UNIVERSITAT POMPEU FABRA

Brenda Laurel
Computers as Theater
Addison-Wesley, 1993, pp. 14-22

"Interactivity and Human Action"

The World's a Stage

For purposes of comparison, let's take a look at the theatre. We have observed that the theatre bears some similarities to interface design in that both deal with the representation of action. Drama, unlike novels or other forms of literature, incorporates the notion of performance; that is, plays are meant to be acted out. A parallel can be seen in interface design. In his book *The Elements of Friendly Software Design* [1982], Paul Heckel remarked, "When I design a product, I think of my program as giving a performance for its user." In the theatre, enactment typically occurs in a performance area called a stage. The stage is populated by one or more actors who portray characters. They perform actions in the physical context provided by the scene and light designers. The performance is typically viewed by a group of observers called an audience.

Part of the technical "magic" that supports the performance is embodied in the scenery and objects on the stage (windows that open and close; teacups that break); the rest happens in the backstage and "wing" areas (where scenery is supported, curtains are opened and closed, and sound effects are produced), the "loft" area above the stage, which accommodates lighting instruments and backdrops or set pieces that can be raised and lowered, and the lighting booth, which is usually above the audience at the back of the auditorium. The magic is created by both people and machines, but who, what, and where they are do not matter to the audience.

It's not just that the technical underpinnings of theatrical performance are unimportant to audience members; when a play is "working, audience members are simply not member aware who of the technical aspects at all. For the audience engaged by and involved in the play, the action on the state is *all there is*. In this sense, plays are like movies: When you are engrossed in one, you forget about the projector, and you may even lose awareness of your own body.

For the actor on stage, the experience is similar in that everything extraneous to the ongoing action is tuned out, with the exception of the audience's audible and visible responses, which are often used by the actors to tweak their performance in real time (this, by the way, reminds us that theatrical audiences are not strictly "passive" and may he said to influence the action). For actor and audience alike, the ultimate "reality" is what is happening in the imaginary world on the stage-the representation.

As researchers grapple with the notion of interaction in the world of computing, they sometimes Compare computer users to theatrical audiences. "Users," the argument goes, are like audience members who are able to have a greater influence on the unfolding action than simply the fine-tuning provided by

conventional audience response. In fact, I used this analogy in my dissertation in an attempt to create a model for interactive fantasy. The users of such a system, I argued, are like audience members who can march up unto the stage and become various characters, altering the action by what they say and do in their roles.
Let's reconsider for a minute. What would it be really like if the audience marched up on the stage? They wouldn't know the script, for starters, and there would be a lot of awkward fumbling for context. Their clothes and skin would look funny under the lights. A state of panic would seize the actors as they attempted to improvise action that could incorporate the interlopers and still yield something that had any dramatic integrity. Or perhaps it would degenerate into a freefor-all, as performances of avant-garde interactive plays in the 1960s often did.

The problem with the audience-as-active-participant idea is that it adds to the clutter, both psychological and physical. The transformation needs to be subtractive rather than additive. People who are participating in the representation aren't audience members anymore. It's not that the audience joins the actors on the stage; it's that they become actors-and the notion of "passive" observers disappears.

In a theatrical view of human-computer activity, the stage is a virtual world. It is populated by agents, both human and computer-generated, and other elements of the representational context (windows, teacups, desktops, or what-haveyou). The technical magic that supports the representation, as in the theatre, is behind the scenes. Whether the magic is created by hardware, software, or wetware is of no consequence; its only value is in what it produces on the "stage." In other words, the representation is all there is. Think of it as existential WYSIWYG.[1]

Theatre as an Interface Metaphor

The idea of human-computer activity suggests a number of interesting corollaries. Since all action is confined to the world of the representation, all agents are situated in the same context, have access to the same objects, and speak the same language. Participants learn what language to speak by noticing what is understood; they learn what objects are and what they do by playing around with them. A good example of this approach is a system called Programming by Rehearsal, developed by Laura Gould and William Finzer at Xerox PARC in 1983 and 1984. The system is a visual programming environment based on a dramatic metaphor. There are some problems with the application of the metaphor per se but the principle of "the representation is all there is" is with powerful results:

> *Two significant obstacles to learning a programming language are mastering the language's syntax and learning the vocabulary. In the Rehearsal World, the designers rarely have to know either the syntax or the vocabulary as most writing of code is done by watching. [Finzer and Fitzer,1984]*

[1] WYSIWYG stands for "what you see is what you get," coined by Warren Teitelman at Xerox PARC. It has been held up as a paradigm for direct manipulation interfaces, but some theorists have contested its value (see, for instance, Ted Nelson's article, "The Right Way to Think About Software Design" in *The Art of Human-Computer Interface Design*.)

A more recent attempt to employ a theatrical metaphor for an authoring system is Ellis Horowitz's SCriptWriter system, developed at the University of Southern California in 1987 and 1988 [Horowitz, 1988]. Horowitz's system further illustrates the distinction between using theatre as an interface metaphor and using it in the deeper way that this book advocates-as a fundamental understanding of what is going on in human-computer interaction.

As a metaphor, Horowitz's system successfully employs notions like "director" (as the code of a program generated by his system) and "rehearsal" (in the same way that Gould's system employs the notion of programming by rehearsal). But Horowitz's interface falls off the edge of its own metaphor in several ways. Programming actions like "cast" and "rehearse" are intermixed with traditional computerese terms like "edit," "list," and "print," failing on the level of consistency. The most disturbing inconsistency is the notion of treating a screen as a "player." His player concatenates the notions of stage, scenery, actors, and dialogue in a concept where the locus of agency is so dispersed as to be invisible. Furthermore, the notion of human agency-the other kind of "player" that may act upon a "stage"-is absent in Horowitz's conceptualization. The system does not support a notion of action that integrates human agency into the whole but rather leaves this aspect of design entirely up to the author.

Interactivity and Human Action

The idea of enabling humans to take action in representational worlds is the powerful component of the programming by rehearsal approach. It is also what is missing in most attempts to use theatre simply as an interface metaphor. A central goal of this book is to suggest ways in which we can use a notion of theatre, not simply as a metaphor but as a way to conceptualize human-computer interaction itself.

Focusing on human agency allows us to simplify another consistently troublesome concept, the notion of "interactivity." People in the computer game business have been arguing about it for over a decade. In 1988, Alexander Associates sponsored INtertainment, the first annual conference bringing together people from all corners of the interactive entertainment business. People came from such diverse industries as personal computers, video games, broadcast and cable television, optical media, museums, and amusement parks. Over the course of the two days, a debate about the meaning of the word "interactive" raged through every session, disrupting carefully planned panels and presentations. People seemed to regard "interactivity" as the unique cultural discovery of the electronic age, and they demanded a coherent definition. Several speakers tried to oblige, but no one succeeded in presenting a definition that achieved general acceptance. Many participants departed angry and dissatisfied. Could it be the "wrong tree" problem again?

[2] Zeno's paradox (called the theory of limits in mathematics) says that you can never get from here to there because you can only get halfway, halfway of halfway, etc. Mathematics offers a solution; so does common sense. But the paradox compelling enough to have interested ogicians and mathematicians for centuries.

In the past, I've barked up that same tree. I posited that interactivity exists on a continuum that could be characterized by three variables: frequency (how often you could interact), range (how many choices were available), and significance (how much the choices really affected matters) [Laurel, 1986a and b]. A not-so-interactive computer game judged by these standards would let you do something only once in a while, would give you only a few things to choose from, and the things you could choose wouldn't make much difference to the whole action. A very interactive computer game (or desktop or flight simulator) would let you do something that really mattered at any time, and it could be anything you could think of-just like real life.

Now I believe that these variables provide only part of the picture. There is another, more rudimentary measure of interactivity: You either feel yourself to be participating in the ongoing action of the representation or you don't. Successful orchestration of the variables of frequency, range, and significance can help to create this feeling, but it can also arise from other sources-for instance, sensory immersion and the tight coupling of kinesthetic input and visual response. If a representation of the surface of the moon lets you walk around and look at things, then it probably feels extremely interactive, whether your virtual excursion has any consequences or not. It enables you to act within a representation that is important. Optimizing frequency and range and significance in human choice-making will remain inadequate as long as we conceive of the human as sitting on the other side of some barrier, poking at the representation with a joystick or a mouse or a virtual hand. You can demonstrate Zeno's paradox on the user's side of the barrier until you're blue in the face, but it's only when you traverse it that things get real.[2]

The experience of interactivity is a thresholdy phenomenon, and it is also highly context-dependent. The search for a definition of interactivity diverts our attention from the real issue: How can people participate as agents within representational contexts? Actors know a lot about that, and so do children playing make-believe. Buried within us in our deepest playful instincts, and surrounding us in the cultural conventions of theatre, film, and narrative, are the most profound and intimate sources of knowledge about interactive representations. A central task is to bring those resources to the fore and to begin to use them in the design of interactive systems.

So now we have at least two reasons to consider theatre as a promising foundation for thinking about and designing human-computer experiences. First, there is significant overlap in the fundamental objective of the two domains-that is, representing action with multiple agents. Second, theatre suggests the basis for a model of human-computer of this book familiar, comprehensible, and evocative. The rest of this book will explore some of the theoretical and practical aspects of theatre that can be directly applied to the task of human-computer experiences. But there are a few More stones to be turned in arranging the groundwork for this discussion.