SYST 542
Decision Support Systems Engineering

Instructor: Kathryn Blackmond Laskey
Fall Semester, 2006

Unit 1: Decision Making and Decision Support
Learning Objectives

• Understand course objectives & requirements
• Define a DSS
• Describe the history of DSS
• Name and define the major functional components of a DSS
• Describe the DSS lifecycle
• Describe the role of DSS in decision making and the kinds of decisions most amenable to DSS
Course Objectives

• Introduce decision support systems
• Provide sound basis for:
  – Designing DSS
  – Managing DSS lifecycle process
  – Evaluating DSS
• Provide systems view of DSS development and integration into organization
Course Requirements

• **Weekly discussion question (30% of grade)**
  – Asynchronous discussion
  – In-class discussion with assigned facilitator
  – Written summary
  – 50% participation, 50% content

• **Project (50% of grade)**
  – Small groups
  – Design and implement DSS for problem of your choice
  – Written report
  – Oral presentation

• **Paper review (15% of grade)**
  – Read a paper from the literature
  – Write report on paper
  – Give oral presentation

• **Lead discussion session (5% of grade)**
A **decision support system** is a computer-based system that supports the decision making process

- Assist decision makers in semi-structured tasks
- Support **not replace** human judgment
- Highly interactive
- Improve effectiveness of human decision makers

“**A decision support system** is a system under the control of one or more decision makers that assists in the process of decision making by providing an organized set of tools to impart structure to portions of the decision-making situation and improve the ultimate effectiveness of the decision outcome”

- Marakas
Why DSS?

• Increasing complexity of decisions
  – Technology
  – Information:
    » “Data, data everywhere, and not the time to think!”
  – Number and complexity of options
  – Pace of change

• Increasing availability of computerized support
  – Inexpensive high-powered computing
  – Better software
  – More efficient software development process

• Increasing usability of computers
  – COTS tools
  – Customization

Computer support for decision making
Rational Decision Making

• Rationality is the use of reason to make the best choice one can in the circumstances
  – What does “best” mean?

• Aspects of rationality (Kant)
  – Cognitive rationality: What to believe?
  – Practical rationality: What to do?
  – Evaluative rationality: What to value?

• GOOD-D mnemonic for rational decision making
  – Identify the goal to be achieved by the decision
  – Identify the options available to the decision maker
  – Evaluate the likely outcomes if each option is chosen
  – Decide which option is best
  – … And then Do it!

• Decision makers need support with all GOOD-D elements
Decision Making Process

1. Define and Structure the Problem
2. Gather, Collect and Fuse Data
3. Generate Options
4. Evaluate Options
5. Select Option(s)
6. Implement Selected Option
Types of Problems

• **Structured**
  – Repetitive
  – Standard solution methods exist
  – Complete automation may be feasible

• **Unstructured**
  – One-time
  – No standard solutions
  – Rely on judgment
  – Automation is usually infeasible

• **Semi-structured**
  – Some elements and/or phases of decision making process have repetitive elements

DSS most useful for repetitive aspects of semi-structured problems
Goal: Use best parts of IS, OR/MS, AI & cognitive science to support more effective decision making

History of DSS

Operations Research
Management Science

Information Systems

AI/Expert Systems

Transaction Processing Systems

Optimization

Expert Systems

Simulation

Knowledge Representation

Judgment & Decision Making

MIS

Human/Computer Interaction

Decision Support Systems

1940’s

2000+

Thanks to Andy Loerch
History: Business Computing

• World War II Era
  – Introduction of computers - Military and scientific applications
  – Computers were for “number crunching”

• 1950’s
  – Business applications
  – Transaction processing systems: billing & payroll
  – Large mainframe computers

• 1960-70’s
  – Use of computers in management
  – Large volumes of data stored in computers
  – Invention of relational databases and SQL
  – Management Information Systems born
  – Automation of paper-and-pencil processes for repeatable tasks

• 1980-90’s
  – Movement toward customization & flexibility
  – Movement toward new user interaction metaphors
  – Increasing emphasis on intelligent systems

• 21st Century
  – Move from “stovepipes” to interoperable systems
  – Distributed systems
  – Web services
History: Artificial Intelligence

• 1950’s
  – Introduction of symbolic computing
  – Newell and Simon: General Problem Solver
  – Differentiation from scientific computing
    » “AI is about symbols and not numbers”

• 1960-70’s
  – First expert systems
    » e.g., HEARSAY I (Speech recognition); MYCIN (Medical diagnosis)
  – Knowledge representation - e.g., frames, rules
  – Fuzzy logic

• 1980-90’s
  – Commercialization of AI
  – Expert system shells
  – Connectionist movement
  – Machine learning
  – Incorporation of methods from decision theory and operations research

• 21st Century
  – Agent-based systems
  – Distributed AI
  – Semantic Web & Intelligent “Web Bots”
History: OR/MS

• **World War II Era**
  – Application of scientific method to operational problems
    » e.g., efficient movement of troops & equipment

• **1950’s**
  – OR/MS established as a discipline
  – Standard methods developed

• **1960-70’s**
  – Expansion of OR/MS
  – Business applications
  – Government: McNamara’s “whiz kids”
  – Problems with appropriate use and acceptance
    » Limited computing power
    » Exclusion of factors not easily quantified
    » Human factors issues in how OR/MS integrated into organizations

• **1980-90’s**
  – Movement toward customization & flexibility
  – Attention to organizational and human factors
  – Incorporation of methods from artificial intelligence
  – Model bases and model management

• **21st Century**
  – Embedded systems
  – Agile, reconfigurable supply chains
  – OR for everyone
    » Excel Solver
    » OR in middle school
  – OR in a Web Services world
Decision Support Trends

- IT is increasingly pervasive
- Users are increasingly computer savvy
- Computer hardware is increasingly smaller and more powerful
- Systems are increasingly interconnected
- The Web is increasingly interwoven into all aspects of our lives
- Demand for usable, flexible, powerful decision support will continue to grow
- Decision support will be embedded into a wide variety of consumer and business products
Discussion

• Give an example of some decision support systems you have encountered
  – What kind of decision was supported?
  – How did it work?
  – How helpful was it?

• What makes for successful decision support?
• What pitfalls should be avoided?
Some Terminology

- **DBMS** - System for storing and retrieving data and processing queries
- **Data warehouse** - Consolidated database, usually gathered from multiple primary sources, organized and optimized for reporting and analysis
- **MIS** - System to provide managers with summaries of decision-relevant information
- **Expert system** - Computerized system that exhibits expert-like behavior in a given problem domain
- **Decision aid** - Automated support to help users conform to some normative ideal of rational decision making
- **DSS** - Provide automated support for any or all aspects of the decision making process
- **EIS (Executive information system)** - A kind of DSS specialized to the needs of top executives
## Traditional Emphasis

<table>
<thead>
<tr>
<th></th>
<th>Data</th>
<th>Dialogue</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBMS/ DW</td>
<td>🍃</td>
<td>🍃</td>
<td>🍃</td>
</tr>
<tr>
<td>MIS</td>
<td>🍃</td>
<td>🍃</td>
<td>🍃</td>
</tr>
<tr>
<td>ES</td>
<td>🍃</td>
<td>🍃</td>
<td>🍃</td>
</tr>
<tr>
<td>DA</td>
<td>🍃</td>
<td>🍃</td>
<td>🍃</td>
</tr>
<tr>
<td>DSS/ EIS</td>
<td>🍃</td>
<td>🍃</td>
<td>🍃</td>
</tr>
</tbody>
</table>
Humans and Computers: Complementary Strengths

• Human decision makers
  – Good at seeing patterns
  – Can work with incomplete problem representations
  – Exercise subtle judgment we do not know how to automate
  – Often unaware of how they perform tasks
  – Poor at integrating large numbers of cues
  – Unreliable and slow at tedious bookkeeping tasks and complex calculations

• Computers
  – Still inferior to humans at pattern recognition, messy unstructured problems
  – Good at integrating large numbers of features
  – Good at tedious bookkeeping
  – Rapid and accurate at complex calculations
The Challenge: Find and Exploit Synergy

• Computers provide cognitive tools
  – You would not build a house without appropriate tools
  – Complex decision problems require cognitive tools to assist with:
    » Collecting and organizing relevant information
    » Weighing multiple factors relevant to choice
    » Integrating large numbers of factors and combining to form overall evaluation
    » Presenting results so rationale for choice is clear
    » Analyzing multiple “what-if” scenarios

• Goal of DSS:
  – Use strengths of computer to augment strengths of human
  – Improve overall effectiveness of decision making process
Danger: The Worst of Both Worlds

• A poorly designed or improperly deployed decision support system can be
  – *Worse* than leaving users unassisted
  – *Worse* than replacing the users with automated system

• Can you explain why?

• How do we keep this from happening?
Achieving the Promise

• Understand the stakeholders
  – Involve stakeholders early and often
  – Listen to feedback (especially negative!)

• Understand the task
  – Objectives to be achieved by decision
  – Current decision making process
  – Human and organizational factors

• Understand the technology
  – What parts of current process can be automated
  – COTS versus custom development
  – Integration of components and non-automated functions

• Understand the DSS development process
  – Co-evolution of process, DSS, human skill sets
  – Why change is resisted
  – Importance of good systems engineering

Iterate! Evaluate! Improve!
DSS Characteristics

• Supported task
  – Structurability
  – Level (strategic / tactical / operational)
  – Decision process phase
  – Application area
  – Real-time / non real-time
• Supported user(s)
  – Type of job
  – Single user / multiple users
    » Distributed?
    » Interactive?
  – Sophistication with computers
  – Mode of interaction

• Level of support
  – Display information?
  – Suggest solutions?
  – Select solutions?
  – Modify suggestions with user feedback?

• Information sources
  – User input
  – Internal database
  – External database
  – Internet (web / email)
  – Sensor observations
Functional Components of a DSS

- Model Management
- Knowledge Engine
- Dialogue Management
- Data Management
- Library of Models
- External Data Sources

User & External Environment
System Life Cycle

1. Identification of Need
2. Concept Definition
3. Preliminary System Design
4. Detailed Configuration Item Design
5. System Integration
6. Production & Manufacturing
7. Training
8. Deployment
9. Operation
10. Maintenance
11. Refinement
12. Retirement

Analysis: Design: Build: Test: Operate

Figure 1.1: Buede, 2000

Systems engineers play major role.
Lifecycle Models

• There are many lifecycle models
  – Can you name some and describe their properties?

• All have phases for:
  – Definition
  – Development
  – Deployment

• Lifecycle model for DSS development must provide for:
  – User involvement and evaluation throughout design & development
  – Iterative evaluation-centered redesign
Stakeholder Involvement

• Most system errors can be traced to poor requirements definition
• Problems caught early are much less expensive to fix
• Communication gaps are inevitable and should be planned for
  – Users understand current process but can’t imagine how technology can change process
  – Developers understand technology but not user’s job
  – Each party thinks its expertise is most important
• Requirements definition must be iterative and evolutionary
  – “I can’t tell you what I want but I’ll know when I see it”
Focus on Constant Improvement

- Improvements happen because we learn from experience
- We can learn from experience only if:
  - We can tell *whether* we have succeeded or failed
  - We can tell *why* we have succeeded or failed
- Good engineering involves interplay between
  - Theory
  - Analysis
  - Reality testing
- Document your experience
  - Identify problems
  - Trace causes
  - Identify lessons learned
  - Incorporate lessons learned into next project
- Many of the most difficult problems are interpersonal & organizational, not technical
Case Study: Planning A Program of Study

• The Problem:
  – Students at GMU must plan a program of study to meet degree requirements
  – Existing PatriotWeb system provides degree evaluation to check whether requirements have been met but no planning function to help students plan courses that meet requirements
  – There are constraints on feasible schedules
    » Requirements (major and concentration)
    » Prerequisites
    » Work and childcare constraints
    » Course time conflicts
    » When courses are offered

• The Users:
  – GMU students (graduate and undergraduate)

• DSS Objective:
  – Provide a decision support tool to help students plan a program of study
Case Study Objectives

• Use program of study DSS as an illustrative example of concepts discussed in course
• Use program of study DSS as focus for asynchronous discussions
• Project groups may (if they choose) do a project on some aspect of the case study
In Summary...
References