ACS-1803 Introduction to Information Systems

Instructor: Trevor Nadeau

Building/Aquiring Information Systems

Lecture Outline 12

Principles and Learning Objectives

- Understand the process used by organizations to manage the development of information systems
- Describe each major phase of the system development life cycle: systems identification, selection, and planning; system analysis; system design; system implementation; and system maintenance
- Understand the factors involved in building a system in-house, along with situations in which it is not feasible

Why do organizations need new Systems?

- Organizations need new IT systems to meet business objectives including:
  - improving competitiveness
  - increasing productivity and efficiency
  - accelerating growth
  - supporting innovation
  - reducing costs
- At times external factor also drive the need to build information systems
  - Regulatory
  - New technologies

Buying Off-the Shelf Software

- Today, most organizations purchase or rent software they need
- Software applications can vary from an unmodified, commercial off-the-shelf (COTS) software package to a custom, written-from-scratch program
- Typically build proprietary software when the requirements and functionality are truly unique, or if there is an opportunity to sell the software.
- --Why?

Systems Development

- Sometimes, a system that is needed by a business is not available as a 'canned' system which can be bought and installed

- Such a system must be custom-built either by systems developers in-house or by external consultants

System Development Life Cycle

- Usually a system is developed in 6 specific SDLC stages:

1. Systems Investigation:
   a) Problem Definition
   b) Feasibility Study
2. System Analysis
3. System Design:
   a) General Design
   b) Detailed Design
4. System Development (Building)
5. System Implementation
6. System Maintenance

Systems Investigation

- The purpose is to gain a clear understanding of the specifics of the problem to be solved or the opportunity to be addressed
- Feasibility analysis: assessment of the technical, economic, legal, operational, and schedule feasibility of a project
- Steps of the investigation phase
  1. Review systems investigation request
  2. Identify and recruit team leader and team members
  3. Develop budget and schedule for investigation
  4. Perform investigation
  5. Perform preliminary feasibility analysis
  6. Prepare draft of investigation report
  7. Review results of investigation with steering team

Systems Analysis

- This phase of systems development involves:
  - Gathering data on the existing system
  - Determining the requirements for the new system
  - Considering alternatives within identified constraints
  - Investigating the feasibility of alternative solutions

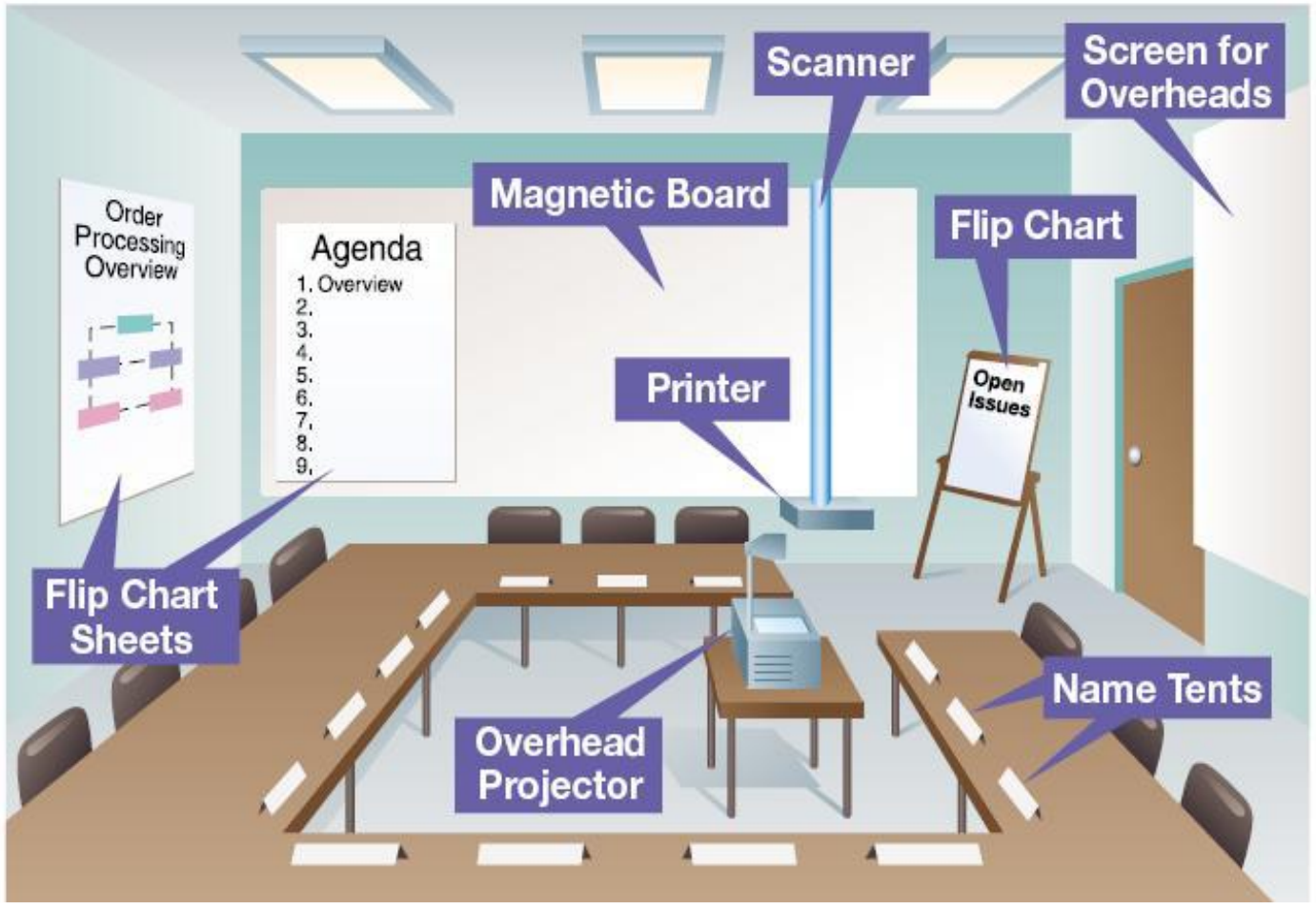
Steps in the systems analysis phase

1. Identify and recruit team leader and team members
2. Develop budget and schedule for systems analysis activities
3. Study existing system
4. Develop prioritized set of requirements
5. Identify and evaluate alternative solutions
6. Perform feasibility analysis
7. Prepare draft of systems analysis report
8. Review results of systems analysis with steering team

Systems Analysis – Critical Success Factors

- Critical Success Factor (CSF) is an element which is necessary for a project to successfully achieve its goal
  - For example, a CSF for a successful system design is user involvement.
- CSF is a means of identifying the tasks and requirements needed for success, and A means to prioritize requirements
- At the lowest level, CSFs become concrete requirements

Systems Analysis – Joint Application Design



- JAD is a special type of a group meeting in which all (most) users meet with an analyst at the same time
- Users jointly define and agree upon system requirements or design dramatically reducing the design time

Systems Design

- Creates a complete set of technical specifications that can be used to construct the information system

Steps in the systems design phase

1. Identify and recruit team leader and team members
2. Develop schedule and budget for systems design activities
3. Design user interface
4. Design system security and controls
5. Design disaster recovery plan
6. Design database
7. Perform feasibility analysis
8. Prepare draft of systems design report
9. Review results of systems design with steering team

System Construction

- The phase of systems development that converts the system design into an operational
- Steps:
  - Acquiring and installing hardware and software
  - Coding and testing software programs
  - Creating and loading data into databases
  - Performing initial program testing

Integration & Testing

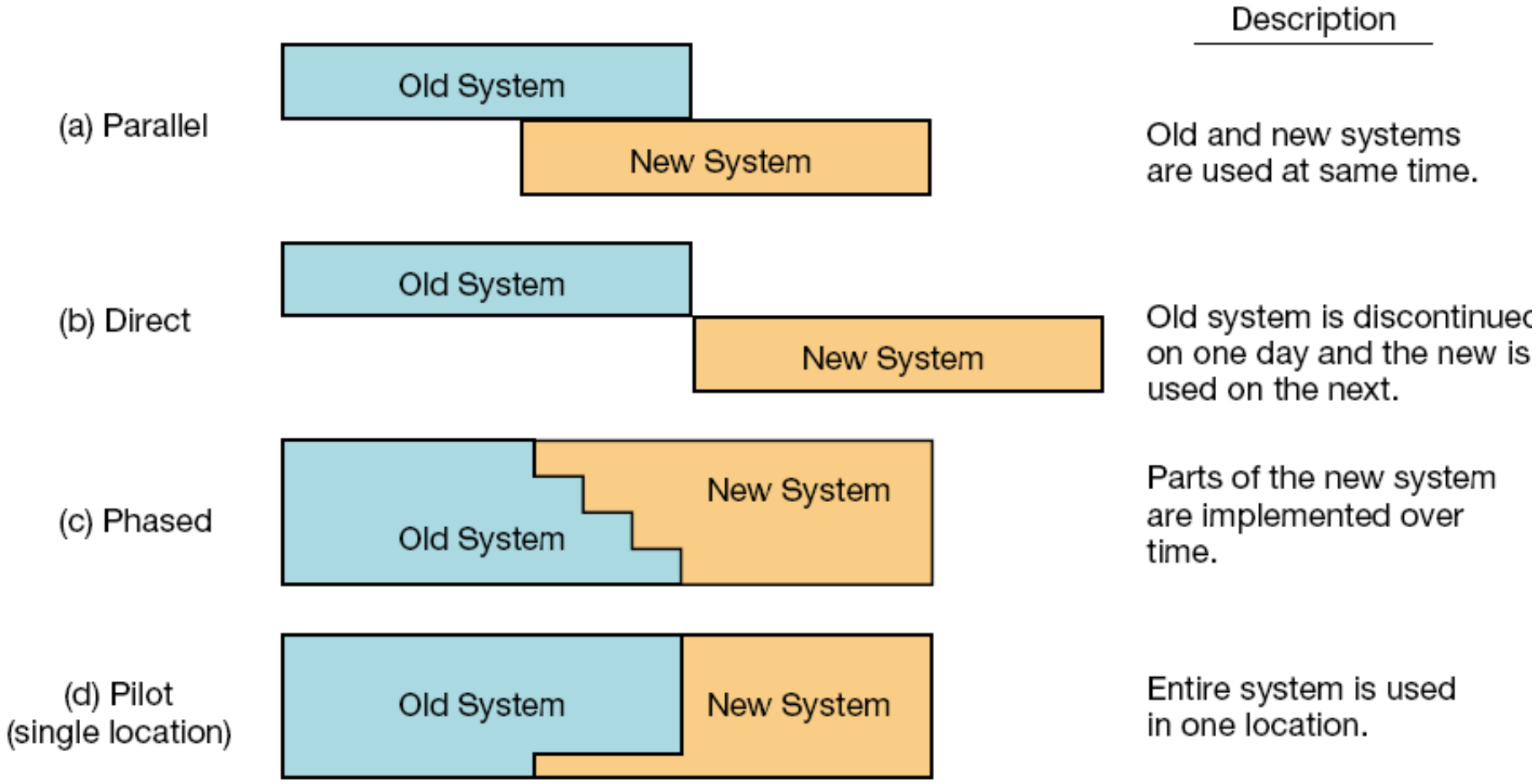
- Types of testing
  - Integration testing
  - System testing
  - Volume testing
  - User acceptance testing

System Implementation

- Successfully introducing an information system into an organization
- The major challenges to successful implementation of an information system are often more behavioral than technical
  - Strong, effective leadership is required to overcome the behavioral resistance
- Steps involved in implementation
  - User preparation
  - Site preparation
  - Installation
  - Cutover

System Implementation: System Conversion

Conversion Approaches – the process of taking information from an old system to populate a new system. This is accomplished through manual and/or automated methods – each comes with a different level of risk



Systems Operation and Maintenance

- Systems operation: using a new or modified system under all kinds of operating conditions
- Systems maintenance: changing and enhancing the system to make it more useful in achieving user and organizational goals

Alternatives to Systems Development

Needs for Alternative to Internal Systems Building

Often it is not feasible to consider building an information system internally. Below are four situations that discourage the idea of internal development:

Limited IS Staff	The IS organization does not have the capability to build the system itself
Limited IS Skill Sets	The IS organization does not have personnel with the correct skill sets to build the system
IS Staff is Overworked	Current IS organization staff demands and priorities make it impossible to build a system
Problem IS Performance	The IS organization does not have the appropriate performance level to build a system

Alternatives to Internal Development

End-user Development
Capitalizing on the sophistication of current users, this technique allows the **system users** to develop their **system functions** using a variety of **tools**

External Acquisition
The purchasing of an **existing system** (hardware, software, databases, network) from an **outside vendor** such as IBM, EDS, or Accenture

Outsourcing
Turning over some or all **responsibility** for an organization's information **systems development** and **operations** to an outside firm

End-User Development - Tools

Tools

Many tools can be used by end users for business applications including some of the following:

- **Personal Computer Tools** – tools common to PCs (e.g. spreadsheets) that allow the creation of macros or automated routines by users
- **Query languages/report generators** – tools that utilize SQL to extract information from databases and generate reports in user-specified formats
- **Graphics generators –** tools that can extract relevant information from databases and create graphs, charts, etc.
- **Decision Support and Modeling Tools** – tools that support complex decision making utilizing multidimensional models
- **Application Generators –** tools that allow users to specify what is to be done and the application decides how it is to be done and generates the program code

End-User Development - Benefits

End-User Development Benefits

Embracing end-user development can help address some common issues with IS development, including:

- **Cost of labour** – IS managers can substitute hardware and the systems for users to develop systems for IS staff
- **Long development time** – allowing users to develop, shortens development lead times
- **Slow modification** – users can react to their own needs
- **Work overload** – leveraging the talent of the end-user staff, in effect, increases the size of the IS staff

End-User Development - Pitfalls

Pitfalls

End-user development is not a panacea. Common issues arise as its use becomes more prevalent, including:

- **Lack of Standards** – if **users do not** adopt **standards** used by **IS professionals** (documentation, error checking, testing procedures), the **IS staff** can experience **problems** when supporting **users**, or troubleshooting **data integrity** and **security problems** when enterprise systems are involved
- **Lack of continuity** – as users leave the organization or department, new employees may not **understand** existing **programs** and lose productivity learning or developing new applications or **"reinventing the wheel"**
- **Appropriate Use of Time** – there is debate whether it is appropriate for **skilled users** and **managers** to be spending time on **systems development**

External Acquisition – Steps to Proceed

Steps in External Acquisition

1. **System identification, selection, and planning** (same as internal method)
2. **Systems analysis** (same as internal method)
3. **Development of a Request for Proposal (RFP)** – a report that is used to tell vendors what the requirements are and how they might be able to meet those requirements (hardware, software, training, etc.)
4. **Proposal evaluation** – may include viewing system demonstrations, evaluating the performance of those systems, and examining criteria important to the organization and judging how the proposed systems respond to those criteria
5. **Vendor selection** – using a scoring system devised to evaluate the competing proposal and then selecting the proposal that best fits the organization's needs

System Development Life Cycle:

1. Systems Investigation:
    a) Problem Definition
    b) Feasibility Study
2. System Analysis/ Requirements Gathering
3. Development of Request for Proposal (RFP)
4. Proposal Evaluation
5. Vendor Selection
    - Product demonstrations, ratings, and site visits.

Solution Review

6. System Design
7. System Development (Functional Customization)
8. System Implementation
9. System Maintenance

Package Evaluation Phase

- Steps in the Package Evaluation Phase include:
  1. Identify potential solutions
  2. Select top contenders
  3. Research top contenders
  4. Perform final evaluation of leading solutions
  5. Make selection
  6. Finalize contract

Package Evaluation Phase

- Identify Potential Solutions
  - Project team should make a preliminary assessment of the software marketplace to determine whether existing packages can meet the organization's needs
  - Request for Information (RFI) is a document that outlines an organization's needs and requests vendors to respond with information about if and how they can meet those needs

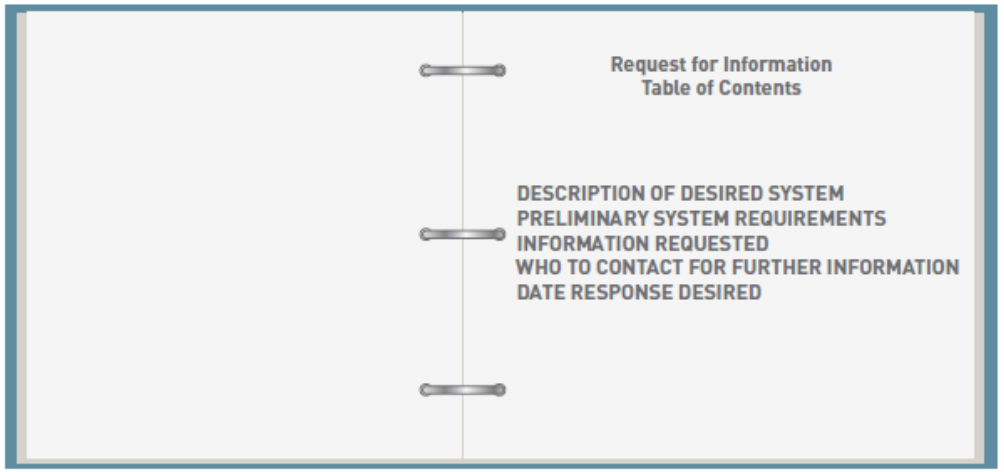


FIGURE 12.20
Recommended table of contents for a request for information
The RFI outlines the desired system and its requirements, identifying key pieces of data that the software vendor must include in the proposal.

Package Evaluation Phase

- Select Top Contenders
  - Project team will review information provided by vendors in response to the RFI
  - Selection will be made based on:
    - How well the vendor's software appears to meet the organization's needs
    - Preliminary cost and timing estimates
    - Information gleaned from references
    - How easy the vendor has been to work with so far

Package Evaluation Phase

- Research Top Contenders
  - Begins with a detailed investigation as well as in-depth discussions with two or three customers of each contender
  - Contenders should be asked to make a final presentation and demonstrate their solution using a performance evaluation test
    - Conducted in a computing environment, with a workload that matches intended operating conditions

Package Evaluation Phase

- Make Selection
  - Weigh factors such as:
    - How well the vendor's solution matches the needs of the users and business
    - The amount of effort required to integrate the new software with existing software
    - Results of the performance evaluation test
    - Relative costs (including any software modifications) and benefits
    - The technical, economic, legal, operational, and schedule feasibility
    - Input from legal and purchasing resources on the legal and financial viability of the contender
    - Feedback from customers on how well the software performs as well as on the quality of the support provided by the vendor

Finalize Contract

- Develop a fair contract when acquiring new computer hardware or software
- Allow at least two months for review and negotiation of a final contract
- Take special precautions in signing contracts with the service provider of cloud-computing or software-as-a-service
- Contract should have provisions for:
  - Monitoring system modification quality and progress
  - Ownership and property rights of the new or modified system
  - Contingency provisions in case something doesn't work as expected
  - Dispute resolution if something goes wrong

Implementation

- Key implementation tasks include:
  - Use data-flow diagrams to map current business processes and requirements to the software, and identify any gaps that must be filled by changing current processes or by modifying the software.
  - Install the software and configure all of its capabilities and options to meet the project requirements.
  - Customize any aspects of the solution needed for the organization.
  - Integrate existing software with the new software.
  - Train end users.
  - Test the software to ensure that it meets all processes and requirements.
  - Convert historical data from the old software so that it can be used by the new software.
  - Roll out the new software to users in a live work environment.
  - Provide for ongoing end-user support and training.

Custom-built vs Off-the-Shelf Software

Factor	Custom-built (make)	Off-the-Shelf (Buy)
Cost	Cost can be difficult to estimate accurately and is frequently higher	The true cost is also difficult to estimate accurately but is likely to be lower
Needs	Custom software is more likely to satisfy your needs	Might not get exactly what you need
Process improvement	Existing business processes will be supported even if they are poor	May simplify or streamline a poor existing business process
Quality	Quality can vary depending on the development team	Can assess the quality before buying
Speed	Can take years to develop	Can acquire it now, but implementation and adoption may also take years
Staffing and support	Requires skilled resources to build and support a custom-built solution	Requires paying the vendor for support, but internal resources are also required
Competitive advantage	Can develop a competitive advantage with good software	No real competitive advantage

Outsourcing – Why Consider?

A firm might outsource some (or all) of its information system service for many reasons:

- **Cost and quality concerns** – current cost and quality of information systems is unacceptable
- **Problems in IS performance** – IS is having trouble meeting acceptable service standards
- **Supplier pressures** – aggressive sales tactics
- **Simplifying, downsizing, and reengineering** – having a need to focus on core processes
- **Financial factors** – turning over IS systems can strengthen a balance sheet
- **Organizational culture** – political or organizational problems that are difficult for IS to overcome
- **Internal irritants** – tension between the IS staff and users

Outsourcing – Types of Arrangements

Outsourcing Arrangements

Not all outsourcing arrangements are the same. They can vary based on the need of the firm and the supplier.

Arrangement Types

- **Basic** – A "cash and carry" relationship in which products and services are purchased on the basis of price and convenience
- **Preferred** – Relationships with a few suppliers where the buyer and supplier set preferences and prices based on mutual benefit (e.g. volume pricing)
- **Strategic** – A relationship where the firm and a vendor are each concerned with, and perhaps have a direct stake in, the success of the other

Outsourcing – Relationship Management

Managing the IS Relationship

Ongoing management of the outsourcing alliance is the single most important aspect of the outsourcing project's success. The following are recommendations for best management:

- **A strong CEO and CIO** should continually manage the legal and professional relationship with the outsourcer
- **Clear, realistic performance measurements** of the systems and of the outsourcing arrangement (e.g. tangible and intangible costs and benefits)
- **The interface between the customer and outsourcer should have multiple levels** (i.e. links to deal with policy and relationship issues)

Systems Development

End of Lecture 12