Applying the Principles of Human Computer Interaction to the Design of Management Flight Simulators

W Brian Kreutzer
Gould-Kreutzer Associates, Inc.
River Court, 10 Rogers Street
Cambridge, MA 02142, USA
Tel: 617 577 1430
Fax. 617 577 0426

David P Kreutzer
Massachusetts Institute of Technology
System Dynamics Group, E40-294
Cambridge, MA 02139, USA
Tel: 617 253 1955

Abstract

Engaging managers in facilitated team model building and using all the principles of system dynamics to obtain high quality models are useful prerequisites to creating a good management flight simulator. No amount of interface improvement should substitute for or mask a poorly designed or understood model. However, there is a great deal of learning going on in the field of Human Computer Interaction that we in the system dynamics community can learn from in order to improve the effectiveness of our management flight simulators and model based learning workshops. This paper will explore how we can apply the principles of Human Computer Interaction to building managment flight simulators.

Applying the Principles of Human Computer Interaction to the Design of Management Flight Simulators

Introduction

> "Thinking about interfaces is thinking too small. Designing human-computer experience isn't about building a better desktop. It's about creating imaginary worlds that have a special relationship to reality--worlds in which we can extend, amplify, and enrich our own capacities to think, feel, and act." (Laurel, 1991)

The field of Human Computer Interaction (HCI) encompasses many different disciplines that have relevance to MFS designers. Designers might benefit from studying art, music, psychology, education, and software development to help them create interfaces that allow the user to learn unimpeded by the interface itself. This is especially true for system dynamics based management flight simulators (MFS) where the focus is to design transparent box simulators (Machuca, 1993). These transparent box simulators allow the user to see and experience the underlying structure of the system as well as to enter input and view output. This makes it important for the interface to be designed in such a way as to make it easy for the user to explore the underpinnings of the simulator.

This paper will describe how applying principles from the field of Human Computer Interaction can help make it easier for the user to buy-in to the MFS. It will also explore the required contents of the audience description, how cognitive psychology can be used to help direct the design process, how story boards can help manage the complexity of the design process, the guidelines available to help design the experience, and how to evaluate the completed design.

Purpose

The first thing to consider when designing the interface is the purpose of the activity. How do you want the user to be changed by the experience? What tasks are going to performed by the user? Should the user walk away with a new understanding of the system? Is the user practicing decision making skills? Before designing the experience, it is critical to know what user reaction the experience is designed to create.

Audience

> "In a sense, the presence of a computer is only incidental to the design; human needs and abilities are the guiding force." (Shneiderman, 1992)

The next step is to learn as much about the intended audience as possible, including psychological, socio-cultural, and knowledge characteristics (Preece, 1993). It helps a great deal if the audience has participated in the construction of the underlying model with some kind of facilitated team model building.

The audience's **psychological profile** should include: the **motivation** of the audience (their reason for being in the training seminar), the needed **prerequisites** (the skill, knowledge, and educational level necessary to use the interface), and the **attitude** your audience has towards computers and simulation (Mallory 1987 & Cox 1993).

The **socio-cultural** profile should include a description of how members of the audience normally make their decisions--individually or in groups--as well as the specific cultural background (different cultures have different reactions to color and icon images).

The **knowledge profile** should describe your audience's educational attainment and work experience. This will indicate how much introductory material will be needed in interface. If the audience consists of graduate level petroleum engineers, descriptions of secondary recovery techniques may not be necessary, whereas, if the audience is high school seniors who are thinking of becoming petroleum engineers, introductory descriptions would be essential.

The audience description created by the three profiles above will be used to aid in choosing the metaphor, sound, and icons that will be used in the interface. By selecting them based on the audience description it's possible that the audience will buy-in to the simulator and learn from it.

Psychology

> "Therefore, let me argue that the actual dawn of user interface design first happened when computer designers finally noticed; not just that end users had functioning minds, but that a better understanding of how those minds worked would completely shift the paradigm of interaction." (Kay 1990)

Cognitive psychology is the branch of psychology that studies how the mind processes information. MFS designers can use cognitive psychology to make sure that the information processing required by the interface is within the users' mental capabilities (Preece, 1993). By studying cognitive psychology, interface designers can create interfaces to be consistent with the way users' think and learn.

Cognitive psychologists tell us that we learn in a number of ways: through active thinking, through doing, through analyzing errors, and through analogy to name just a few (Preece, 1993). When users are confronted with a system (via an interface) they attempt to make sense of the behavior produced by that system through the creation of a cognitive or mental models (**learning through active thinking**). They explore the system further by varying the inputs and examining the outputs of the computer model, and adjust their mental models to explain that behavior (**learning through doing**). During this process, however, they occasionally create flawed mental models. By continuing to experiment with the MFS, they can adjust their mental models for a closer fit with the real world system the models represents (**learning through analyzing our errors**). The designers can guide this learning process by selecting good metaphors (**learning through analogy**). A metaphor not only guides the experience, but, if chosen properly, also ties into the user's previous conceptual model, helping the user buy-in to the process.

Tools

> "...well designed artifacts tell people what functions they perform and how they perform them..." (Rhienfrank, 1992)

According to Kevin Cox (1993) a good tool has the following characteristics; user control, transparency, flexibility, and learnability. These characteristics not only apply to the interface but also to the tools used to produce them.

The **user** should be in **control**, the actor, not the acted upon. This means that the user should be in charge of as much of the interface experience as your purpose allows. As much as possible the interface should be **transparent** to the user. It should not hinder the users from doing the things their conceptual models say they should be able to do. This is one of the reasons it is so important to have a very clear description of the audience and why their conceptual models must be clearly identified before designing the MFS. If the interface designer doesn't understand the conceptual models of the users, there is no way to make the MFS transparent. However, the conceptual model described in the audience description is an average of all your users. The design of the MFS must be **flexible** to acknowledge the conceptual models of all users. Finally, the interface should be simple enough so that it is easy for the user to **learn** it. A good test for learnability is--if you have to explain it, change it.

Designing a MFS so that it contains all of the characteristics listed above is a complex task. To make this task even more complex, it must be remembered that a MFS is not a collection of static screens but an interactive experience. In order to manage this complex task it is helpful to use a design tool created by film makers who are also concerned with the design of experiences. Movie and cartoon makers use story boards to help them handle the complexity of producing films (Laybourne 1979). Story boards consist of a series of rough 8x10 sketches that represent each of the major scenes in the movie or cartoon. They are then posted on the wall in the order the audience will experience them in the completed film. The same can be done with interface design. Draw pictures of the major activities the user will encounter while using the software and then post them on the wall in the order

they will be experienced. This will serve as a guide to the design of the flow of the management flight simulator. This helps ensure that the design will be of the experience, concentrating on action rather than on static screens.

The design of the simulator should be separated from the implementation of the simulator (Rubinstien, 1984). This is important. If the interface is designed during the implementation, if the current design doesn't work out, both design and programming time have been wasted. Another benefit of designing the interface first is that the documentation can begin as part of the design. The documentation can be created on a parallel track. Obviously at this point one can't do screen shots and describe the interface in detail, but the designer can get a head start on the general structure and make up of the documentation.

Guidelines

> "Making software is like making movies because both are about how moving presentations affect the mind and feelings of the viewer. The talented software designer subtly calculates the overall structure and how it will affect the viewer or user; not merely putting parts together. Technical concerns are merely preliminary, the substratum; what counts are the artistic planning, execution, and the reunified tuning of all the parts." (Nelson, 1990)

There have been a number of "guidelines" for designing user interfaces published over the years. The following are useful to the design of system dynamics based MFS: learnability, user confidence, information feedback, color, sound, and display.

1) Learnability

The MFS should be **easy to learn and use**. This can be facilitated by designing the external myth of the program to be in tune with the conceptual model (mental model) of the user (Rubinstein, 1984). The external myth can be defined as the presentation and the impression the program gives to the user as opposed to the underlying code. This impression is created by the combination of metaphor, color, sound and many other methods. If the external myth of the program is consistent with the mental model of the user then it is possible that it will be easier for the user to suspend disbelief and enthusiastically explore the experience.

> "...a system dynamics model, if it is to be effective, must communicate with and modify the prior mental models. Only people's beliefs, that is, their mental models, will determine action. Computer models must relate to and improve mental models if the computer models are to fill an effective role." (Forrester, 1992)

There are several other ways to make the interface enhance the learnability, they are:

a) Make sure that the **metaphors** used in the interface are not only consistent with the external myth but also with the real world. Metaphors should be chosen from the real world with care (Erickson, 1990). Their purpose is to relate what users are doing in the interface with a similar task in the real world so it will be easier to learn and use that task in the interface.

b) Make sure that the commands and the method of input and output are kept **simple.** This can be accomplished by moving the interaction away from the keyboard, using the mouse as much as possible. This emphasizes seeing and pointing rather than remembering and typing (Apple, 1987). Ideally the user should never have to use the keyboard at all, they should only use the mouse. Input can be done through buttons and slide bars; navigation and feature selection can be done through clicking on icons.

c) Make sure that communication from the computer follows the rules of human conversation (Rubinstein, 1984). We shouldn't have to learn a new language to understand what the computer is trying to communicate. Feedback should be in the vernacular and should be simple and to the point.

2) User confidence

There are many ways in which user confidence can be affected. Naturally if the interface is flawed the user's confidence will be eroded. However, even in systems that are virtually "bug" free, user confidence can be undermined by impressions of **instability** and **inconsistencies** (Apple, 1987). One way to overcome these impressions is to aim for consistency. Similar activities should be done in similar ways and identical activities should be done in exactly the same way.

3) Information feedback

Physicists tell us that **for every action there is an opposite and equal reaction**. This should also be true of interfaces. Anytime the user does something the interface should respond in such a way that the user knows action has been received. This can be as simple as temporarily highlighting buttons to show they've been pressed to showing "Please Wait" screens when the action requested cannot be done immediately.

At the same time any action the user takes should be easily reversible (Cox, 1993). A destroy the universe button should warrant at least one "Are you sure?" query and most definitely have an "Undo" menu option.

4) Color

Color can be used to do a number of things: "Soothe or strike to the eye, add accents to uninteresting displays, facilitate subtle discriminations in complex displays, emphasize the logical organization of information, draw attention to warnings, and evoke strong emotional reactions of joy excitement, fear, or anger." (Shneiderman, 1992).

While color can certainly spice up an interface and carry additional information, if not chosen properly it can also can affect the user's MFS experience with unintended consequences. We respond to color in a number of ways: biologically, visually, emotionally, aesthetically, and psychically (Birren 1978). Colors should be chosen carefully so as to generate the desired reaction.

Before actually selecting color it should be remembered that approximately 8% of the male population and .5% of the female population is color blind, and that as people age they become less sensitive to color (Salomon, 1990). The audience description should be studied carefully before selecting what, if any, colors to use.

Some designers recommend that if color is used it should be limited to four colors (Cox, 1993, Shneiderman, 1992, and Chijiiwa, 1987) and the color should be used to add meaning, being sure to be consistent with the meaning. Be aware that there are certain combinations of color which the human eye cannot focus on simultaneously, because their wavelengths are so different (red and blue for instance). When these colors are in juxtaposition they produce a distracting shimmer (Kosslyn, 1994).

5) Sound

Another tool for communicating information is sound. Sounds, however, can also be annoying and distracting. Great care must be given to planning its use, and even after careful planning the user should always be given the ability to turn the sound off. This means that sound cannot be the primary carrier of information. It should only be used to emphasize and draw attention to information communicated in other ways.

Sound can also be used to communicate information, to confirm that something has been done (e.g. the simulation is done or the button has been pressed), and to aid navigation (Cox, 1993).

Interface designers have divided sounds into several different types. The two main types are auditory icons and earcons. Auditory icons, like their visual counterparts are an auditory representation of what is being done. A monkey screeching would be an auditory icon equivalent to a picture of a monkey. Earcons on the other hand are "composed of short motives, which are short, rhythmic sequences of pitches with variable intensity, timbre and register" (Brewster, 1993). As such, they convey only the meaning you tell the user they convey. The chime you receive which tells you

that you have electronic mail would be an example of an earcon. With earcons especially, consistency of use is a must.

6) Display

There are a number of things to keep in mind when designing the displays of the MFS, whether they're animation, output, input, debriefing, or briefing screens.

The first thing to remember is not to overwhelm the user with visual displays (Rubinstein, 1984). Trying to crowd too much information onto one screen, may save hard disk space but that savings will be more than offset with the confusion and frustration it causes the user. Displays should be simple, clear, and uncluttered.

The designer should also attempt to incorporate the **user's representation of data** (Rubinstein, 1984). Not only will this be familiar, which will reduce the need for explanation and help screens, but also it is another way of tying the MFS external myth to the user's conceptual model.

Finally the designer should **respect the rules of good paper presentation** (Rubinstein 1984). This includes choices and use of color, compositional elements, icons etc.

Evaluation

> "It is necessary to have a testing methodology that is closely integrated with the development process and that brings discipline to the manner in which the what-to-test, when-to-stop, and who-does-the-work questions are answered. Testing requirements at each step of development have to be defined and, in effect, a "testing life cycle" has to exist side-by-side with the development life cycle." (Hetzel, 1984)

There are a number of usability testing issues that should be kept in mind throughout the interface design process: learnability, throughput, flexibility, and attitude (Shackel, 1990).

The **learnability** of an interface is tested by determining whether or not the interface makes it easy for the user to learn how to use it. The interface itself should be transparent to the task the user wishes to accomplish. If the interface gets in the way of this task by requiring that the user continually refer to the user guide, it has failed the learnability test.

Throughput tests how quickly users are able to accomplish the tasks the designers have set before them. Does the interface hinder their accomplishment?

Flexibility is concerned with the life cycle of the MFS. As a user gains more experience with the MFS does the interface grow with them? That is, are there shortcuts for experienced users? Are there ways of turning off the hand holding that is essential for beginners?

What type of **attitude** do users have when using the interface? Is there a lot of cursing going on or criticism of the underlying assumptions of the model when the simulator is being used?

Norman describes three areas of general design which can be applied to the design of interfaces; visibility, mapping, and feedback (Norman, 1988).

The main questions to ask to ensure **visibility** are: Are all the relevant parts of the process visible? Does the user have to go hunting or work hard to find the controls to do the things that need to be done? If so, the system needs to be redesigned so that the controls are easy to find.

Mapping describes the process of linking the controls to the functions. For instance, buttons that take the user to the next screen traditionally point to the right. This is mainly because a book is read by going forward moving from the left to the right. If a reader needs to return to the previously read pages he or she goes back, moving to the left. That being the case if the "Next Page" buttons of our interface point to the left, it confuses and frustrates the user. This is another area where it is important to know the user. The above example only works in western cultures where we read books

from front to back. Some cultures read their books from back to front and would find it confusing and frustrating for the next page button to be in pointing to the right.

Anything a user does should have some visible **feedback**. If a button is pressed, at the very least, it should be momentarily hi-lighted. This is especially important if there are any delays in the system. If there are, and there is no feedback to the user, the designer can be sure that the button will be pressed a number of times before the process is completed. If the programmers didn't foresee this eventuality the process will be repeated the number of times the button was accidentally pressed. This can be very annoying and frustrating to the user.

Conclusion

User buy-in is crucial for success in the development of management flight simulators. One way to achieve this is to design your MFS so that its external myth corresponds to the users' mental models. In order to do this you need to identify the users' mental models and design the interface's external myth accordingly. This can be done by creating a profile of the user and using it to direct your design throughout each step of the entire design process.

Bibliography

Apple, (1987). *Human Interface Guidelines, The Apple Desktop Interface*, Reading, MA, Addison-Wesley.

Birren, Faber, (1978). *Color and Human Response*, New York, Van Nostrand Reinhold.

Brewster, S., Wright, P., & Edwards, A. (1993). "An Evaluation of Earcons for Use in Auditory Human-Computer Interfaces," in Ashlund, S. et al (eds.) *Human Factors in Computing Systems*, Amsterdam, The Association for Computer Machinery.

Chijiiwa, Hideaki, (1987). *Color Harmony: A Guide to Creative Color Combinations*, Rockport, MA, Rockport Publishers.

Cox, Keven and Walker, David, (1990). *User Interface Design (2nd Edition)*, New York, Prentice Hall.

Erickson, Thomas, (1990). "Working with Interface Metaphors," pages 65-73 in Laurel, Brenda (ed), *The Art of Human-Computer Interface Design*, Reading, MA, Addison-Wesley.

Forrester, Jay (1992). "System Dynamics and the Lessons of 35 Years," in Greene, Kenyon (1993) *A Systems-Based Approach to Policymaking*. Boston, MA, Kluwer Academic Publishers.

Heckel, P. (1984). *The Elements of Friendly Software Design,* New York, Warner Books.

Hetzel, Bill., (1984). *The Complete Guide to Software Testing*, Wellesley, MA, QED Information Sciences.

Kay, Alan (1990). "User Interface: A Personal View," pages 191-207 in Laurel, Brenda (ed), (1990). *The Art of Human-Computer Interface Design*, Reading, MA, Addison-Wesley.

Kossleyn, Stephen K. (1994). *Elements of Graph Design.* New York, W. H. Freeman and Company.

Laurel, Brenda, (1991). *Computers as Theatre,* Reading, MA, Addison-Wesley.

Laybourne, Kit, (1979). *The Animation Book: A Complete Guide to Animated Filmmaking from Flip-Books to Sound Cartoons*. New York, Crown Publishers, Inc.

Machuca, J., Machuca, M., Ruiz, A., Ruiz, J., (1993). "Systems Thinking Learning for Management Education. What are Our Ideas and How are We Going about It in Sevilla?" in Zepeda et al (ed), *Proceedings of the International System Dynamics Conference*, Cancun (Mexico).

Mallory, W. J. (1987). "Technical Skills," in Craig, Robert (ed.) *Training and Development Handbook: A Guide to Human Resource Development* (Third Edition), New York, McGraw-Hill Book Company.

Nelson, Theodore (1990). "The Right Way to Think About Software Design." in Laurel, Brenda (ed), (1990). *The Art of Human-Computer Interface Design*, Reading, MA, Addison-Wesley.

Norman, Donald, (1988). *The Design of Everyday Things*, New York, DoubleDay Currency.

Preece, Jenny, (1993). *A Guide to Usability: Human Factors in Computing*, Reading, MA, Addison-Wesley.

Rheinfrank, John, Hartman, William, & Wasserman, Arnold (1992) "Design for Usability: Crafting a Strategy for the Design of a New Generation of Xerox Copiers," in Adler, Paul and Winograd, Terry. *Usability: Turning Technologies into Tools*, New York, Oxford University Press.

Rubinstein, Richard and Hersh, Harry, (1984). *The Human Factor: Designing Computer Systems for People*, Bedford, MA, Digital Press

Russo, Patricia, and Boor, Stephen, (1993). "How Fluent is your Interface? Desiging for International Users." in *Human Factors in Computing Systems: INTERCHI'93 Conference Proceedings.* Reading, MA, Addison-Wesley.

Salomon, Gitta, (1990). "New Uses for Color," pages 269-278 in Laurel, Brenda (ed), *The Art of Human-Computer Interface Design*, Reading, MA, Addison-Wesley.

Shackel, B. (1990). "Human Factors and Usability," pages 27-41 in Preece et al (1990) (eds) *Human-Computer Interaction: Selected Readings*, London, Prentice Hall.

Shneiderman, Ben, (1992). *Designing the User Interface: Strategies for Effective Human-Computer Interaction (2nd Edition)*, Reading, MA, Addison-Wesley.