5

Systems Analysis

Overview

Chapter 5 builds on the introduction to information system development from Chapter 3. Specifically, this chapter provides a more detailed look at the phases that are collectively called *systems* analysis. This systems analysis process reinforces the information system building blocks that were developed over Chapters 1-4.

The book's hypothetical methodology (called FAST: The Framework for the Application of System Techniques) is used to teach an implementation of systems analysis process. This chapter provides a "task-level" overview of each phase.

The methodology as presented supports all contemporary paradigms including structured analysis, information engineering, object-oriented analysis, and accelerated development. Then, as the readers progress through the book, they will learn how to "plug and play" various tools and techniques into the methodological framework provided by FAST.

It is very important to recognize that it is not the purpose of Chapter 5 to actually teach the tools and techniques of systems analysis. It only seeks to place those tools and techniques into the context of a process. Subsequently, Chapters 6-10 teach the students the actual tools and methods introduced in Chapter 5.

Chapter to Course Sequencing

This chapter should follow either Chapter 3 or Chapter 4 in all but exceptional course scenarios. Chapter 3 introduced the full life cycle of phases. Chapter 4 placed those phases into a project management framework.

For sequel courses that focus on systems design, this chapter may be assigned as a review with no planned course lectures.

What's Different Here and Why?

The following changes have been made to the seventh edition of the information systems development chapter:

1. As with all chapters, we have streamlined the SoundStage episode into a quick narrative introduction to the concepts presented the chapter.
2. The sixth edition's discussions of structured analysis and information engineering have been combined into a single discussion of "traditional

approaches." Though the distinction between structured analysis and information engineering has been retained, this revision better contrasts the traditional approaches with the newer object-oriented approach.

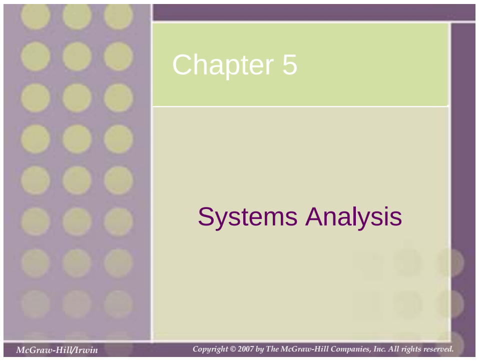
3. We have added a new key term for scope to reinforce this concept.

4. We have added the Context Diagram as a tool under Task 2.1 - Understand the Problem Domain. Some OO methodologists use this as a first step. Following any methodology, a context diagram can be a good first modeling exercise. We have also added a context diagram as an optional assignment in Milestone 2 of the seventh edition case studies.

Lesson Planning Notes for Slides

The following instructor notes, keyed to slide images from the PowerPoint repository, are intended to help instructors integrate the slides into their individual lesson plans for this chapter.

Slide 1



Chapter 5

Systems Analysis

McGraw-Hill/Irwin

Copyright © 2007 by The McGraw-Hill Companies, Inc. All rights reserved.

slide appearance after initial mouse click in slide show mode

This repository of slides is intended to support the named chapter. The slide repository should be used as follows:

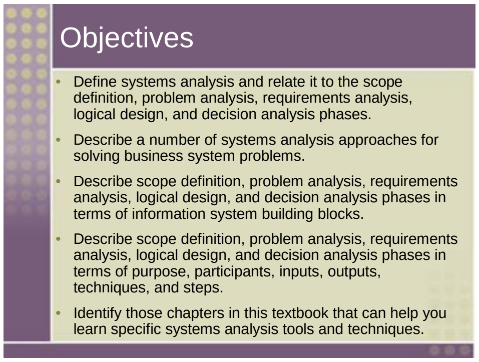
Copy the file to a unique name for your course and unit.

Edit the file by deleting those slides you don’t want to cover, editing other slides as appropriate to your course, and adding slides as desired.

Print the slides to produce transparency masters or print directly to film or present the slides using a computer image projector.

Most slides include instructor notes. In recent versions of PowerPoint, notes by default display in a window under the slide. The instructor notes are also reprinted below.

Slide 2



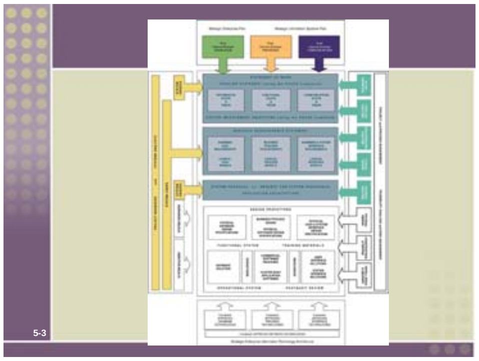
Objectives

- Define systems analysis and relate it to the scope definition, problem analysis, requirements analysis, logical design, and decision analysis phases.
- Describe a number of systems analysis approaches for solving business system problems.
- Describe scope definition, problem analysis, requirements analysis, logical design, and decision analysis phases in terms of information system building blocks.
- Describe scope definition, problem analysis, requirements analysis, logical design, and decision analysis phases in terms of purpose, participants, inputs, outputs, techniques, and steps.
- Identify those chapters in this textbook that can help you learn specific systems analysis tools and techniques.

Teaching Notes

Although some of the tools and techniques of systems analysis are previewed in this chapter, it is not the intent of this chapter to teach those tools and techniques. This chapter teaches only the *process* of systems analysis. The tools and techniques will be taught in the subsequent six chapters.

Slide 3

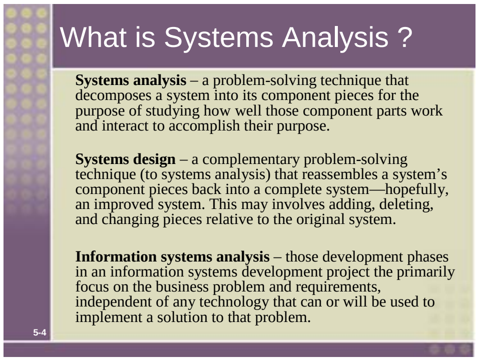


5-3

Teaching Notes

This slide shows the how this chapter's content fits with the building blocks framework used throughout the textbook. This chapter covers at a high level all the systems analysis phases from the perspective of analysts, users, and owners.

Slide 4



What is Systems Analysis ?

Systems analysis – a problem-solving technique that decomposes a system into its component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose.

Systems design – a complementary problem-solving technique (to systems analysis) that reassembles a system's component pieces back into a complete system—hopefully, an improved system. This may involve adding, deleting, and changing pieces relative to the original system.

Information systems analysis – those development phases in an information systems development project the primarily focus on the business problem and requirements, independent of any technology that can or will be used to implement a solution to that problem.

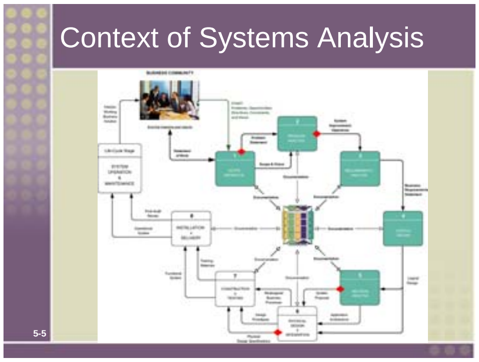
5-4

Teaching Notes

Systems modeling corresponds precisely with this classical definition of systems analysis and design.

Systems design is sometimes called Systems Synthesis

Slide 5

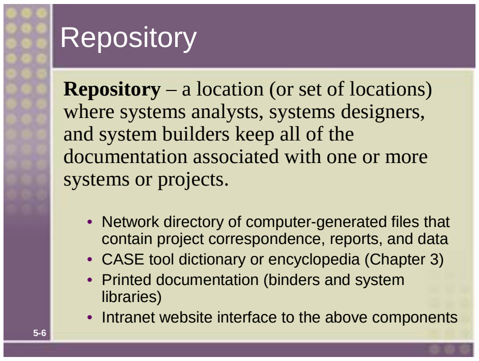


Context of Systems Analysis

5-5

Teaching Notes
This context comes directly from Chapter 3. The blue processes and the blue and black data flows define systems analysis.

Slide 6



Repository

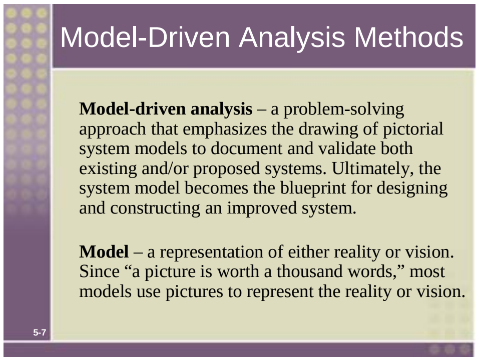
Repository – a location (or set of locations) where systems analysts, systems designers, and system builders keep all of the documentation associated with one or more systems or projects.

- Network directory of computer-generated files that contain project correspondence, reports, and data
- CASE tool dictionary or encyclopedia (Chapter 3)
- Printed documentation (binders and system libraries)
- Intranet website interface to the above components

5-6

No additional notes

Slide 7



Model-Driven Analysis Methods

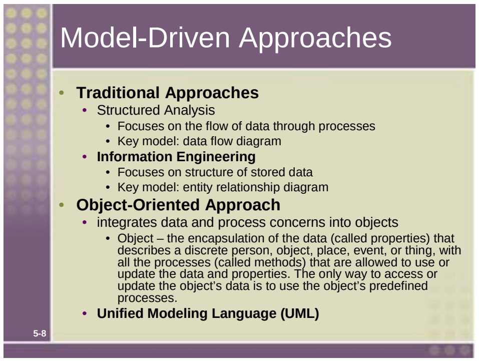
Model-driven analysis – a problem-solving approach that emphasizes the drawing of pictorial system models to document and validate both existing and/or proposed systems. Ultimately, the system model becomes the blueprint for designing and constructing an improved system.

Model – a representation of either reality or vision. Since "a picture is worth a thousand words," most models use pictures to represent the reality or vision.

5-7

Teaching Notes
Some books use the term "computer technology." We prefer the more contemporary term, "information technology" as a superset of computer technology.

Slide 8



Model-Driven Approaches

- **Traditional Approaches**
  - Structured Analysis
    - Focuses on the flow of data through processes
    - Key model: data flow diagram
  - **Information Engineering**
    - Focuses on structure of stored data
    - Key model: entity relationship diagram
- **Object-Oriented Approach**
  - integrates data and process concerns into objects
    - Object – the encapsulation of the data (called properties) that describes a discrete person, object, place, event, or thing, with all the processes (called methods) that are allowed to use or update the data and properties. The only way to access or update the object's data is to use the object's predefined processes.
  - **Unified Modeling Language (UML)**

5-8

Conversion Notes

The sixth edition treated structured analysis, information engineering, and object-oriented as three different model-driven approaches. The seventh edition groups the first two as traditional approaches.

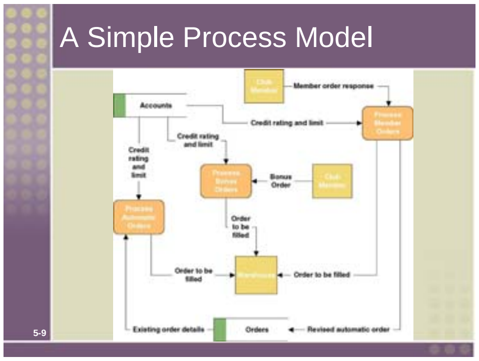
Teaching Notes

These different methods are just different approaches to the same thing. Structure analysis emphasizes processes, Information Engineering emphasizes data, and Object-oriented analysis emphasizes the integration of processes and data.

Information engineering is more complex and comprehensive than the oversimplified presentation in this edition's chapter. But we have found few organizations that still practice pure IE. But many organizations still practice data-driven analysis and design.

Slide 9

A Simple Process Model



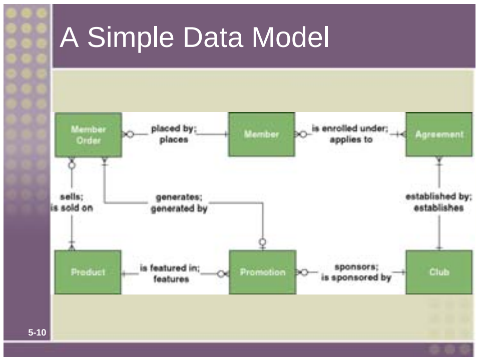
5-9

Teaching Notes

It is not the intent to teach the tool in this chapter. DFDs will be taught in Chapter 9.

Slide 10

A Simple Data Model

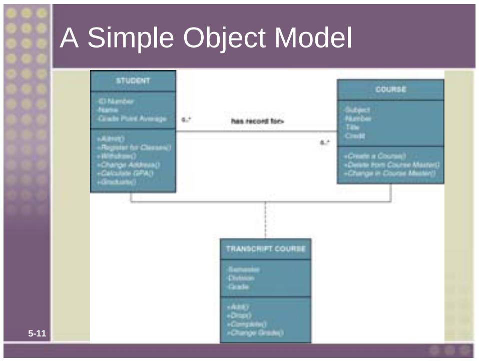


5-10

Teaching Notes

It is not the intent to teach the tool in this chapter. ERDs will be taught in Chapter 8.

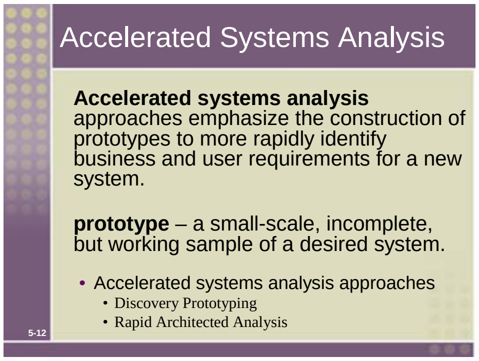
Slide 11



A Simple Object Model

Teaching Notes
It is not the intent to teach the tool in this chapter. Object modeling will be taught in Chapters 10 and 18.

Slide 12



Accelerated Systems Analysis

Accelerated systems analysis approaches emphasize the construction of prototypes to more rapidly identify business and user requirements for a new system.

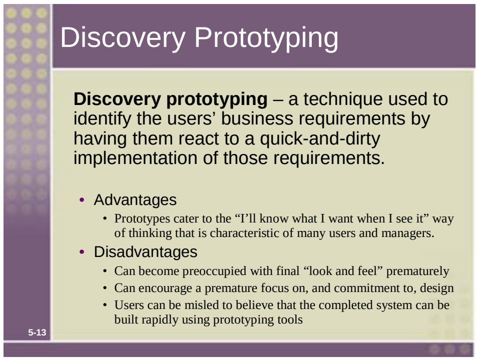
prototype – a small-scale, incomplete, but working sample of a desired system.

- Accelerated systems analysis approaches
  - Discovery Prototyping
  - Rapid Architected Analysis

5-12

Teaching Notes
Prototypes are “incomplete” in that they lack error checking, data validation, security, and processing completeness

Slide 13



Discovery Prototyping

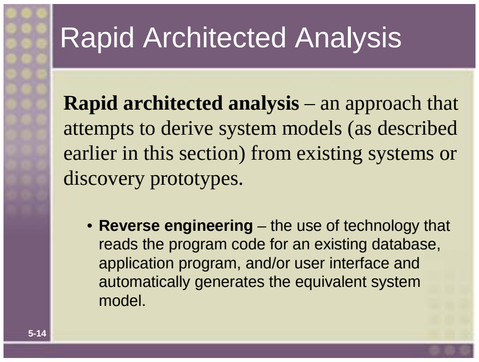
Discovery prototyping – a technique used to identify the users’ business requirements by having them react to a quick-and-dirty implementation of those requirements.

- Advantages
  - Prototypes cater to the “I’ll know what I want when I see it” way of thinking that is characteristic of many users and managers.
- Disadvantages
  - Can become preoccupied with final “look and feel” prematurely
  - Can encourage a premature focus on, and commitment to, design
  - Users can be misled to believe that the completed system can be built rapidly using prototyping tools

5-13

Teaching Notes
Discovery prototyping is sometimes called *requirements prototyping*.

Slide 14



Rapid Architected Analysis

Rapid architected analysis – an approach that attempts to derive system models (as described earlier in this section) from existing systems or discovery prototypes.

- **Reverse engineering** – the use of technology that reads the program code for an existing database, application program, and/or user interface and automatically generates the equivalent system model.

5-14

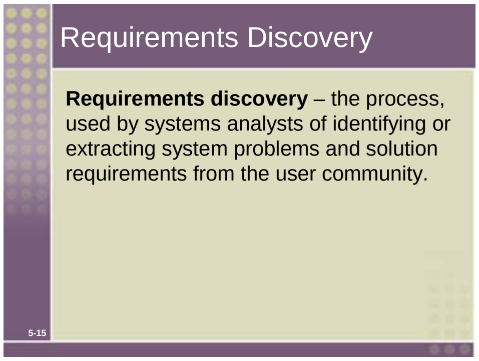
Teaching Notes

It is difficult, if not impossible, to adequately do discovery prototyping without some formal design. This is the realm of rapid architected analysis.

Some might consider rapid architecture analysis to be a model-driven approach since it results in system models. We elected to classify it as an accelerated analysis approach because of the technique used to build those models.

If you have Microsoft Visio Professional or a similar tool, you can demonstrate reverse engineering to transform a database from Access or another DBMS students are familiar with into a data model.

Slide 15



Requirements Discovery

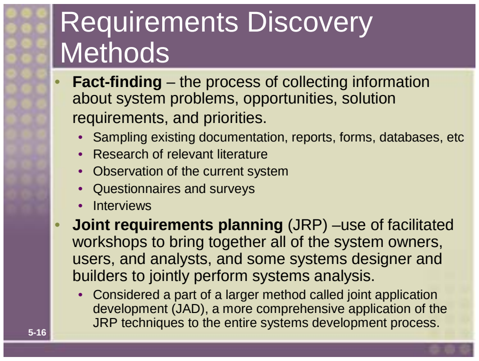
Requirements discovery – the process, used by systems analysts of identifying or extracting system problems and solution requirements from the user community.

5-15

Conversion Notes

In the sixth edition this slide and the following slide were combined. They were split in the seventh edition for the sake of readability.

Slide 16



Requirements Discovery Methods

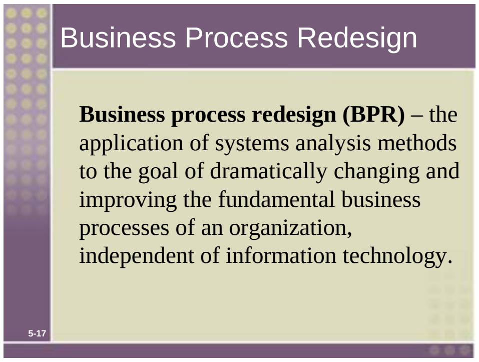
- **Fact-finding** – the process of collecting information about system problems, opportunities, solution requirements, and priorities.
  - Sampling existing documentation, reports, forms, databases, etc
  - Research of relevant literature
  - Observation of the current system
  - Questionnaires and surveys
  - Interviews
- **Joint requirements planning** (JRP) –use of facilitated workshops to bring together all of the system owners, users, and analysts, and some systems designer and builders to jointly perform systems analysis.
  - Considered a part of a larger method called joint application development (JAD), a more comprehensive application of the JRP techniques to the entire systems development process.

5-16

Teaching Notes

Fact-finding is also called information gathering.

Slide 17



Business Process Redesign

Business process redesign (BPR) – the application of systems analysis methods to the goal of dramatically changing and improving the fundamental business processes of an organization, independent of information technology.

5-17

Teaching Notes

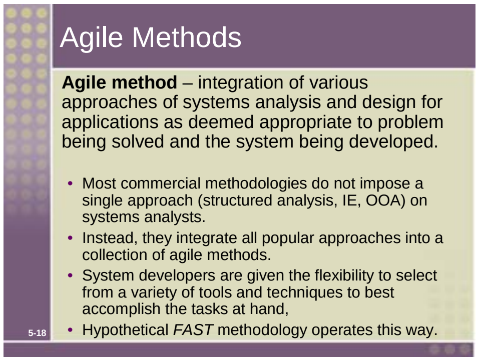
BPR is not a competing systems analysis methods. BPR is an application of systems analysis methods.

BPR can be used in redesigning completely manual processes.

It is not uncommon for IS projects to include a study of existing business processes to identify problems, bureaucracy, and inefficiencies that can be addressed in requirements for new and improved information systems.

Changes in processes brought about through BPR generally trigger needed changes in information systems.

Slide 18



Agile Methods

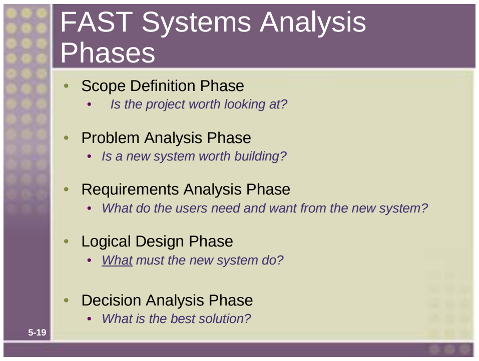
Agile method – integration of various approaches of systems analysis and design for applications as deemed appropriate to problem being solved and the system being developed.

- Most commercial methodologies do not impose a single approach (structured analysis, IE, OOA) on systems analysts.
- Instead, they integrate all popular approaches into a collection of agile methods.
- System developers are given the flexibility to select from a variety of tools and techniques to best accomplish the tasks at hand,
- Hypothetical *FAST* methodology operates this way.

5-18

No additional notes.

Slide 19



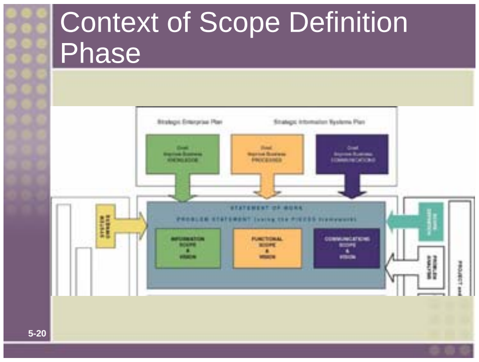
FAST Systems Analysis Phases

- Scope Definition Phase
  - *Is the project worth looking at?*
- Problem Analysis Phase
  - *Is a new system worth building?*
- Requirements Analysis Phase
  - *What do the users need and want from the new system?*
- Logical Design Phase
  - *What must the new system do?*
- Decision Analysis Phase
  - *What is the best solution?*

5-19

No additional notes.

Slide 20

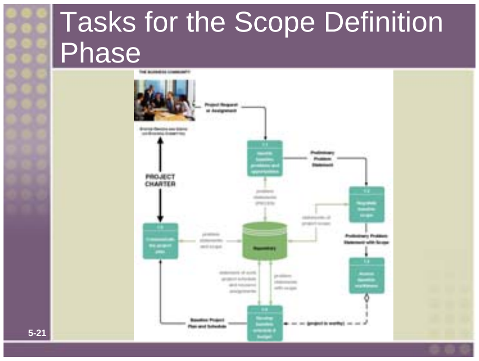


Context of Scope Definition Phase

Strategic Enterprise Plan
Strategic Information Systems Plan
STATEMENT OF WORK
PROBLEM STATEMENT (using the PIECES framework)
INFORMATION SCOPE & VISION
FUNCTIONAL SCOPE & VISION
COMMUNICATIONS SCOPE & VISION
SYSTEM OWNERS
PROBLEM ANALYSIS
PROJECT
5-20

Teaching Notes
The focus is on system owner perspectives.

Slide 21

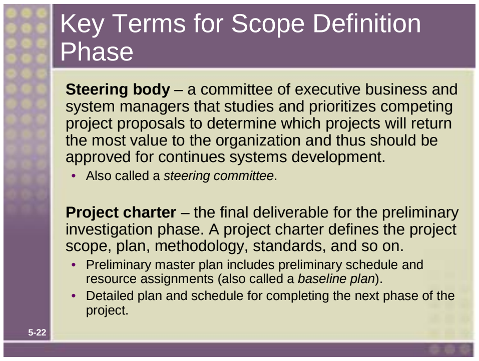


Tasks for the Scope Definition Phase

THE BUSINESS COMMUNITY
Project Request or Assignment
PROJECT CHARTER
Preliminary Problem Statement
Preliminary Problem Statement with Scope
Repository
Baseline Project Plan and Schedule
(project is worthy)
5-21

Teaching Notes
This is called a task diagram for a phase. This is a look "inside" a phase. It decomposes the phase into its component tasks .
It is only a guideline. Each project will adapt these tasks to the project at hand. Tasks may be added, split, or deleted according to the methodology and route used.
The dashed line is a control flow (as contrasted to a solid data flow). In this case, it represents a decision that determines whether the next task is necessary.

Slide 22



Key Terms for Scope Definition Phase

Steering body – a committee of executive business and system managers that studies and prioritizes competing project proposals to determine which projects will return the most value to the organization and thus should be approved for continues systems development.

- Also called a *steering committee*.

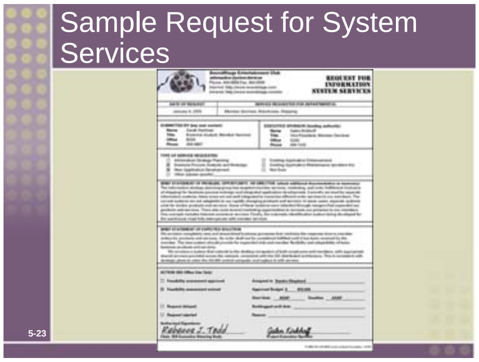
Project charter – the final deliverable for the preliminary investigation phase. A project charter defines the project scope, plan, methodology, standards, and so on.

- Preliminary master plan includes preliminary schedule and resource assignments (also called a *baseline plan*).
- Detailed plan and schedule for completing the next phase of the project.

5-22

Teaching Notes
Completion of the project charter represents the first milestone in a project
The project, in most cases, must be approved by the steering body before it can proceed

Slide 23



Sample Request for System Services

SoundStage Entertainment Club

REQUEST FOR INFORMATION SYSTEM SERVICES

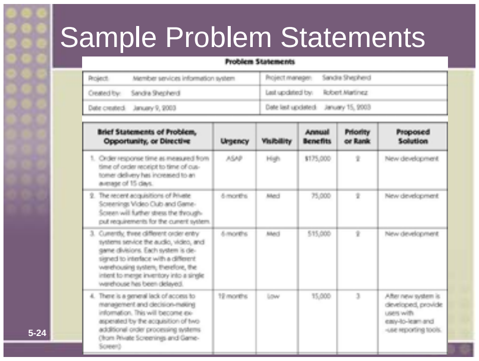
Rebecca I. Todd

Galen Kirkhoff

5-23

Teaching Notes
Not all organizations have a formal document to initiate projects.

Slide 24



Sample Problem Statements

Problem Statements

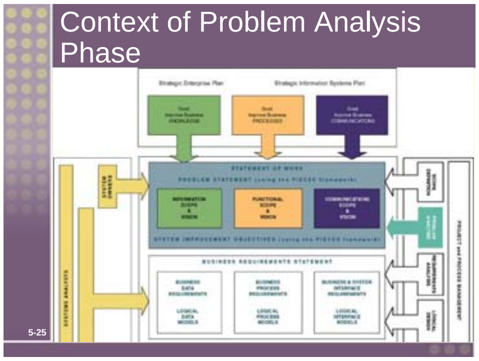
Project:	Member services information system	Project manager:	Sandra Shepherd
Created by:	Sandra Shepherd	Last updated by:	Robert Martinez
Date created:	January 9, 2003	Date last updated:	January 15, 2003

Brief Statements of Problem, Opportunity, or Directive	Urgency	Visibility	Annual Benefits	Priority or Rank	Proposed Solution
1. Order response time as measured from time of order receipt to time of customer delivery has increased to an average of 15 days.	ASAP	High	$175,000	2	New development
2. The recent acquisitions of Private Screenings Video Club and GameScreen will further stress the throughput requirements for the current system.	6 months	Med	75,000	2	New development
3. Currently, three different order entry systems service the audio, video, and game divisions. Each system is designed to interface with a different warehousing system; therefore, the intent to merge inventory into a single warehouse has been delayed.	6 months	Med	515,000	2	New development
4. There is a general lack of access to management and decision-making information. This will become exasperated by the acquisition of two additional order processing systems (from Private Screenings and GameScreen)	12 months	Low	15,000	3	After new system is developed, provide users with easy-to-learn and -use reporting tools.

5-24

Teaching Notes
Alternatively, this information could be documented in a business memo or report.
Define columns for students and walk through sample
For readability, this is a shortened version from what appears in the text. Refer students to the text for more identified problems, opportunities, and/or directives.

Slide 25



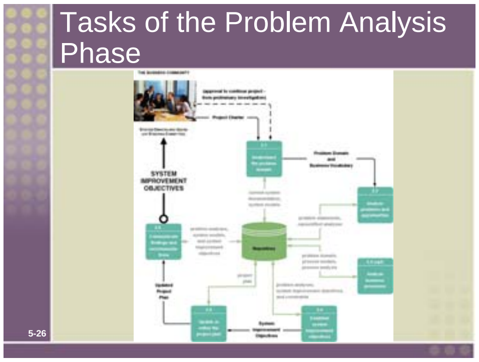
Context of Problem Analysis Phase

Strategic Enterprise Plan
Strategic Information Systems Plan
Goal Improve Business KNOWLEDGE
Goal Improve Business PROCESSES
Goal Improve Business COMMUNICATIONS
SYSTEM OWNERS
SYSTEMS ANALYSTS
STATEMENT OF WORK
PROBLEM STATEMENT (using the PIECES framework)
INFORMATION SCOPE & VISION
FUNCTIONAL SCOPE & VISION
COMMUNICATIONS SCOPE & VISION
SYSTEM IMPROVEMENT OBJECTIVES (using the PIECES framework)
BUSINESS REQUIREMENTS STATEMENT
BUSINESS DATA REQUIREMENTS
LOGICAL DATA MODELS
BUSINESS PROCESS REQUIREMENTS
LOGICAL PROCESS MODELS
BUSINESS & SYSTEM INTERFACE REQUIREMENTS
LOGICAL INTERFACE MODELS
SCOPE DEFINITION
PROBLEM ANALYSIS
REQUIREMENTS ANALYSIS
LOGICAL DESIGN
PROJECT and PROCESS MANAGEMENT

5-25

Teaching Notes
The focus is on both system owner and system user perspectives.
We are looking at the building blocks of the existing system.

Slide 26

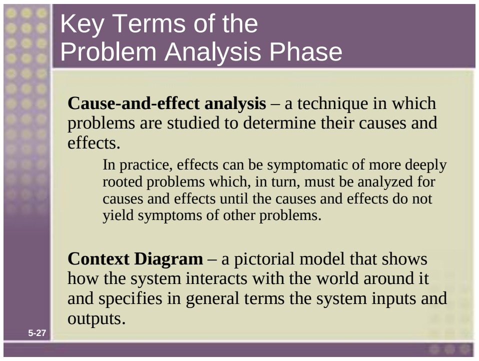


Tasks of the Problem Analysis Phase

THE BUSINESS COMMUNITY
(approval to continue project – from preliminary investigation)
Project Charter
Problem Domain and Business Vocabulary
SYSTEM IMPROVEMENT OBJECTIVES
Repository
Updated Project Plan
System Improvement Objectives
5-26

No additional notes

Slide 27



Key Terms of the Problem Analysis Phase

Cause-and-effect analysis – a technique in which problems are studied to determine their causes and effects.

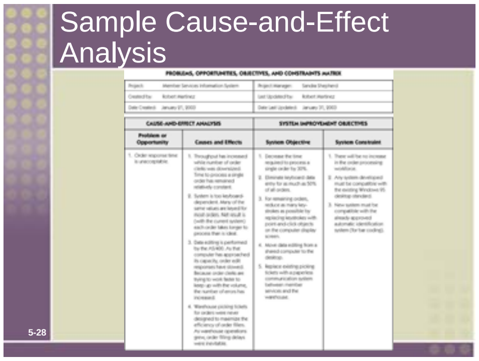
In practice, effects can be symptomatic of more deeply rooted problems which, in turn, must be analyzed for causes and effects until the causes and effects do not yield symptoms of other problems.

Context Diagram – a pictorial model that shows how the system interacts with the world around it and specifies in general terms the system inputs and outputs.

5-27

Teaching Notes
Analyze a problem using cause-and-effect analysis.
If you know “fishbone diagrams”, demonstrate cause-and-effect analysis using the diagrams.

Slide 28



Sample Cause-and-Effect Analysis

PROBLEMS, OPPORTUNITIES, OBJECTIVES, AND CONSTRAINTS MATRIX

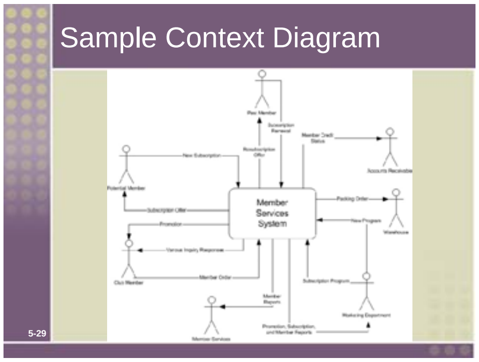
Project:	Member Services Information System	Project manager:	Sandra Shepherd
Created by:	Robert Martinez	Last Updated by:	Robert Martinez
Date Created:	January 27, 2003	Date Last Updated:	January 31, 2003

CAUSE-AND-EFFECT ANALYSIS		SYSTEM IMPROVEMENT OBJECTIVES	
Problem or Opportunity	Causes and Effects	System Objective	System Constraint
1. Order response time is unacceptable.	1. Throughput has increased while number of order clerks was downsized. Time to process a single order has remained relatively constant. 2. System is too keyboard-dependent. Many of the same values are keyed for most orders. Net result is (with the current system) each order takes longer to process than is ideal. 3. Data editing is performed by the AS/400. As that computer has approached its capacity, order edit responses have slowed. Because order clerks are trying to work faster to keep up with the volume, the number of errors has increased. 4. Warehouse picking tickets for orders were never designed to maximize the efficiency of order fillers. As warehouse operations grew, order filling delays were inevitable.	1. Decrease the time required to process a single order by 30%. 2. Eliminate keyboard data entry for as much as 50% of all orders. 3. For remaining orders, reduce as many keystrokes as possible by replacing keystrokes with point-and-click objects on the computer display screen. 4. Move data editing from a shared computer to the desktop. 5. Replace existing picking tickets with a paperless communication system between member services and the warehouse.	1. There will be no increase in the order processing workforce. 2. Any system developed must be compatible with the existing Windows 95 desktop standard. 3. New system must be compatible with the already approved automatic identification system (for bar coding).

5-28

No additional notes

Slide 29

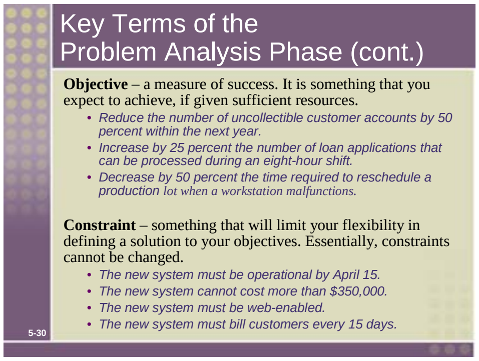


Sample Context Diagram

5-29

Conversion Notes
The context diagram is new to this chapter in the fifth edition. But it is a useful tool for understanding the present system at a high level.

Slide 30



Key Terms of the Problem Analysis Phase (cont.)

Objective – a measure of success. It is something that you expect to achieve, if given sufficient resources.

- *Reduce the number of uncollectible customer accounts by 50 percent within the next year.*
- *Increase by 25 percent the number of loan applications that can be processed during an eight-hour shift.*
- *Decrease by 50 percent the time required to reschedule a production lot when a workstation malfunctions.*

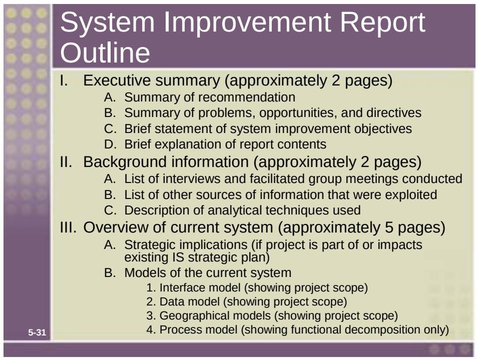
Constraint – something that will limit your flexibility in defining a solution to your objectives. Essentially, constraints cannot be changed.

- *The new system must be operational by April 15.*
- *The new system cannot cost more than $350,000.*
- *The new system must be web-enabled.*
- *The new system must bill customers every 15 days.*

5-30

Teaching Notes
The criteria for success of an information system should be measured in terms of objectives. Formulate objectives as an in-class exercise.

Slide 31



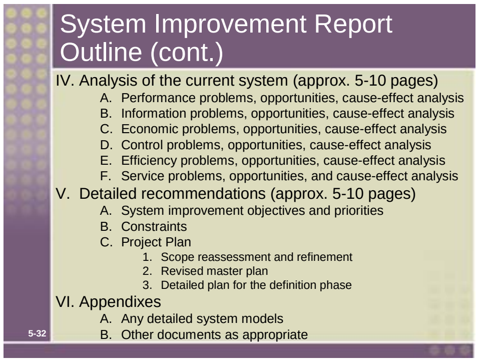
System Improvement Report Outline

I. Executive summary (approximately 2 pages)
  A. Summary of recommendation
  B. Summary of problems, opportunities, and directives
  C. Brief statement of system improvement objectives
  D. Brief explanation of report contents
II. Background information (approximately 2 pages)
  A. List of interviews and facilitated group meetings conducted
  B. List of other sources of information that were exploited
  C. Description of analytical techniques used
III. Overview of current system (approximately 5 pages)
  A. Strategic implications (if project is part of or impacts existing IS strategic plan)
  B. Models of the current system
    1. Interface model (showing project scope)
    2. Data model (showing project scope)
    3. Geographical models (showing project scope)
    4. Process model (showing functional decomposition only)

5-31

Teaching Notes
The focus is on system user perspectives. Requirements can be expressed in narrative, model, and prototype forms, or any combination thereof.

Slide 32



System Improvement Report Outline (cont.)

IV. Analysis of the current system (approx. 5-10 pages)
   A. Performance problems, opportunities, cause-effect analysis
   B. Information problems, opportunities, cause-effect analysis
   C. Economic problems, opportunities, cause-effect analysis
   D. Control problems, opportunities, cause-effect analysis
   E. Efficiency problems, opportunities, cause-effect analysis
   F. Service problems, opportunities, and cause-effect analysis

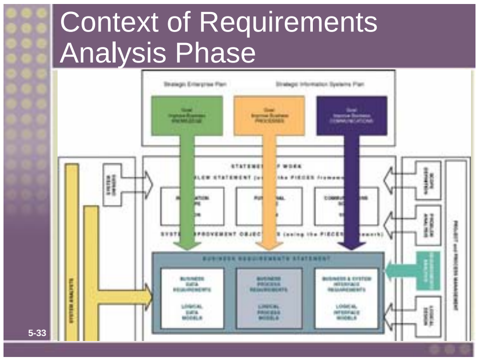
V. Detailed recommendations (approx. 5-10 pages)
   A. System improvement objectives and priorities
   B. Constraints
   C. Project Plan
      1. Scope reassessment and refinement
      2. Revised master plan
      3. Detailed plan for the definition phase

VI. Appendixes
   A. Any detailed system models
   B. Other documents as appropriate

5-32

No additional notes.

Slide 33



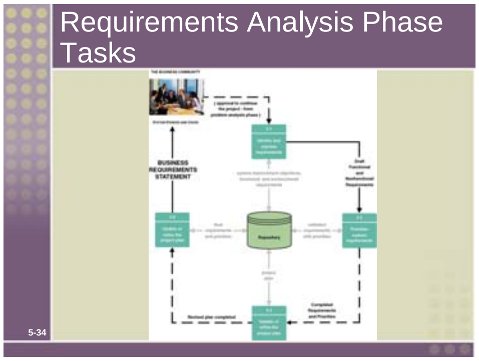
Context of Requirements Analysis Phase

Strategic Enterprise Plan
Strategic Information Systems Plan
Goal Improve Business KNOWLEDGE
Goal Improve Business PROCESSES
Goal Improve Business COMMUNICATIONS
SYSTEM OWNERS
SCOPE DEFINITION
PROBLEM ANALYSIS
BUSINESS REQUIREMENTS STATEMENT
BUSINESS DATA REQUIREMENTS
LOGICAL DATA MODELS
BUSINESS PROCESS REQUIREMENTS
LOGICAL PROCESS MODELS
BUSINESS & SYSTEM INTERFACE REQUIREMENTS
LOGICAL INTERFACE MODELS
SYSTEM ANALYSTS
LOGICAL DESIGN
PROJECT and PROCESS MANAGEMENT

5-33

Teaching Notes
The focus is on system user perspectives. Requirements can be expressed in narrative, model, and prototype forms, or any combination thereof.

Slide 34



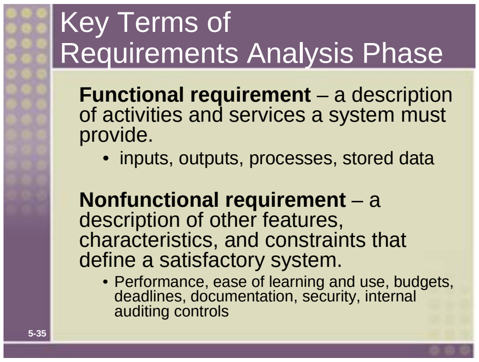
Requirements Analysis Phase Tasks

THE BUSINESS COMMUNITY
(approval to continue the project - from problem analysis phase)
BUSINESS REQUIREMENTS STATEMENT
Draft Functional and Nonfunctional Requirements
Repository
Revised plan completed
Completed Requirements and Priorities

5-34

Teaching Notes
Some of the tasks are completed in parallel.

Slide 35



Key Terms of Requirements Analysis Phase

Functional requirement – a description of activities and services a system must provide.

- inputs, outputs, processes, stored data

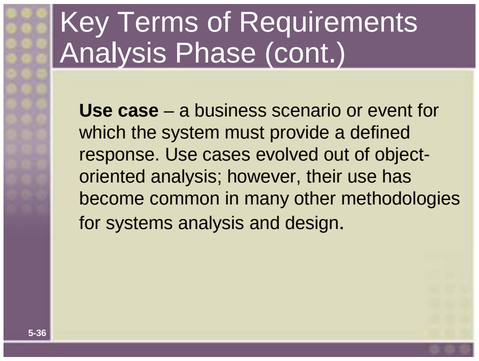
Nonfunctional requirement – a description of other features, characteristics, and constraints that define a satisfactory system.

- Performance, ease of learning and use, budgets, deadlines, documentation, security, internal auditing controls

5-35

Teaching Notes
It is not that functional requirements are mandatory and nonfunctional requirements are optional. The list of example nonfunctional requirements includes mandatory items.
The difference is that functional requirements describe functions that the system must perform. Nonfunctional requirements are not functions to be done but qualities that the system must have if it is to be successful.

Slide 36



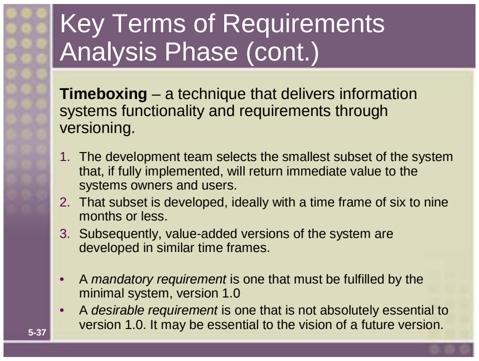
Key Terms of Requirements Analysis Phase (cont.)

Use case – a business scenario or event for which the system must provide a defined response. Use cases evolved out of object-oriented analysis; however, their use has become common in many other methodologies for systems analysis and design.

5-36

No additional notes.

Slide 37



Key Terms of Requirements Analysis Phase (cont.)

Timeboxing – a technique that delivers information systems functionality and requirements through versioning.

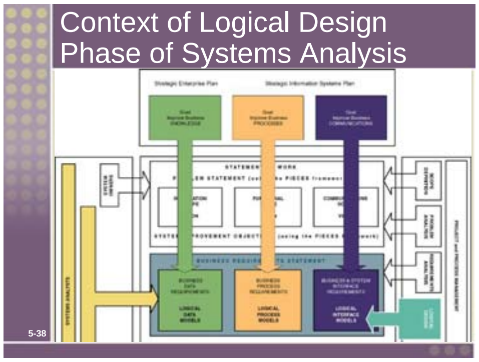
1. The development team selects the smallest subset of the system that, if fully implemented, will return immediate value to the systems owners and users.
2. That subset is developed, ideally with a time frame of six to nine months or less.
3. Subsequently, value-added versions of the system are developed in similar time frames.

- A *mandatory requirement* is one that must be fulfilled by the minimal system, version 1.0
- A *desirable requirement* is one that is not absolutely essential to version 1.0. It may be essential to the vision of a future version.

5-37

No additional notes

Slide 38

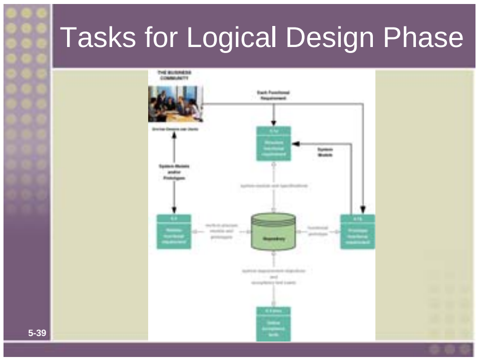


Context of Logical Design Phase of Systems Analysis

Strategic Enterprise Plan
Strategic Information Systems Plan
STATEMENT OF WORK
BUSINESS REQUIREMENTS STATEMENT
BUSINESS PROCESS REQUIREMENTS
LOGICAL DATA MODELS
LOGICAL PROCESS MODELS
LOGICAL INTERFACE MODELS
SYSTEMS ANALYSTS
5-38

Teaching Notes
The focus is on system user perspectives.

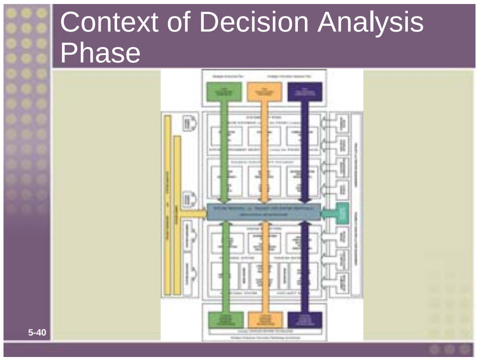
Slide 39



Tasks for Logical Design Phase

No additional notes.

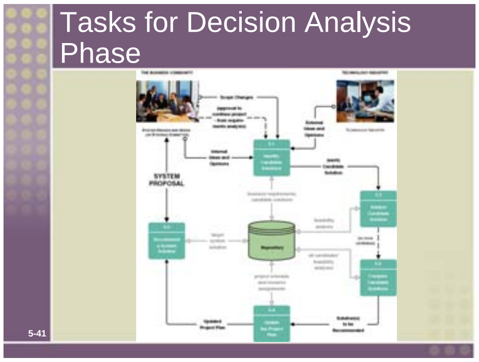
Slide 40



Context of Decision Analysis Phase

Teaching Notes
The focus is on system user and system designer perspectives.
Notice the transition to technical concerns leading to a system proposal that includes data, process, and interface elements.

Slide 41



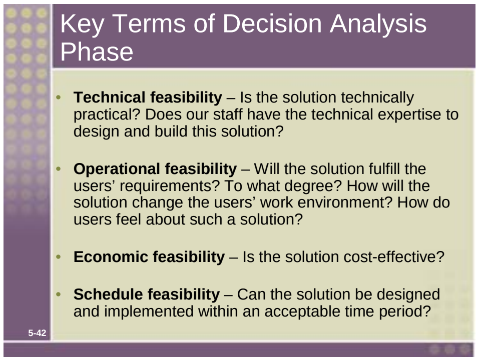
Tasks for Decision Analysis Phase

THE BUSINESS COMMUNITY
TECHNOLOGY INDUSTRY
Scope Changes
(approval to continue project - from requirements analysis)
External Ideas and Opinions
Internal Ideas and Opinions
5.1
Identify Candidate Solutions
Identify Candidate Solution
SYSTEM PROPOSAL
5.2
Analyze Candidate Solutions
Repository
5.5
Recommend a System Solution
5.3
Compare Candidate Solutions
5.4
Update the Project Plan
Updated Project Plan
Solution(s) to be Recommended

5-41

No additional notes.

Slide 42



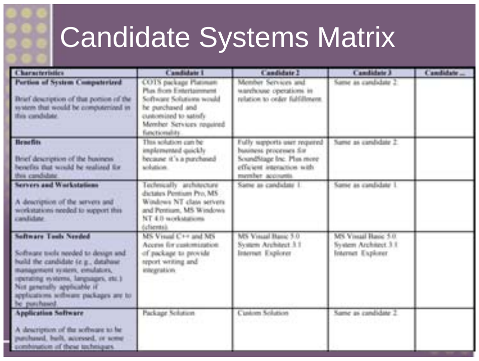
Key Terms of Decision Analysis Phase

- **Technical feasibility** – Is the solution technically practical? Does our staff have the technical expertise to design and build this solution?
- **Operational feasibility** – Will the solution fulfill the users' requirements? To what degree? How will the solution change the users' work environment? How do users feel about such a solution?
- **Economic feasibility** – Is the solution cost-effective?
- **Schedule feasibility** – Can the solution be designed and implemented within an acceptable time period?

5-42

No additional notes

Slide 43

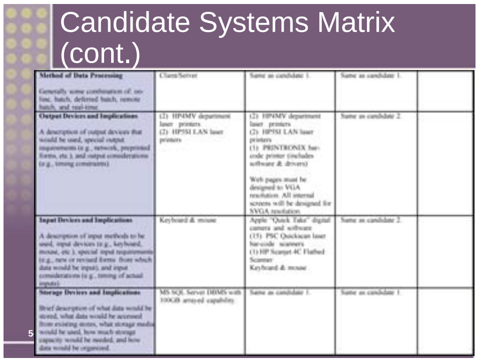


Candidate Systems Matrix

Characteristics	Candidate 1	Candidate 2	Candidate 3	Candidate ...
Portion of System Computerized Brief description of that portion of the system that would be computerized in this candidate.	COTS package Platinum Plus from Entertainment Software Solutions would be purchased and customized to satisfy Member Services required functionality.	Member Services and warehouse operations in relation to order fulfillment.	Same as candidate 2.	
Benefits Brief description of the business benefits that would be realized for this candidate.	This solution can be implemented quickly because it's a purchased solution.	Fully supports user required business processes for SoundStage Inc. Plus more efficient interaction with member accounts.	Same as candidate 2.	
Servers and Workstations A description of the servers and workstations needed to support this candidate.	Technically architecture dictates Pentium Pro, MS Windows NT class servers and Pentium, MS Windows NT 4.0 workstations (clients).	Same as candidate 1.	Same as candidate 1.	
Software Tools Needed Software tools needed to design and build the candidate (e.g., database management system, emulators, operating systems, languages, etc.) Not generally applicable if applications software packages are to be purchased.	MS Visual C++ and MS Access for customization of package to provide report writing and integration.	MS Visual Basic 5.0 System Architect 3.1 Internet Explorer	MS Visual Basic 5.0 System Architect 3.1 Internet Explorer	
Application Software A description of the software to be purchased, built, accessed, or some combination of these techniques.	Package Solution	Custom Solution	Same as candidate 2.	

No additional notes

Slide 44



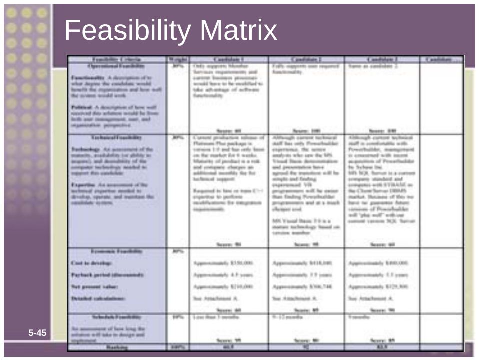
Candidate Systems Matrix (cont.)

Method of Data Processing Generally some combination of: on-line, batch, deferred batch, remote batch, and real-time.	Client/Server	Same as candidate 1.	Same as candidate 1.	
Output Devices and Implications A description of output devices that would be used, special output requirements (e.g., network, preprinted forms, etc.), and output considerations (e.g., timing constraints)	(2) HP4MV department laser printers (2) HP5SI LAN laser printers	(2) HP4MV department laser printers (2) HP5SI LAN laser printers (1) PRINTRONIX bar-code printer (includes software & drivers) Web pages must be designed to VGA resolution. All internal screens will be designed for SVGA resolution.	Same as candidate 2.	
Input Devices and Implications A description of input methods to be used, input devices (e.g., keyboard, mouse, etc.), special input requirements (e.g., new or revised forms from which data would be input), and input considerations (e.g., timing of actual inputs).	Keyboard & mouse	Apple "Quick Take" digital camera and software (15) PSC Quickscan laser bar-code scanners (1) HP Scanjet 4C Flatbed Scanner Keyboard & mouse	Same as candidate 2.	
Storage Devices and Implications Brief description of what data would be stored, what data would be accessed from existing stores, what storage media would be used, how much storage capacity would be needed, and how data would be organized.	MS SQL Server DBMS with 100GB arrayed capability.	Same as candidate 1.	Same as candidate 1.	

5

No additional notes

Slide 45



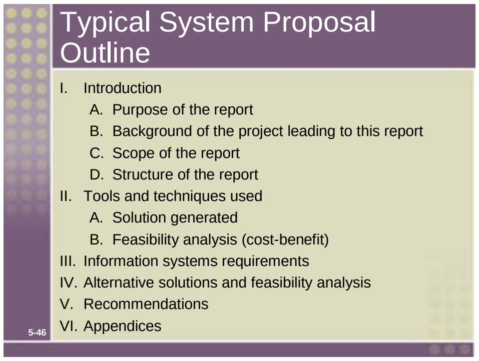
Feasibility Matrix

Feasibility Criteria	Weight	Candidate 1	Candidate 2	Candidate 3	Candidate ...
Operational Feasibility **Functionality.** A description of to what degree the candidate would benefit the organization and how well the system would work. **Political.** A description of how well received this solution would be from both user management, user, and organization perspective.	30%	Only supports Member Services requirements and current business processes would have to be modified to take advantage of software functionality. **Score: 60**	Fully supports user required functionality. **Score: 100**	Same as candidate 2. **Score: 100**	
Technical Feasibility **Technology.** An assessment of the maturity, availability (or ability to acquire), and desirability of the computer technology needed to support this candidate. **Expertise.** An assessment of the technical expertise needed to develop, operate, and maintain the candidate system.	30%	Current production release of Platinum Plus package is version 1.0 and has only been on the market for 6 weeks. Maturity of product is a risk and company charges an additional monthly fee for technical support. Required to hire or train C++ expertise to perform modifications for integration requirements. **Score: 50**	Although current technical staff has only Powerbuilder experience, the senior analysts who saw the MS Visual Basic demonstration and presentation have agreed the transition will be simple and finding experienced VB programmers will be easier than finding Powerbuilder programmers and at a much cheaper cost. MS Visual Basic 5.0 is a mature technology based on version number. **Score: 95**	Although current technical staff is comfortable with Powerbuilder, management is concerned with recent acquisition of Powerbuilder by Sybase Inc. MS SQL Server is a current company standard and competes with SYBASE in the Client/Server DBMS market. Because of this we have no guarantee future versions of Powerbuilder will "play well" with our current version SQL Server. **Score: 60**	
Economic Feasibility **Cost to develop:** **Payback period (discounted):** **Net present value:** **Detailed calculations:**	30%	Approximately $350,000. Approximately 4.5 years. Approximately $210,000. See Attachment A. **Score: 60**	Approximately $418,040. Approximately 3.5 years. Approximately $306,748. See Attachment A. **Score: 85**	Approximately $400,000. Approximately 3.3 years. Approximately $325,500. See Attachment A. **Score: 90**	
Schedule Feasibility An assessment of how long the solution will take to design and implement.	10%	Less than 3 months. **Score: 95**	9-12 months **Score: 80**	9 months **Score: 85**	
Ranking	100%	60.5	92	83.5	

5-45

No additional notes

Slide 46



Typical System Proposal Outline

I. Introduction
   A. Purpose of the report
   B. Background of the project leading to this report
   C. Scope of the report
   D. Structure of the report
II. Tools and techniques used
   A. Solution generated
   B. Feasibility analysis (cost-benefit)
III. Information systems requirements
IV. Alternative solutions and feasibility analysis
V. Recommendations
VI. Appendices

5-46

No additional notes.

Answers to End of Chapter Questions and Exercises

Review Question

1. System analysis is driven by the business concerns of system owners and system users. It should address the knowledge, process, and communication building blocks from the perspective of system owners and system users

2. Model-driven analysis is a problem-solving approach that emphasizes the drawing of pictorial system models to document and validate existing and/or proposed systems. Ultimately, the system model becomes the blueprint for designing and constructing an improved system.

   Model-driven analysis is used because “a picture is worth a thousand words” in communicating business problems, requirements and solutions. Using models to represent reality or vision is easier for system users and owners to understand.

   Examples of models included flowcharts, hierarchy chars and organization charts.

3. Structured analysis is a process-centered technique that focuses on the flow of data through business and software processes. Structured analysis emphasizes the process building blocks in the information system framework.

4. Information engineering is a data-centered technique that focuses on the structure of stored data in a system. Information engineering emphasizes studying and conducting the requirements analysis of knowledge requirements before doing of the process and communication requirements. Information engineering is based on the idea that knowledge and data are corporate resources that should be planned and managed.

5. Object-oriented analysis has become popular because it eliminates the artificial separation of concerns about data and processes that limits other older systems development approaches. Although most systems analysis methods have made attempted to synchronize data and process models, the results have been only partially successful. Object-oriented analysis has solved this problem by encapsulating both data (called properties) with all the processes (called methods) used to create, read, update and delete the data.

6. Scope definition phase, problem analysis phase, requirements analysis phase, logical design phase, and decision analysis phase.

7. The scope definition phase determines whether a project is worth looking at. To determine this, we must define the scope of the project and the perceived

problems, opportunities, and directives that triggered the project. If the project is found to be worthwhile, the scope definition phase must also establish the project plan relative to project scale, development strategy, schedule, resource requirements, and budget.

8. 1) Identify baseline problems and opportunities
   2) Negotiate baseline scope
   3) Assess baseline project worthiness
   4) Develop baseline schedule and budget
   5) Communicate the project plan

9. Communicating the project plan to an audience is triggered once the baseline project plan and schedule is completed.

   The audience to whom the project plan is presented for consideration will generally be some type of steering body. The steering body is a committee of executive business and system managers that studies and prioritizes competing project proposals to determine which projects will return the most value to the organization and thus should be approved for continued systems development.

   It is important because in most organizations, there are more potential projects than resources to staff and fund those projects. In order for your project to commence, you must present to the steering body and be prepared to defend the importance and critical nature of the project. .

10. Many inexperienced systems analysts try to solve problems before they really analyze them. They see the problem in terms of a solution, e.g., by stating "we need to," or "we want to." Experienced and skilled systems analysts first analyze each perceived problem for causes and effects. Cause-effect analysis will lead to a deeper understanding of the true nature of the problem, and can lead to a less obvious but more effective solution. New systems analysts can increase their effectiveness by critically self-analyzing their problem-solving approach, and by gaining experience under the mentorship of a more experienced systems analyst.

11. Use cases are a widely-used modeling tool for identifying and expressing the functional requirements of a system. Each use case is a business scenario or event for which the system must provide a defined response. Use cases evolved out of object-oriented analysis; however, their use as a modeling tool has become common in many other methodologies for system analysis and design.

12. Timeboxing is a technique that delivers information systems functionality and requirements through versioning. Timeboxing is popular because it attempts to divide requirements into small "chunks" that can be more readily implemented. The development team selects the smallest subset of the sys-

tem that, if fully implemented, will return immediate value to the system owners and users.

13. As an option, discovery prototypes can be built when system modeling can’t be used because users are having difficulty expressing or visualizing their business requirements. Prototyping is typically used in the requirements analysis phase to build sample inputs and outputs.

14. The decision analysis phase is needed because this is the point in the process where we identify candidate solutions, analyze them, and recommend a target system to be designed, constructed, and implemented.

    Even if someone has previously proposed a technical solution, alternative solutions, or even better ones, always exist. Thus, during the decision analysis phase, you identify options, analyze those options, and the sell the best solution based on the analysis.

15. Candidate solutions can be derived from design ideas and opinions of the system owners and users. Others may come from various sources including systems analysts, systems designers, technical consultants, and other IT professionals. The task here is to identify alternatives from a broad range of stakeholders. This task is concerned with the process of identifying different candidate solutions, not evaluating their viability.

Problem and Exercises

1. There is no universally accepted definition of systems analysis, nor is there any general consensus as to when systems analysis begins and when it ends. Different methodologies and different organizations may employ different definitions, and different starting and ending points. What is important to understand is the definition and parameters of the methodology employed by your organization regarding systems analysis. Further, to ensure that all project stakeholders for your organization share in the same common understanding.

2.

	CENTRICITY (data, process, etc.)	**TYPE OF MODELS USED**	**ESSENTIAL DIFFERENCES**
		Data flow dia-	Attempts to synchronize

STRUCTURED ANALYSIS	Process-centric	grams (DFD) first, then entity relationship diagrams (ERD)	separate data and process models by looking at processes first.
INFORMATION ENGINEERING AND DATA MODELING	Data-centric	Entity relationship diagrams first, then data flow diagrams	Attempts to synchronize separate data and process models by looking at data first
OBJECT-ORIENTED ANALYSIS	Object-centric	Object models via Unified Modeling Language (UML)	Removes separation between data and process

3. The two most popular approaches to accelerated analysis are discovery prototyping and Rapid Architected Analysis.

   Discovery prototyping uses RAD-type tools such as Microsoft Access to quickly build user interfaces such as data input screens and reports in order to help identify business requirements and to get feedback from system users and owners. In effect, these models are like façades used in a movie set; they look very realistic, but there's really no structure, i.e., detailed database and programming, behind the facade. That is both the strength and weakness of prototyping – a systems analyst can very quickly create prototype models to identify business requirements, but that same ease and speed of creating elegant-looking interfaces can lead to false expectations by the system users and owners regarding the actual system to be built.

   Rapid Architected Analysis is similar to discover prototyping in that it is also focused on quickly building system models in order to get feedback. But Rapid Architected Analysis builds these models from existing systems, usually legacy ones, or from prototypes by using reverse-engineering tools, such as CASE.

   Experience systems analysts rarely attempt to use prototyping as a stand-alone design method that completely replaces model-driven design; in fact, the Rapid Architected Analysis approach actually combines both prototyping and model-driven approaches through its reverse-engineering process.

4. The central question to keep in mind throughout the scope definition phase is “Is this project worth looking at?” To answer the question, the project team must answer and/or define the following:
    1. What is the scope of the project?
    2. What was the trigger for this project, i.e., what are the perceived problems, directives or mandates, and/or opportunities to be resolved or taken advantage of?
    3. For each problem, directive or opportunity, what is the assessment of their urgency, visibility, tangible benefits and priority?
    4. Will resolving the problems, carrying out the directives, and/or taking advantage of the opportunities have a great enough ‘pay-off’ to justify the effort, cost and risk of conducting the project?

    Typically, the scope definition phase consists of five tasks:
    1. Identifying the project triggers, i.e., the problems, directives, and opportunities
    2. Negotiating and reaching consensus on the baseline scope of the project.
    3. Determining whether the project is worth conducting
    4. Developing the initial (baseline) project schedule and budget
    5. Getting approval for the project from a steering committee, where applicable, and launching and communicating the project.

5. Many stakeholders, as well as new systems analysts, approach problem-solving by framing the problem based upon the perceived solution. In effect, they are trying to solve the problem without analyzing or understanding what the real problem is. Cause-and-effect analysis is one method that avoids this situation by examining whether the effect is really a symptom of a deeper underlying problem. Cause-and-effect analysis is done on an iterative basis until examining the effect does not result in finding symptoms of deeper problems.

6. There should be a system improvement objective established for each serious problem verified during problem analysis. Each system improvement objective should be a specific, measurable statement of business performance that the new system will be expected to meet. Where applicable, constraints that may prevent the system improvement objective from being realized should also be identified.

    Examples of system improvement objectives are:

1. The new system shall allow authorized users to automatically generate and print the Quarterly Report of Sales by Region from their personal computers without submitting a service request to a programmer.
2. Locating a custom record shall not take more than 2.5 seconds during peak call times at 10:00 a.m. and 3:00 p.m.
3. The new system shall automatically e-mail an early renewal notice to subscribers 90 days before their subscriptions expire.

Regarding whether the system improvement objectives developed by your project team are examples of good system improvement objectives:

1. Reduce the time required to process the order
   No, because there are no measurable terms specified
2. The new system must use Oracle to store data
   No, because this is a system constraint, not a system objective
3. The data input screens must be redesigned so they are more user-friendly
   No, because this is a requirement (and a very vague one), not an objective.
4. The customer satisfaction rate with the online ordering process must be increased by 10%
   Yes, because it is a precise, measurable and statement of business performance

The first three objectives might be rewritten as follows:

1. The new system must reduce the time to process a sales order to an average of no more than one business day.
2. The new system must eliminate storing redundant and maintaining duplicate files.
3. The new system must be designed so that employees can learn to use the data input screens in an average of two hours or less, and they can enter sales orders in an average of five minutes or less.

Constraints can be any one or more of four categories: schedule, cost, technology and policy. Examples of constraints for the preceding objectives could be:

1. The personnel budget for the sales department will not increase (Cost).
2. The new system must utilize the existing Oracle-based architecture (Technology)
3. The new system will not require purchasing screens that are larger than the current screen standard for the organization (Cost).

7. The requirements phase is not about the how – that comes later when the technical solution is designed. The requirements phase is focused on the <u>what</u> of defining business requirements. To identify the business requirements, the analyst should be asking instead “<u>what</u> do the users require (or want) from the new system?”

In short, the analyst is trying to devise a technical solution before the requirements are known. It doesn't matter whether the analyst is inexperienced or deliberately trying to take a shortcut; cutting short the requirements analysis phase (or skipping it entirely) invariably leads to project that are challenged or that fail.

8. Identifying and expressing system requirements accurately and completely is one of the most difficult tasks in any project. To assist conceptually, requirements are generally divided into two categories – functional and nonfunctional.

   Functional requirements consist of the concrete activities and services that make up a system, i.e., the inputs, processes, outputs and data stores, and that are needed to meet the system improvement objectives.

   Nonfunctional requirements refer to the behavioral attributes of the system, i.e., performance factors such as speed of operations and response time, cost of operations, ease of use, security controls, quality management, documentation, etc.

   Traditionally, requirements are expressed in outline or table format. In its most basic format, this outline or table lists the inputs, processes, outputs and data stores needed for each system improvement objective.

   Use cases, which were originally developed as a modeling tool for object-oriented analysis, have come into increasingly common usage for a wide range of methodologies. Use cases approach system requirements by modeling the different business scenarios that occur and the responses that must be provided by the system.

9. Prioritizing requirements is an important safety measure in case the project falls behind schedule or over budget. Prioritizing also helps stakeholders to focus on the real target. The point at which the requirements should be prioritized is early on in the project, during the requirements analysis phase, rather than waiting until there is a project crisis.

   Timeboxing is a frequently used technique that divides the requirements into "chunks" or subsets that can be developed and released in successive versions The requirements are divided into two groups: 1) mandatory requirements, which are absolutely essential to meeting the system improvement objectives and thus not ranked, and 2) desirable requirements, which are not absolutely essential and thus can be ranked.

   The first subset, which is often termed version 1.0, generally contains only the mandatory requirements group. The desirable requirements are then put into succeeding versions that can be developed and released in six to nine month increments.

   One way to test whether a mandatory requirement is truly mandatory is attempt to rank it. If it can be ranked, then it is really a desirable requirement, not a mandatory requirement.

10. Although our jobs might be easier if scope was static, that almost never is the case in the real business world. Changes may occur in the business environment, mission, and/or processes during the course of the project Not only is scope inherently dynamic, our understanding of the scope tends to change and evolve during the requirements analysis process. Therefore, it is important to review the project plan at this point, and ensure that the schedule and budget reflect any changes in the scope or in our understanding of it.

    For many of the same reasons, stakeholders almost invariably perceive a need to add or change requirements subsequent to the requirements analysis phase. Given that reality, a change management process is absolutely imperative, not only to track and document requests, but also specifying the procedures for requesting changes and the criteria for evaluating them. Absent a formal change management process, the project runs the risk of requirements anarchy.

11. It is never too early to start working on system testing. The system models and prototypes lend themselves very well to defining and developing test cases, since they contain the data and business rules, as well as the processing requirements, against which you are ultimately going to test. In fact, developing test cases at this point is actually an excellent way to validate the functional requirements. If a functional requirement is incorrect or missing, the earlier you find it, the easier and cheaper it is to fix!

12. The requirement analysis phase defines and codifies the business requirements for the new system. The deliverable for this phase is a business requirement document that identifies the functional and nonfunctional requirements which will satisfy the system improvement objectives.

    A logical design phase further documents business requirements using system models that illustrate data structures, business processes, data flows and uses interfaces.

13. Skipping or rushing through tasks in order to get back on schedule almost inevitably leads to trouble further downstream in the project. Each task in a phase builds an important foundation for the next task, and each phase does the same for the next phase. Projects are inherently risky; an experienced project manager does not add risk for the sake of expediency. Additionally, the logical design phase, through the diagrams and documents that it produces, is effectively the last opportunity before the system is built for system owners and users to validate the functional requirements, and to make corrections or clarifications where needed.

14. The decision analysis phase can be perceived as representing a transition in progress from the business concerns to the technological concerns. This is reflected in the roles of the various stakeholders.

As with other phases and tasks, the systems analyst generally acts as a facilitator in identifying candidate solutions.

System owners and users provide their thoughts and suggestions, but are generally not directly involved in this task. Of course, as the owners and users, their suggestions should never be dismissed out of hand even if they may be technologically naïve; it is important to remember they are partners in the solution.

System designers and builders will generally suggest a number of potential solutions, based upon their areas of expertise and knowledge. Some candidate solutions may also be limited and predefined, since the organization may require the candidate solutions to be compatible with the organization's existing and/or approved technology architecture. This is particularly true in public sector organizations and large private sector companies.

Ultimately, the systems analyst should capture the characteristics of the different candidate solutions using a matrix similar to Figure 5-19 in the textbook.

15. Typically, you would use at least four sets of criteria for your evaluation:
    1. Technical feasibility, i.e., is it practical to design, build, implement and maintain this system from a technical standpoint?
    2. Operational feasibility, i.e., does it meet users' needs and expectations, and if so, how well does it do so? Do the users feel position about the candidate solution, and how much will it change how they work?
    3. Economic feasibility, i.e., do the one-time and ongoing costs compare favorably to the anticipated benefits?
    4. Schedule feasibility, i.e., Can the candidate solution be developed and implemented within the needed or desired timeframe?

    In general, you would get input regarding the candidate solutions from a representative group of stakeholders. The type of input would depend upon whether they are from the business or technological side of the organization. Systems owners and users would typically provide input regarding operational, economic and schedule feasibility, while systems designers and builders provide input regarding the technical feasibility. Where appropriate, external experts may also provide input.

    Comparisons should not be drawn between candidate solutions until each individual candidate solution is analyzed and evaluated. The purpose is to avoid any premature decision or bias.

    Most methodologies provide as a deliverable some type of feasibility analysis matrix, so that decision makers are able to readily compare the different candidate solutions.

Project and Research

1. The student should be able to develop and describe the information requested in Question 1a – 1c from different viewpoints and perspectives. For Question 1d, the student should be able to succinctly describe problems and opportunities in terms of their urgency, visibility, benefits, priorities, and possible solutions using the format given in the textbook. Responses should be logical, consistent and reasonable.

2. Responses should follow the procedures and models described in Tasks 2.1 through 2.4. The problem domain and business vocabulary should be described and/or defined in business terms. The cause-and-effect analysis should be plausible and the cause should reflect the root problem. Process models should resemble data flow diagrams, and should also include system volumes, response times, and bottleneck points. System improvement objectives should be specific, measurable, and state the actual business objective, not a requirement.

3. The Executive Summary should follow the format shown in Figure 5-12, and be no more than two pages in length. The Executive Summary should be concise, consistent with the information developed in the preceding tasks, and sufficiently complete so that an informed decision to be made by the executive steering committee without referring to the body of the report.

4. Responses for Question 4a should indicate an understanding of what system requirements are, the process for identifying them, and their relationship to system improvement objectives. Responses for Question 4b should indicate an understanding of the prioritization process and the essential difference between mandatory and desirable requirements.

5. a. Answer: Response should be consistent with the steps described in Task 5.1 Response should be cognizant of the roles of different stakeholders in this task and also that this intent of this task is identification only; evaluation does not yet take place.

   b. Response should be consistent with the matrix shown in Figure 5-19. Candidate solutions are open-ended, but the description of their characteristics in the matrix should be accurate and complete. Also, response should not be evaluative.

6. a. Response should be consistent with the steps described in Task 5.2. Response should be cognizant that feasibility analysis extends beyond costs

and benefits. Response should also emphasis the need to ensure that evaluation of each candidate solution is independent; alternatives should not be compared against each other at this point to avoid the risk of losing objectivity and making a premature decision.

b. Response should be consistent with the matrix shown in Figure 5-20, and each candidate solution should be identified against the four sets of criteria described in
Task 5.2. The evaluation of each candidate solution should stand alone and be plausible.

Minicases

1. Answers will vary. However, some thought and analysis should be given to the opportunity for understanding the customer that has been created by this technology. The retailer company now has an even more powerful way to collect information and to interpret complex behavior. Should the company also employ this tactic? The arguments can be made for and against – so look for a logical argument in the students' answer. For example, the CIO could chose not to employ the technology due to expected consumer backlash. Privacy is a concern whenever videoing is used. The company could, for example, survey consumers to determine strength of feeling on the matter. If consumers feel strongly, the company could not only not use videomining, the could employ signaling (hypercompetition) to let potential customers know that the competition is using this "privacy invading" technology.

2. Okuno instituted five activities to facilitate change acceptance:

- Draft – the very best employee(s) from a group were drafted into another group, so that they could learn many aspects of the business process
- Price Control – Each department was required to "purchase" its input materials from another department, and "sell" the creation to the department 'downstream' in the manufacturing process. The departments had to manage payroll and other costs of development. This was used to teach economic and monetary principles to the largely non-college educated employee base.
- Tornado – managers were kept from entering the plant on a given day and for a total of three days, and then were taken to other companies to learn good business practices. The employees, given no warning of their managers leave, were forced to resolve business problems themselves and complete assigned tasks. The outcome was empowered employees and better trained managers.

- Hangen (cutting in half) – Each team was cut in half – one half was required to complete the needed work, and the other half was sidelined, but watched and gave advice. This provided for brainstorming on how to improve efficiency.
- Manager/team meetings – meetings were established to promote the free-flow of ideas and concerns.

Change acceptance is critical to the success of an IT implementation. No matter how good a system or program is, if people don't use it, then it is a failure.

3. Issues are, but are not limited to: intangible costs that should include lost sales due to customer fears of privacy loss, potential brand image damage as well as potential lost productivity from employees. Intangible benefits will largely be increased sales due to better marketing and product choice, as well as product positioning in the store. Costs should include employee training, legal costs, hardware, software, and labor for analyzing the videos. The discount rate is determined by risk and expected reward, stability of the company, and overall economic health of the society.

4. Note: Make sure the tasks are inclusive of all aspects of the video-mining project. Examples are: hardware and software purchases, installation, networking and hiring of needed talent to analyze the information.

Team and Individual Exercises

1. Example: Wireless networks in a bank are deeply constrained due to legal requirements for data security. Over time, as security of wireless increases, this will be less of an issue.

2. As a team, brainstorm some ways to enhance employee change acceptance of new information systems or business processes.

3. Think of an example when Business Process Improvement is more appropriate than Business Process Reengineering. Share with the class.