Designing Learning Environments

Steve Peterson

Thayer School of Engineering, Dartmouth College
High Performance Systems, Inc.
13 Dartmouth College Highway
Lyme, NH 03768
603-795-4857

ABSTRACT

For the past 18 months, individuals at High Performance Systems have been engaged in the development of *Learning Environments* — personal computer based arenas for learner-directed learning. These learning environments have been used in a variety of settings, in grades 8-12, in colleges, and in corporations. Our experience suggests that learning environments can be of great utility in facilitating the learning process.

In this paper, I summarize what we have learned from our development activities. I begin by developing a generic learning environment framework. Then, I illustrate this generic framework with two examples. The first example learning environment is developed from the classic "Deer on the Kaibab Plateau" model. The second learning environment is motivated by the "Tracking Experiment" described by Richardson (1984). I conclude the paper with a set of learning environment design principles.

Introduction

Over the past two years, the words "learning environment" have gradually become part of the lexicon of the system dynamics community. Increasingly, learning environments (and the associated concepts "learning laboratories" and "management flight simulators") have become the topics for discussions, papers, and research in system dynamics. In this conference, for example, several presentations will focus on learning environments.

But there is still much confusion associated with the words. What *is* a learning environment? What constitutes an effective learning environment design? And, are there any examples that demonstrate the utility of learning environments?

This paper represents an attempt to clarify the concept of a learning environment. Over the past 18 months, individuals at High Performance Systems, Inc. have been engaged in the development of learning environments. During that time, we have had numerous opportunities to test our ideas by working with individuals in a variety of classroom and corporate settings. From our experience, we are learning what it takes to design an effective learning environment. The purpose of this paper is to report what we have learned.

I begin the paper by developing a generic learning environment framework. This framework is one that has evolved over time. It is what we currently use as a starting point for our own learning environment development activities. It also serves as the template for learning environments created with STELLA®Stack (Peterson, 1990, 19-55) . Second, I illustrate the generic framework with two examples. The first example is a STELLAStack learning environment which focuses on the management of deer populations on the Kaibab plateau. The second example investigates the effects of delays, biases, and distortions in a feedback control system. This latter example was motivated by the "Tracking Experiments" described by George Richardson in his Ph.D. thesis (Richardson 1984, 374-375). Finally, I conclude the paper by providing a set of learning environment design principles.

A Generic Learning Environment Framework

When we use the words "learning environment", we refer to a personal computer based arena in which people are free to learn using their own style, at their own pace, and in a sequence that they determine. One or more simulation models provide the "engine" for learning within the environment. A learning environment thus is a vehicle for active, learner-directed learning. An effective learning environment will facilitate learners as they *reconstruct* the substance of the environment.

In our own experience, an *effective* learning environment has two fundamental characteristics. First, it provides tools that facilitate the development of five critical thinking skills. These skills are: Dynamic Thinking, Generic Thinking, Structural Thinking, Operational Thinking and Scientific Thinking. These five skills are ones that "good" systems thinkers use consistently in their thinking activities. They are discussed in detail by my colleague Barry Richmond in "Systems Thinking: A Critical Set of Critical Thinking Skills for the 90's and Beyond" (Richmond 1990). Second, an effective environment engages the user in the learning process. By providing a rich context, as well as acontextual "puzzles", the environment will tap the inherent motivation of the learner.

Operationally, an effective learning environment uses different activities to weave together tools, context, and puzzles. The environment provides a way for users to seamlessly navigate among these activities. In the more effective learning environments that we have seen or designed, these activities fall into the basic categories shown in the generic framework on the next page.

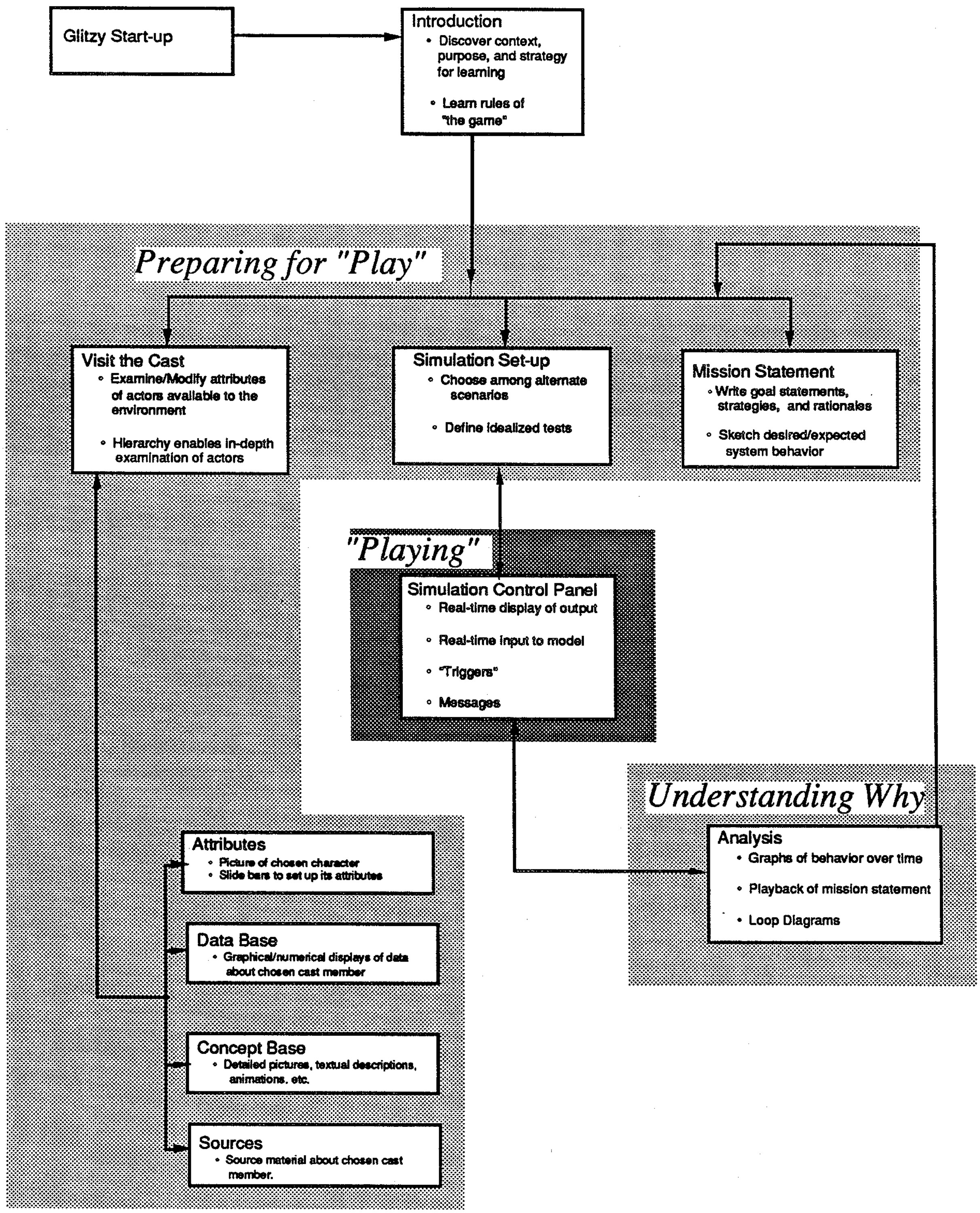
As the diagram indicates, this generic learning environment framework contains a clear, logical structure. Within the broad confines of this progression, users have the latitude to move about as they wish, at their own pace, using their own style. Beginning with a glitzy start-up (essential for members of the video age!), users drop into an Introduction. From here, users engage in three distinct "tracks": Preparing for "Play", "Playing", and "Understanding Why" . The diagram indicates how the user can navigate within, and among, the different tracks. Let's briefly consider what's involved in this generic learning environment.

It is impossible to overstate the importance of a good introduction to a learning environment. The introductory screens allow the user to discover the context of the environment, the purpose of the environment, and the strategies available for learning in the environment. Without a solid context, a clear purpose, and an understanding of the strategies to be employed, the user is quite likely to become frustrated. Reactions such as "Why am I doing this?" or "What am I supposed to be doing now?" are clear signals that your introductory material needs to be enhanced.

Preparing for Play. As the diagram shows, "Preparing for Play" incorporates three major activities. These activities (Visit the Cast, Mission Statement, and Simulation Set-up) are engaged in *before* the user conducts a simulation. The activities enable users to learn more about the substance of the environment. They help users make explicit their thinking about the issues addressed in the environment.

Visit the Cast: The "Visit the Cast" activity enables users to learn more about the actors — both animate and inanimate — available in the learning environment. An important Visit the Cast activity consists of the examination and modification of the attributes of these actors. It is useful, for example, for a learning environment in a corporate context to enable users to specify attributes such as learning curves for newly hired individuals. The process of modifying attributes encourages the user to "own" the assumptions in the application. While engaged in this process, the user cannot help but ask "what if" questions about the assumptions that are being altered.

A Generic Learning Environment Framework



The "visit" is also an appropriate time to learn more about the issues at hand. Because a learning environment is likely to generate numerous substantive questions, it is important to provide easy access to relevant information. Pictures, tables, graphs, textual descriptions, and source material can easily be incorporated into a learning environment. This information, when provided "as needed", can go far to help users develop their understanding of how the system works. Using a tool such as HyperCard, supplementary material (specific to a particular actor) can be arranged in a hierarchical fashion. Although lodged fairly deep within the environment, such material can still remain within easy grasp of users.

Mission Statement: After Visiting the Cast, users should be encouraged to record a Mission Statement. This activity is an important one — it fosters both dynamic thinking and scientific thinking. A mission statement consists of a sketch over time of desired (or expected) output from a key variable in the underlying model, as well as a written *a priori* rationale for the sketched behavior pattern.

Simulation Set-up: Simulation Set-up is the final activity in the "Preparing for Play" track. In this activity, users will establish environmental backdrops or scenarios, against which the simulation will be run, Set-up thus provides another opportunity to challenge one's thinking by asking "what if" questions. Alternate scenarios also serve the important purpose of allowing users to test the relative importance of external forces versus internal relationships in the genesis of a system's behavior.

In the set-up activity, users might also define idealized tests for the environment. These tests can be important for the learning process, since they provide a degree of experimental control that is often not possible in real world systems.

Playing. An underlying simulation engine is what distinguishes a learning environment from other computer aided instruction applications. In our own work, simulation engines are typically created using the STELLA software. In the "Play" track, users access this engine to run a simulation. As the diagram indicates, this activity occurs on the simulation "control panel". On the control panel, users will receive real-time display of model output. They also will be able to submit real-time inputs to the model.

"Flying the airplane" is an excellent way to test understanding of the system under investigation. Active participation in the management of the system creates realism that can draw the user into the learning environment. If facilitated carefully, this activity stimulates the user to think through the dynamic implications of the corrective actions taken while "in flight".

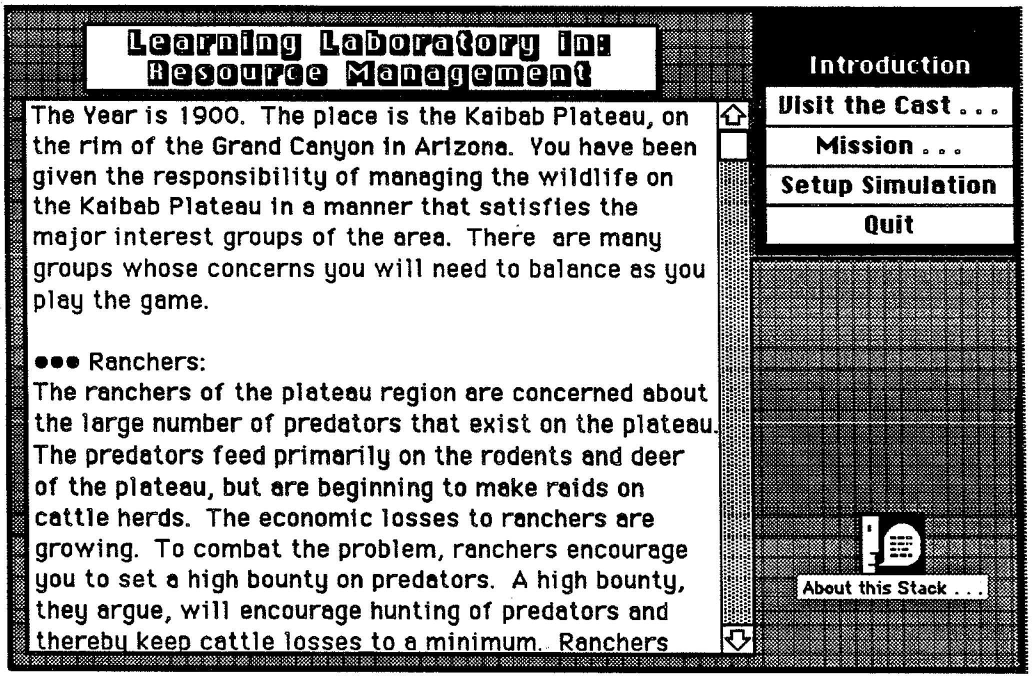
Understanding Why. The final track in the generic learning environment framework is accessible after a simulation run is completed. "Understanding Why" activities fall under the broad heading of "analysis". A graphical playback of simulation results superimposed on the mission pattern enables the user to see what happened, in comparison with what was desired or expected. Any discrepancies between desired and actual create a strong incentive to figure out why. Finally, text, pictures, simple loop diagrams, and tools for testing simpler simulation engines can help users along as they seek to understand why the simulation unfolded as it did.

Depending on the purpose of the learning environment, it may be unnecessary to incorporate all the activities in generic learning environment framework. What's presented here is by no means revealed truth. The framework does, however, work well — both in business and in education. The basic activities, and the basic progression are common to virtually all effective learning environments that we have seen or designed ourselves.

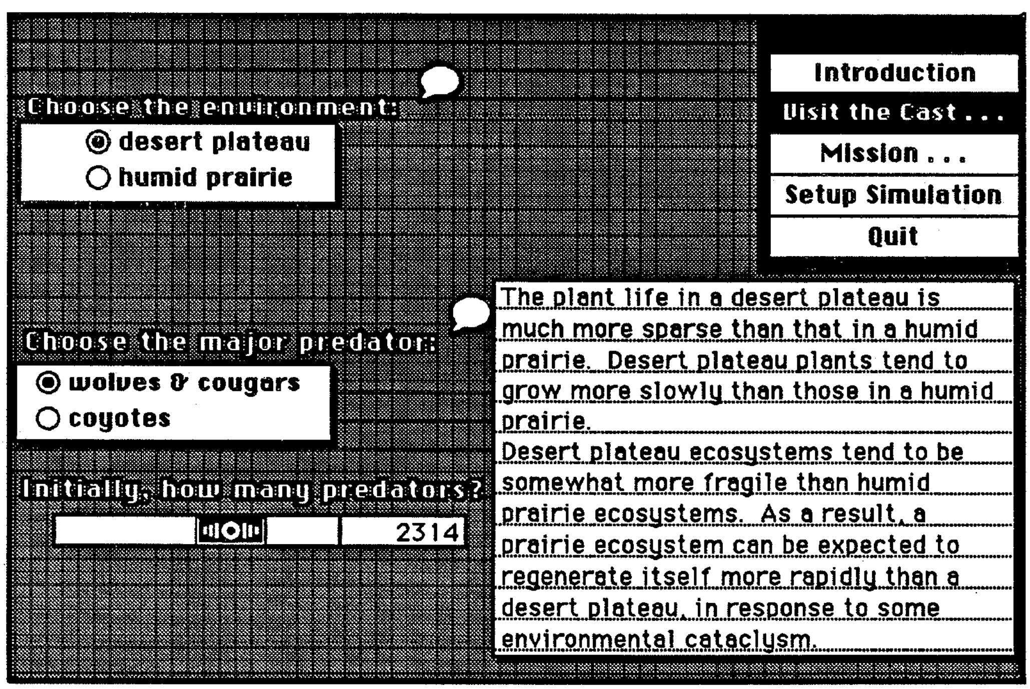
Example 1. Learning Laboratory in Resource Management:

A STELLAStack application named "Learning Laboratory in Resource Management" (Peterson, 1990, 13-16) provides a first illustration of the generic learning environment framework. This application was developed from the classic "Deer on the Kaibab Plateau" model often used in system dynamics classes.

Upon opening the stack, you'll find yourself looking at the introduction card shown below. This card contains a scrollable field that indicates what the user must do to play, and underscores the concerns of the major interest groups in the area. As they read through the text, students will begin to "feel" the pressures faced by real resource managers. This "feeling" will deepen as they continue using the application.



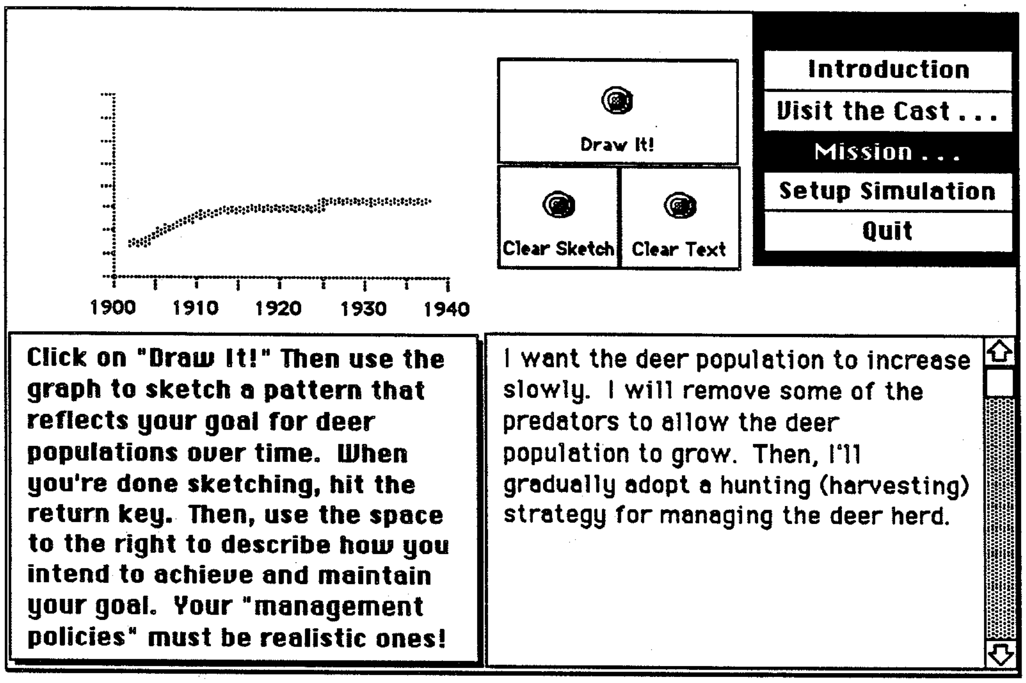
The right side of the Intro card contains a set of buttons. These buttons allow the user to navigate among the cards in the stack. A click on the "Visit the Cast . . ." button, for example, leads to the card shown below.



Before playing (or between rounds) the user can use the buttons and slide bar to define the environmental characteristics of the Kaibab Plateau. It is possible to define the region as a humid prairie, or a desert plateau, by clicking on the appropriate radio button. Depending on which button is "on," a different set of parameter values is sent to the underlying STELLA model. Similarly, one can define the type of predator that inhabits the area, and how many predators are in the area initially.

Note also the two "tell about" balloons, located near the two sets of buttons. A click on one of these balloons will reveal information about the implications of the user's choice. The field to the left of the card shows environmental information, for example.

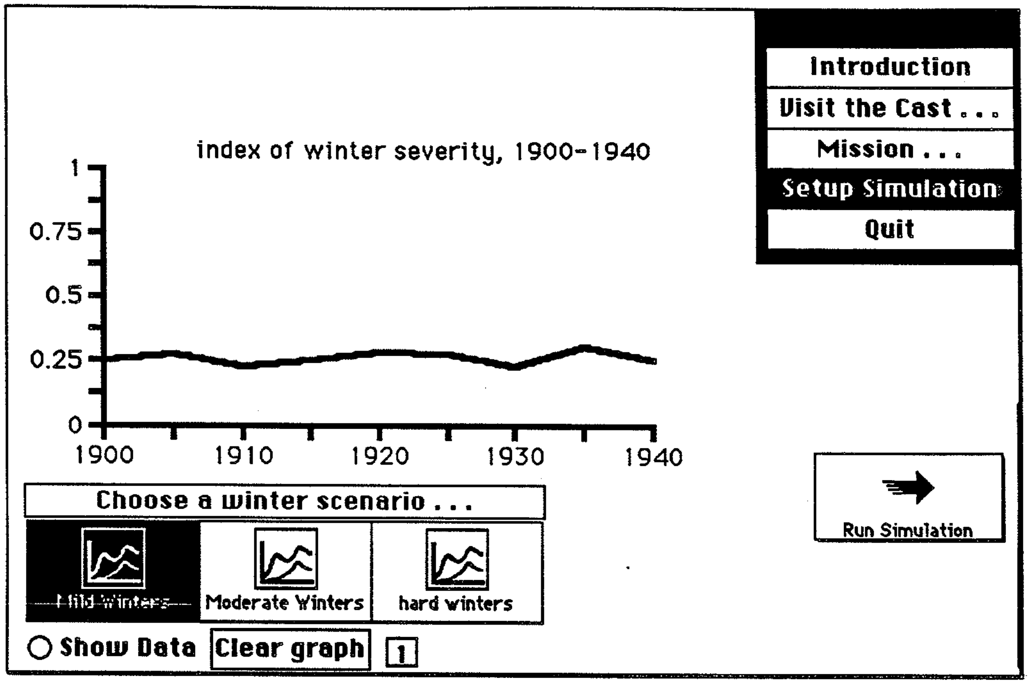
Once the cast has been visited, it's time to define a mission statement. A click on "Mission . . ." will take the user to the appropriate card, shown below. As the card shows, defining a mission statement entails, in this case, sketching over time the desired level of deer population. It also entails writing a statement of how the user intends to achieve this desired pattern. The concept of a mission statement is vitally important for learning. By specifying a graph over time, one is "putting a stake in the ground". Thus, the user is nudged into the role of an active learner. And, by writing out strategies, users are asked to be explicit about how they think the ecosystem works. When confronted with simulation results, students will be able to use any discrepancies as an aid to gaining a better understanding.



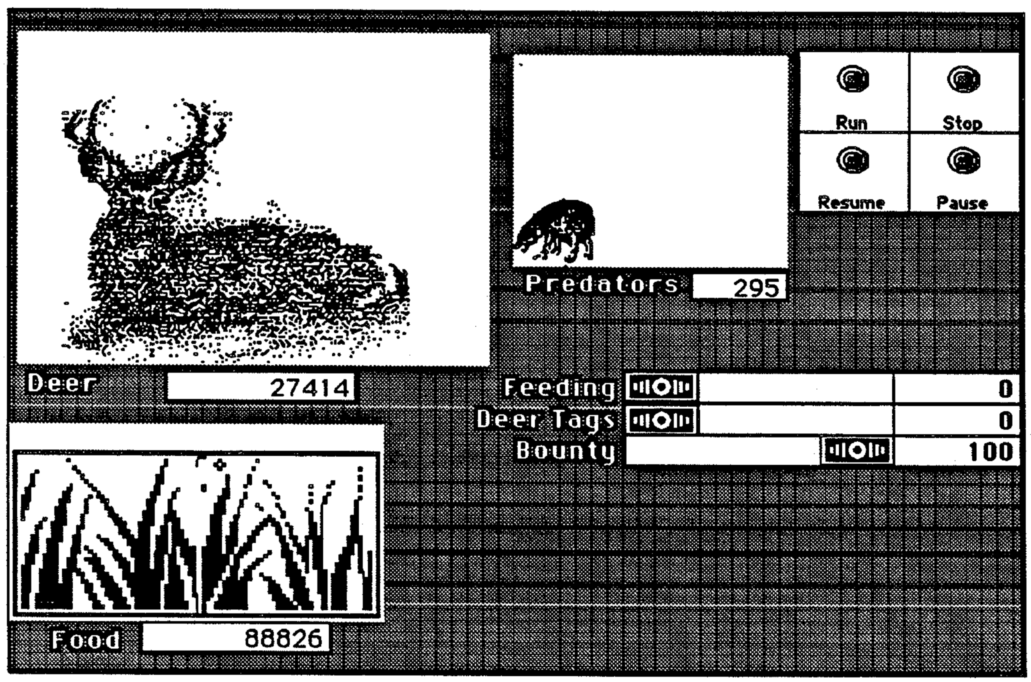
Finally, before running the simulation, it's necessary to set up an environmental backdrop within which the model will be run. This is done on the "Setup Simulation" card, as shown on the next page.

As the card shows, I've chosen a set of mild winters for the period 1900-1940. Users can choose among mild, moderate, or hard winters during the period. Doing so will enable one to test the effects of the environment on the internal dynamics of the ecosystem.

Once environmental conditions have been defined as desired, a click on "Run Simulation will take the user to the control panel. It's also shown atop the next page. On this card, users will send resource management decisions to the STELLA model by moving the slide bars for

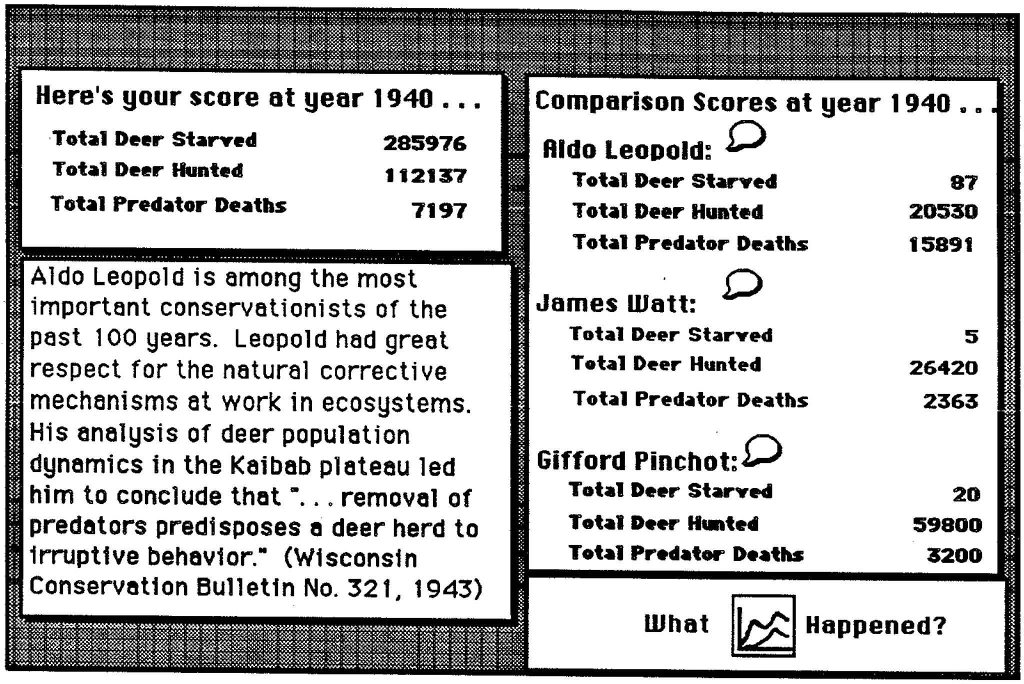


Feeding, Deer Tags, and Bounty back and forth. In return, they will receive information about the resulting state of the Kaibab plateau. The pictures of predators, deer, and food will change in size as the amount of each changes in the system.

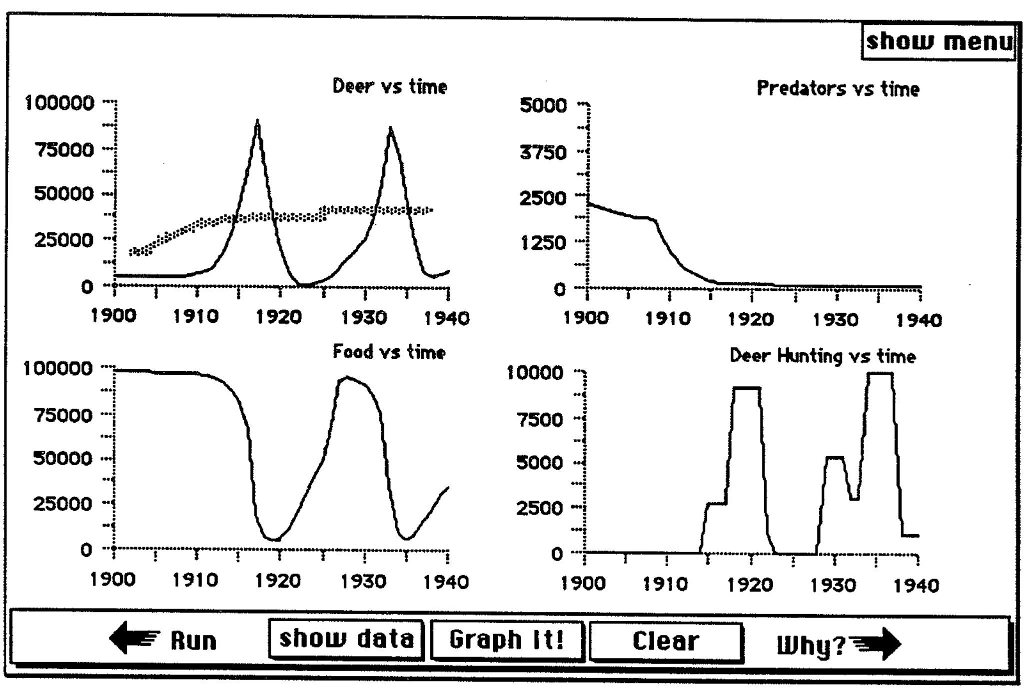


On the right side of the card is a set of buttons that give the user control over the simulation. Another set of buttons will appear at the bottom of the card when the simulation is stopped. These will allow for navigation through the stack. After playing the game, for example, you might want to look at your score. A click on "Analyze Simulation" will bring you to a card that looks like the one shown atop the next page.

On the scorecard, you can compare the results of your management policies with the results from simulated policies of famous (or infamous) American resource managers. You also can learn a bit more about these people, by clicking on their "tell about" balloons.



The results that you get, especially when compared with the results for Aldo, James, and Gifford, are likely to leave you somewhat puzzled. A click on "What Happened?" will help to ease your puzzlement. You will be taken to a plotting card. On this card, a click on "Graph It!" will reveal plots like the ones shown below.



The first plot clearly shows differences between what a user *thought* would happen (as reflected in their mission statement sketch), and what actually *did* happen. The plots also provide the information needed to begin to understand *why* things happened as they did.

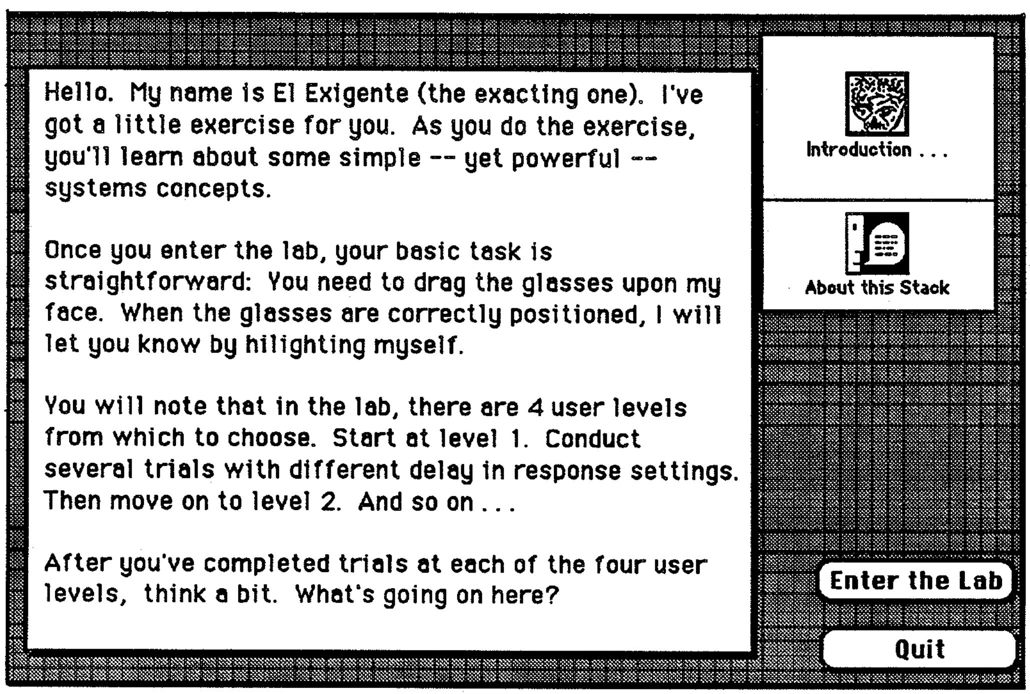
For further help, you might want to click on "Why?" Doing so, will lead you through a sequence of simple causal loop diagrams that help to explain the basic population control mechanisms at work on the Kaibab plateau. Working through these diagrams will help you to gain a better appreciation for the interdependencies in this system. Armed with a greater appreciation and understanding, you then might try playing another round (attempting to improve your performance).

Example 2. The Feedback Stack

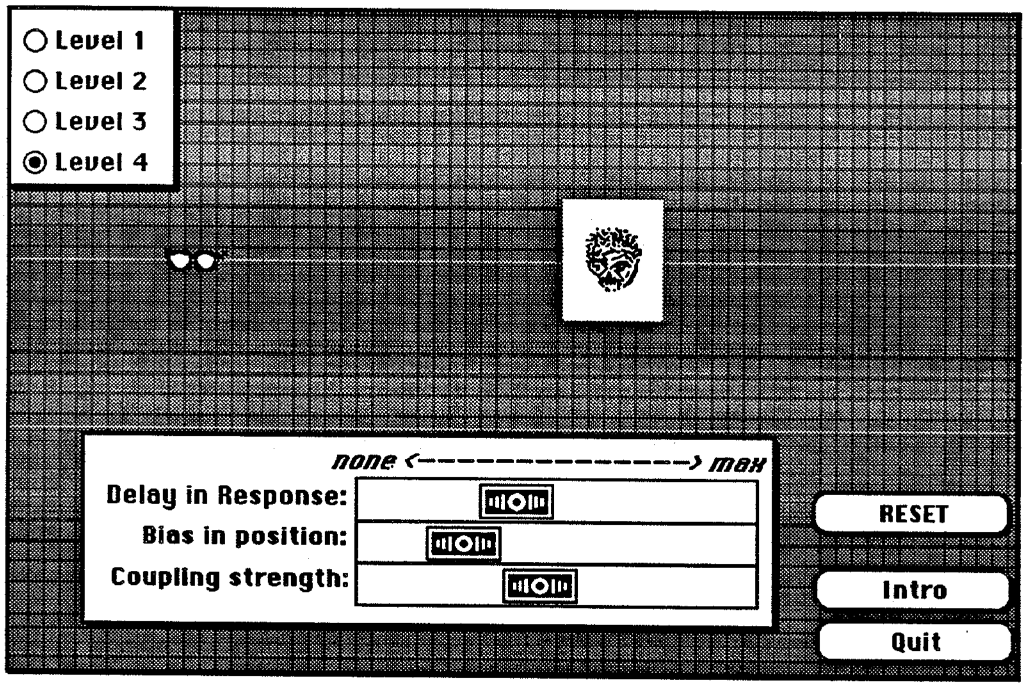
A HyperCard Stack named " The Feedback Stack" provides a second illustration of the generic learning environment framework. This application was motivated by a section in George Richardson's Ph.D. thesis (Richardson 1984, pp. 374-375), in which he discusses a classical tracking experiment in psychology. In that experiment, a subject attempts to track a moving spot on a screen with a cursor controlled by a joystick.

The Feedback Stack evolves this experiment somewhat. The stack is deceptively simple in its design — it incorporates only a small subset of the learning environment activities described earlier. Nonetheless, it is rich in its utility. Our brief experience with the stack indicates that the application goes far to exercise skills of dynamic thinking, generic thinking, and scientific thinking. Indeed, it can serve to spring-load exercises devoted to the development of structural thinking and operational thinking skills.

The stack consists of two cards. The first is an introductory card, shown below. This introductory card sets the context for the exercise. As the card shows, the basic task to be completed in the Feedback Stack is to drag a pair of glasses onto the face of El Exigente. Once El Exigente senses that the goal is successfully sought, he "highlights" himself.



Once the user enters the laboratory (the second card of the stack), he or she can progress through four different user levels. The first level enables the user to modify the delay in the feedback control loop of which he or she is a part. The second level allows for different biases in the perception of the goal being sought. With a higher bias, there is a larger distance between the actual position of El Exigente's face and what's being perceived by the user. The third level allows for adjustment in the strength of coupling between the movement of the glasses, and the movement of El Exigente. With high coupling strengths, a one unit movement of the glasses will result in a *greater than* one unit movement of El Exigente — in the same direction. Finally, the fourth level (shown also on the next page) allows the user to combine different delays, biases, and coupling strengths.



A user experimenting within this learning environment soon discovers that goal-seeking activity can quickly become quite difficult to bring to fruition when delays, biases, and floating goals are an integral part of the process. By conducting controlled experiments within the environment, a user can quickly develop a visceral appreciation for the role that each provides in the dynamic behavior generated by a variety of real-world systems.

Summary: Learning Environment Design Principles

The simple, two card interface to the Feedback Stack contrasts sharply with the interface to the Learning Laboratory in Resource Management. Nonetheless, both stacks share a common design philosophy. While there certainly is no one "right way" to design a learning environment, in our own efforts we have discovered a set of principles which help us to keep on track as we develop learning environments for business and education. These are presented below:

1. Be Clear About Your Purpose: Unless you clearly articulate the purpose of your learning environment, its chances of success are slim. State your objectives. Then, design your environment to meet those objectives. In particular, it is useful to target your learning environments toward the development of the critical thinking skills (Dynamic Thinking, Generic Thinking, Structural Thinking, Operational Thinking, and Scientific Thinking) discussed earlier.

2. Use Context to Your Advantage: Context can be an extremely powerful motivating force for learners. Case studies and design exercises based on real issues and problems are compelling. They can fire the imagination of even the most recalcitrant learners. Strive to position your learning environments near to the day-to-day lives of those who use your environments.

3. Keep the User in Control: An effective learning environment will allow a user to direct the learning process. In order to direct, the user must feel in control of the environment. The user interface must be simple and transparent. Icons should be clear and consistent. Users need to know where they are, how to get where they want to go, and what to do once they get there. It's essential, therefore, to establish a clear framework for your learning environments.

4. Be Parsimonious: With HyperCard in particular, it is quite easy to go overboard in the display of model output. It's just as easy to be excessive in the provision of policy levers. Resist the temptation. The best learners may, in real time, juggle 5 - 7 performance indicator variables in their heads, and perhaps be able to manipulate, 3 - 5 policy levers. Most can accommodate less on both scores. If your environment provides too many indicators and levers, your users will go on "overload".

5. Move Beyond Video Games: One of the biggest risks that a learning environment runs is being approached like a video game. That is, because most people enjoy a good game, many users are seduced into trying to "beat the machine". They push buttons and pull slide bars just as if they were playing a Nintendo game. Little substantive learning occurs in this exchange. Users may learn how to "win", but they usually don't have a clue as to what's really going on below the surface.

As a designer of a learning environments, you must be aware of this risk. It's also your responsibility to provide some way for users to discover "why". If all you provide is the opportunity to enjoy a "fun experience of pushing buttons", don't expect much learning to occur. You can minimize the risk either by incorporating analysis tools (such as idealized tests, plots, and loop diagrams) within the environment, or by providing a rigorous de-brief afterward.

Learning environments are emerging as powerful tools for developing the capacity of individuals to think rigorously about dynamic phenomena. By designing them carefully, members of the system dynamics community can go far to disseminate the framework, tools, and technologies associated with systems thinking. The generic framework presented and illustrated here provides one clear design structure for facilitating learner directed learning about dynamic systems.

References:

Peterson, S. 1990. *A User's Guide to STELLAStack.* Lyme, NH: High Performance Systems, Inc.

Richardson, G.P. 1984 *The Evolution of the Feedback Concept in American Social Science.* Albany: State University of New York at Albany

Richmond, B.M. 1990. "Systems Thinking: A Critical Set of Critical Thinking Skills for the 90's and Beyond". Proceedings of the 1990 International System Dynamics Conference.