Designing Management Simulators

David P. Kreutzer and Janet M. Gould
Massachusetts Institute of Technology
System Dynamics Group, E40-294
Cambridge, MA 02139
(617) 253-1955

W. Brian Kreutzer
Gould-Kreutzer Associates, Inc.
River Court, 10 Rogers Street
Cambridge, MA 02142
(617) 577-1430

Abstract: The form of a management flight simulator should follow from the functions it serves for the user. Interfaces designed to facilitate educational interventions should differ in functional form from interfaces designed to provide support systems for executives making real time decisions or conducting scenario planning exercises. Designers should consider the purpose of the interface, the nature of the interaction, the characteristics of the users, the context of use, and the style of presentation before developing the software application. This paper provides examples of how radically different design criteria lead a design team to choose different forms for several management flight simulators and executive-information systems.

"Form ever follows function."
-Louis Henri Sullivan

Introduction

Software has played a critical role in the development of the field of system dynamics. DYNAMO, created in 1958, was the first software that allowed system dynamics model builders to simulate the solutions to system dynamics models. Now there are many other general purpose simulation languages for creating models, stand-alone managerial microworlds, custom interfaces, and systems-thinking-based executive information systems. This greater availability of software, and the increasing diversity of applications for system dynamics models, creates many new design challenges. Models do not need to stand alone. Interfaces are now designed to assist the user as well as the model builder. Interfaces can be simple with a few lines of text or more sophisticated with text, graphics, sound, and animation.

Now the designer must ask questions that were previously considered irrelevant to the model-building process. Should the model be linked to a customized interface? If so, what design will help the user understand elements of the model without explanation? Will the use of animation, video-disk based images, and sound effects improve the user's understanding of the model? Should the learning pathways in the interface be highly structured or open ended? Is it useful to offer links to spreadsheets, databases, and networks? How should the designer determine when new features add or detract from functionality? This paper will illustrate an emerging design philosophy for software interfaces by providing several examples where different circumstances of use, context, audience, and purpose led to radically different design choices.

Simulations, Management Flight Simulators, and Executive-Information Systems

The criteria for evaluating the functionality and style of an interface depends on the purpose and style of interaction needed for the users. Although a great deal of overlap exists, it is useful to distinguish among 3 types of interfaces: 1) those that are used as analytic tools for model builders, 2) those that are used as management flight simulators and in learning laboratories to facilitate learning, 3) and those that are used as executive-information systems designed to improve decision making in real-time data-rich environments.

A model builder using a general purpose simulation language may begin a project by interviewing the end-user (e.g., a manager). The information from the interview may be converted into causal-loop and flow diagrams leading to written equations. The expert model builder can create the simulation, test the model for robustness, discover leverage points, then write a report, or return to the manager and communicate the results. Interface enhancements to general purpose software such as array mappers, documentors, automated causal-loop or flow diagrams, and database managers for storing rerun datasets can improve functionality for such a model builder.

The design criteria for creating management flight simulators or microworlds are substantially different than those for general purpose simulation languages. The purpose of a microworld is to facilitate an exploration of a model and its dynamics rather than to create the model. If the interface is going to be part of a learning laboratory--complete with an introductory workshop, briefing materials, and discussion--then the lessons to be learned are fairly well understood. For example, the People Express simulator (Sterman, 1988) with the accompanying written materials, provides the user with a learning environment for exploring the dynamics of and for learning some of the lessons from the simulator without having to actually build the model.

Recent developments in linking simulation engines to databases and spreadsheets creates another category of interface software. Executive information and strategy systems are designed to support a manager who is interested in using a model of her own system. Strategy systems have models behind the interface that the user may never use explicitly. The interface is designed so that the manager can gain access to information in a familiar format. Data is derived from the model and can be used by the manager to understand, for example, the organization's position in an industry.

Software for Developing Microworlds or Management flight simulators

The success of Sterman's People Express management flight simulator has intrigued many managers. Management flight simulators offer managers a "practice field" in which to learn by experimenting with multiple decisions over multiple trials with the simulator. For example, a simulator of the high-definition television industry allows a user to play out a "competitive dynamics" scenario by struggling to out perform the competition by building market share and increasing profitability. Or users can try launching a new product by setting their goal to be the high volume, low-cost producer, and watching the results of their policy unfold over time. If the strategy fails, they can start the simulation again and plan another strategy.

The core of a management flight simulator is a dynamic model that captures the structure of a particular company or industry and simulates its behavior over time. These models are linked to an interactive computer interface that make it easy to test various strategies and policies. Many examples of managerial microworlds exist, including stand-alone training simulators and decision-support systems. There is a variety of software now available for developing management flight simulators. MicroWorld Creator, STELLAStack, and HyperCard are described here.

MicroWorld Creator™

MicroWorld Creator ™ was designed by Diehl (1992) and first used for the development of the People Express as well as other simulators. The primary advantage of MicroWorld Creator is the rapidity with which a designer can develop the interface from initial ideas to working prototype, and then use the microworld in a learning lab. While it is relatively easy to intrigue a manager with the prospect of an exploratory laboratory for decision-making, it is much more difficult to promise delivery on a modest budget. But, by accepting the standard MicroWorld interface, the design process is simplified by an easy modification of the MicroWorld's environment to fit the manager's needs. The designer is able, therefore, to create a high-quality interface that might otherwise be too expensive to pursue. As an example, a highly successful project was developed on a tight deadline, over a weekend, following a training course. MicroWorld Creator was chosen for its rapid prototyping.

> The microworld was created for a large national oil company which had had stable production for a decade, but decided to expand production by 50% over the next five years. By simultaneously implementing a comprehensive total quality program they hoped to achieve this expansion with only a 15% increase in their labor force. Such an ambitious new strategy would require unprecedented change in the corporate culture. Many employees spent most of their careers in field stations and rarely, if ever, saw the corporate headquarters. Not surprisingly, given the independence of each sector of the new strategy of greatly increasing production levels and efficiency was meeting resistance and increasing cross-functional competition for resources.
>
> The human resource manager attended 1 of 6 introductory systems thinking courses conducted at the company. He expressed his interest in the People Express simulator. He asked if would be possible to build a simulator to help convey the importance of productivity improvement to new employees as part of the company's week-long training program for new employees. He saw the project as an experiment in using management flight simulators for rapid training and culture change. He had three hypotheses: 1) that using management flight simulators would improve the learning process and increase the enjoyment of the training programs; 2) that using the simulator would increase the trainees' understanding and identification with company goals and the total quality management program rather than with local subcultures; 3) that improving an employee's understanding of the whole company would increase communication laterally and vertically, thereby facilitating participation in and development of a more decentralized strategy process.
>
> The discussions with the human resource manager began in the middle of a week long systems thinking course. The initial prototypes of the STELLA model and MicroWorld interface were created over the weekend and received approval on Monday. The simulator would not have been possible at that time with any other software. Although other software choices have many more technical features, they also have much longer design cycles and would have been too expensive for this project (Kreutzer et al., 1992)

HyperCard™ and STELLAStack™

Because of the popularity of the gaming concept, High Performance Systems, Inc. developed a software package called STELLAStack™ which allows modelers to easily connect their STELLA™ models to the HyperCard™ and SuperCard™ prototyping languages. (This product has since been discontinued, but is still widely used. Many of its features have been incorporated into iThink™). HyperCard is a powerful prototyping tool that allows the creation of sophisticated interfaces. It is the best choice when the user needs extensive help buttons, on-line documentation, sound, and animation.

Examples of Management Flight Simulators

Flight simulators have been created with MicroWorld Creator and STELLAStack. The simulators described here were created with STELLAStack and HyperCard.

Scenario Impact Simulator™

Examples of advanced HyperCard features can be seen in the Scenario Impact Simulator™ created in the fall of 1989 by Gould-Kreutzer Associates, Inc. It was designed to allow the user to test a variety of strategies against various scenarios. In order to ensure that the users paid attention to the discrepancies between the results they expected and the actual simulated results, the users were asked to draw the output they anticipated on a time series plot. They could also enter a verbal description of the strategy they pursue. The user drew the expected output curves on the computer screen. After the simulation was run, the actual behavior of the system was then plotted on top of the anticipated behavior.

Because the model was quite large, a method of demystifying it had to be used. The chosen method was the use of hierarchical flow diagrams. The user at first saw an overview chart of the entire model. By clicking on any part of the chart the user was able to explore the model's structure. The users could move through the hierarchical structure until they were at the level of the actual STELLA model. This permitted the user to explore the structure of the model that was used to simulate the scenarios under study. This interface also had a comparative plot feature, which allowed the user to save the results of up to 10 runs, and plot them against one another. The user was also given the choice of playing against another person or the computer.

COPEX™

COPEX™ (Kemeny and Kreutzer, 1992) was jointly developed by Gould-Kreutzer Associates, Inc. and Innovation Associates, Inc. It was created to illustrate the "Fixes that Fail" archetype described in the appendix of *The Fifth Discipline* (Senge, 1990). Since the audience for this was expected to have limited system dynamics experience, the interface needed to be designed to allow access to a wide variety of information that the user may not have acquired from other reading or experience. This was accomplished by making almost everything on the screen a "button" that would display additional information when selected. But, the simulation also had other purposes, not the least of which was showing that the structure of the system determines its behavior. This was shown by plotting the output of the simulation on top of a causal loop diagram that represented the mental model of the on-screen characters. On-screen characters had been created with each character having an opinion about the best approach to managing the system.

Marketing Flight Simulator™

Flashy graphics and arcade-style sound effects distract the user from learning about the system that is simulated, but there are times when they can be used effectively. A management flight simulator, the Marketing Flight Simulator™, was created for a computer exposition for a major computer manufacturing firm. Hundreds of 19-inch color monitors were expected to be vying for audience attention at the exposition. The managers wanted their simulator to draw people in to win a competition. A system dynamics model was created of the manager's problems and an arcade-style interface was developed complete with alert screens, warning flashers, sound effects, and color graphics. A cockpit metaphor was chosen to help bring order to the collage of colors and sounds.

Sound effects are an excellent way of conveying information quickly and efficiently. It is also a quick way to aggravate your user if the effects are too repetitive. As with explanatory screens, it would be wise to include a set-up screen that allows the user to turn off the sound effects. Since this means that the information conveyed by the sound will no longer be available, it is a good idea to have sound complement other methods of communication rather than being the chief carrier of that information.

A Management Flight Simulator of the Beer Game

The microcomputer version of the Beer Game, developed by Gould-Kreutzer Associates, Inc. and Innovation Associates, Inc. was designed to allow the user to experience playing the board game version of the Beer Game as well as to allow them to explore the dynamics of the system that controls the behavior of the game. This simulator is particularly relevant for individuals who have already played the Beer Game and would like to challenge their understanding of the system, by playing the computer version. The microcomputer version of the Beer Game includes: the ability to set the decision rules of each position, two modes of play - simulation and gaming, the ability to set the customer order stream, and graphical and tabular display of the output.

Executive-information Systems

While the management flight simulator may be an ideal tool for management training, the model-based executive-information system is an ideal tool for management operations. Model-based executive-information systems (EIS) differ from management flight Simulators in both depth and detail. While management flight simulators usually contain fairly dynamically sophisticated models, they are often very highly aggregated and lack specific details necessary at the operational level. In the model-based EIS, however, the model contains data to closely match the real world. For example, the People Express management flight simulator represents all the aircraft for all the routes of all the competing companies as one aggregate stock. A People Express model-based EIS would represent aircraft by specific airline, routing, and type, thus allowing managers to make real-world decisions with the model. If an airline manager must choose whether to purchase a contract of ten new Boeing 767's for service in the year 1993, seven MD11's in the year 1995, or pick-up used aircraft of a variety of types over a two-year period, she needs a tool that can specifically input exactly what the options are and immediately see their implications. While all three options might have the same effect in a management flight simulator, on an operational level they might have extremely different effects on costs, routing, and reliability.

Because of the high-degree of customization necessary to produce an operational model-based EIS, they tend to be somewhat more costly than management flight simulators.

Combining model development, graphical user interface development, and data handling together with end-user interviews and design sessions into a complete model-based EIS package can take a firm three to nine months to design, build, install, and calibrate.

Executive-Strategy Systems

An executive-strategy system consists of three parts: a networked database, a graphic interface, and a computer model that captures the relevant dynamic issues of a firm and its competitive environment. The development process starts with an existing model of the company. Adding on an interface and data-reading capabilities a model is created with accurate information and an easy-to-use format. Like the executive-information system, executive-strategy systems allow users to retrieve, view, and analyze data. But the more sophisticated simulation capabilities give the added ability to test alternative strategies, plot scenarios, and simulate possible future trends.

Examples of Executive-Information and Executive Strategy Systems

The following simulators are examples of systems designed for managers who are attempting to make decisions about the management of their organizations.

Sharebuilder™

Share Builder™, designed by Gould-Kreutzer Associates, Inc. and MicroWorlds, Inc. is a powerful new tool for corporate strategy development for product industries. It examines pricing, research and development, marketing, customer service, capacity expansion, and cost reduction issues for companies competing in a market with a specific, similar product line. With this tool, the results of various long-term strategies can be rapidly played-out over time to allow managers and planners to improve their strategic thinking and fully explore all options and scenarios before committing to a particular future path. The underlying model is dynamically sophisticated, and capable of displaying multiple modes of market growth, product life cycles, competitive dynamics, and long-term, short-term trade-offs in corporate policy. All of these complex, real-world dynamics are consistently tracked and displayed through a full-color graphical user interface. This interface makes the model very accessible and easy to use and understand. Once custom-calibrated for a particular product line, Share Builder™ can add tremendous value to that company's planning process.

Chemicals Strategy System™

The Chemicals Strategy System™ by MicroStrategy, Inc. is an example of a model-based executive-information system (EIS). Very simply, it tracks historic and forecasted supply, demand, and price data for a specific global chemical market. The model underneath is arrayed by specific company, producer sight, world region, and end-use market segment. This high level of disaggregation allows specific real-world information to be easily added to the dynamic model. The model can then accurately provide global capacities, demands, and regional economics, generate future pricing, margin, and return on investment information on a global scale. By using known data, the model can then derive the unknown data with integrity and internal consistency. For example, if it was announced that a major competitor in Europe had a terrible explosion at their largest chemical production facility last Tuesday, making it dysfunctional for 18 months, the Chemicals Strategy System™ could immediately produce the global response including new pricing information, opening of new plants or round-out of existing facilities, and how market share and profitability might be effected for each region of the world.

For operations managers of a global chemical (or any other commodity), this model-based EIS could prove to be extremely valuable for the user, but extremely disadvantageous to the competition who does not have the EIS.

Global Petroleum Simulator™

The Global Petroleum Simulator™ is currently being constructed by Gould-Kreutzer Associates, Inc. This model-based EIS combines the familiarity of the traditional database and spreadsheet with the power of computer simulation and a sophisticated graphical user interface. This system's main advantage is its ease of use. By opening the global upstream database the user finds a full-color world map on the computer display. "Drill-down" through each world region, country, and specific oil field with a series of "point and click" commands and the user obtains detailed reports from the database. The heavy use of graphics and charts readily turns the dry data into living information for useful analysis. When combined with downstream, refinery, and end use databases, this EIS should be the state-of-the-art in data presentation and analysis for the petroleum industry. A global petroleum industry model will round-out the Global Petroleum Simulator™ into probably one of the most advanced model-based EIS of its kind.

Conclusions

Can interfaces make simulation models more powerful tools for learning? Perhaps, but such learning improvements do not happen automatically by adding more technical functionality or colorful graphics. Without a careful consideration of how the design of the software interface serves the purpose of the user, such "improvements" can in fact be detrimental. If the look and feel of the software as well as the technical functionality are designed to support the learning process the impact can be powerful.

Some considerations to keep in mind are the following:

1. Do not use a flashy interface to hide a mediocre model.

An interface is only as good as the model it controls. Since the value of an interface is dependent on the quality of the model, no matter how flashy an interface is, it is worthless if the model it controls is poor.

2. Remember the purpose of the interface.

A critical element of all systems thinking related software seems to be making assumptions and mental models visible. Visibility enhances testability, contributes to group participation, and provides a record for later examination. Technologically, it is now possible to take visibility to new extremes. There is a very real temptation when designing interfaces to become so engrossed in the arcade-style of the interface that flashy graphics and sound effects become an end in themselves and begin to interfere with the intended purpose of the interface. Again, it is vital that everything in the interface supports the stated purpose of that interface.

With these caveats in mind however, we have seen several examples in this paper where flashy sound effects, extended graphic metaphor, and technologically sophisticated add-ons, not only contributed to the functionality of the interface, but were critical elements to the success of the software.

References

Apple, Inc. 1987. *Human Interface Guidelines: The Apple Desktop Interface*. Reading: Addison Wesley Publishing Company.

Bakken, B., Gould, J., and Kim. D. 1992. Experimentation in learning organizations: a management flight simulator approach. *European Journal of Operational Research* 59(1).

Barney, G., Kreutzer, W. B., and Garrett, M. J. 1991. *Managing A Nation: The Microcomputer Software Catalog* 2nd Edition. Boulder, CO: Westview Press.

Diehl, E. W. 1992. Participatory simulation software for managers: the design philosophy behind MicroWorld Creator. *European Journal of Operational Research* 59(1).

Kemeny, J. M. and Kreutzer, W. B. 1992. An archetype based management team flight simulator. *1992 System Dynamics Conference Proceedings*. The Netherlands: Utrecht University.

Kim, D. 1989. Learning laboratories: designing reflective learning environments. in *Computer Based Management of Complex Systems*, (editors P. M. Milling and E. O. K. Zahn). Berlin: Springer-Verlag.

Kreutzer, W. B. 1993. How to choose systems thinking software. *System Thinker* 3(10). Cambridge, MA: Pegasus Communications.

Kreutzer, W. B. 1993a. A system dynamics approach to executive information systems. *System Thinker* 3(7). Cambridge, MA: Pegasus Communications.

Kreutzer, W. B. 1990. Ithink™: the visual thinking tool. *The Systems Thinker* 1(7). Cambridge, MA: Pegasus Communications.

Kreutzer, W. B. 1990a. Let the games begin. *The System Thinker* 1(6). Cambridge, MA: Pegasus Communications.

Kreutzer, W. B. 1988. Software to run a country with. *Whole Earth Review* No. 9.

Kreutzer, W. B., Kreutzer, D. P., and Gould, J. M. 1992. The quahog oil production simulator: a case study of the rapid development of a management flight simulator for training. *1992 System Dynamics Conference Proceedings*. The Netherlands: Utrecht University.

Meadows, D. 1989. Gaming to implement system dynamics models. In P.M. Milling, & E.K. Zahn (Ed.), *Computer-based management of complex systems* . Berlin: Springer-Verlag.

Nelson, T. H. 1990. The right way to think about software design. In B. Laurel (Ed.) *The Art of Human-Computer Interface Design*. Reading, MA: Addison-Wesley.

Senge, P. M. 1990. *The Fifth Discipline: The Art & Practice of the Learning Organization.* New York: Doubleday.

Senge, P.M., & Lannon, C. 1990. Managerial microworlds. *Technology Review* 93(5).

Sterman J. D. 1988. *People Express Management Flight Simulator*, software and documentation available from author, Sloan School of Management, MIT, E52-562, Cambridge, MA 02139, USA