Simulations vs. Simulation

• Distinction between a simulator and Simulations lies in one being a “surface” model while the other is a “deep” model
  – A design question that must be answered.

Simulations Vs. Simulator

Method

• The construction is linear, one simulation at a time
• The construction is incremental: each simulation adds new features to the model
Simulations Vs. Simulator

Size of the Project

- Only a few tasks to simulate
- Numerous tasks

Simulations Vs. Simulator

Budget

- Increases linearly with the number of simulations
- Increases logarithmically (opposite to exponentially) with the number of simulations; the last simulation costs less than the first one

Simulations Vs. Simulator

Skills of the Team

- Easier for non-computer expert
- Can be done without simulation authoring tools
- The most economical solution with computer expert and/or using simulation authoring tools
Simulations Vs. Simulator Tools

- Does not need specialized tools
- Better with simulation or object-oriented authoring tools

Simulations Vs. Simulator Management

- Easier to control: once the first simulation is done, it's done!
- Needs professional and computer-educated management

Simulations Vs. Simulator End User Acceptance

- Less user control, little latitude for choice of action
- Wide scope of action
Simulations Vs. Simulator

Advantages

- Well adapted for drill and practice
- Can work under different modes: demo, tutorial, free discovery, troubleshooting

Simulations Vs. Simulator

Risks

- Users with high expectations (Nintendo generation) can be frustrated
- Simulation goes into limbo

Can a Non-Programmer Design a Simulation?

- Yes . . .
- And No . . .
- Depends on the tools one has and the degree of complexity wanted/needed.
Design Reusable Components

• Cuts cost -- reusing the software
• Using them as building blocks can speed development

Structure Documents

• Titles, headlines, body
• Using a structured language such as SGML (Standard Generalized Markup Language) helps

Prototyping -- a product in process

• Create the skeleton of the model and user interface
• Validate it
• Revise the prototype and revalidate
• Refine and revalidate
• Deliver the final product
Simulations

Problem centered
• Building a model
• Analyzing outcomes
• Testing a theory
• Getting data from statistical distributions
• No user action

User centered
• Piloting the program
• Analyzing data
• Learning/teaching new skills
• Testing skills
• The learner
• Accepts user actions

Simulation Techniques

Problem-Centered Simulations
• Main feature -- Problem centered
• User’s main activity -- Building a model and Analyzing outcomes
• Role of the Model -- Testing a theory
• Who is in command -- Getting a theory
• Simulation loop -- No user action

User-Centered Simulations
• Main feature -- User centered
• User’s main activity -- Piloting the model and Analyzing data
• Role of the model -- Learning/teaching new skills and Testing skills
• Who is in command -- The learner
• Simulation loop -- Accepts user actions
Training Simulations is a Team Effort

- Requires:
  - Manager
  - Simulation Designer(s)
  - Training Specialist(s)
  - Subject Matter Expert(s)
  - Programmer(s)
  - Graphic Artist(s)
  - Editor(s)

Assignment

- Proposal Due March 3.
  - (date correction from earlier posting)
  - Email it as an attachment.

- Read Empathy-Insight
  - Email answers to me before class

- Keep working on Learning Activity 2