Developing Business Information Systems

Chapter 13
Information Systems are dynamic, are ever changing, and must be constantly upgraded.
The System Life Cycle

- Death
- Birth
- Development
- Production
Application Software: How Do We Get It?

- Purchase a proprietary software package
- Use an application service provider (ASP)
- Develop a custom information system
System Development: Structured System Design

- Encourages top-down design
- Structure chart

Payroll System

Employee Database
  - Creation
  - Maintenance

Report Process
  - Hourly
  - Salary

General Ledger
**System Development: Diagrams**

**Data Flow Diagram:** Focus is on information flow

**Entity Relationship Diagram:** Focus is on attributes of entities and relationship between them
System Development: Flowcharting

Combination of symbols and flow lines portrays the logic of the program

- Main Program
- Loop
- Subroutines

- Computer Process
- Predefined Process

- Decision
  - Yes: Terminal Point
  - No: Printed Output

- Online Storage
- Input/Output
- Display
The CASE Tool Kit

- Computer-Aided Software Engineering
- Software Engineers
- Custom Programs

Courtesy of Federal Express Corporation. All rights reserved.
CASE Tool Kit Contains

- Design tools
  - User interface
  - Screen generator
  - Layout
  - Report generator
- Information repository tools
- Program development tools
  - Application Generator
  - Generation of text data
Feeling Dizzy?
Information Overload?
Prototyping

- Developing Prototype Systems
  - Scalable Systems
  - System Specifications (Specs)
  - Functional Specifications

- Three Objectives
  - Analyze current situation
  - Identify information needs
  - Develop a model of the target system

- Rapid Application Development (RAD)

Serendipitous Surfing: Business
Define System Specifications

- Existing system review
- System objectives
- Design constraints
- Requirements definition
Create Prototype

- Rough out logic of system
- Decide how elements fit together
- Define the I/O interfaces
- Create software
Prototyping Process Phases

3

Refine Prototype

• System evaluated
• Suggested improvements
Develop Operational System

- Non-scalable prototype: develop another system
- Scalable prototype: implement system
System Conversion

- Unit Testing
- Systems Testing
- Approaches to System Conversion
  - Parallel Conversion
  - Direct Conversion
  - Phased Conversion
  - Pilot Conversion
System Implementation

System becomes operational!

System Maintenance starts

Patches necessary
Programming: Languages

- Machine Language
- Procedure-Oriented: COBOL
- Object-Oriented (OOP): C++
- Fourth Generation (4GL)
- Visual Programming: Visual BASIC
Programming: Writing the Program

- System specification review
- Program identification and description
- Coding, testing, and documentation
End of Chapter 13