Chapter 1
Assuming the Role of the Systems Analyst

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Why System Analysis?
- Information systems
- Phases of analysis and design
- System maintenance
- CASE tools
- Alternate methodologies

Why System Analysis?

- Understand what humans need to analyze:
  - Data entry or dataflow
  - Process or transform data
  - Store data
  - Output information

- Support users and business
  - Analyze, design and implement improvements to the system (new or existing)
- Improve a business through the use of computerized information systems

Information

- Information is an organizational resource, which must be managed as carefully as other resources.(labor & materials)
- Costs are associated with information processing. Like production, distribution, security and storage.
- Information processing must be managed to take full advantage.

Categories

Information systems fall into one of the following eight categories:

- Transaction processing systems (TPS).
- Office automation systems (OAS).
- Knowledge work systems (KWS).
- Management information systems (MIS).
- Decision support systems (DSS).
- Expert systems (ES) and Artificial Intelligence (AI).
- Group decision support systems (GDSS) and Computer-Supported Collaborative Work Systems.
- Executive support systems (EES).

Categories

Transaction processing systems (TPS).

TPSs are computerized information system that process large amounts of data for routine business transaction such as payroll and inventory.

Office automation systems (OAS).

OASs for supporting data workers, who do not usually create new knowledge but rather analyze information so as to transform data or manipulate it in some way.

Such as word processing, spreadsheets, e-scheduling and communication through voice mail, e-mail video conference

Categories

Knowledge work systems (KWS).

It's like OASs but KWS support professional workers

Such as scientist, engineers and doctors.

Management information systems (MIS).

MIS are computerized information systems to help interaction between people and computers by requiring people, S/W and H/W.

Include decision analysis and decision making

Categories

Decision support systems (DSS).

DSS are similar to MIS but DDS emphasizes the support of decision making in all it's phases.

Expert systems (ES) and Artificial Intelligence (AI).

An expert systems effectively captures and uses the knowledge of an expert for solving a particular problem in an organization

Categories

Group decision support systems (GDSS) and Computer-Supported Collaborative Work Systems.

GDSSs are intended to bring a group together to solve a problem with the help of various support such as questionnaires, brainstorming and scenario creation

Executive support systems (EES).

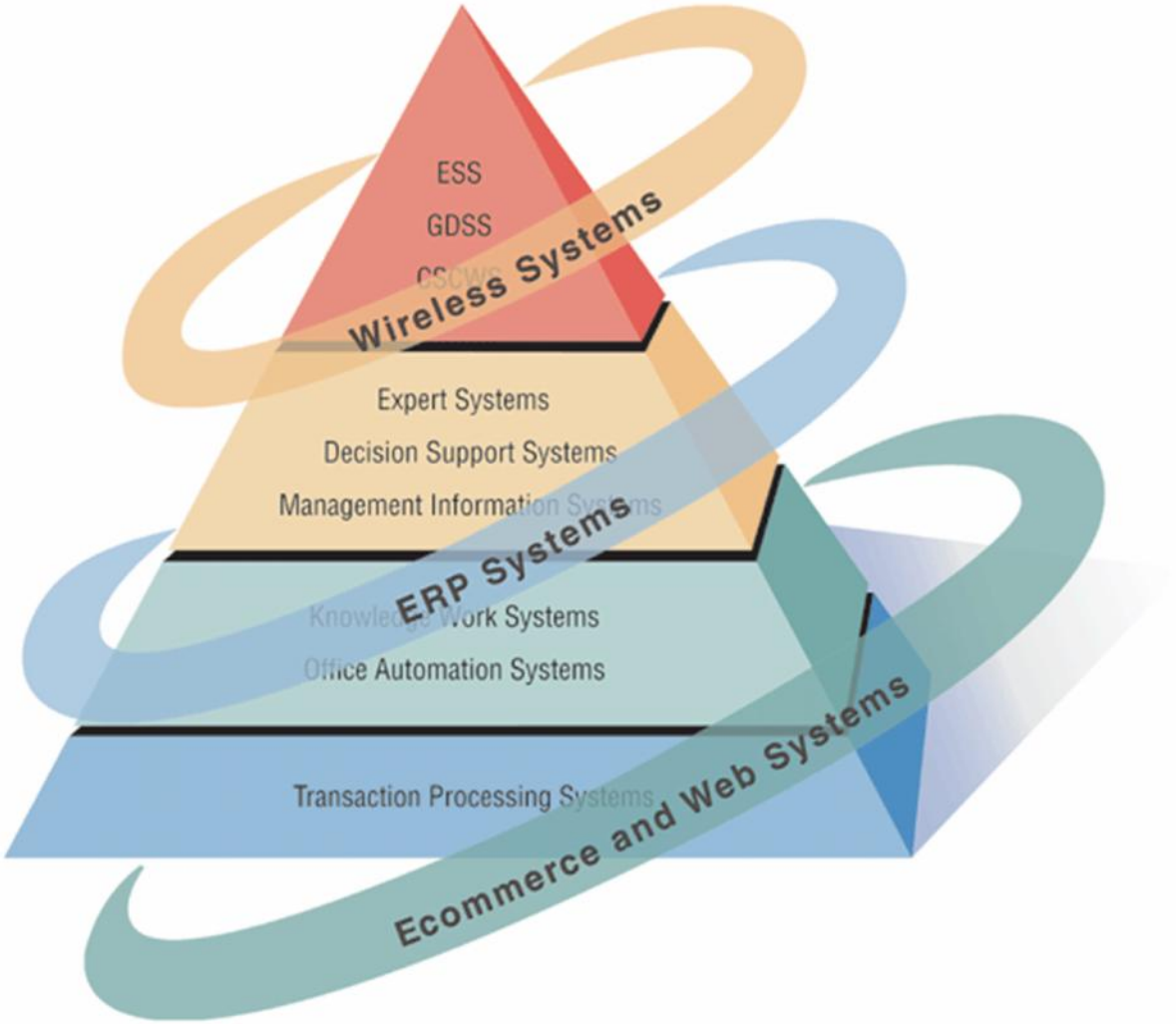
EES help executives organize their interaction with external environment by providing communication support.

New Technologies

New technologies are being integrated into traditional systems:

- Ecommerce uses the Web to perform business activities.
- Enterprise Resource Planning (ERP) has the goal of integrating many different information systems within the corporation.
- Wireless and handheld devices, including mobile commerce (mcommerce).
- Open source software.

Figure 1.2 Systems analysts need to be aware that integrating technologies affect all types of systems.



Advantages of Using the Web

- The benefits of using the Web are:
- Increasing awareness of the availability of the service, product, industry, person, or group.
- 24-hour access for users.
- Standard interface design.
- Creating a global system.

Nature of Analysis and Design

Systems analysis and design is a systematic approach to:

- Identifying problems, opportunities, and objectives.
- Analyzing the information flows in organizations.
- Designing computerized information systems to solve a problem.

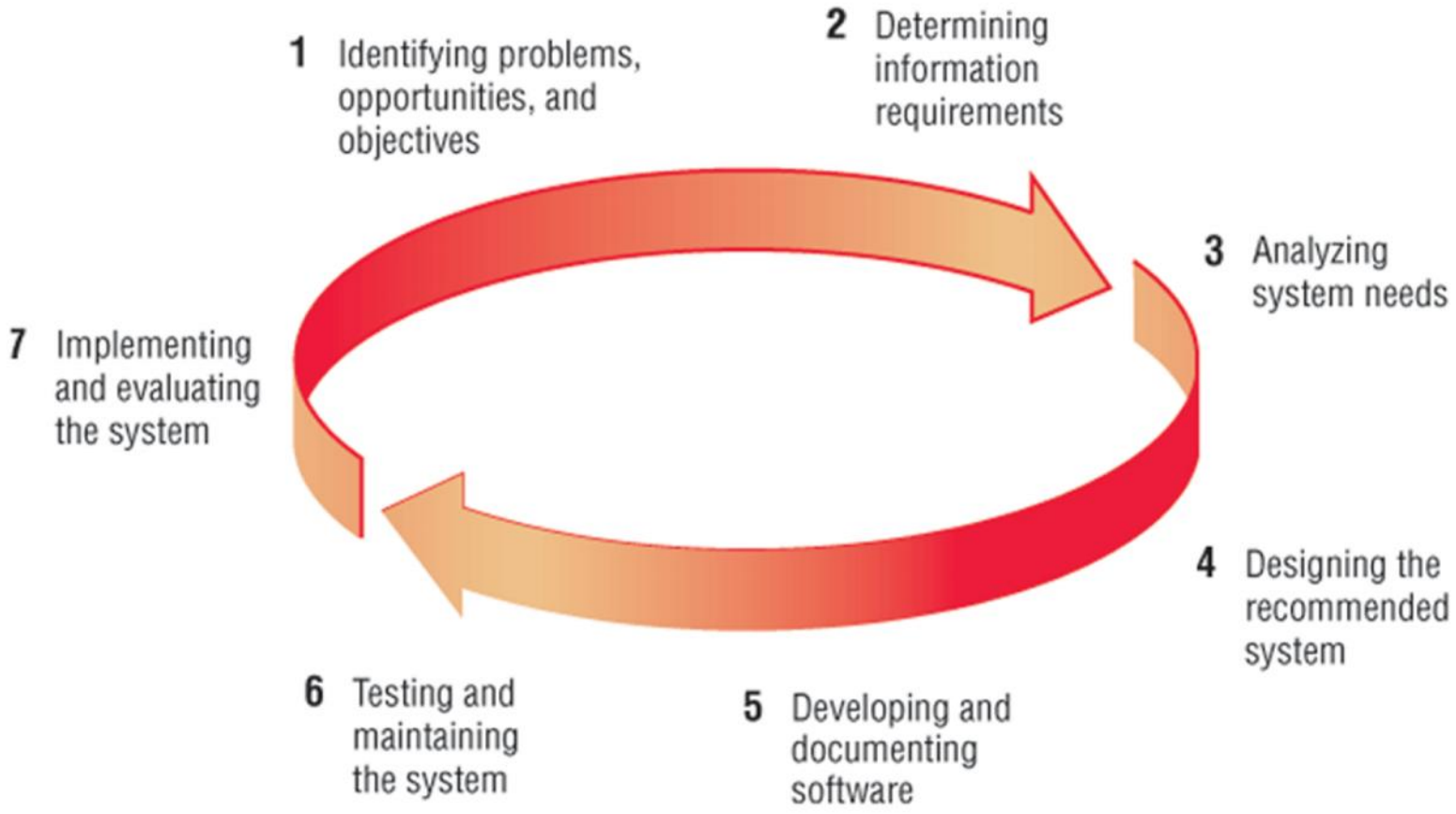
Systems Analyst

- Systems analysts act as:
  - Outside consultants to businesses.
  - Supporting experts within a business.
  - As change agents.
- Analysts are problem solvers, and require communication skills.
- Analysts must be ethical with users and customers.

Systems Development Life Cycle (SDLC)

- The systems development life cycle is a systematic approach to solving business problems.
- It is divided into seven phases.
- Each phase has unique activities.

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



Phase 1

- Identifying:
  - Problems.
  - Opportunities.
  - Objectives.
- Personnel involved:
  - Analyst.
  - User management.
  - Systems management.

Phase 1

- Activities:
  - Interviewing .
  - Summarizing the knowledge.
  - Estimate the scope of project.
  - Documenting the result.
- The output:
  - Feasibility study which containing :
    - Problem definition.
    - Summarizing the objectives.

Phase 2

- Determining information requirements:
  - Interview management, operations personnel.
  - Gather systems/operating documents.
  - Use questionnaires.
  - Observe the system and personnel involved.
- Learn the who, what, where, when, and how, and the why for each of these.

Phase 2 (Continued)

- Personnel involved:
  - Analyst.
  - User management.
  - User operations workers.
  - Systems management.

Phase 3

- Analyzing system needs:
  - Create data flow diagrams.
  - Document procedural logic for data flow diagram processes.
  - Complete the data dictionary.(list of all data items used in the system)
  - Make structured decisions.(Methods : structured English, Decision tables, Decision trees)
  - Prepare and present the system proposal.
  - Recommend the optimal solution to management.

Phase 3 (Continued)

- Personnel involved:
  - Analyst.
  - User management.
  - Systems management.

Phase 4

- Designing the recommended system:
  - Design the user interface.
    - Design output.
    - Design input.
  - Design system controls to protect the system and data.
  - Design files and/or database.
  - Produce program specifications packets. For programmers

Phase 4 (Continued)

- Personnel involved:
  - Analyst.
  - System designer.
  - User management.
  - User operations workers.
  - Systems management.

Phase 5

- Developing and documenting software:
  - Design computer programs using structured techniques for designing and documented software like structure charts, Nassi-Schneiderman charts, and pseudocode.
  - Walkthrough program design.
  - Write computer programs.
  - Document software with help files, procedure manuals, and Web sites with Frequently Asked Questions.

Phase 5 (Continued)

- Personnel involved:
  - Analyst.
  - System designer.
  - Programmers.
  - Systems management.

Phase 6

- Testing and maintaining the system:
  - Test and debug computer programs.
  - Test the computer system.
  - Enhance system.

Phase 6 (Continued)

- Personnel involved:
  - Analyst.
  - System designer.
  - Programmers.
  - Systems management.

Phase 7

- Implementing and evaluating the system:
  - Plan conversion. (Old to New)
  - Train users.
  - Purchase and install new equipment.
  - Convert files.
  - Install system.
  - Review and evaluate system.

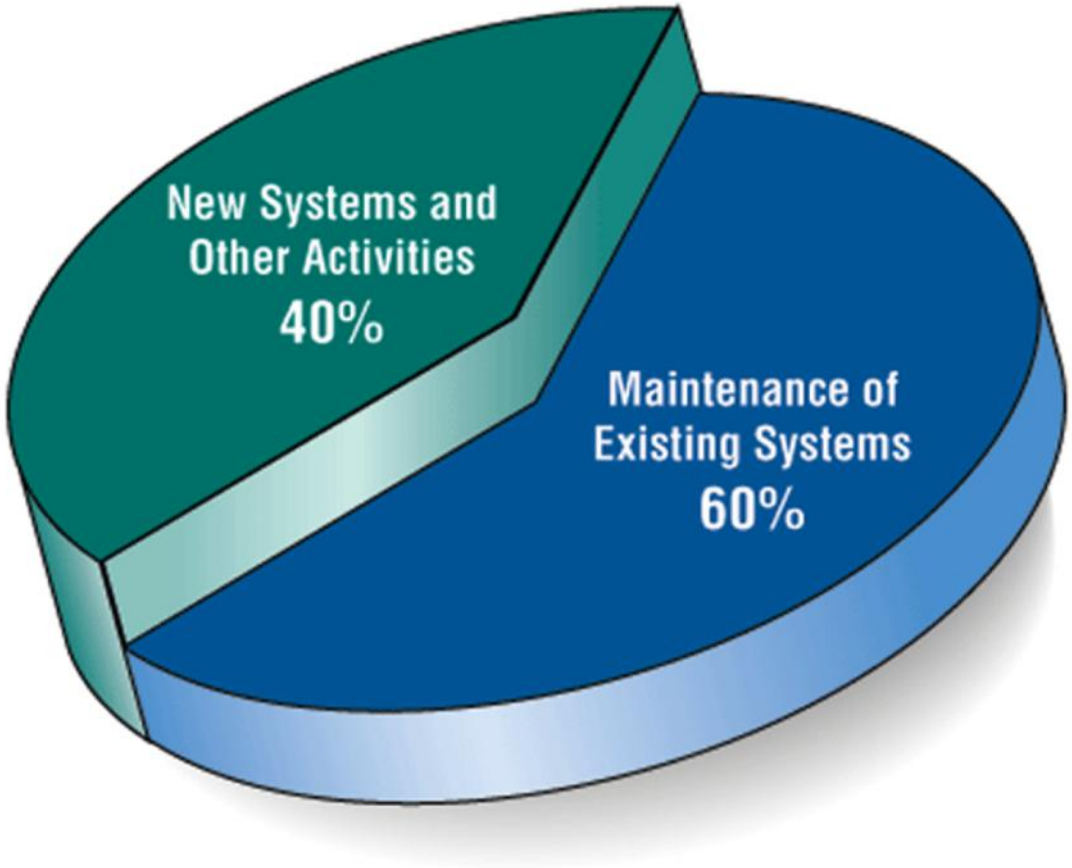
Phase 7 (Continued)

- Personnel involved:
  - Analyst.
  - System designer.
  - Programmers.
  - User management.
  - User operations workers.
  - Systems management.

System Maintenance

- System maintenance is:
  - Removing undetected errors, and
  - Enhancing existing software.
- Time spent on maintenance typically ranges from 48-60 percent of total time.

Figure 1.4 Some researchers estimate that the amount of time spent on system maintenance may be as much as 60 percent of the total time spent on systems projects.

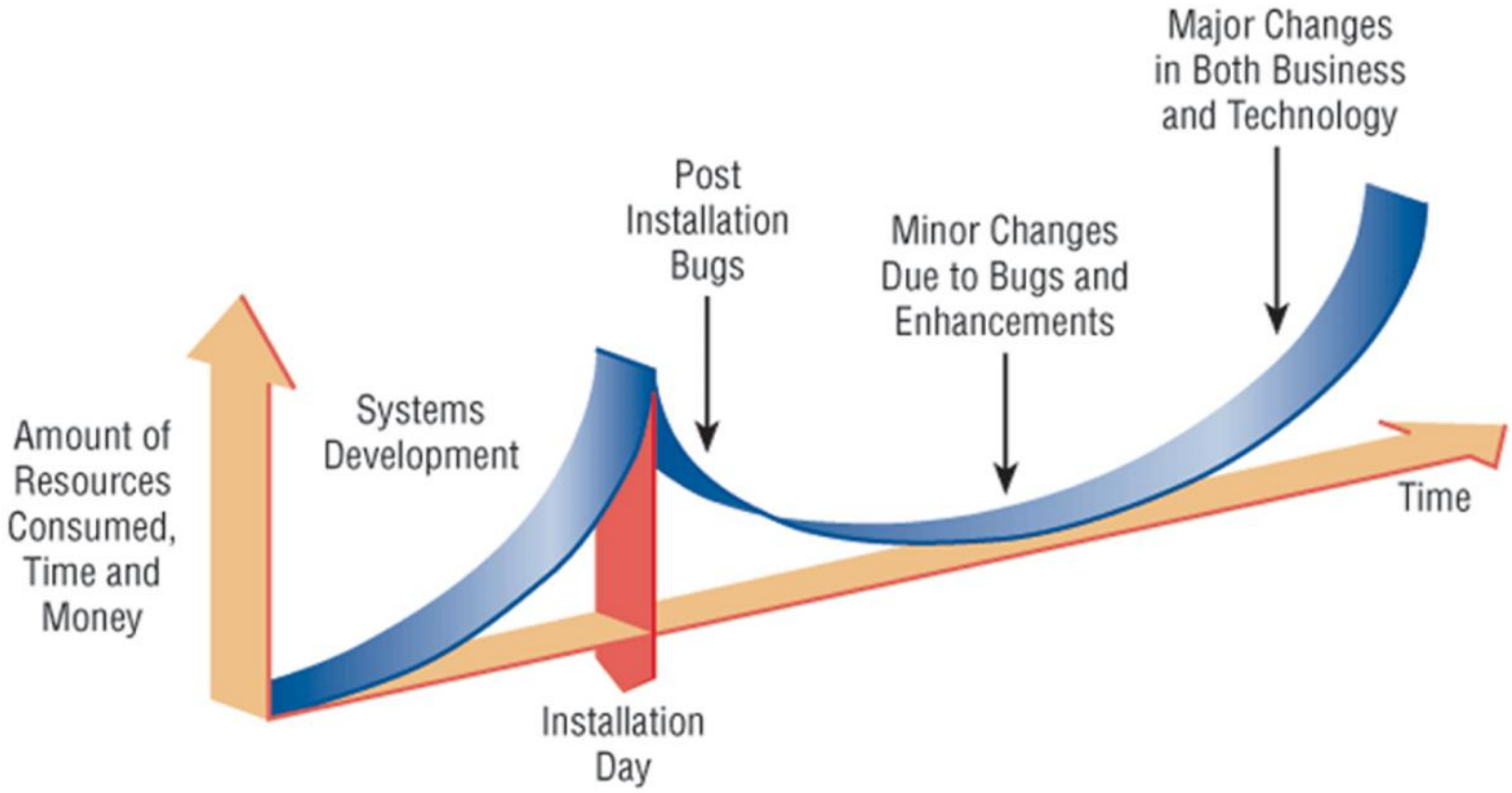


System Enhancements

Systems are enhanced for the following reasons:

- Adding additional features to the system.
- Business and governmental requirements change over time.
- Technology, hardware, and software are rapidly changing.

Figure 1.5 Resource consumption over the system life.



CASE Tools

- CASE tools are automated, computer-based software packages for systems analysis and design.
- Four reasons for using CASE tools are:
  - To increase analyst productivity.
  - Facilitate communication among analysts and users.
  - Providing continuity between life cycle phases.
  - To assess the impact of maintenance.

CASE Tool Categories

CASE tools may be divided into several categories

- Upper CASE (also called front-end CASE) tools, used to perform analysis and design.
- Lower CASE (also called back-end CASE). These tools generate computer language source code from CASE design.
- Integrated CASE, performing both upper and lower CASE functions.

Upper CASE

Upper CASE tools:

- Create and modify the system design.
- Store data in a project repository.

  The repository is a collection of records, elements, diagrams, screens, reports, and other project information.
- These CASE tools model organizational requirements and define system boundaries.

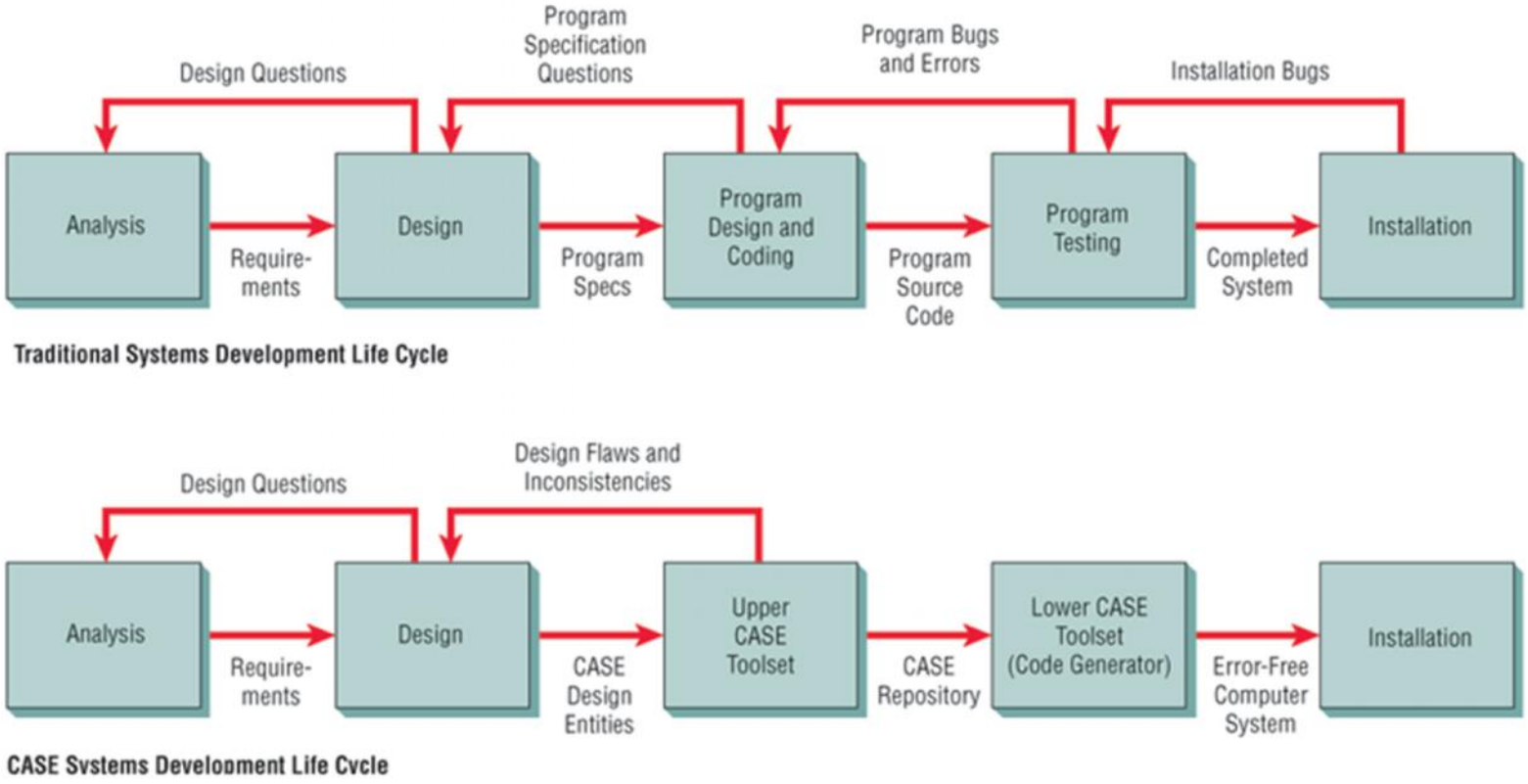
Lower CASE

- Lower CASE tools generate computer source code from the CASE design.
- Source code may usually be generated in several languages.

Advantages of Generating Code

- Time to develop new systems decreases.
- The time to maintain generated code is less than to maintain traditional systems.
- Computer programs may be generated in more than one language.
- CASE design may be purchased from third-party vendors and tailored to organizational needs.
- Generated code is free from errors.

Figure 1.7 Traditional versus CASE systems development life cycles.



Reverse Engineering

- Reverse engineering is generating the CASE design from computer program code.
- Source code is examined, analyzed, and converted into repository entities.

Reverse Engineering (Continued)

- Reverse engineering produces (depending on the tool set used):
  - Data structures and elements, describing the files, records, and field.
  - Screen designs, if the program is online.
  - Report layouts .
  - A structure chart showing the hierarchy of the modules in the program.
  - Database design and relationships.

Advantages of Reverse Engineering

Reverse Engineering has the following advantages:

- Reduced system maintenance time.
- Produce Program documentation.
- Structured programs may be generated from older programs.
- Future system maintenance is easier to implement.
- Unused portions of programs may be eliminated.

Object-Oriented Analysis and Design

- Object-oriented (O-O) analysis and design is used to build object-oriented programs.
- O-O programming examines the objects of a system.
- Objects are grouped into classes for optimal reuse and maintainability.

Extreme Programming (XP)

Extreme Programming (XP) is actually disciplined approach to software development. About eight years old.

XP is successful because:

- Stresses customer satisfaction.
- XP help your developers to respond to changing customer requirements, even late in the life cycle.
- This methodology also enhance team work. Managers, customers, and developers to delivering quality software.
- XP implements a simple.

Extreme Programming (XP) (Continued)

- Extreme programming values are:
  - Communication.
  - Simplicity.
  - Feedback.
  - Courage.

Home Work
E-Auction

www.elance.com

- Elance is the leading site for online work where businesses connect with independent professionals to get work done.

Chapter 2
Understanding Organizational Style and Its Impact on Information Systems

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Organizational environment
- Nature of systems
- Context-level data flow diagram
- Entity-relationship diagram
- Levels of management
- Organizational culture

Organizations

- Organizations are composed of interrelated and interdependent subsystems.
- System and subsystem boundaries and environments impact on information system analysis and design.

Organizational Environment

- Community environment

  The environment of the community in which the organization is physically located, which is shaped by the size of its population and its demographic profile, including factors such as education and average income

- Economic environment

  Influenced by market factors and competition

- Political environment

  State and local government

Open and Closed Systems

Systems are described as either

- Open (art departments)
  - Free-flowing information.
  - Output from one system becomes input to another. (art departments)
- Closed with restricted access to information
  - Limited by numerous rules.

  (defense department)

Virtual Organizations

- A virtual organization has parts of the organization in different physical locations.
- Computer networks and communications technology are used to work on projects.

Virtual Organization Advantages

Advantages of a virtual organization are:

- Reduced costs of physical facilities.
- More rapid response to customer needs.
- Flexibility for employees to care for children or aging parents.

Levels of Management

Figure 2.13 Management in organizations exists on three horizontal levels: operational control, managerial planning and control, and strategic management.

Managerial Control

- The three levels of managerial control are:
  - Operations management.
  - Middle management.
  - Strategic management.

Operations Management

- Make decisions using predetermined rules that have predictable outcomes make decisions.
- Oversee the operating details of the organization.
- dependent on internal information.

Middle Management

- Make short-term planning and control decisions about resources and organizational objectives.
- Decisions may be partly operational and partly strategic.
- Decisions are dependent on internal information, both historical and prediction oriented.

Strategic Management

- Look outward from the organization to the future.
- Make decisions that will guide middle and operations managers.
- Work in highly uncertain decision-making environment.
- Define the organization as a whole.
- Often make one-time decisions.

Managerial Levels

Each of the three levels of management have:

- Different organization structure.
- Leadership style.
- Technological considerations.
- Organization culture.
- Human interaction.
- All carry implications for the analysis and design of information systems.

Organizational Culture

- Organizations have cultures and subcultures.
- Learn from verbal and nonverbal symbolism.

Verbal & Nonverbal Symbolism

Verbal Symbolism

- Using language to convey.

Nonverbal Symbolism

- Clothing worn
- Office placement and decorations

Chapter 3
Determining Feasibility and Managing Analysis and Design Activities

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Project initiation
- Determining project feasibility
- Project scheduling
- Managing project activities
- Manage systems analysis team members

Project Initiation

Projects are initiated for two reasons:

- Problems that need themselves to systems solutions.
- Improvement through
  - Upgrading systems.
  - Altering systems.
  - Installing new systems.

Organizational Problems

Identify problems by looking for the following signs:

- Check output against performance criteria
  - Too many errors.
  - Work completed slowly.
  - Work done incorrectly.
  - Work done incompletely.
  - Work not done at all.

Organizational Problems (Continued)

- Listen to feedback from vendors, customers, and suppliers
  - Complaints.
  - Suggestions for improvement.
  - Loss of sales.
  - Lower sales.

Possibilities for Improvement

Many possible objectives exist including:

- Speeding up a process.
- Streamlining a process.
- Combining processes.
- Reducing errors in input.
- Reducing redundant storage.
- Reducing redundant output.
- Improving system and subsystem integration.

Project Selection

Five specific criteria for project selection:

- Supported by management.
- Have resources for that project.
- It moves the business toward to its goals.
- Practicable. (We can do it)
- Important enough to be considered over other projects.

Feasibility Impact Grid (FIG)

- A feasibility impact grid (FIG) is used to assess the impact of any improvements to the existing system.
- It can increase awareness of the impacts made on the achievement of corporate objectives

Feasibility Impact Grid (FIG) (Continued)

- Current or proposed systems are listed on the left.
- Objectives are listed on the top.
- Red arrows indicate a positive impact.
- Green arrows indicate implementation.

Figure 3.2 page 54

Figure 3.3 page 55

Feasibility Impact Grid (FIG)

	System Components	Process Objectives: Speeding Up a Process	Streamlining a Process	Combining Processes	Reducing Errors in Input	Reducing Redundant Data Storage	Reducing Redundant Output	Improving Integration of Systems
Ecommerce Systems	Online catalog	✓	✓				✓	✓
	Online order processing	✓	✓		✓	✓		✓
	Online technical support		✓					
	Banner advertisements							
	Web-based intelligent push agent							
MIS	Inventory management		✓					✓
	Production scheduling		✓					✓
	Monthly sales reports				✓			✓
	Regional sales analysis				✓			✓
	Logistics management					✓		✓

Feasibility Impact Grid (FIG)

		Corporate Objectives						
	System Components	Corporate Profits	Competitive Strategy	Cooperative Ventures	Internal Operations Support	Internal Decision Support	Customer Service	Employee Morale
Ecommerce Systems	Online catalog		✓				✓	✓
	Online order processing		✓	✓			✓	✓
	Online technical support		✓				✓	✓
	Banner advertisements	✓		✓			✓	
	Web-based intelligent push agent						✓	
MIS	Inventory management	✓	✓		✓	✓	✓	
	Production scheduling	✓	✓		✓	✓	✓	✓
	Monthly sales reports	✓	✓		✓	✓		✓
	Regional sales analysis	✓			✓	✓		✓
	Logistics management	✓	✓		✓	✓		

Feasibility

- A feasibility study assesses the operational, technical, and economic of the proposed project.
- There are three types of feasibility:
  - Technical feasibility.
  - Economic feasibility.
  - Operational feasibility.

Technical Feasibility

- Technical feasibility assesses whether the current technical resources are sufficient for the new system.
- If they are not available, can they be upgraded to provide the level of technology necessary for the new system.

Economic Feasibility

- Economic feasibility determines whether the time and money are available to develop the system.
- Includes the purchase of:
  - New equipment.
  - Hardware.
  - Software.

Operational Feasibility

- Operational feasibility determines if the human resources are available to operate the system once it has been installed.
- Users that do not want a new system may prevent it from becoming operationally feasible.

Activity Planning

- Activity planning includes:
  - Selecting a systems analysis team.
  - Estimating time required to complete each task.
  - Scheduling the project.
- Two tools for project planning and control are Gantt charts and PERT diagrams.

Estimating Time

- Project is broken down into phases.
- Project phases are broken down into tasks or activities.
- Finally project is broken down into steps or even smaller units.
- Time is estimated for each task or activity.

Estimating Time

Phase	Activity
Analysis	Data gathering Data flow and decision analysis Proposal preparation
Design	Data entry design Input design Output design Data organization
Implementation	Implementation Evaluation

Estimating Time

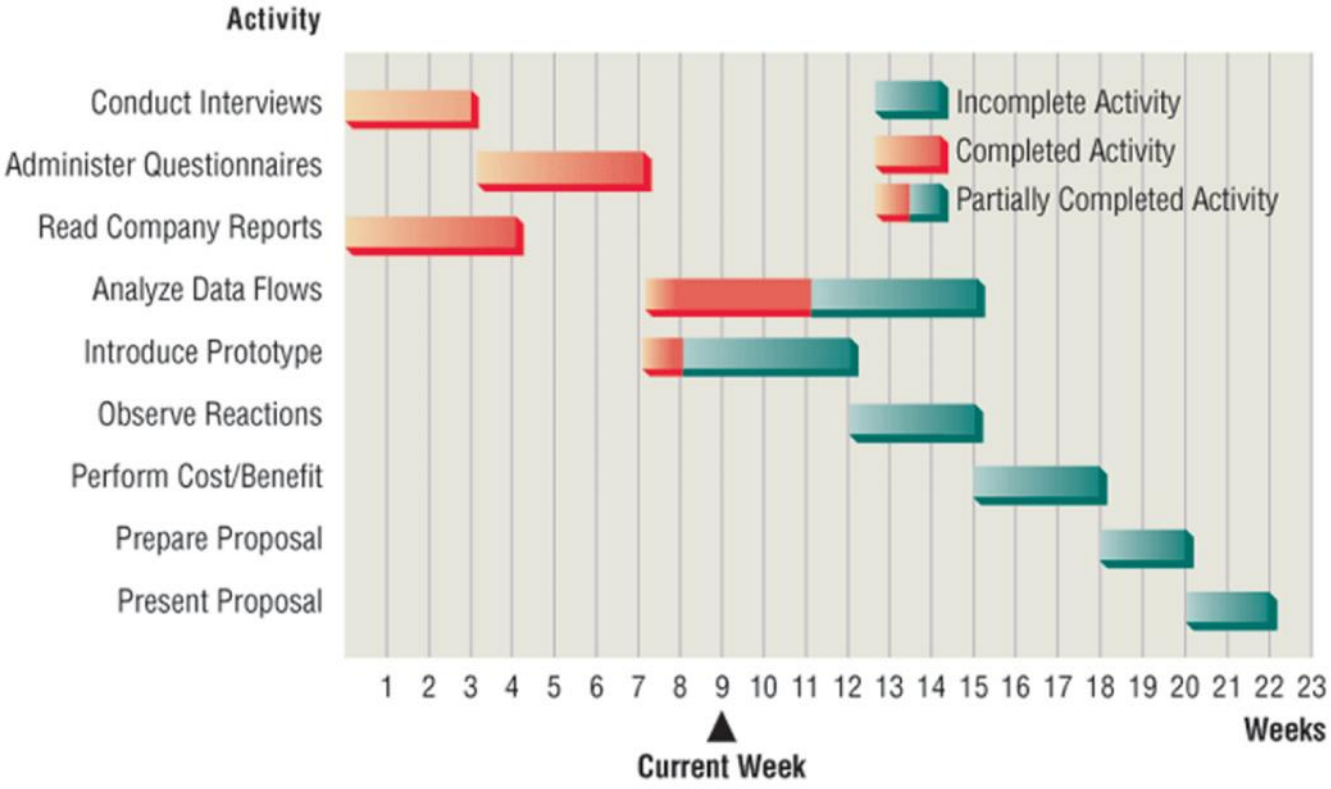
Activity	Detailed Activity	Weeks Required
Data gathering	Conduct interviews	3
	Administer questionnaires	4
	Read company reports	4
	Introduce prototype	5
	Observe reactions to prototype	3
Data flow and decision analysis	Analyze data flow	8
Proposal preparation	Perform cost/benefit analysis	3
	Prepare proposal	2
	Present proposal	2

Gantt Charts

- A Gantt chart is an easy way to schedule tasks.
- It is a chart on which bars represent each task or activity. The length of each bar represents the relative length of the task.
- Easy to construct and use.
- Shows activities over a period of time.

Gantt Chart Example

Figure 3.7 Using a two-dimensional Gantt chart for planning activities that can be accomplished in parallel.



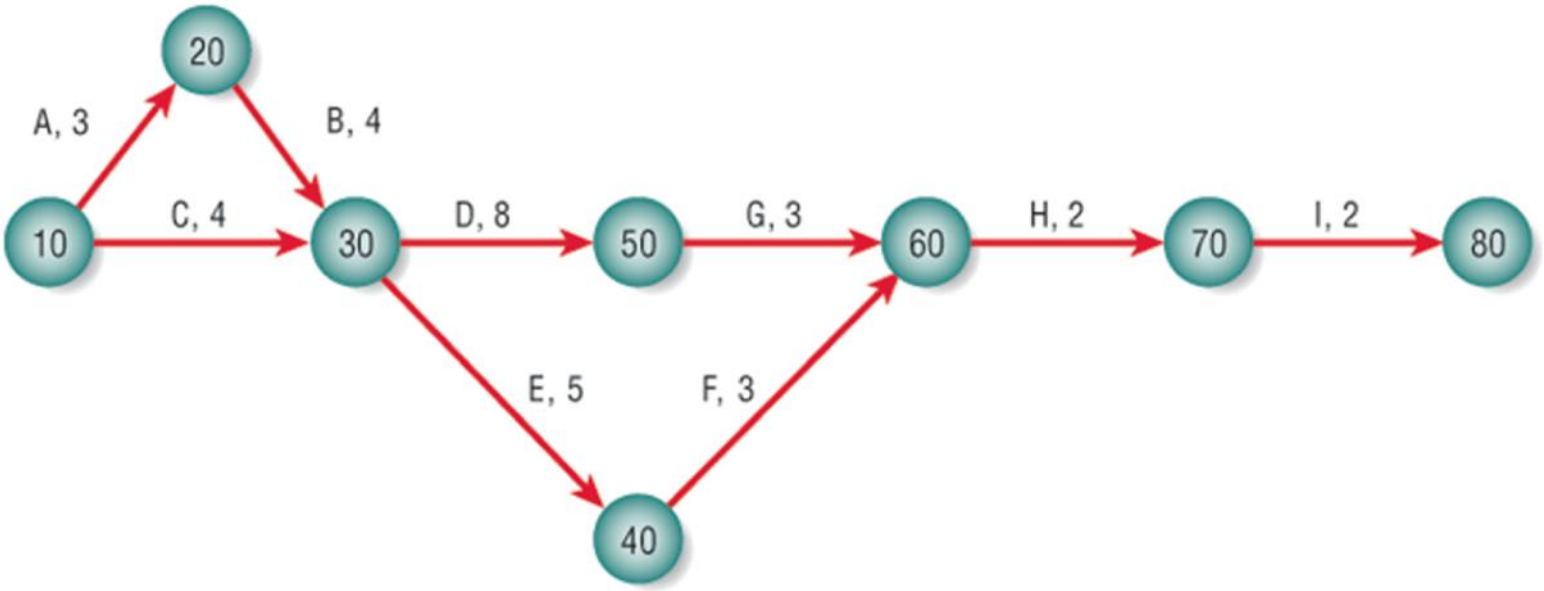
PERT Diagram

PERT-Program Evaluation and Review Technique

- PERT diagrams show precedence, activities that must be completed before the next activities may be started.
- Once a diagram is drawn it is possible to identify the critical path, the longest path through the activities.
- Monitoring critical path will identify shortest time to complete the project.

PERT Diagram Example

Figure 3.11 A completed PERT diagram for the analysis phase of a systems project.



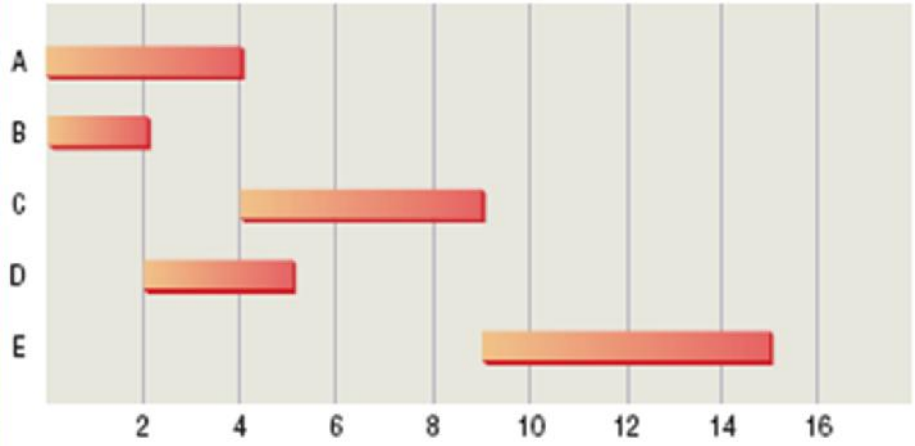
PERT Diagram Advantages

- Easy identification of the order of precedence
- Easy identification of the critical path and thus critical activities
- Easy determination of slack time

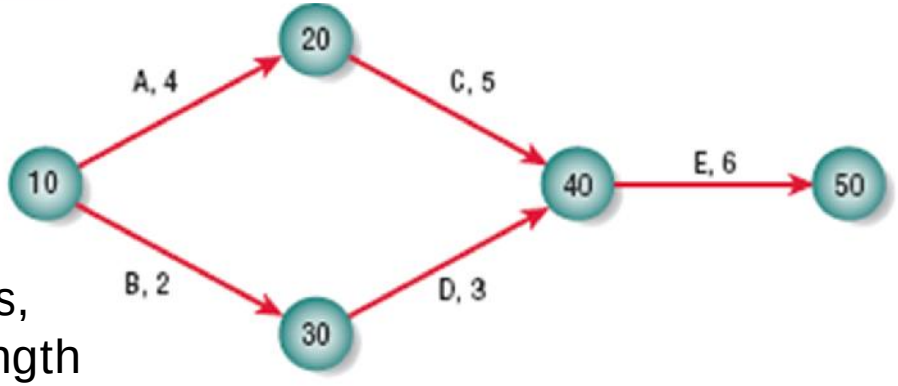
Gantt chart and PERT digram

Activity		Predecessor	Duration
A	Conduct interviews	None	3
B	Administer questionnaires	A	4
C	Read company reports	None	4
D	Analyze data flow	B, C	8
E	Introduce prototype	B, C	5
F	Observe reactions to prototype	E	3
G	Perform cost/benefit analysis	D	3
H	Prepare proposal	F, G	2
I	Present proposal	H	2

Gantt chart and PERT digram



Activity		Predecessor	Duration
A	Conduct interviews	None	3
B	Administer questionnaires	A	4
C	Read company reports	None	4
D	Analyze data flow	B, C	8
E	Introduce prototype	B, C	5
F	Observe reactions to prototype	E	3
G	Perform cost/benefit analysis	D	3
H	Prepare proposal	F, G	2
I	Present proposal	H	2



10–20–40–50 has a length of 15 days, whereas path 10–30–40–50 has a length of 11 days.

Team Management

- Teams often have two leaders:
  - One who leads members to accomplish tasks.
  - One concerned with social relationships.
- The systems analyst must manage:
  - Team members.
  - Their activities.
  - Their time and resources.

Ecommerce Project Management

Ecommerce and traditional software project management differences:

- The data used by ecommerce systems is scattered across the organization.
- Ecommerce systems need a staff with a wide variety of skills.
- Security is of utmost importance.

Timeboxing

- Timeboxing sets an absolute due date for project delivery.
- The most critical features are developed first and implemented by the due date.
- Other features are added later.

Project Failures

Project failures may be prevented by:

- Training.
- Experience.
- Learning why other projects have failed.

Extreme Programming Activities

The activities of extreme programming are:

- Coding.
- Testing.
- Listening.
- Designing.

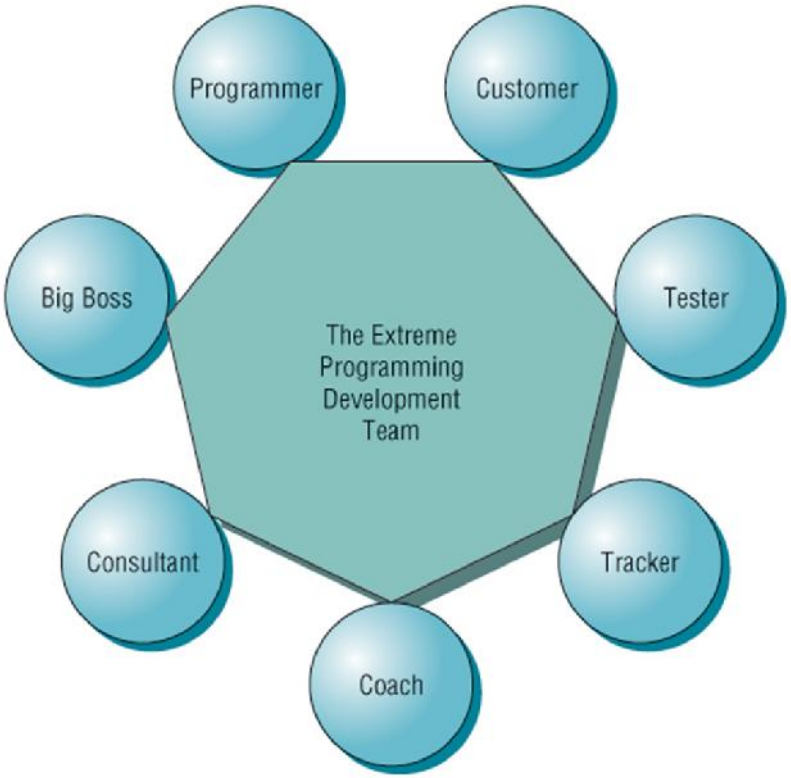
Extreme Programming Core Practices

There are four core practices in extreme programming:

- A short release time.
- Working a 40-hour week.
- Having an onsite customer.
- Pair programming.

Roles in Extreme Programming

Figure 3.17 Roles in the XP development process include members from inside of the development team as well as at least one onsite customer.



Roles in Extreme Programming

There are a 7roles played in XP:

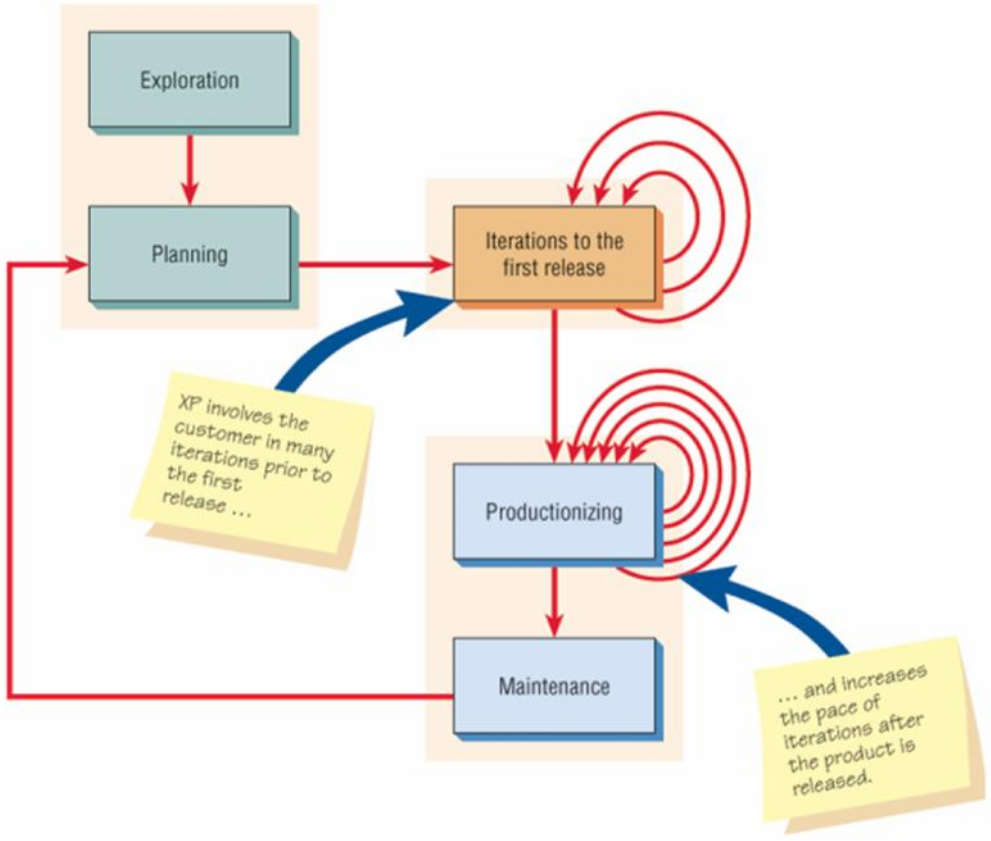
- Programmer.
- Customer.
- Tester.
- Tracker.
- Coach.
- Consultant.
- Big Boss.

XP Development Process

- XP projects are interactive and incremental.
- The five Stages of XP development are:
  - Exploration.
  - Planning.
  - Iterations to the first release.
  - Production.
  - Maintenance.

XP Development Process

Figure 3.19 The five stages of the XP development process shows that frequent iterations are essential to successful system development.



Chapter 4
Information Gathering: Interactive Methods

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Question format
- Interviewing techniques
- Joint Application Design (JAD)
- Questionnaires

Interviewing

- Interviewing is an important method for collecting data on information system requirements.
- Interviews reveal information about:
  - Interviewee opinions.
  - Interviewee feelings.
  - About the current state of the system.
  - Organizational and personal goals.
  - Informal procedures.

Planning the Interview

Five steps in planning the interview are:

- Reading background material.
- Establishing interview objectives.
- Deciding whom to interview.
- Preparing the interviewee.
- Deciding on question types and structure.

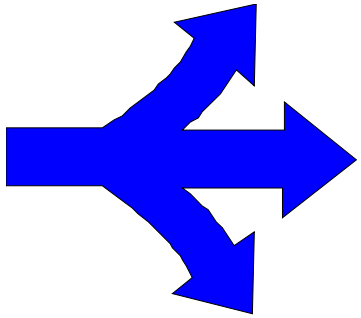
Question Types

There are two basic types of interview questions:

- Open-ended.
- Closed.

Open-Ended Questions

- Open-ended interview questions allow interviewees to respond how they wish, and to what length they wish.
- Open-ended questions are appropriate when the analyst is interested in breadth and depth of reply.



Advantages of Open-Ended Questions

Eight benefits of open-ended questions are:

1. Puts the interviewee at ease.
2. Allows the interviewer to pick up on the interviewee's vocabulary.
   > Reflect education, beliefs.
3. Provides richness of detail.
4. Reveals to further questioning that may have gone untapped.

Advantages of Open-Ended Questions

Eight Benefits of open-ended questions are: (continued)

5. Provides more interest for the interviewee.
6. Allows more spontaneity.
7. Makes phrasing easier for the interviewer.
8. Useful if the interviewer is unprepared.

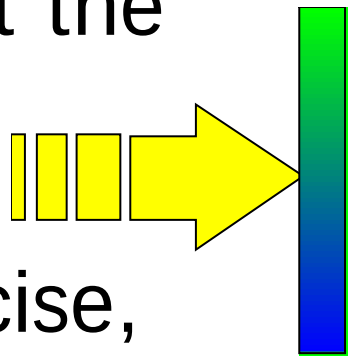
Disadvantages of Open-Ended Questions

The five drawbacks include:

- May result in too much irrelevant detail.
- Possibly losing control of the interview.
- May take too much time for the amount of useful information gained.
- Possibly giving the impression that the interviewer is on a "fishing expedition"

Closed Interview Questions

- Closed interview questions limit the number of possible responses.
- Closed interview questions are appropriate for generating precise, reliable data that is easy to analyze.
- The methodology is efficient, and it requires little skill for interviewers to administer.



Benefits of Closed Interview Questions

Six benefits are:

1. Saving interview time.
2. Easily comparing interviews.
3. Getting to the point.
4. Keeping control of the interview.
5. Covering a large area quickly.
6. Getting to related data.

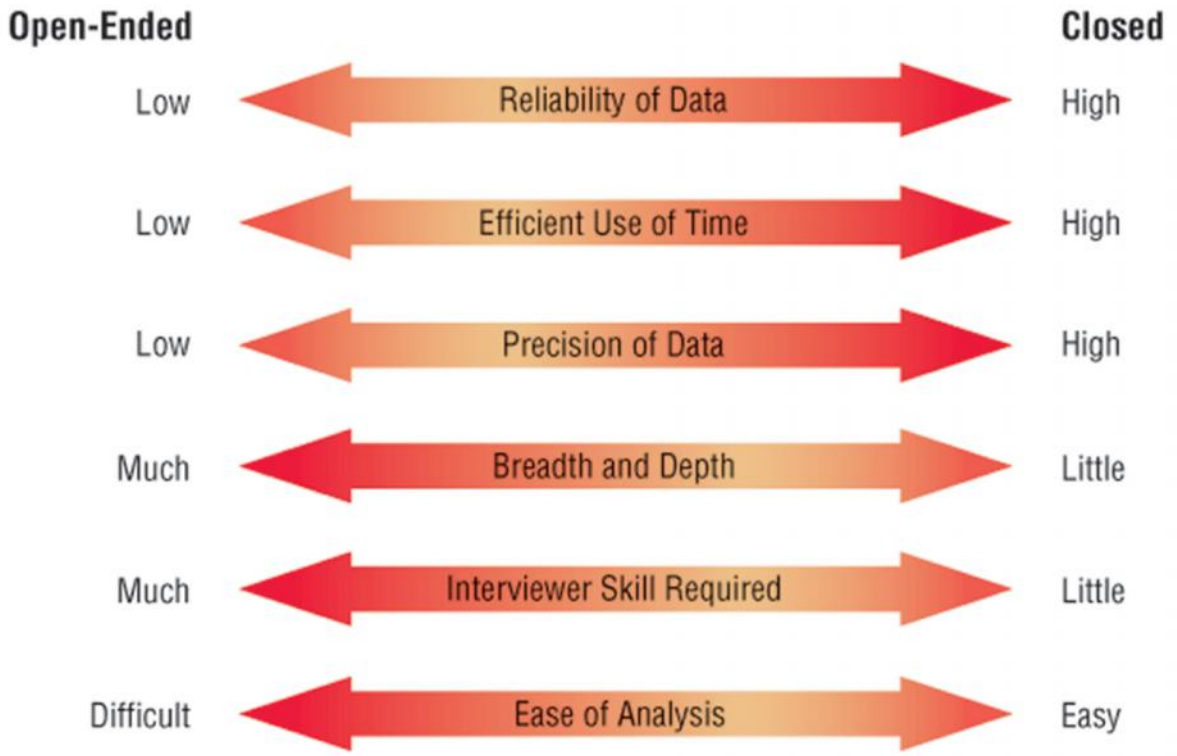
Disadvantages of Closed Interview Questions

Four drawbacks of closed interview questions include:

1. Boring for the interviewee.
2. Failure to obtain rich detailing.
3. Missing main ideas.
4. Failing to build relation between interviewer and interviewee.

Attributes of Open-ended and Closed Questions (Comparison)

Figure 4.5 Attributes of open-ended and closed questions.



Bipolar Questions and Probes

- Bipolar questions are those that may be answered with a ‘yes’ or ‘no’ or ‘agree’ or ‘disagree’.

Probing Questions

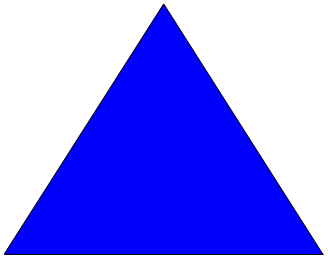
- Probing questions elicit more detail about previous questions.
- The purpose of probing questions is:
  - To get more meaning.
  - To clarify.
  - To draw out and expand on the interviewee's point.

Question Sequencing

The three basic ways of structuring interviews are :

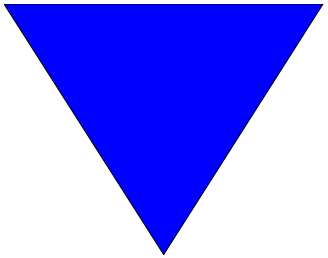
1. Pyramid, starting with closed questions and working toward open-ended questions.
2. Funnel, starting with open-ended questions and working toward closed questions.
3. Diamond, starting with closed, moving toward open-ended, and ending with closed questions.

Pyramid Structure



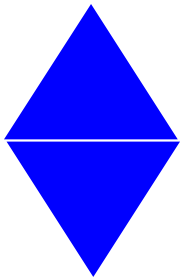
- Begins with very detailed, often closed questions
- Expands by allowing open-ended questions

Funnel Structure



- Begins with generalized, open-ended questions
- Concludes by narrowing the possible responses using closed questions

Diamond Structure



- A diamond-shaped structure begins in a very specific way
- Then more general issues are examined
- Concludes with specific questions
- Combines the strength of both the pyramid and funnel structures
- Takes longer than the other structures

Closing the Interview

- Always ask “Is there anything else that you would like to add?”
- Summarize and provide feedback on your impressions.
- Ask whom you should talk with next.
- Set up any future appointments.
- Thank them for their time and shake hands.

Interview Report

- Write as soon as possible after the interview.
- Provide an initial summary, then more detail.
- Review the report with the respondent.

Joint Application Design (JAD)

- Joint Application Design (JAD) can replace a series of interviews with the user community.
- JAD is a technique that allows the analyst to accomplish requirements analysis and design the user interface with the users in a group setting.

Benefits of JAD

The potential benefits of using JAD are:

- Time is saved, compared with traditional interviewing.
- Rapid development of systems.
- Improved user ownership of the system.
- Creative idea production is improved.

Drawbacks of Using JAD

Potential drawbacks of using JAD are:

- JAD requires a large block of time to be available for all session participants.
- If preparation is incomplete, the session may not go very well.
- If the follow-up report is incomplete, the session may not be successful.
- The organizational skills and culture may not be conducive to a JAD session.

Questionnaires

Questionnaires are useful in gathering information from key organization members about:

- Attitudes.
- Beliefs.
- Behaviors.
- Characteristics.

When to Use Questionnaires

Questionnaires are valuable if:

- Organization members are widely dispersed.
- Many members are involved with the project.
- Exploratory work is needed.
- Problem solving prior to interviews is necessary.

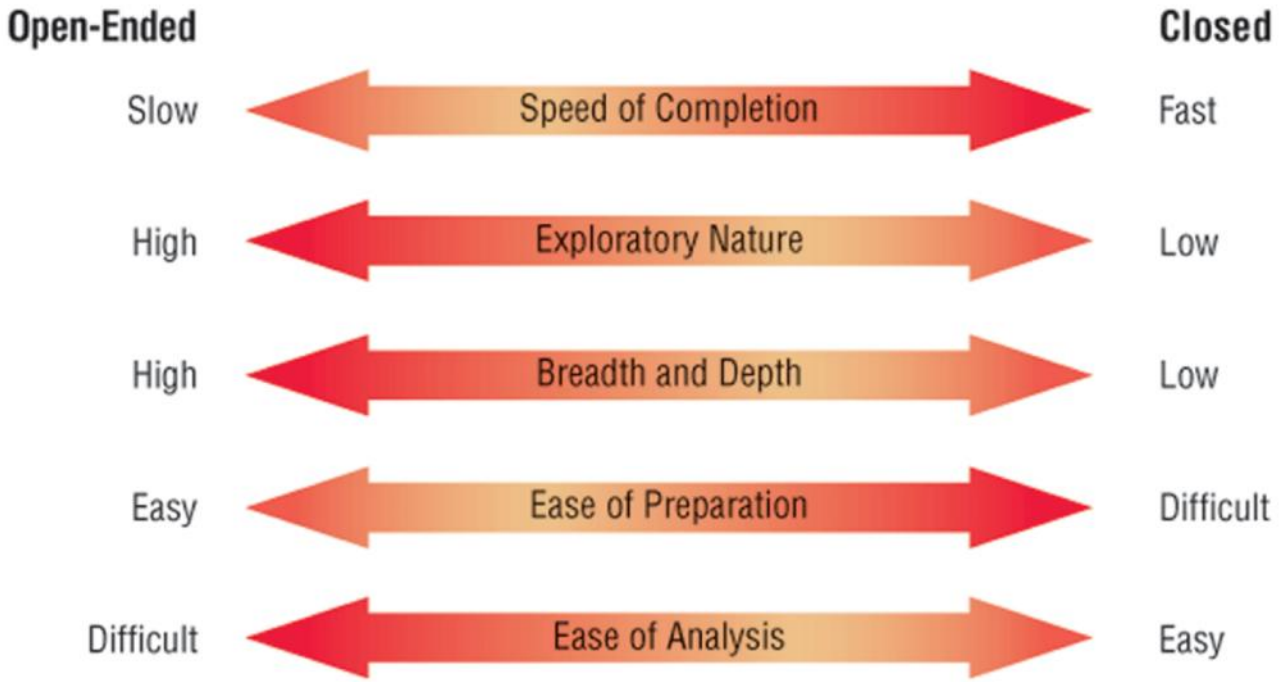
Question Types

Questions are designed as either:

- Open-ended
  - Well suited for getting opinions.
- Closed
  - Use when all the options may be listed.

Open-Ended and Closed Questions

Figure 4.12 Trade-offs between the use of open-ended and closed questions on questionnaires.



Questionnaire Language

Questionnaire language should be:

- Simple.
- Specific.
- Free of bias.
- Technically accurate.
- Addressed to those who are knowledgeable.
- Appropriate for the reading level of the respondent.

Measurement Scales

- The two different forms of measurement scales are :
  - Nominal.
  - Interval.

Nominal Scales

- Nominal scales are used to classify things into categories.
- It is the weakest form of measurement.

What type of software do you use the most?
1 = Word Processor
2 = Spreadsheet
3 = Database
4 = An Email Program

Interval Scales

- An interval scale is used when the intervals are equal.
- Examples of interval scales include the Fahrenheit or centigrade scale.

How useful is the support given by the Technical Support Group?

NOT USEFUL AT ALL				EXTREMELY USEFUL
1	2	3	4	5

Designing the Questionnaire

Good response rates can be achieved with consistent control of questionnaire.

- Allow large white space.
- Allow large space to write or type in responses.
- Make it easy for respondents to clearly mark their answers.
- Be consistent in style.

Order of Questions

- Place most important questions first.
- Group items of similar content together.
- Introduce less complex questions first.

Web Form Questionnaires

Controls (fields) used on Web forms:

- Single line text box.
- Scrolling text box, used for one or more paragraphs of text.
- Check box for yes-no or true-false answers.
- Radio button for mutually exclusive yes-no or true-false answers.
- Drop-down menu for selection from a list.
- Submit or Clear buttons.

Methods of Administering the Questionnaire

Methods of administering the questionnaire include:

- Convening all concerned respondents together at one time.
- Personally administering the questionnaire.
- Mailing questionnaires.
- Administering over the Web or via email.

Electronically Submitting Questionnaires

Administering a questionnaire electronically has the following benefits:

- Reduced costs.
- Collecting and storing the results electronically.

End

Using Dataflow Diagrams

7

Systems Analysis and Design, 7e

Kendall & Kendall

Data Flow Diagrams

- Graphically characterize data processes and flows in a business system
- Represent:
  - System inputs
  - Processes
  - outputs

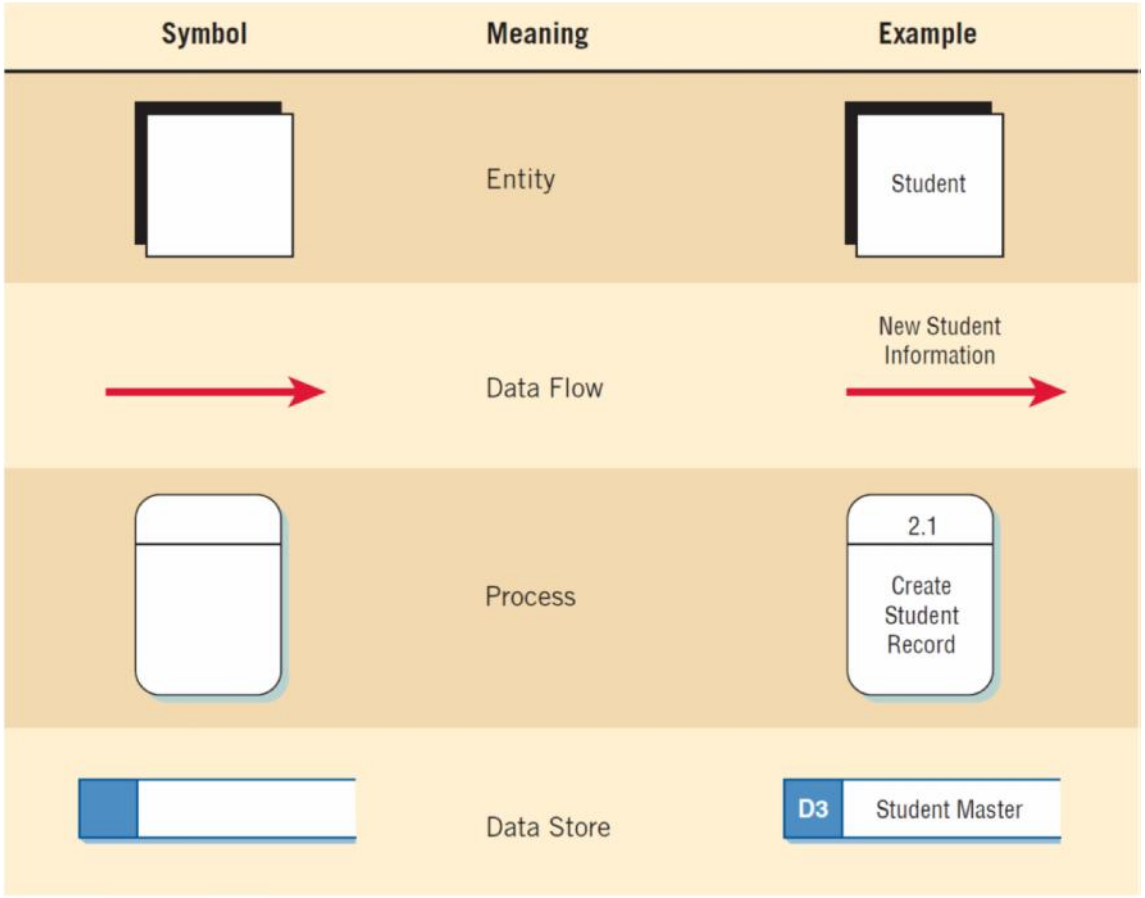
Major Topics

- Data flow diagram symbols
- Data flow diagram levels
- Creating data flow diagrams
- Physical and logical data flow diagrams
- Communicating Using Data Flow Diagrams

Advantages of the Data Flow Approach

- Analysis the technical implementation too early
- Understanding of the interrelatedness of systems and subsystems
- Communicating current system knowledge to users
- Analysis of the proposed system

Figure 7.1 The four basic symbols used in data flow diagrams, their meanings, and examples



External Entities

- Represent another department, a business, a person, or a machine
- A source or destination of data, outside the boundaries of the system
- Should be named with a noun

Data Flow

- Shows movement of data from one point to another
- Described with a noun
- Arrowhead indicates the flow direction
- Represents data about a person, place, or thing

Process

- Denotes a change in or transformation of data
- Represents work being performed in the system
- Naming:
  - Assign the name of the whole system when naming a high-level process
  - To name a major subsystem attach the word subsystem to the name
  - Use the form verb-adjective-noun for detailed processes

Data Store

- A depository for data that allows examination, addition, and retrieval of data
- Named with a noun, describing the data
- Data stores are usually given a unique reference number, such as D1, D2, D3
- Represents a:
  - Filing cabinet
  - Database
  - Computerized file

Creating the Context Diagram

- The highest level in a data flow diagram
- Contains only one process, representing the entire system
- The process is given the number 0
- All external entities, as well as Major data flows are shown

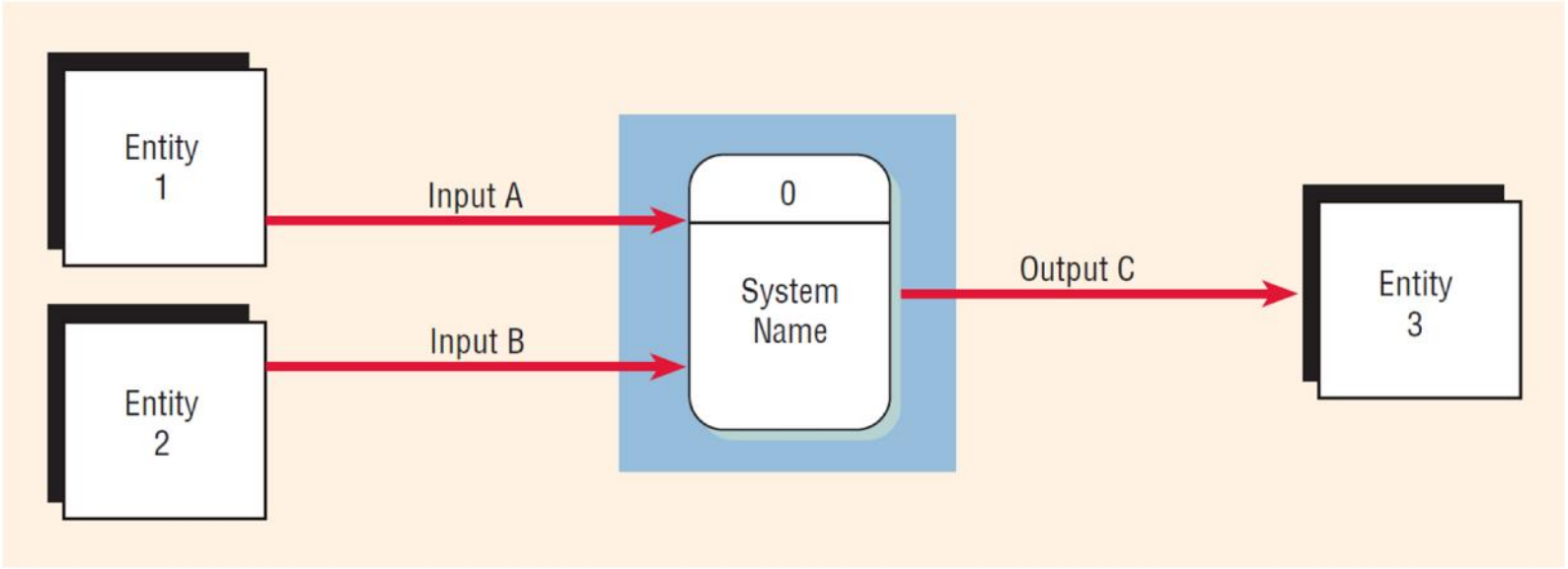
Creating the Context Diagram

Basically the context diagram consists of:

1. one process – depicting the entire system
2. external entities
3. data flows from the external entities to the process

The diagram does not contain any data stores.

Figure 7.3 Context diagram



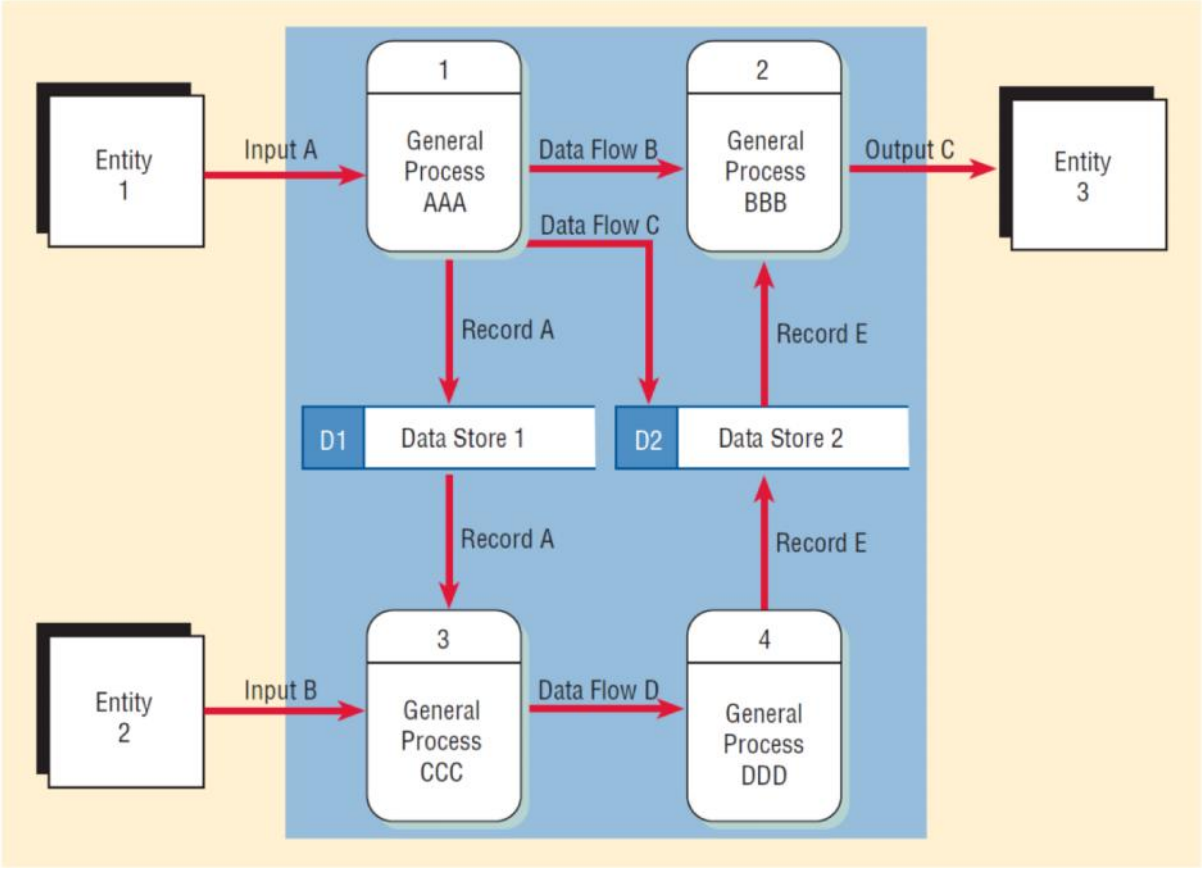
Drawing Diagram 0

- The explosion of the context diagram
- May include up to nine processes
  - Including more than nine processes will result in a cluttered diagram that is difficult to understand.
- Each process is numbered
- Major data stores and all external entities are included

Drawing Diagram 0 (Continued)

- Start with the data flow from an entity on the input side
- Work backwards from an output data flow
- Examine the data flow to or from a data store
- Analyze a well-defined process
- Take note of any fuzzy areas

Figure 7.3 Note the greater detail in diagram 0



Data Flow Diagram Levels

- Data flow diagrams are built in layers
- The top level is the Context level
- Each process may explode to a lower level
- The lower level diagram number is the same as the parent process number
- Processes that do not create a child diagram are called primitive

Creating Child Diagrams

- The process on Diagram 0 that is exploded is called the parent process, and the diagram that results is called the child diagram.

- A child diagram cannot produce output or receive input that the parent process does not also produce or receive

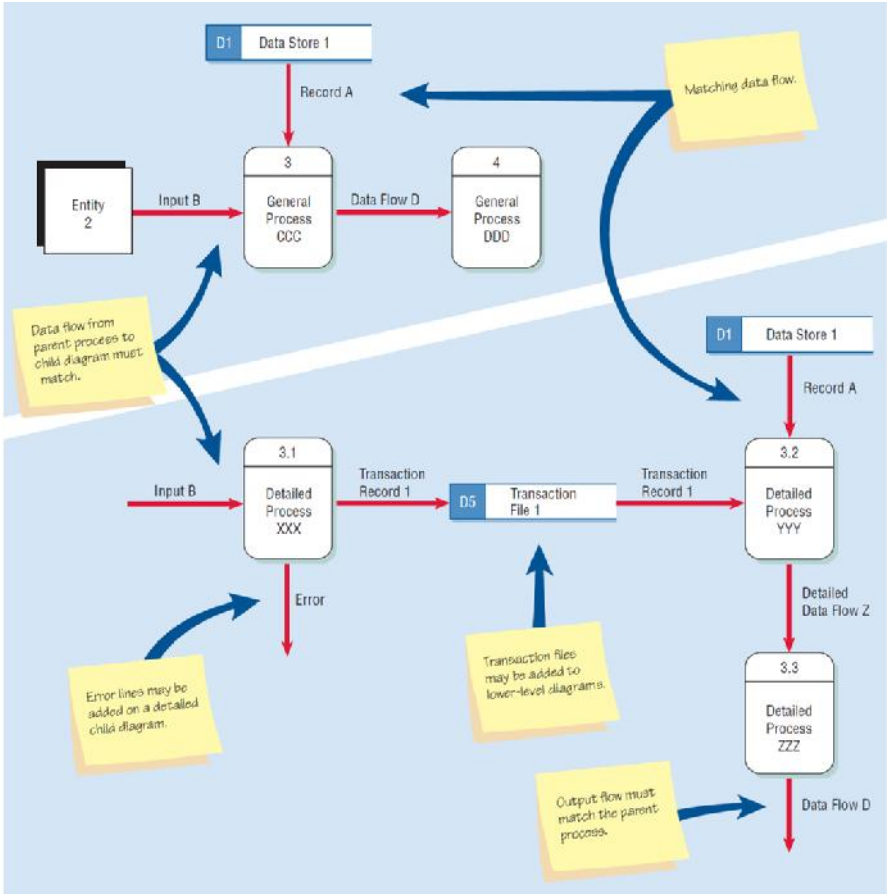
Creating Child Diagrams (Continued)

- The child process is given the same number as the parent process
  - On Diagram 3, the processes would be numbered 3.1, 3.2, 3.3, and so on.
  - This allows the analyst to trace a series of processes through many levels of explosion.

Creating Child Diagrams (Continued)

- Entities are usually not shown on the child diagrams below Diagram 0
- If the parent process has data flow connecting to a data store, the child diagram may include the data store as well

Figure 7.4 Differences between the parent diagram (above) and the child diagram (below)



DFD Example

Draw the DFD for a distance education university. The enrolment process works as follows:

Students send in an application form containing their personal details, and their desired course The university checks that the course is available and that the student has necessary academic qualifications. If the course is available the student is enrolled in the course, and the university confirms the enrolment by sending a confirmation letter to the student. If the course is unavailable the student is sent a rejection letter.

DFD Example

Step 1
Read the problem description carefully looking for:

- People / organizations / things that supply information to or use information from the system => external entities (EE)
- Actions/doing words/verbs => Processes (P)
- Movement/exchange of information/data between external entities to processes, and processes to processes => data flows (DF)
- Store/record information/data => data stores(DS)

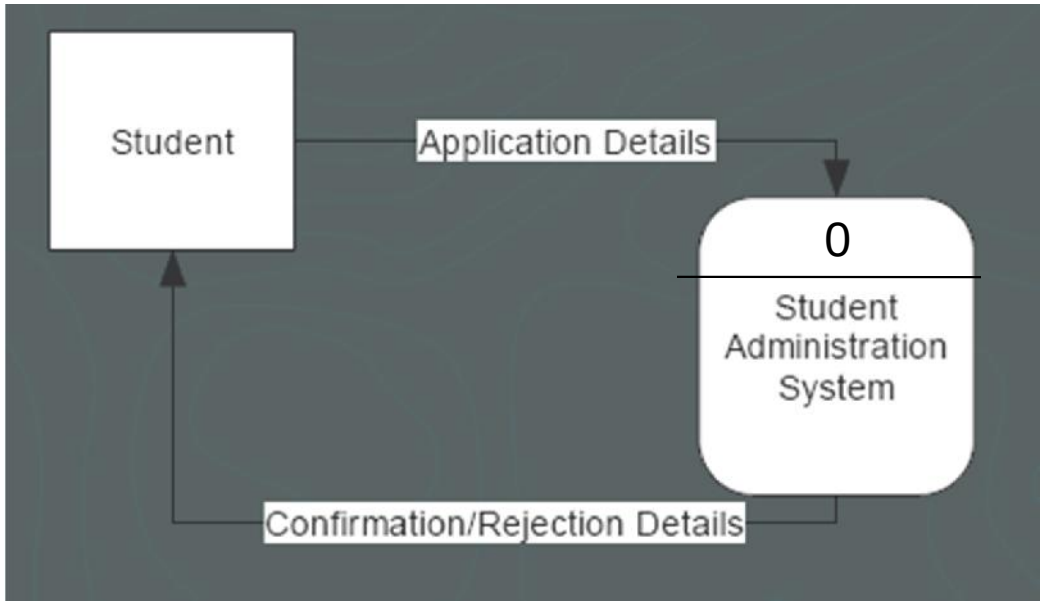
DFD Example

Output :

- A student (EE) sends in an application form (DF) containing their personal details, and their desired course.
- The university checks (P) that the course is available.
- If the course is available the student is enrolled (P) in the course, and the university confirms (P) the enrolment by sending a confirmation letter (DF) that they are registered for the course to the student.
- Or if the course is unavailable the student is sent a rejection letter (DF).

DFD Example
Context diagram

External entity - Student
Process - Student Administration process application
Data Flows - Application Form, Confirmation/Rejection Letter



DFD Example

System/Level 0 DFD

External entity - Student

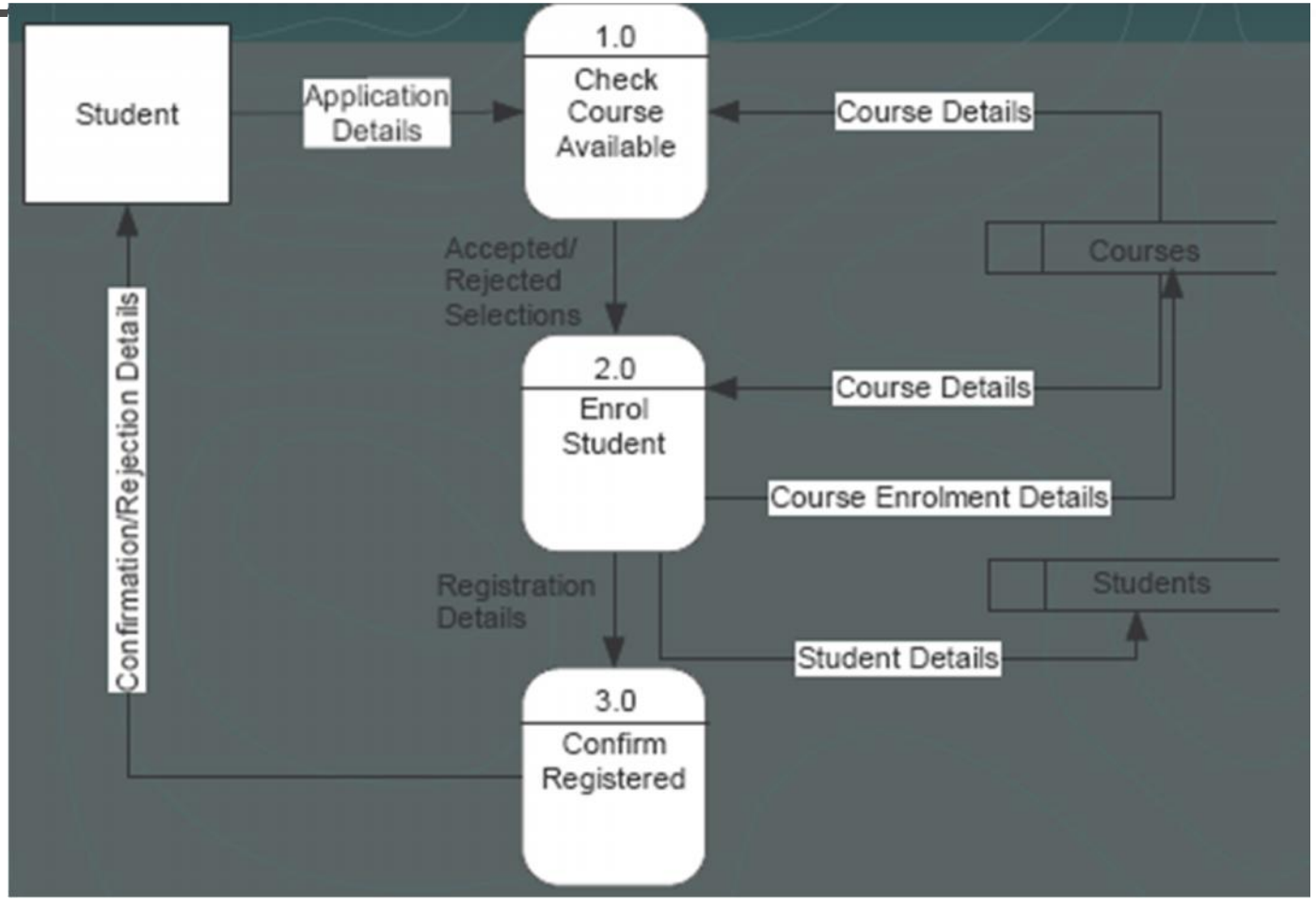
Processes - Check available, Enroll student, Confirm Registration

Data Flows - Application Form, Course Details, Course Enrolment Details, Student Details, Confirmation/Rejection Letter

Data Stores - Courses, Students.

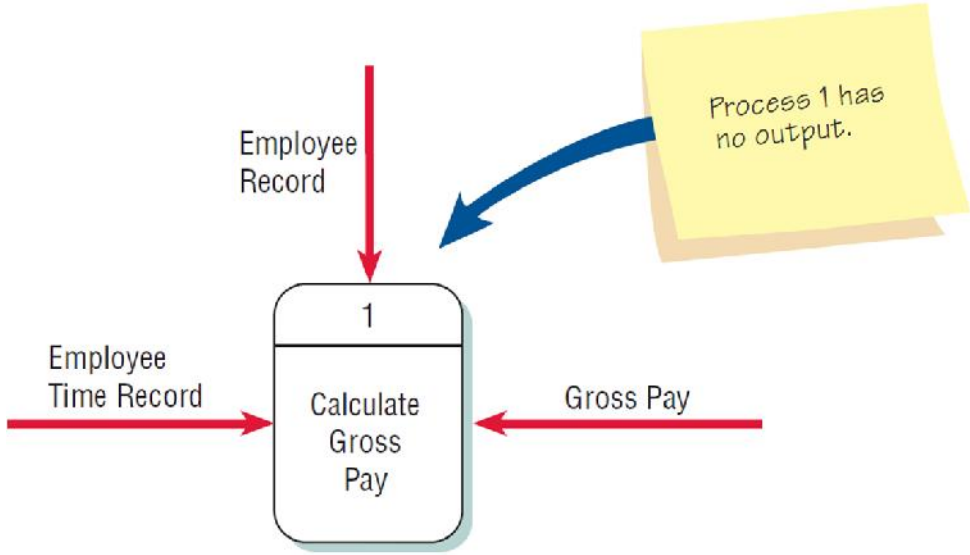
DFD Example

System / Level 0 DFD



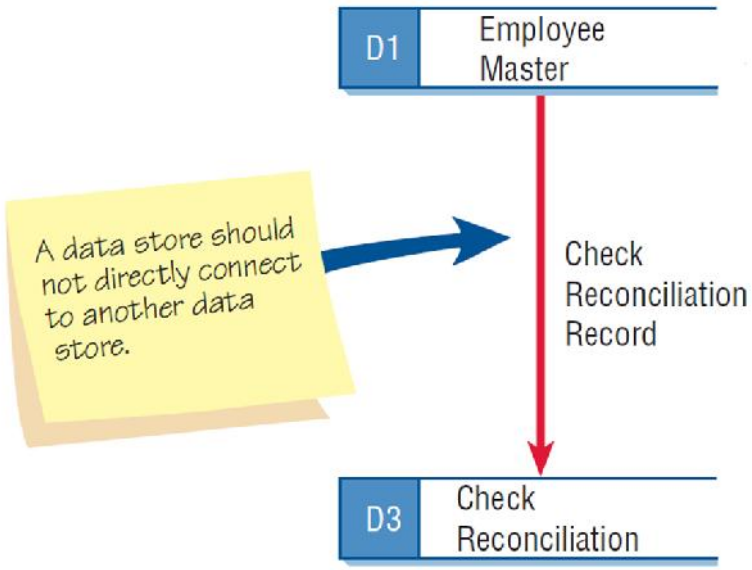
Checking the Diagrams for Errors

- Forgetting to include a data flow or pointing an arrow in the wrong direction



Checking the Diagrams for Errors (Continued)

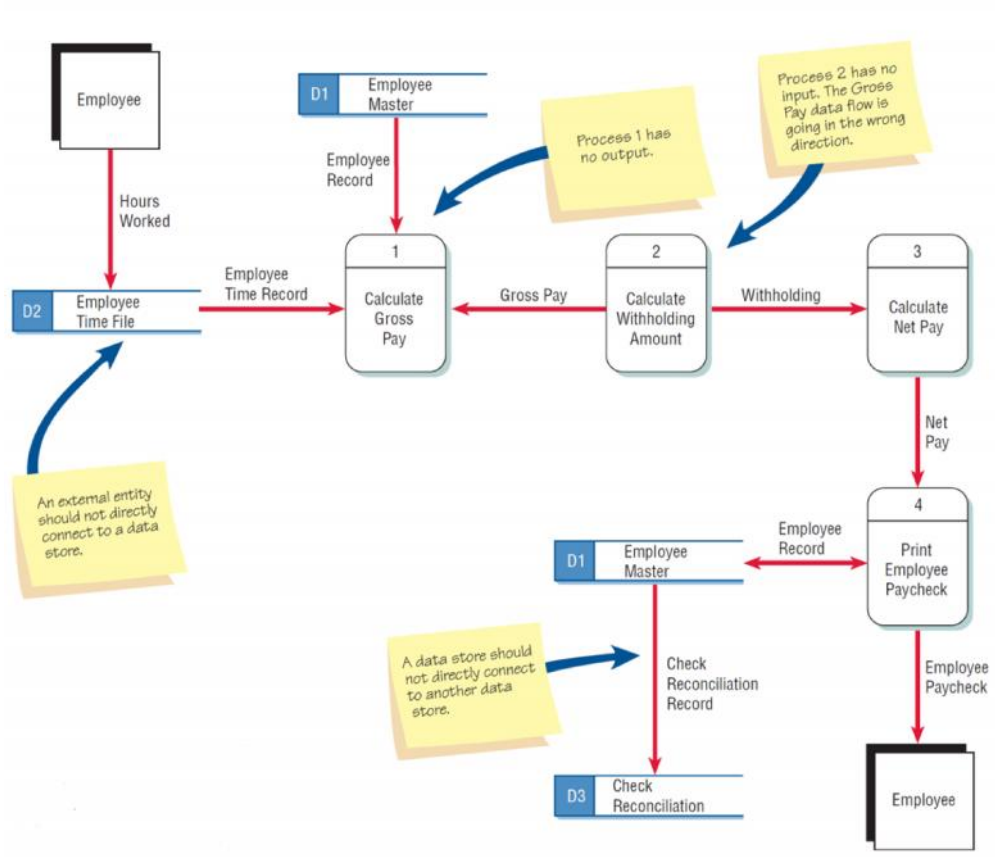
- Connecting data stores and external entities directly to each other



Checking the Diagrams for Errors (Continued)

- Incorrectly labeling processes or data flow
- Including more than nine processes on a data flow diagram
- Omitting data flow

Figure 7.5 Typical errors that can occur in a data flow diagram (payroll example)



End

Logical and Physical Data Flow Diagrams

- Logical
  - Focuses on the business and how the business operates
  - Not concerned with how the system will be constructed
  - Describes the business events that take place and the data required and produced by each event
- Physical
  - Shows how the system will be implemented
  - Depicts the system

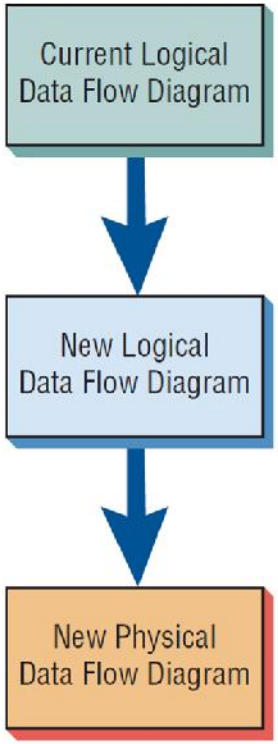
Figure 7.7 Features common of logical and physical data flow diagrams

Design Feature	Logical	Physical
What the model depicts	How the business operates.	How the system will be implemented (or how the current system operates).
What the processes represent	Business activities.	Programs, program modules, and manual procedures.
What the data stores represent	Collections of data regardless of how the data are stored.	Physical files and databases, manual files.
Type of data stores	Show data stores representing permanent data collections.	Master files, transition files. Any processes that operate at two different times must be connected by a data store.
System controls	Show business controls.	Show controls for validating input data, for obtaining a record (record found status), for ensuring successful completion of a process, and for system security (example: journal records).

The progression of creating data flow diagrams is:

1. Analyze the current system (current logical DFD).
2. Add features the new system should include (the proposed logical DFD).
3. Finally the best methods for implementing the new system should be developed (the physical DFD).

Figure 7.8 The progression of models from logical to physical



Derive the logical data flow diagram for the current system by examining the physical data flow diagram and isolating unique business activities.

Create the logical data flow diagram for the new system by adding the input, output, and processes required in the new system to the logical data flow diagram for the current system.

Derive the physical data flow diagram by examining processes on the new logical diagram. Determine where the user interfaces should exist, the nature of the processes, and necessary data stores.

Reasons for Developing Logical Data Flow Diagrams

- Better communication with users
- More stable systems
- Better understanding of the business by analysts
- Flexibility and maintenance
- Elimination of redundancy and easier creation of the physical model

Reasons for Developing Physical Data Flow Diagrams

- Clarifying which processes are performed by humans and which are automated
- Describing processes in more detail
- Sequencing processes that have to be done in a particular order
- Identifying temporary data stores
- Specifying actual names of files and printouts
- Adding controls to ensure the processes are done properly

Figure 7.10 Physical data flow diagrams contain many items not found in logical data flow diagrams

Contents of Physical Data Flow Diagrams

- Manual processes
- Processes for adding, deleting, changing, and updating records
- Data entry and verifying processes
- Validation processes for ensuring accurate data input
- Sequencing processes to rearrange the order of records
- Processes to produce every unique system output
- Intermediate data stores
- Actual file names used to store data
- Controls to signify completion of tasks or error conditions

Summary

- Data flow diagrams
  - Structured analysis and design tools that allow the analyst to comprehend the system and subsystems visually as a set of interrelated data flows
- DFD symbols
  - Rounded rectangle
  - Double square
  - An arrow
  - Open-ended rectangle

Summary (Continued)

- Creating the logical DFD
  - Context-level data flow diagram
  - Level 0 logical data flow diagram
  - Child diagrams
- Creating the physical DFD
  - Create from the logical data flow diagram

Chapter 9
Describing Process Specifications and Structured Decisions

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Process specifications
- Business rules
- Structured English
- Decision tables
- Decision trees

Process Specifications

- Process specifications are created for primitive processes and some higher level processes on a data flow diagram.

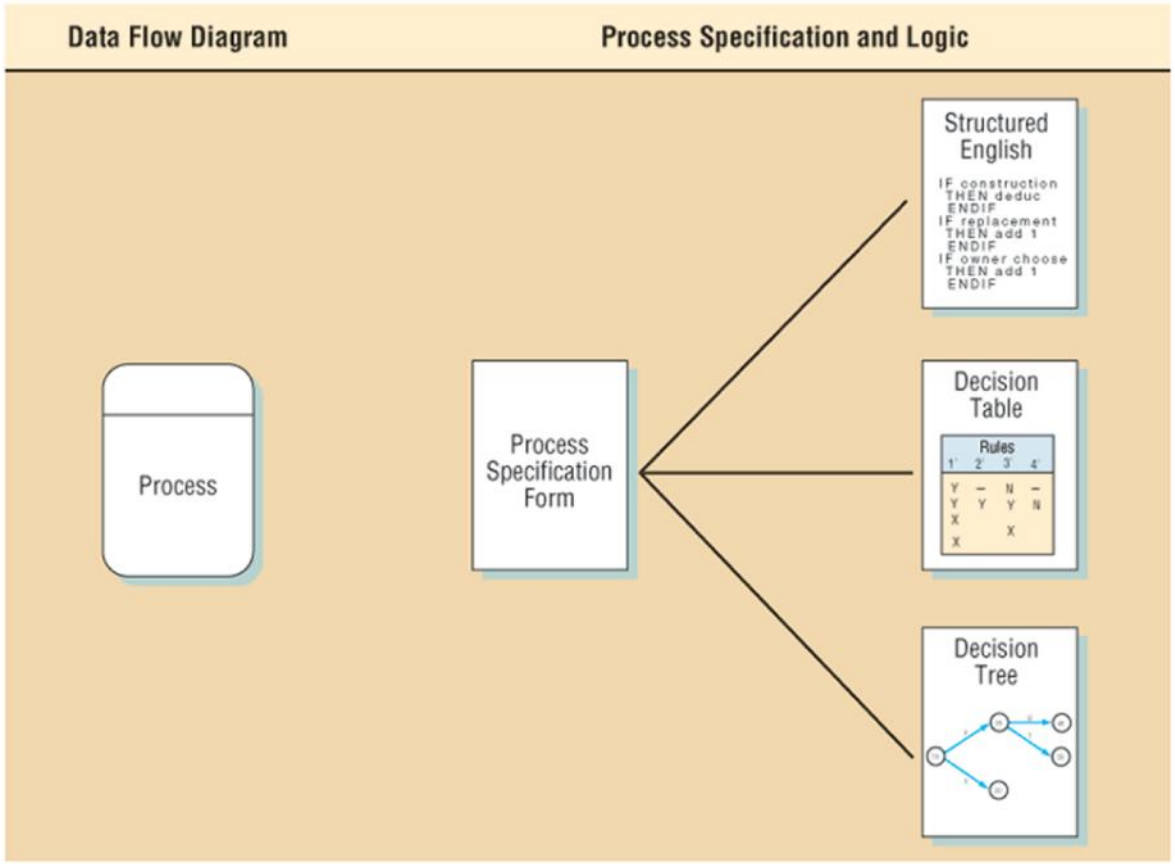
Goal of Creating Process Specifications

The goals of producing process specifications are:

- Reduce process ambiguity.
- Obtain a precise description of what is accomplished.
- Validate the system design, including data flow diagrams.

Data Flow Diagram and Process Specifications

Figure 9.1 How process specifications relate to the data flow diagram.



Process Specification Format

- Process specifications link the process to the DFD and the data dictionary.
- The following information should be entered:
  1. The process number, which must match the process ID on the data flow diagram.
     - This allows an analyst to work or review any process.

Process Specification Format (Continued)

2. The process name, the same as displays within the process symbol on the DFD.
3. A brief description of what the process accomplishes.
4. A list of input and output data flow, using the names found on the data flow diagram.
5. Data names used in the formulae or logic should match the data dictionary, for consistency and good communication.

Process Specification Format (Continued)

6. An indication of the type of process, whether it is batch, online, or manual.
   - All online processes require screen designs.
   - All manual processes should have well-defined procedures for employees performing the process tasks.
7. If the process has prewritten code for it, include the name of the subprogram or function.

Process Specification Format (Continued)

8. A description of the process logic.
   - This should state policy and business rules, not computer language pseudocode.
9. A reference to further information, such as a structured English description, a decision table, or tree depicting the logic.
10. List any unresolved issues.
    - These issues form the basis of the questions used for a follow-up interview.

Business Rules

Business rules include the following:

- Definitions of business terms
- Business conditions and actions
- Data integrity constraints
- Logical inferences
- Processing sequences
- Relationships among facts about the business

Process Specification Example Part 1

Number 1
Name Add Customer Order

Description Key and add the Customer Order.
The order should be edited for correct information.
Customer and Item master files are updated.

Input Data Flow

Customer Order Form from the Customer
Customer Record from data store D1, Customer Master File
Item Record from data store D2, Item Master File

Process Specification Example Part 2

Output Data Flow

- Pending Order to data store D3, Order File
- Backordered Item Record to the Inventory Control Department
- Updated Customer and Item records

Type of process Online

Structured English

- Structured English is based on structured logic and Simple English statements such as add, multiply, move, and so on.
- It is an appropriate technique for analyzing the system when structured decisions are not complex.

Steps to Use Structured English

- The following steps are needed:
  - Express all logic in terms of sequential structures, decision structures, case structures, or iterations.
  - Use and capitalize accepted keywords such as IF, THEN, ELSE, DO, and PERFORM.
  - Indent blocks of statements to show their hierarchy (nesting) clearly.

Steps to Use Structured English (Continued)

- Be careful when using "and" and "or ”.
- Avoid confusion when using logical comparisons such as "greater than" and "greater than or equal to”.

Structured English

Figure 9.5 Examples of logic expressed in a sequential structure, a decision structure, a case structure, and an iteration.

Structured English Type	Example
Sequential Structure A block of instructions in which no branching occurs	Action #1 Action #2 Action #3
Decision Structure Only IF a condition is true, complete the following statements; otherwise, jump to the ELSE	IF Condition A is True THEN implement Action A ELSE implement Action B ENDIF
Case Structure A special type of decision structure in which the cases are mutually exclusive (if one occurs, the others cannot)	IF Case #1 implement Action #1 ELSE IF Case #2 Implement Action #2 ELSE IF Case #3 Implement Action #3 ELSE IF Case #4 Implement Action #4 ELSE print error ENDIF
Iteration Blocks of statements that are repeated until done	DO WHILE there are customers. Action #1 ENDDO

Advantages of Structured English

- Clarifying the logic and relationships found in human languages
- An effective communication tool, and easy to teach and understand

Decision Tables

- Decision tables provide a way to examine, describe, and document decisions using a table.
- They are used to:
  - Describe the conditions.
  - Identify possible decision alternatives.
  - Indicate actions should be performed.
  - Describe actions.

Decision Table Format

Figure 9.9 The standard format used for presenting a decision table.

Conditions and Actions	Rules
Conditions	Condition Alternatives
Actions	Action Entries

Decision Table Example

Figure 9.11 Constructing a decision table for deciding which catalog to send to customers who order only from selected catalogs.

Conditions and Actions	Rules 1	2	3	4	5	6	7	8
Customer ordered from Fall catalog	Y	Y	Y	Y	N	N	N	N
Customer ordered from Christmas catalog	Y	Y	N	N	Y	Y	N	N
Customer ordered from specialty catalog	Y	N	Y	N	Y	N	Y	N
Send out this year's Christmas catalog		X		X		X		X
Send out specialty catalog			X				X	
Send out both catalogs	X				X			

Decision Trees

- Decision trees are used when complex branching occurs in a structured decision process.
- Trees are also useful when it is essential to keep a group of decisions in a particular sequence.

Drawing Decision Trees

- First, identify all conditions and actions and the order and timing of these (if they are critical).
- Second, begin building the tree from left to right while making sure you are complete in listing all possible alternatives before moving over to the right.

Decision Tree Example

Figure 9.18 Drawing a decision tree to show the noncash purchase approval actions for a department store.

Decision Tree Advantages

Three advantages over a decision table are:

- The order of checking conditions and executing actions is immediately noticeable.
- Second, conditions and actions of decision trees are found on some branches but not on others.
- Third, compared to decision tables, decision trees are more readily understood by others in the organization.

Selecting a Structured Decision Analysis Technique

Guidelines are as follows:

- Use structured English when there are many repetitious actions or when communication to end users is important.
- Use decision tables when complex combination of conditions, actions, and rules are found.
- Use decision trees when the sequence of conditions and actions is critical or when not every condition is relevant to every action

Chapter 10
Preparing The Systems Proposal

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Systems proposal
- Determining hardware/software needs
- Tangible and intangible costs and benefits
- Systems proposal
- Using tables, graphs, and figures

Systems Proposal

In order to prepare the systems proposal analysts must use a systematic approach to:

1. Ascertain hardware and software needs.
2. Identify and forecast costs and benefits.
3. Compare costs and benefits.
4. Choose the most appropriate alternative.

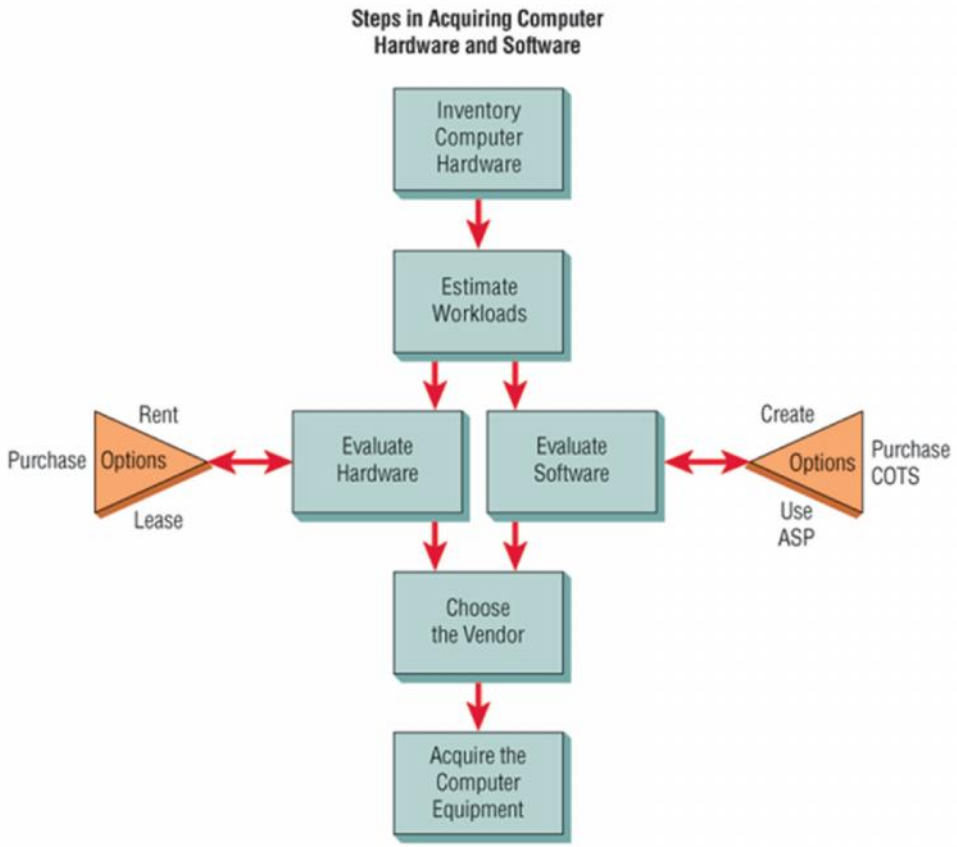
1. Ascertaining Hardware and Software Needs

Steps used to determine hardware and software needs:

1. Inventory computer hardware currently available.
2. Estimate current and projected workload for the system.
3. Evaluate the performance of hardware and software.
4. Choose the vendor according to the evaluation.
5. Obtain hardware and software from the vendor.

Steps in Acquiring Computer Hardware and Software

Figure 10.1 Steps in choosing hardware and software.



Hardware Inventory

When inventorying hardware you should document the following:

- Type of equipment.
- Status of equipment operation.
- Estimated age of equipment.
- Physical location of equipment.
- Department or person responsible for equipment.
- Any other important information.

Evaluating Hardware

Criteria for evaluating hardware:

- Time required for average transactions (including time for input and output).
- Total volume capacity of the system.
- Idle time of the central processing unit.
- Size of memory provided.

People that Evaluate Hardware

The people involved:

- Management.
- Users.
- Systems analysts.
- Help desk worker.

Purchasing, Leasing, or Renting Decision

There are three options for obtaining computer equipment:

- Buying.
- Leasing.
- Rental.

Buying

Advantages	Disadvantages
Cheaper than leasing or renting over the long run	Initial cost is high
Ability to change system	Risk of obsolescence
Provides tax advantages of accelerated depreciation	Risk of being stuck if choice is wrong
Full control	Full responsibility

Leasing

Advantages	Disadvantages
No capital is tied up	Company doesn't own the system when lease expires
No financing is required	Usually a heavy penalty for terminating the lease
Leases are lower than rental payments	Leases are more expensive than buying

Renting

Advantages	Disadvantages
No capital is tied up	Company doesn't own the computer
No financing is required	
Easy to change systems	Cost is very high because vendor assumes the risk (most expensive option)
Maintenance and insurance are usually included	

Evaluating Hardware Support

When evaluating hardware vendors, the selection needs to consider:

- Hardware support.
- Software support.
- Installation and training support.
- Maintenance support.
- Performance of the hardware.

Software Alternatives

Software may be:

- Custom created in-house.
- Purchased as COTS (commercial off-the-shelf) software.
- Provided by an application service provider (ASP).

Creating Custom Software

Advantages	Disadvantages
Specific response to specialized business needs	May be significantly higher initial costs compared to others
Innovation may give competitive edge	Ongoing maintenance
In-house staff for maintenance	Necessity of hiring or working with a development team
Pride of ownership	

Purchasing COTS Packages

Advantages	Disadvantages
Refined in the commercial world	Programming focused, not business focused
Increased reliability and functionality	Must accept existing features
Often lower initial cost	Limited customization
Already in use by other firms	Uncertain financial future of vendor
Help and training comes with software	Less ownership and commitment

Using an ASP

Advantages	Disadvantages
Organizations can focus on what they do best	Loss of control of data, systems, IT employees and schedules
No need to hire, train, retain a large IT staff	Concerns over financial viability, stability of ASP
There is no expenditure of employee time on IT tasks	Security, confidentiality, and privacy concerns
	Loss of potential strategic advantage regarding innovativeness

Software Evaluation

- Use the following to evaluating software packages:
  - Performance efficiency
  - Ease of use
  - Flexibility
  - Quality of documentation
  - Manufacturer support

Items in the Systems Proposal

- When preparing a systems proposal, systems analysts should arrange the following ten items in order:
  1. Cover letter.
  2. Title page of project.
  3. Table of contents.
  4. Executive summary (including recommendation).

Items in the Systems Proposal

5. Outline of systems study with appropriate documentation.
6. Detailed results of the systems study.
7. Systems alternatives (three or four possible solutions).
8. Systems analysts recommendations.
9. Summary.

Items in the Systems Proposal (Continued)

10. Appendices
    - Summary of phases.
    - Correspondence. (e-mail, Quotations)
    - Other material as needed.

Guidelines for Using Tables

Some guidelines to use tables effectively are:

- Integrate it into the body of the proposal.
- Try to fit the entire table vertically on a single page.
- Number and title the table at the top of the page.

Guidelines for Using Tables

Some guidelines to use tables effectively are (continued):

- Make the title descriptive and meaningful.
- Label each row and column.
- Use footnotes if necessary to explain detailed information contained in the table.

Guidelines for Using Graphs

Some guidelines for using graphs are:

- Choose a style of graph that communicates your intended meaning well.
- Integrate the graph into the proposal body.
- Give the graph a sequential figure number and a meaningful title.
- Label each axis, any lines, columns, bars, and pieces of the pie on the graph.

Types of Graphs

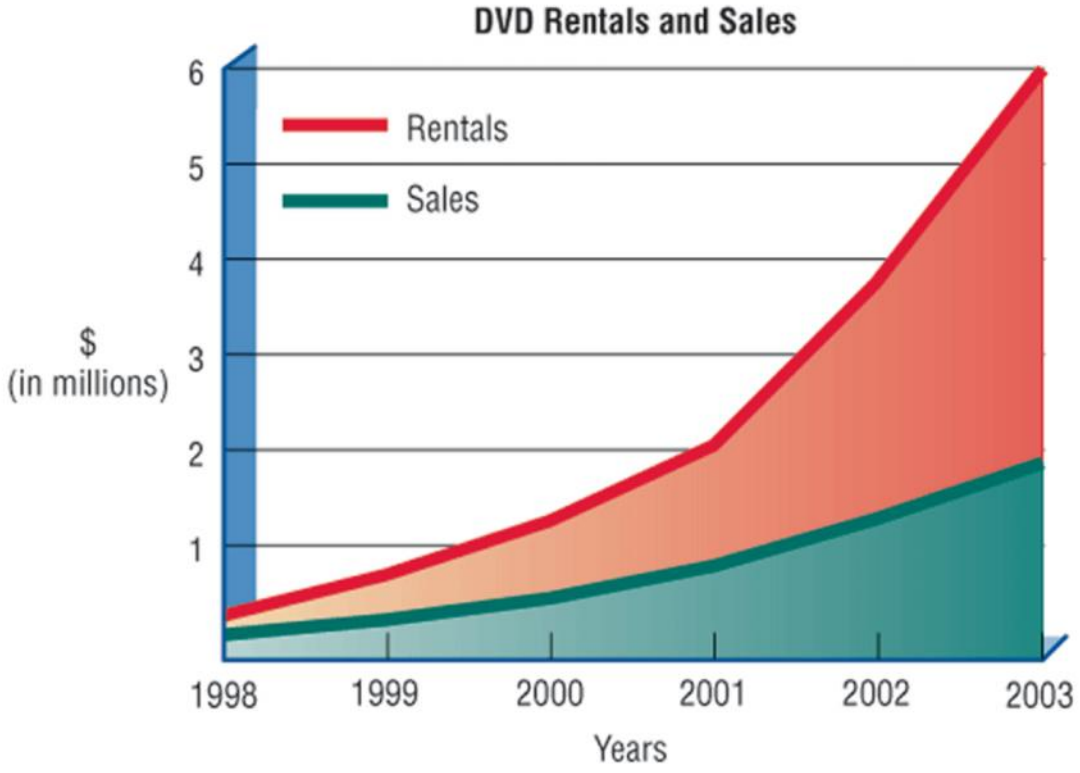
- Line graphs
- Column charts
- Bar charts
- Pie charts

Line Graphs

- Used to show change over time
- Changes of up to five variables on a single graph
- May show when lines intersect

Line Chart Example

Figure 10.20
An area chart is a form of line graph that may make more of an impact.



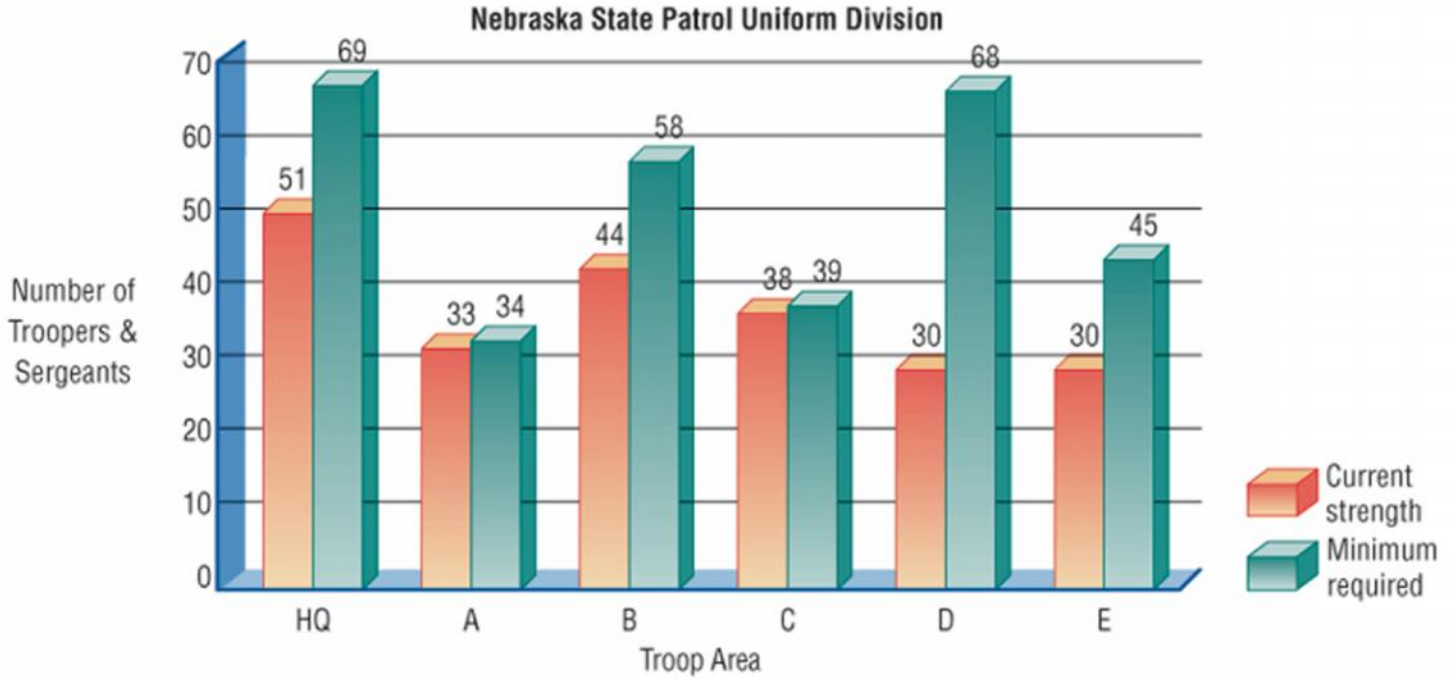
Column Charts

- Show a comparison between two or more variables
- Compare different variables at a particular point in time
- Easier to understand than line graphs

Column Chart Example

Figure 10.21

More than one variable can be displayed on a column chart by shading or coloring the column bars.

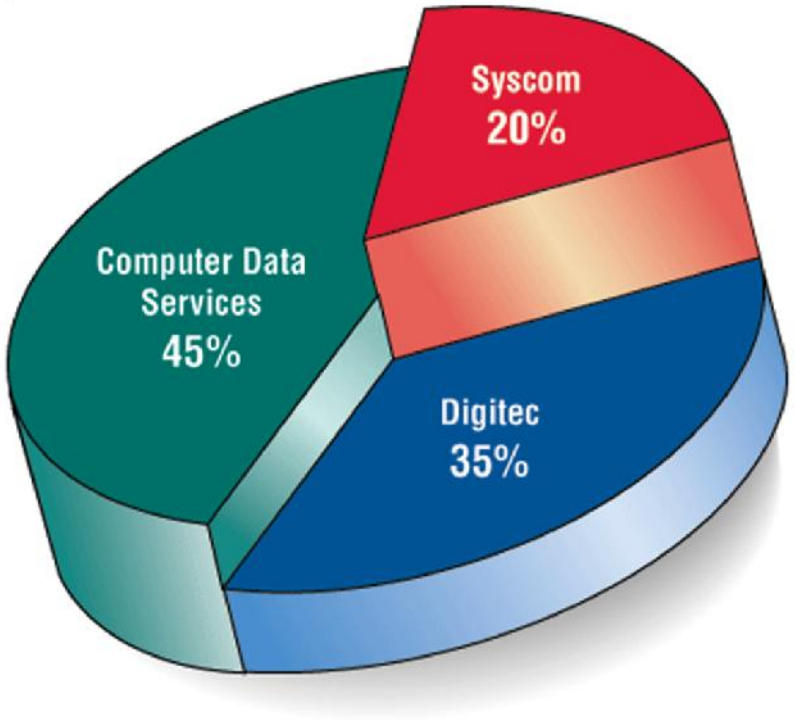


Pie Charts

- Used to show how 100 percent of a commodity is divided at a particular point in time
- Easier to read than 100 percent stacked column charts or 100 percent subdivided bar charts

Pie Chart Example

Figure 10.24 A pie chart is a visually appealing way to display how 100 percent of an entity is divided up at a particular time.



Oral Presentations

When delivering the oral presentation, keep in mind the principles of delivery:

- Project loudly enough so that the audience can hear you.
- Look at each person in the audience as you speak.
- Make visuals large enough so that the audience can see them.

http://www.componentsource.com

End