitting** is when a composite data flow at a higher level is split and different parts go to different processes in the lower level DFD.
- The DFD remains balanced because the same data is involved, but split into two parts.

Balancing DFDs (Cont.)

FIGURE 7-11
Example of data flow splitting



Balancing DFDs: More DFD Rules

TABLE 7-3 Advanced Rules Governing Data Flow Diagramming

Q. A composite data flow on one level can be split into component data flows at the next level, but no new data can be added and all data in the composite must be accounted for in one or more subflows.

R. The inputs to a process must be sufficient to produce the outputs (including data placed in data stores) from the process. Thus, all outputs can be produced, and all data in inputs move somewhere: to another process or to a data store outside the process or onto a more detailed DFD showing a decomposition of that process.

S. At the lowest level of DFDs, new data flows may be added to represent data that are transmitted under exceptional conditions; these data flows typically represent error messages (e.g., "Customer not known; do you want to create a new customer?") or confirmation notices (e.g., "Do you want to delete this record?").

T. To avoid having data flow lines cross each other, you may repeat data stores or sources/sinks on a DFD. Use an additional symbol, like a double line on the middle vertical line of a data store symbol or a diagonal line in a corner of a sink/source square, to indicate a repeated symbol.

(*Source:* Adapted from Celko, 1987.)

Four Different Types of DFDs

- **Current Physical**
  - Process labels identify technology (people or systems) used to process the data.
  - Data flows and data stores identify actual name of the physical media.
- **Current Logical**
  - Physical aspects of system are removed as much as possible.
  - Current system is reduced to data and processes that transform them.

Four Different Types of DFDs (Cont.)

- **New Logical**
  - Includes additional functions
  - Obsolete functions are removed.
  - Inefficient data flows are reorganized.
- **New Physical**
  - Represents the physical implementation of the new system

Guidelines for Drawing DFDs

- **Completeness**
  - DFD must include all components necessary for system.
  - Each component must be fully described in the project dictionary or CASE repository.
- **Consistency**
  - The extent to which information contained on one level of a set of nested DFDs is also included on other levels

Guidelines for Drawing DFDs (Cont.)

- **Timing**
  - Time is not represented well on DFDs.
  - Best to draw DFDs as if the system has never started and will never stop.
- **Iterative Development**
  - Analyst should expect to redraw diagram several times before reaching the closest approximation to the system being modeled.

Guidelines for Drawing DFDs (Cont.)

- **Primitive DFDs**
  - Lowest logical level of decomposition
  - Decision has to be made when to stop decomposition

Guidelines for Drawing DFDs (Cont.)

- Rules for stopping decomposition
  - When each process has been reduced to a single decision, calculation or database operation
  - When each data store represents data about a single entity

Guidelines for Drawing DFDs (Cont.)

- Rules for stopping decomposition, cont.
  - When the system user does not care to see any more detail
  - When every data flow does not need to be split further to show that data are handled in various ways

Guidelines for Drawing DFDs (Cont.)

- Rules for stopping decomposition, cont.
  - When you believe that you have shown each business form or transaction, online display and report as a single data flow
  - When you believe that there is a separate process for each choice on all lowest-level menu options

Using DFDs as Analysis Tools

- **Gap Analysis** is the process of discovering discrepancies between two or more sets of data flow diagrams or discrepancies within a single DFD.
- Inefficiencies in a system can often be identified through DFDs.

Using DFDs in BPR

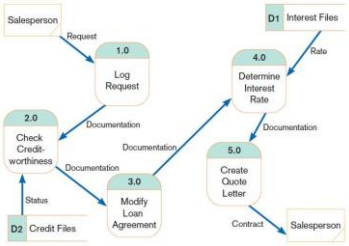


FIGURE 7-16
IBM Credit Corporation's primary work process before BPR (*Source: Based on Hammer and Champy,* 1993.)

Using DFDs in BPR (Cont.)

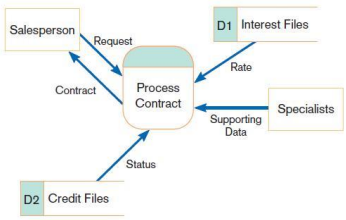


FIGURE 7-17
IBM Credit Corporation's primary work process after BPR (*Source: Based on Hammer and Champy,* 1993.)

Modeling Logic with Decision Tables

- **Decision table**: a matrix representation of the logic of a decision which specifies the possible conditions for the decision and the resulting actions
- Best used for complicated decision logic

Modeling Logic with Decision Tables (Cont.)

	Conditions/ Courses of Action	Rules					
		1	2	3	4	5	6
Condition Stubs	Employee type	S	H	S	H	S	H
	Hours worked	<40	<40	40	40	>40	>40
Action Stubs	Pay base salary	X		X		X	
	Calculate hourly wage		X		X		X
	Calculate overtime						X
	Produce absence report		X				

FIGURE 7-18
Complete decision table for payroll system example

Modeling Logic with Decision Tables (Cont.)

- **Condition stubs**: that part of a decision table that lists the conditions relevant to the decision
- **Action stubs**: that part of a decision table that lists the actions that result for a given set of conditions

Modeling Logic with Decision Tables (Cont.)

- **Rules**: that part of a decision table that specifies which actions are to be followed for a given set of conditions
- **Indifferent condition**: in a decision table, a condition whose value does not affect which actions are taken for two or more rules

Modeling Logic with Decision Tables (Cont.)

- Procedure for Creating Decision Tables
  - Name the condition and the values that each condition can assume.
  - Name all possible actions that can occur.
  - List all possible rules.
  - Define the actions for each rule.
  - Simplify the table.

Modeling Logic with Decision Tables (Cont.)

Conditions/ Courses of Action	Rules			
	1	2	3	4
Employee type	S	H	H	H
Hours worked	–	<40	40	>40
Pay base salary	X			
Calculate hourly wage		X	X	X
Calculate overtime				X
Produce Absence Report		X		

FIGURE 7-19
Reduced decision table for payroll system example

Electronic Commerce Application: Process Modeling using Data Flow Diagrams

- Process modeling for Pine Valley Furniture's WebStore
  - Completed JAD session
  - Began translating the WebStore system structure into data flow diagrams
    - Identified six high-level processes

Electronic Commerce Application: Process Modeling using Data Flow Diagrams (Cont.)

TABLE 7-4 System Structure of the WebStore and Corresponding Level-0 Processes

WebStore System	Processes
❑ Main Page	Information Display (minor/no processes)
• Product Line (Catalog)	1.0 Browse Catalog
✓ Desks	2.0 Select Item for Purchase
✓ Chairs	
✓ Tables	
✓ File Cabinets	
• Shopping Cart	3.0 Display Shopping Cart
• Checkout	4.0 Check Out Process Order
• Account Profile	5.0 Add/Modify Account Profile
• Order Status/History	6.0 Order Status Request
• Customer Comments	Information Display (minor/no processes)
❑ Company Information	
❑ Feedback	
❑ Contact Information	

Electronic Commerce Application: Process Modeling using Data Flow Diagrams

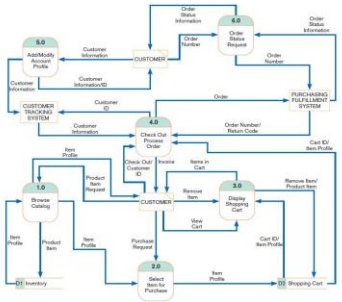


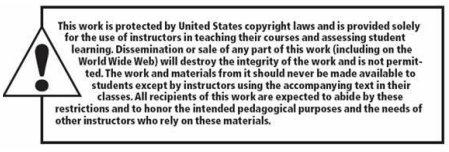
FIGURE 7-22 Level-0 data flow diagram for the WebStore

Summary

- In this chapter you learned how to:
  - ✓ Understand logical process modeling via data flow diagrams (DFDs).
  - ✓ Draw data flow diagrams of well-structured process models.
  - ✓ Decompose data flow diagrams into lower-level diagrams.

Summary (Cont.)

- ✓ Balance high-level and low-level data flow diagrams.
- ✓ Use data flow diagrams for analyzing information systems.
- ✓ Use decision tables to represent the logic of choice in conditional statements.



Modern Systems Analysis and Design

Seventh Edition

Jeffrey A. Hoffer
Joey F. George
Joseph S. Valacich

Chapter 7 Appendix A
Object-Oriented Analysis and Design: Use Cases

Learning Objectives

- ✓ Explain use cases and use case diagrams and how they can be used to model system functionality.
- ✓ Present the basic aspects of how to create written use cases.
- ✓ Discuss process modeling with use cases for electronic commerce application.

Use Cases

- A **use case** is a depiction of a system's behavior or functionality under various conditions as the system responds to requests from users.
- An **actor** is an external entity that interacts with the system.

Use Cases (Cont.)

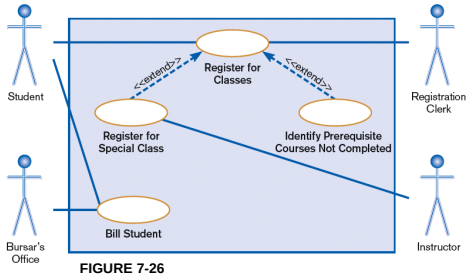


FIGURE 7-26
A use case diagram for a university registration system

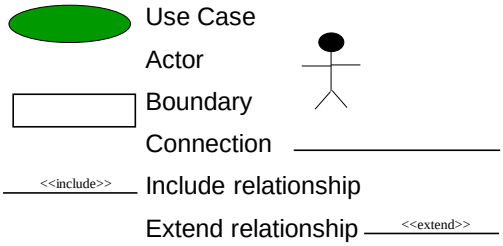
Use Cases (Cont.)

- Most actors represent user roles, but actors can also be external systems.
- An actor is a role, not a specific user; one user may play many roles, and an actor may represent many users.
- A use case model consists of actors and use cases.

Use Cases diagrams

- **Use case diagram:** a picture showing system behavior along with the key actors that interact with the system
- ***Abstract use case*** is when a use case is initiated by another use case.
- A use case represents complete functionality.

Definitions and Symbols



Definitions and Symbols (Cont.)

- **Actor** is a role, not an individual.
  - Involved with the functioning of the system at some basic level
  - Represented by stick figures
- **Use case** represents a single system function.
  - Represented as an eclipse

Definitions and Symbols (Cont.)

- **System boundary** includes all the relevant use cases.
  - A boundary is the dividing line between the system and its environment.
  - Use cases are within the boundary.
  - Actors are outside of the boundary.
  - Represented as a box

Definitions and Symbols (Cont.)

- **Connection** is an association between an actor and a use case.
  - Depicts a usage relationship
  - Connection does not indicate data flow
  - Actors are connected to use cases with lines.
  - Use cases are connected to each other with arrows.

Definitions and Symbols (Cont.)

- *Extend relationship* is an association between two use cases where one adds new behaviors or actions to the other.
  - Extends a use case by adding new behavior or actions
  - Specialized use case extends the general use case.

Definitions and Symbols (Cont.)

- *Include relationship* is an association between two use cases where one use case uses the functionality contained in the other.
  - Indicates a use case that is used (invoked) by another use case
  - Links to general purpose functions, used by many other use cases

Definitions and Symbols (Cont.)

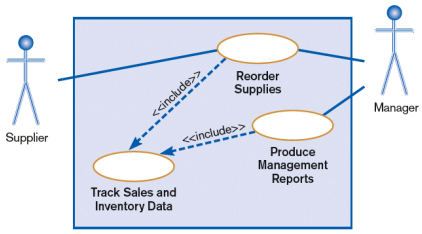


FIGURE 7-27
A use case diagram featuring an include relationship

Written Use Cases

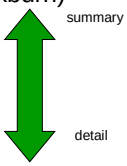
- Document containing detailed specifications for a use case
- Contents can be written as simple text or in a specified format
- Step-by-step description of what must occur in a successful use case

Figure 7-29
A template for writing use cases (*Source:* Cockburn, Alistair, *Writing Effective Use Cases*, 1st ed., © 2001. Reprinted and Electronically reproduced by permission of Pearson Education, Inc. Upper Saddle River, New Jersey.)

Use Case Title:
Primary Actor:
Level:
Stakeholders:
Precondition:
Minimal Guarantee:
Success Guarantee:
Trigger:
Main Success Scenario:
Extensions:

Level of Use Case

- Refers to degree of detail in the use case description
- Five suggested levels (Cockburn)
  1. White – as seen from clouds
  2. Kite – "birds-eye view"
  3. Blue – sea-level view
  4. Fish – below sea-level
  5. Black – bottom of the sea



Sample Format for Written Use Case

- ***Title*** – descriptive name, matches name in use case diagram
- ***Primary actor*** – usually a user role
- ***Stakeholders*** – any group or individual with an interest in the function of the use case

Sample Format for Written Use Case (Continued)

- ***Precondition*** – conditions that must be satisfied in order to execute the use case
- ***Minimal guarantee*** – outputs that can be expected if the service attempt failed
- ***Success guarantee*** – outputs that can be expected if the service succeeds

Sample Format for Written Use Case (Continued)

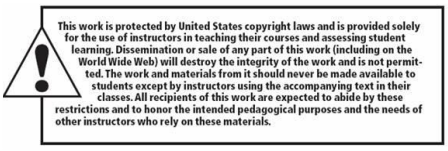
- ***Trigger*** – an event or action that initiates the use case
- ***Main success scenario*** – description of sequence of interactions between actor and use case during the use case execution
- ***Extensions*** – detailed description of how errors are handled

Use Case Title: Buying a PVF Product at WebStore
Primary Actor: Customer
Level: Kite (summary)
Stakeholders: Customer, shipping clerk
Precondition: Customer accesses the WebStore website
Minimal Guarantee: Rollback of any uncompleted transaction
Success Guarantees: Order filled
Trigger: Customer accesses WebStore homepage
Main Success Scenario: 1. Customer browses catalog. 2. Customer places order for desired product(s). 3. Shipping clerk fills order. 4. Customer checks status of order.
Extensions: 1a. Catalog is not available. 1a1. Customer quits site. 1a2. Customer takes action to gain access to catalog. 2a. Order transaction is interrupted. 2a1. Transaction rolled back. Customer starts again. 2a2. Transaction rolled back. Customer quits site. 3a. Item is out of stock. 3a1. Shipping clerk notifies customer. Customer waits for stock to be replenished. 3a2. Shipping clerk notifies customer. Customer cancels order. 4a. Order status is not available. 4a1. Customer quits site. 4a2. Customer takes action to gain access to order status.

Figure 7-34
Jim Woo's kite level written use case for buying a product at PVF's WebStore (*Source:* George, Hoffer, Valacich, Batra, 2006. *Object-Oriented Systems Analysis and Design*, 2nd ed. Upper Saddle River, NJ: Prentice Hall.)

Summary

- In Appendix A you learned how to:
  - ✓ Explain use cases and use case diagrams and how they can be used to model system functionality.
  - ✓ Present the basic aspects of how to create written use cases.
  - ✓ Discuss process modeling with use cases for electronic commerce application.



Modern Systems Analysis and Design

Seventh Edition

Jeffrey A. Hoffer
Joey F. George
Joseph S. Valacich

Chapter 7 Appendix B

Object-Oriented Analysis and Design: Activity Diagrams

Learning Objectives

- ✓ Understand how to represent system logic with activity diagrams.

Process Modeling: Activity Diagrams

- **Activity Diagrams**
  - Show the conditional logic for the sequence of system activities needed to accomplish a business process.
  - Clearly show parallel and alternative behaviors.
  - Can be used to show the logic of a use case.

Use Activity Diagrams to:

- Depict the flow of control from activity to activity.
- Help in use case analysis to understand what actions need to take place.
- Help in identifying extensions in a use case.
- Model work flow and business processes.
- Model the sequential and concurrent steps in a computation process.

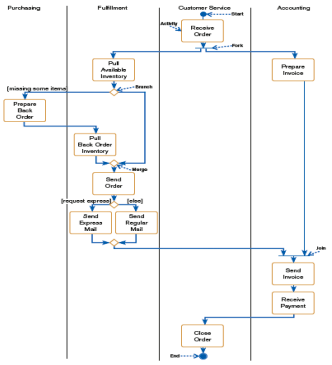


FIGURE 7-36
Activity diagram for a customer order process

Process Modeling: Activity Diagrams (Cont.)

- Elements of Activity Diagrams:
  - **Activity**: a behavior that an object carries out while in a particular state
  - **Branch**: a diamond symbol containing a condition whose results provide transitions to different paths of activities
  - **Merge**: a circular symbol where different paths converge

Process Modeling: Activity Diagrams (Cont.)

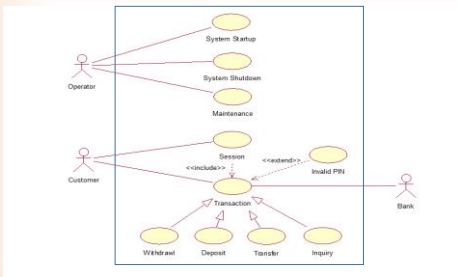
- **Fork**: the beginning of parallel activities
- **Join**: the end of parallel activities
- **Swimlanes**: columns representing different organizational units of the system

Summary

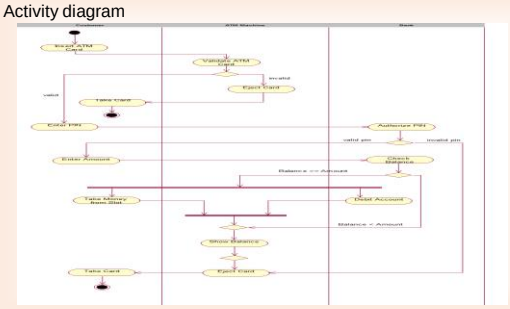
- In this appendix you learned how to:
  - ✓ Understand how to represent system logic with activity diagrams.

ATM application

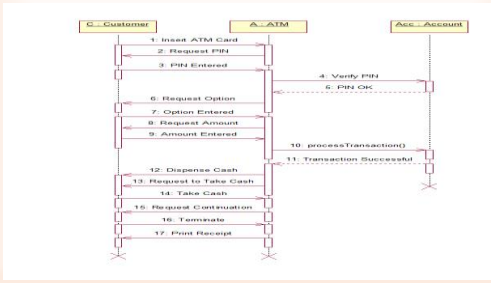
USE CASE DIAGRAM



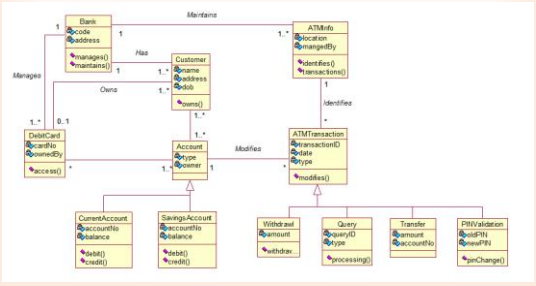
Activity diagram



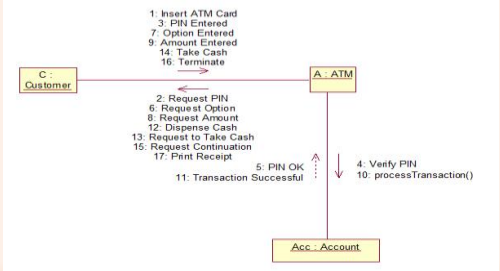
Sequences diagram



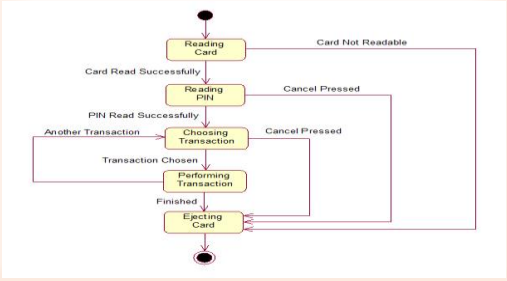
Class diagram



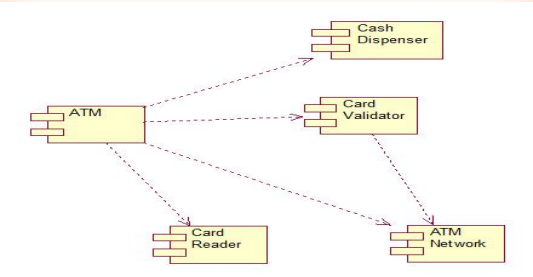
Collaboration diagram



Statechart diagram



Component diagram



Deployment diagram