Information Systems
ISM 3011

Fall 2003
Unit 12A

Assignment for Next Class

- Read and prepare case studies 1, 2, and 3 (p. 552 – 554).

Systems Investigation and Analysis

Chapter 12

An Overview of Systems Development

Participants in Systems Development

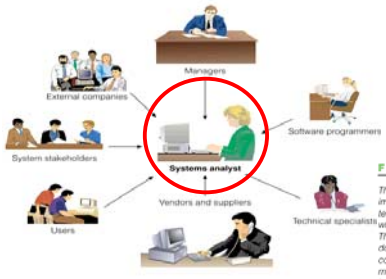


FIGURE 12-1
The systems analyst plays an important role in the development team and is often the only person who sees the system in its totality. The one-way arrows in this figure do not mean that there is no direct communication between other team members. Instead, these arrows just indicate the pivotal role of the systems analyst—an individual who is often called on to be a facilitator, moderator, negotiator, and interpreter for development activities.

Initiating Systems Development

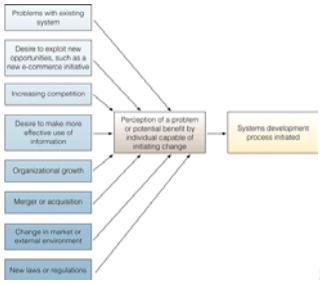


FIGURE 12-2
Typical Reasons to Initiate a Systems Development Project

Trends in Systems Development and ERP

- ERP vendor as one-stop provider
- Applications to integrate with ERP systems
- External consulting

Systems Development Life Cycle (SDLC)

The Traditional Systems Development Life Cycle

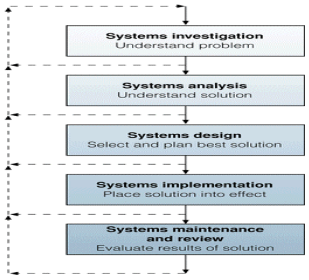


FIGURE 12-6

The Traditional Systems Development Life Cycle
Sometimes, information learned in a particular phase requires cycling back to a previous phase.

Systems Development Life Cycle - Problems

FIGURE 12-5

The later that system changes are made in the SDLC, the more expensive these changes become.

Advantages and Disadvantages of Traditional SDLC

TABLE 12-1

Advantages and Disadvantages of Traditional SDLC

Advantages	Disadvantages
Formal review at the end of each phase allows maximum management control.	Users get a system that meets the needs as understood by the developers; this may not be what was really needed.
This approach creates considerable system documentation.	Documentation is expensive and time-consuming to create. It is also difficult to keep current.
Formal documentation ensures that system requirements can be traced back to stated business needs.	Often, user needs go unstated or are misunderstood.
It produces many intermediate products that can be reviewed to see whether they meet the users' needs and conform to standards.	Users cannot easily review intermediate products and evaluate whether a particular product (e.g., data flow diagram) meets their business requirements.

Prototyping

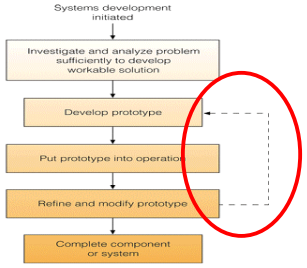


FIGURE 12-8

Prototyping is a popular technique in systems development. Each generation of prototype is a refinement of the previous generation based on user feedback.

Prototyping

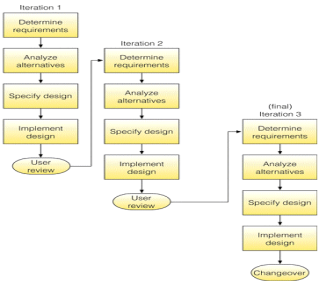


FIGURE 12-7
Prototyping Is an Iterative Approach to Systems Development

Advantages and Disadvantages of Prototyping

TABLE 12-2
Advantages and Disadvantages of Prototyping

Advantages	Disadvantages
Users can try the system and provide constructive feedback during development.	Each iteration builds on the previous one. The final solution may be only incrementally better than the initial solution.
An operational prototype can be produced in weeks.	Formal end-of-phase reviews may not occur. Thus, it is very difficult to contain the scope of the prototype, and the project never seems to end.
As solutions emerge, users become more positive about the process and the results.	System documentation is often absent or incomplete, since the primary focus is on development of the prototype.
Prototyping enables early detection of errors and omissions.	System backup and recovery, performance, and security issues can be overlooked in the haste to develop a prototype.

Rapid Application Development (RAD)

TABLE 12-3
Advantages and Disadvantages of RAD

Advantages	Disadvantages
For appropriate projects, this approach puts an application into production sooner than any other approach.	This intense SDLC can burn out systems developers and other project participants.
Documentation is produced as a by-product of completing project tasks.	This approach requires systems analysts and users to be skilled in RAD system development tools and RAD techniques.
RAD forces teamwork and lots of interaction between users and stakeholders.	RAD requires a larger percentage of stakeholders' and users' time than other approaches.

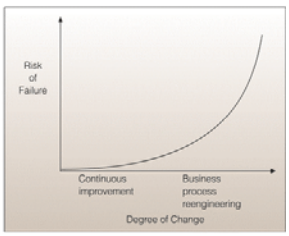
Extreme Programming

- An integrated set of techniques based on simplicity and feedback.
  - Pair Programming
  - Testing
  - 40-Hour Week
  - Code is integrated once per day

http://www.xprogramming.org

Factors Affecting Systems Development Success

FIGURE 12-9
Degree of change can greatly affect the probability of a project's success.



Project Management

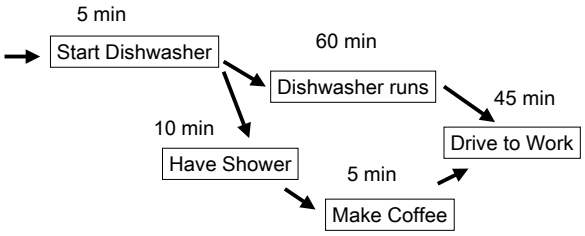
- Project schedule
- Project milestone
- Project deadline
- Critical path

Use of Project Management Tools

FIGURE 12-11
Sample Gant Chart
A Gantt chart shows progress through systems development activities by putting a bar through appropriate cells.

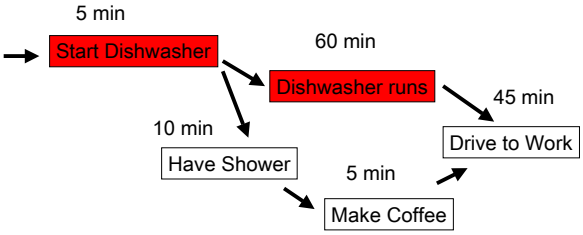
Dr. Martin Hepp

Critical Path Method (CPM)



Dr. Martin Hepp 20

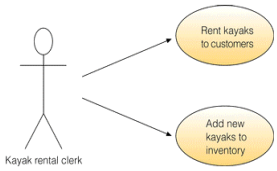
Critical Path Method (CPM)



Dr. Martin Hepp 21

Object-Oriented Systems Investigation

FIGURE 12-14
Use Case Diagram for a Kayak Rental Application



Dr. Martin Hepp 22

Systems Analysis

Dr. Martin Hepp 23

Identifying Sources of Data

Internal Sources	External Sources
Users, stakeholders, and managers	Customers
Organization charts	Suppliers
Forms and documents	Stockholders
Procedure manuals and policies	Government agencies
Financial reports	Competitors
IS manuals	Outside groups
Other measures of business process	Journals, etc.
	Consultants

FIGURE 12-16
Internal and External Sources of Data for Systems Analysis

Dr. Martin Hepp

Collecting Data

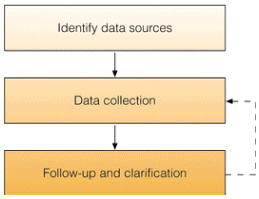
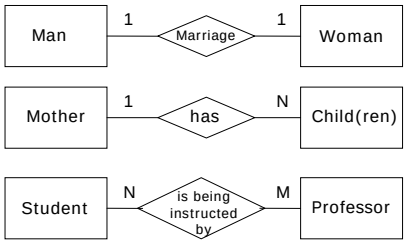


FIGURE 12.17
The Steps in Data Collection

Dr. Martin Hepp

Entity Relationship Modeling



Dr. Martin Hepp 26

Thank you!

The slides will be available on the internet at
http://ruby.fgcu.edu/courses/mhepp/
(-> CRN80040)

Dr. Martin Hepp 27