
Computers
as
Theatre
Brenda
Laurel
now
featuring
Post-Virtual
Reality

Foreword

> Movies did not flourish until the engineers lost control to artists—or more precisely, to the communications craftsmen. The same thing is happening now with personal computers.[1]

> Think of the computer, not as a tool, but as a medium.[2]

This is a challenging book, one that offers an entirely new perspective upon the development of the modern technologies of computation and communication. The main theme of this multithemed book is that these technologies offer new opportunities for creative, interactive experiences and, in particular, for new forms of drama. But these new opportunities will come to pass only if control of the technology is taken from the technologist and given to those who understand human beings, human interaction, communication, pleasure, and pain.

It is time for the engineers to go back to engineering. To develop these new technologies, we need a new breed of creative individuals, most likely those associated with poetry, writing, and theatrical direction. Who, asks Laurel, who better understands human interaction than the dramatist? The dramatic arts have a tradition of several thousand years in thought, study, and experimentation with human experience and with a variety of modes of interaction. Today, in the absence of informed guidance, we stumble toward a new technological era made possible by the emerging technolo-

[1]From P. Heckel, *The Elements of Friendly Software Design.* New York: Warner Books, 1984, p. 5.

[2]From B. Laurel, *Computers as Theatre,* Chapter 5

gies of the computer, video, telephone, and high-quality sound.

Alas, the stumbling is not guided by any understanding of the nature of interaction. Instead, it is more like the tale describing the groping of those legendary blind men touching an elephant. Not only does each "explorer" have a different understanding of what kind of beast this elephant might be, but there is no interaction among them, no way in which they can make sense of their individual experiences.

Evidence of both the new technologies and of our many stumbles is all about us—from the automatic bank tellers, "programmable" ovens, video recorders, and thermostats that confuse our daily lives, to the plethora of office programs that are starting to dominate our business lives.

The key word in finding an illuminating path through the technological maze is "interaction." These new technologies all have one thing in common: They can aid our interaction with others, with knowledge, information, and experience, and even with the devices themselves. When we look toward what is known about the nature of interaction, why not turn to those who manage it best—to those from the world of drama, of the stage, of the theatre?

Today, the technology is provided by the technologists. It is, therefore, no wonder that most new devices, including computers and their software applications emphasize technology over all else. Each program or device comes with multiple "features," new complexities to learn, new functions that can be performed. Even the games sired by these new technologies emphasize technological aspects over dramatic them, adding sounds and colors, motion and spectacle all in the service of technology that amazes and astounds.

Do you enjoy the experience of using these new technologies? If not, why not? perhaps it never even occurred to you that the concepts of "enjoy" or "experience" could apply to the new devices, with perhaps the exception of the television set or the game whose functions are, after all, to entertain. But shouldn't you? After all, the purpose of most of these technologies is to assist us in doing our daily activities, per-group, working on a project from inception to completion, with peak moments of experience in the midst that determine the success or failure of the project.

We need to view each of our activities in a larger framework. Then each single device can be built in a way that makes sense within the whole framework of the day or task, so that each device can fit gracefully into its place in the total activity and add to our enhancement of the task.

Some of you who work in the field called "human-computer interaction" may be disappointed at first reading. You may miss controlled studies and specific specifications. You will not find anything you can apply to your work today, or even tomorrow. You will have to think of characters and "points of view." And some of the most advanced notions of today are challenged and replaced.

For example, the three major emphases in current modern computer system design can be summarized as follows:

- Design consistent objects and environments.
- Develop a metaphor for the task, tools, and actions and make all activities consistent with that metaphor.
- Think of the computer as a tool.

Now consider three of Laurel's "Rules of Thumb," discussed in Chapter 5:

- Focus on designing the action. The design of objects, environments, and characters is all subsidiary to this central goal.
- Interface metaphors have limited usefulness. What you gain now you may have to pay for later.
- Think of the computer not as a tool, but as a medium.

The current emphases make sense in today's context where design is for the small, for self-contained minute-by-minute activities. But Laurel has a more cohesive goal in mind. She emphasizes that which is large, the totality where activities are intertwined with one another, where each individual activity makes sense only when viewed with respect to the others, and where the whole set may be of considerable duration. Laurel is concerned with the total experience. If new technologies are to enhance that experience, they must be designed to take that larger, more global view.

This book will challenge you to think about new concepts and new approaches. The theme is to go beyond that which was possible until now. To think of human interaction as drama is to think broadly, to take a wide overview, and to emphasize the thematic aspects of our daily activities and needs. By so doing, we free ourselves from small, mundane considerations. Most importantly of all, this book forces us to take a wider, broader perspective on the nature of human activity. Technology can enrich our experience and increase our enjoyment, but only if properly conceived and properly applied.

Donald A. Norman
University of California, San Diego

Chapter One

The Nature of the Beast

Representing Action

In 1962, the first computer game was invented by some hackers at MIT. It was called *Spacewar* and it ran on a DEC PDP-1, the world's first minicomputer, connected to a CRT display. One of the game's designers explained that the game was born when a group sat around trying to figure out "what would be interesting displays" they could create for the CRT with some pattern-generating software they had developed. "We decided that probably, you could make a two-dimensional maneuvering sort of thing, and decided that naturally the obvious thing to do was spaceships." The MIT hackers weren't the only ones to invent *Spacewar*. As Alan Kay noted, "the game of *Spacewar* blossoms spontaneously wherever there is a graphics display connected to a computer" [Brand, 1974].

Why was *Spacewar* the "natural" thing to build with this new technology? Why not a pie chart or an automated kaleidoscope or a desktop? Its designers identified *action* as the key ingredient and conceived *Spacewar* as a game that could provide a good balance between thinking and doing for its players. They regarded the computer as a machine naturally suited for representing things that you could see, control, and play with. Its interesting potential lay not in its ability to perform calculations but in its capacity to *represent action in which humans could participate*.

Why don't we look at everything computers do that way? Consider the following question:

Q: What is being represented by the Macintosh interface?

1. A desktop.
2. Something that's kind of like a desktop
3. Someone doing something in an environment that's kind of like a desktop.

Number three is the only answer that comes close. The human is an indispensable ingredient of the representation, since it is only through a person's actions that all dimensions of the representation can be manifest. To put it another way, a computer-based representation without a human participant is like the sound of a tree falling in the proverbial uninhabited forest.

There are two major reasons for belaboring such a seemingly obvious point. First, it wasn't always true—and the design disciplines for applications and interfaces still bear the marks of that former time. Second, reconceptualizing what computers do as representing action with human participants suggests a design philosophy that diverges significantly from much of the contemporary thinking about interfaces.

Interface Evolution

"Interface" has become a trendy (and lucrative) concept over the last several years—a phenomenon that is largely attributable to the introduction of the Apple Macintosh. Interface design is concerned with making computer systems and applications easy to use (or at least usable) by humans. When we think of human-computer interfaces today, we are likely to visualize icons and menu bars, or perhaps command lines and blinking cursors. But it wasn't always so.

John Walker, founder and president of Autodesk, Inc., provides an illuminating account of the "generations" of user interface design [Walker, 1990]. In the beginning, says Walker, there was a one-on-one relationship between a person and a nputers do that way?

cintosh interface?

op.
ironment that's kind

hat comes close. The
of the representation,
ns that all dimensions
put it another way, a
a human participant
roverbial uninhabited

aboring such a seem-
lways true—and the
nterfaces still bear the
conceptualizing what
h human participants
ges significantly from
at interfaces.

icrative) concept over
non that is largely
ne Apple Macintosh.
ing computer systems
st usable) by humans.
terfaces today, we are
or perhaps command
always so.
nt of Autodesk, Inc.,
"generations" of user
eginning, says Walker,
tween a person and a

computer through the knobs and dials on the front of massive early machines like the ENIAC. The advent of punch cards and batch processing replaced this direct human-computer interaction with a transaction mediated by a computer operator. Time-sharing and the use of "glass teletypes" reintroduced direct human-computer interaction and led to the command-line and menu-oriented interfaces with which the senior citizens of computing (people over thirty) are probably familiar. Walker attributes the notion of "conversationality" in human-computer interfaces to this kind of interaction, where a person does something and a computer responds—a tit-for-tat interaction.

This simplistic notion of conversation led many early interface specialists to develop a model of interaction that treats human and computer as two distinct parties whose "conversation" is mediated by the screen. But as advances in linguistics have demonstrated, there is more to conversation than tit-for-tat. Dialogue is not just linearized turn-taking in which I say something, you go think about it and then you say something, I go think about it, and so on. An alternative model of conversation employs the notion of *common ground*, described by Herbert H. Clark and Susan E. Brennan [1990]:

> It takes two people working together to play a duet, shake hands, play chess, waltz, teach, or make love. To succeed, the two of them have to coordinate both the content and process of what they are doing. Alan and Barbara, on the piano, must come to play the same Mozart duet. This is coordination of content. They must also synchronize their entrances and exits, coordinate how loud to play forte and pianissimo, and otherwise adjust to each other's tempo and dynamics. This is coordination of process. They cannot even begin to coordinate on content without assuming a vast amount of shared information or common ground—that is, mutual knowledge, mutual beliefs, and mutual assumptions [Clark and Carlson, 1982; Clark and Marshall, 1981; Lewis, 1969; Schelling, 1960]. And to coordinate on process, they need to update, or revise, their common ground moment by moment. All collective actions are built on common ground and its accumulation. [Clark and Brennan, 1990]

In her work in applying the notion of common ground to human-computer interfaces, Brennan [1990a] suggests that common ground is a jointly inhabited "space" where meaning takes shape through the collaboration and successive approximations of the participants. Brennan's ongoing work is aimed at designing human-computer interfaces so that they offer means for establishing common ground ("grounding") that are similar to those that people use in human-to-human conversation—for example, interruptions, questions, utterances, and gestures that indicate whether something is being understood [Brennan, 1990b].

Contemporary graphical interfaces, as exemplified by the Macintosh, explicitly represent part of what is in the "common ground" of interaction through the appearance and behavior of objects on the screen. Some of what goes on in the representation is exclusively attributable to either the person or the computer, and some of what happens is a fortuitous artifact of a collaboration in which the traits, goals, and behaviors of both are inseparably intertwined.

The notion of common ground not only provides a superior representation of the conversational process but also supports the idea that an interface is not simply the means whereby a person and a computer represent themselves to one another; rather it is a shared context for action in which both are agents. (This book will employ the noun "agent" to mean *one who initiates action*, a definition consistent with Aristotle's use of the concept in the *Poetics*. Insurance agents, real estate agents, and secret agents are examples of a kind of agency that is more complex—and vaguely ominous. The subject of "interface agents" is discussed later in Chapter 5.) When the old tit-for-tat paradigm intrudes, the "conversation" is likely to break down, once again relegating person and computer to opposite sides of a "mystic gulf" filled with hidden processes, arbitrary understandings and misunderstandings, and power relationships that are competitive rather than cooperative. "Mistakes," unanticipated outcomes, and error messages are typical evidence of such a breakdown in communication, where the common ground becomes a sea of misunderstanding.

ıtion of common ground to
nan [1990a] suggests that
ed "space" where meaning
on and successive approxi-
n's ongoing work is aimed
terfaces so that they offer
;round ("grounding") that
e in human-to-human con-
ons, questions, utterances,
something is being under-

ıces, as exemplified by the
rt of what is in the "com-
ıugh the appearance and
ɔme of what goes on in the
ıtable to either the person
at happens is a fortuitous
he traits, goals, and behav-
ned.
ıot only provides a superi-
ɔnal process but also sup-
is not simply the means
r represent themselves to
ɔntext for action in which
ıploy the noun "agent" to
efinition consistent with
Poetics. Insurance agents,
are examples of a kind of
d vaguely ominous. The
ussed later in Chapter 5.)
intrudes, the "conversa-
again relegating person
ı "mystic gulf" filled with
standings and misunder-
hat are competitive rather
ıticipated outcomes, and
of such a breakdown in
ground becomes a sea of

The notion of interface metaphors was introduced to provide people with a conceptual scheme that would guard against such misunderstandings by deploying familiar objects and environments as stakes in the common ground. But even "good" metaphors don't always work. For instance, in an informal survey of Macintosh-literate university students, many people failed to employ the word "desktop" anywhere in their description of the Finder.[1] Where an interface metaphor diverges significantly from its real-world referent, people proceed by accounting for the behaviors of particular "objects" on the screen with ad hoc explanations of system operation, which are often incorrect—a "naive physics" of computing [see Owen 1986]. In such cases, metaphors do not serve as "stakes in the common ground," but rather as cognitive mediators whose labels may be somewhat less arcane (but possibly more ambiguous) than a computer scientist's jargon.

Although interface metaphors can fail in many ways (as discussed later in Chapter 5), their growing prevalence, especially in graphical interfaces, has expanded the domain of interface design to admit contributions from specialists in graphic and industrial design, linguistics, psychology, education, and other disciplines. An important contribution of the metaphorical approach has been to make interface design an *interdisciplinary* concern. The next section focuses on two of those "interdisciplines": psychology and graphic design.

Interface Interdisciplines

Psychology is a familiar domain to dramatists, actors, and other theatre artists because of its focus on human behavior. Understanding how psychology and theatre are alike and

[1] The Macintosh Finder is an application for managing people's file systems and for launching other applications. It comes with the system and is automatically launched when the machine is turned on. The Finder was designed on the basis of a "desktop metaphor," employing graphical icons to represent individual files as "documents" and hierarchical organizational units as "folders."

different may illuminate the distinct contributions that each can make in the field of human-computer interaction.

The two disciplines have several elements in common. Both concern themselves with how agents relate to one another in the process of communicating, solving problems, building things, having fun—the whole range of human activity. Both interpret human behavior in terms of goals, obstacles, conflicts, discoveries, changes of mind, successes, and failures. Both domains have important contributions to make to interface theory and design. Both attempt to observe and understand human behavior, but they employ that understanding to different ends: In general, psychology attempts to *describe what goes on in the real world* with all its fuzziness and loose ends, while theatre attempts to *represent something that might go on*, simplified for the purposes of logical and affective clarity. Psychology is devoted to the end of explaining human behavior, while drama attempts to represent it in a form that provides intellectual and emotional closure. Theatre is informed by psychology (both professional and amateur flavors), but it turns a trick that is outside of psychology's province through the art of representing action. By taking a look at some of the key ideas that psychology has contributed to interface design, we may be able to identify some ways in which theatrical knowledge can extend and complement them.

Psychologists have been involved in the quest to understand and shape human-computer interaction almost since the beginning of computing, through such disciplines as human factors and computer-aided instruction.[2] In the 1970s

[2] The literature on "human factors" and other psychological perspectives on human-computer interaction is huge. It is beyond the scope and purpose of this book to provide even a cursory survey of the entire domain. The work mentioned in this chapter is selected in terms of its relevance to the thesis of this particular book. Interested readers may wish to review *The Human Factor: Designing Computer Systems for People* by Richard Rubinstein and Harry Hersh [1984], which includes an excellent bibliography, *Readings in Human-Computer Interaction: A Multidisciplinary Approach* by Ronald M. Baecker and William A.S. Buxton [1987], or the various proceedings of ACM SIGCHI and the Human Factors Society.

and on through the 1980s, cognitive psychologists developed perspectives on human-computer interaction that were more critically focused on interface design than those of their colleagues in other branches of psychology. The work of Donald A. Norman, founder of the Institute for Cognitive Psychology at the University of California at San Diego, is especially illuminating. In the 1980s, Norman built a lab at UCSD that fostered some of the most innovative and germane thinking about human-computer interaction to date [see Norman and Draper, 1986, for a collection of essays by members and associates of this group]. Norman's perspective is highly task-oriented. In his book, *The Psychology of Everyday Things* [1988], Norman drives home the point that the design of an effective interface—whether for a computer or a doorknob—must begin with an analysis of what a person is trying to *do*, rather than with a metaphor or a notion of what the screen should display.

Norman's emphasis on action as the stuff that interfaces both *enable* and *represent* bores a tunnel out of the labyrinth of metaphor and brings us back out into the light, where *what is going on* is larger, more complex, and more fundamental than the way the human and the computer "talk" to each other about it.

Norman's insights dovetail nicely with those of the "common ground" linguists, suggesting a notion of the interface that is more than screen-deep. The interface becomes the arena for the performance of some task in which both human and computer have a role. What is represented in the interface is not only the task's environment and tools but also the process of interaction—the contributions made by both parties and evidence of the task's evolution. I believe that Norman's analysis supports the view that interface design should concern itself with representing *whole actions with multiple agents*. This is, by the way, precisely the definition of theatre.

Norman has also been a key figure in the development of another pivotal interface concept, the idea of *direct manipulation*. Direct manipulation interfaces employ a psychologist's knowledge of how people relate to objects in the real world in the belief that people can carry that knowledge across to the manipulation of virtual[3] objects that represent computational entities and processes

The term *direct manipulation* was coined by Ben Shneiderman of the University of Maryland, who listed three key criteria:

1. Continuous representation of the object of interest
2. Physical actions or labeled button presses instead of complex syntax
3. Rapid incremental reversible operations whose impact on the object of interest is immediately visible [Shneiderman, 1987]

Shneiderman reports that direct-manipulation interfaces can "generate a glowing enthusiasm among users that is in marked contrast with the more common reaction of grudging acceptance or outright hostility" [Shneiderman, 1987]. In a cognitive analysis of how direct manipulation works, Hutchins, Hollan, and Norman [1986] suggest that direct manipulation as defined may provide only a partial explanation of such positive feelings. They posit a companion effect, labeled *direct engagement*, a *feeling* that occurs "when a user experiences direct interaction with the objects in a domain" [the notion of direct engagement is introduced in Laurel, 1986b]. Hutchins et al. add the requirements that input expressions be able to make use of previous output expressions, that the system create the illusion of instantaneous response (except where inappropriate to the domain), and that the interface be unobtrusive.

It seems likely that direct manipulation and direct engagement are head and tail of the same coin (or two handfuls of the same elephant)—one focusing on the qualities of action and the other focusing on subjective response. The basic issue

---

[3] The adjective *virtual* describes things—worlds, phenomena, etc.—that look and feel like reality but that lack the traditional physical substance. A virtual object, for instance, may be one that has no real-world equivalent, but the persuasiveness of its representation allows us to respond to it *as if* it were real.

that represent computational

tion was coined by Ben
Maryland, who listed three

the object of interest.
ton presses instead of

perations whose impact on
ately visible [Shneiderman,

ct-manipulation interfaces
sm among users that is in
mon reaction of grudging
Shneiderman, 1987] In a
t manipulation works,
1986] suggest that direct
de only a partial explana-
posit a companion effect,
hat occurs "when a user
the objects in a domain"
is introduced in Laurel,
equirements that input
previous output expres-
lusion of instantaneous
ate to the domain), and

ation and direct engage-
oin (or two handfuls of
the qualities of action
esponse. The basic issue

lds, phenomena, etc —that
ditional physical substance.
e that has no real-world
presentation allows us to

is what is required to produce the feeling of taking action within a representational world, stripped of the "metacontext" of the interface as a discrete concern. Hutchins et al. sum it up this way: "Although we believe this feeling of direct engagement to be of critical importance, in fact, we know little about the actual requirements for producing it." Nevertheless, their analysis as well as Shneiderman's [1987] provide many valuable insights and useful examples of the phenomenon.

If we remove Shneiderman's clause regarding labeled button presses (because in many cases buttons are the artifacts of a pernicious interface metacontext), then the sense of directness can be boiled down to continuous representation, "physical" action, and apparent instantaneity of response. Apparent instantaneity depends upon both processing speed and the elimination of representations of intermediate activities in design. In the analyses of both Shneiderman and Hutchins et al., continuous representation and physical action depend heavily upon graphical representation. In fact, Hutchins et al. identify the granddaddy of direct manipulation as Ivan Sutherland's graphical design program, *Sketchpad* [Sutherland, 1963]. Graphical (and, by extension, multisensory) representations are fundamental to both the physical and emotional aspects of directness in interaction. Hence, it is worthwhile to examine the role and contributions of graphic design in the interface domain.

In many ways, the role of the graphic designer in human-computer interaction is parallel to the role of the theatrical scene designer. Both create representations of objects and environments that provide a context for action. In the case of theatre, the scene designer provides objects like teacups and chairs ("props"), canvas-covered wooden frames that are painted to look like walls ("flats"), and decorative things like draperies and rugs ("set dressing"). The behaviors of these elements is also designed—doors open, make-believe bombs explode, trick chairs break in barroom brawls. The lighting designer uses elements like color, intensity, and direction to illuminate the action and its environment and to focus our attention on key areas and events.

Both scene and light designers use such elements as line, shadow, color, texture, and style to suggest such contextual information as place, historical period, time of day, season, mood, and atmosphere Theatrical designers also employ metaphor (and amplify the metaphors provided by the playwright) in the design of both realistic and nonrealistic pieces: the looming cityscape around Willy Loman's house in *Death of a Salesman* metaphorically represents his isolation and the death of his dreams; abstract webs of gauzy fabric suggest the multiple layers of illusion in the personality of Peer Gynt.

Likewise, in the world of interfaces, the graphic designer renders the objects and environments in which the action of the application or system will occur, imparting behaviors to some objects (like zoom-boxes and pop-up menus) and representing both concrete and ephemeral aspects of context through the use of such elements as line, shadow, color, intensity, texture, and style. Such familiar metaphors as desktops and windows provide behavioral and contextual cues about the nature of the activity that they support.

Both theatrical design and graphical interface design are aimed at creating representations of *worlds that are like reality only different*. But a scene design is not a whole play—for that we also need representations of character and action. Likewise, the element of graphical design is only part of the whole representation that we call an interface.

Throw the Baggage Out

The previous section picks up some of the more promising threads in the evolving discipline of interface design. It also suggests that these elements alone may not be sufficient in defining the nature of human-computer interaction or in realizing it effectively, and it recommends theatre as an additional perspective. But it may not be productive for theatre people simply to join all the other cooks in the kitchen. I want to take the argument a step further and suggest that the concept of *interface* itself is a hopeless hash, and that we might do better to throw it out and begin afresh.

-computer interaction, in . The triangles represent nerated types, and the other nvironment The shape of otlight, to suggest that all d."

etaphor

ty suggests a number of
is confined to the world
situated in the same con-
and speak the same lan-
age to speak by noticing
objects are and what they
A good example of this
ming by Rehearsal, devel-
Finzer at Xerox PARC in
l programming environ-
There are some problems
per se,[5] but the principle

oining a group of performers
This is a case where a novel
aking point This particular
een stage and performer and
sed "only" metaphorically.

of "the representation is all there is" is applied consistently 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]

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?

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

s missing in most attempts
e metaphor. A central goal
vhich we can use a notion
· but as a way to conceptu-
self.
ws us to simplify another
the notion of "interact-
business have been argu-
988, Alexander Associates
nnual conference bringing
the interactive entertain-
uch diverse industries as
roadcast and cable televi-
amusement parks. Over
about the meaning of the
very session, disrupting
ations. People seemed to
cultural discovery of the
d a coherent definition.
no one succeeded in pre-
eneral acceptance. Many
satisfied. Could it be the

ame tree. I posited that
it could be characterized
ften you could interact),
lable), and significance
matters) [Laurel, 1986a
game judged by these
g only once in a while,
choose from, and the
ake much difference to
mputer game (or desk-
do something that real-
be anything you could

provide only part of the
imentary measure of
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.[6]

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 activity that is familiar, comprehensible, and evocative. The rest of this book

---

[6]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, then halfway of halfway, etc. Mathematics offers a solution; so does common sense. But the paradox is compelling enough to have interested logicians and mathematicians for centuries.

will explore some of the theoretical and practical aspects of theatre that can be directly applied to the task of designing human-computer experiences. But there are a few more stones to be turned in arranging the groundwork for this discussion.

Is Drama Serious Enough?

Because theatre is a form of entertainment, many people see it as fundamentally "non-serious." I have found that computer-science–oriented developers exhibit a high resistance to a theatrical approach to designing human-computer activity on the grounds that it somehow trivializes "serious" applications. Graphic designers undoubtedly have had to wrestle with the same sort of bias, design being seen not as a task of representation but one of mere decoration. Decoration is suspect because it may get in the way of the serious work to be done. (The same argument was used a few decades ago to ban bright colors, potted plants, and chatchkas from the workplace—but that's another story.) The fact of the matter is that graphics is an indispensable part of the representation itself, as amply demonstrated by the Macintosh and other contemporary computing environments.

The no-frills view that permeates thinking about the interfaces of "serious" applications is the result of a fundamental misunderstanding of the nature of seriousness in representations. The idea that theatre is "really not real" and is therefore unsuited as an approach to serious human-computer activities is misguided, because those activities are "really not real" in precisely the same ways. Without the representation, there is nothing at all—and theatre gives good representation.

Human-computer activity may be divided into two broad categories: productive and experiential [Laurel, 1986b]. Experiential activities, such as computer games, are undertaken purely for the experience afforded by the activity as you engage in it, while productive activities such as word processing have outcomes in the real world that are somehow beyond the experience of the activity itself. They are often myths and stories, such as the tragedies of Agamemnon, Orestes, and Oedipus. They communicated philosophical and religious ideas as well as providing the occasion for the collective experience of emotion. Quite simply, Greek drama was the way that Greek culture publicly thought and felt about the most important issues of humanity, including ethics, morality, government, and religion. To call drama merely "entertainment" in this context is to miss most of the picture. Imagine how our own culture would be transformed if the basic fare of television, for instance, were the same subjects as those treated by the Greek theatre. What if television were the way for our whole culture to consider matters of deepest import? The human need for such forums has not changed, but television seems to be the only pervasive means American society has devised for meeting it. A case can therefore be made that the trivial content and impoverished range of points of view that characterize commercial television diminish the human spirit by diminishing what we think about and how we think about it. The Greeks employed drama and theatre as *tools for thought*, in much the same way that we employ computers today—or at least in the ways that we envision employing them in the not-too-distant future.

In science as well as in art, the Greeks of the fifth and fourth centuries B.C. were discovering and inventing a world of unprecedented scope and order through the rapidly evolving tools of philosophy. In exploring the nature of the drama and other arts, Aristotle employed the same conception of causality to which he attributed the forms of living things, and that is a good place to begin.

The Four Causes, or Why Things Are the Way They Are

How does a representation—a play or a human-computer activity—get to be the way it is? What defines its nature, its shape, its particulars? What forces are at work? If you are tempted to balk at this excursion into the deepest regions of

tragedies of Agamemnon,
ommunicated philosophical
oviding the occasion for the
Quite simply, Greek drama
e publicly thought and felt
es of humanity, including
nd religion. To call drama
ontext is to miss most of the
ulture would be transformed
nstance, were the same sub-
k theatre. What if television
ture to consider matters of
d for such forums has not
e the only pervasive means
or meeting it. A case can
content and impoverished
acterize commercial televi-
by diminishing what we
ut it. The Greeks employed
ght, in much the same way
or at least in the ways that
not-too-distant future.
e Greeks of the fifth and
ing and inventing a world
through the rapidly evolv-
g the nature of the drama
l the same conception of
e forms of living things,

Things Are the

y or a human-computer
hat defines its nature, its
are at work? If you are
o the deepest regions of

theory, let me remind you of the reason for taking it: Understanding how things work is necessary if we are to know how to make them. When a made thing is flawed or unsuccessful, it may not be due to poor craftsmanship. Architects have designed and built beautiful buildings that wouldn't stand up; playwrights have written plays with mellifluous words and solid dramatic structure that closed after one night in New Jersey; and engineers have designed software with lovely screens and loads of "functionality" that leaves people pounding on their keyboards in frustration. The reason for failure is often a lack of understanding about how a thing works, what its nature is, and what it will try to be and do—whether you want it to or not—because of its intrinsic form.

The four causes are forces that operate concurrently and interactively during the process of creation. Although Aristotle also applied them to living organisms, our discussion will be restricted to the realm of made things. We will begin with definitions of the four causes and then apply them, first to drama, and then to human-computer activity.[1]

- *Formal cause*: The formal cause of a thing is the form or shape of what it is trying to be. For example, the formal cause of a building is the architect's notion of what its form will be when it is finished. Those formal properties of "building-ness" (or "church-ness," or "house-ness," etc.) that are independent of any particular instance of a building (or church or house) and which define what a building is are one component of the formal cause. They are filtered through the mind of the architect, where they are particularized by various design contingencies (there needs to be sunlight in the morning room; the conference room needs to accommodate a group of fifty, etc.), as well as his or her own values, tastes, and ideas.

[1] I have employed the traditional terminology, not out of a desire to promote philosophical jargon but because it is quite difficult to find synonyms that do these concepts justice, and also because more casual terminology can lead to confusion downstream.

Formal cause operates through an idea or vision of the completed whole, which will undergo change and elaboration as the process of creation unfolds; that is, there is a reciprocal relationship between the formal cause and the work in progress. The formal cause for a thing may be muddy or clear, constant or highly evolutionary, but it is always present.

- *Material cause:* The material cause of a thing is what it is made of. So, to pursue the architecture example, the material cause of a building includes stones or concrete or wood, glass, nails, mortar, and so on. Note that the properties of the materials influence the properties of the structure; for example, wood is more flexible than steel, but steel is stronger.
- *Efficient cause:* The efficient cause of a thing is the way in which it is actually made. This includes both the maker and the tools. For instance, two buildings with the same architectural plan and the same materials created by different builders with different skills and tools will differ in terms of their efficient cause.
- *End cause:* The end cause of a thing is its purpose—what it is intended to *do* in the world once it is completed. In architecture, a building is intended to accommodate people, living or working or playing or performing operas or whatever, according to the kind of building it is.

Now let's apply these four causes to the theatre:

- *Formal cause:* The completed plot—that is, the *whole action* that the playwright is trying to represent. The whole action subsumes notions of form and genre and the patterns that define them.
- *Material cause:* The stuff a play is made up of—the sounds and sights of the actors as they move about on the stage. Note that the material of a play is not words, as one might think from reading a script. That's because plays are intended to be acted out, and there's more to enactment than words. The *enactment* is the performance—that which unfolds before the eyes and ears of the audience.

gh an idea or vision of the
ndergo change and elabo-
unfolds; that is, there is a
the formal cause and the
cause for a thing may be
hly evolutionary, but it is

se of a thing is what it is
ecture example, the mate-
es stones or concrete or
o on. Note that the prop-
he properties of the struc-
e flexible than steel, but

e of a thing is the way in
includes both the maker
buildings with the same
materials created by dif-
lls and tools will differ in

ng is its purpose—what it
once it is completed. In
ed to accommodate peo-
or performing operas or
f building it is.

the theatre:

—that is, the *whole action*
o represent. The whole
and genre and the pat-

made up of—the sounds
nove about on the stage.
not words, as one might
hat's because plays are
ere's more to enactment
erformance—that which
f the audience.

- *Efficient cause*: The skills, tools, and techniques of the playwright, actors, and other artists who contribute to the finished play.
- *End cause*: The pleasurable arousal and expression of a particular set of emotions in the audience (*catharsis*).

As mentioned in the section on catharsis in Chapter 1, "pleasurable" is a key word in understanding catharsis; emotions aroused by plays are not experienced in the same way as emotions aroused by "real" events, and even the most negative emotions can be pleasurable in a dramatic context (the success of such film genres as suspense and horror depends on this fact). Various historical periods have added "riders" like political consciousness-raising or moral instruction to the end cause. It is safe to say that since emotion depends upon the successful communication of content, then some level of communication is implicit in the end cause. We will explore this aspect further in the discussion of causality and universality in Chapter 3.

The Four Causes of Human-Computer Activity

How can we define these four causes for human-computer activities? In this discussion it is difficult to avoid using computer-related terminology, which is in many cases already loaded with connotations that are not always appropriate. Among these terms are *functionality*, *program*, *application*, *representation*, and *agent*.

In computerese, *functionality* refers to the things that a program does—a spreadsheet can make calculations of certain types, for instance, and a word processor can do such things as move text around, display different fonts, and check spelling. Interface designers often describe their task as representing a program's functionality [for example, see Rubinstein and Hersh, 1984, p. 19]. But this understanding of functionality brings us to the tree falling in the forest again. A spreadsheet's ability to crunch numbers in certain ways is only *potential* until a person gives it some numbers to crunch and tells it how to crunch them, in fine or gross detail. Thus the definition of functionality needs to be reconceived as *what a person can do with a program*, rather than what a program has the capacity to do. This definition lands us back in the territory of action with human and computer agents. It also contains a word we haven't used before: "program."

A program is a set of instructions that defines the potential actions that make up a human-computer activity and their representations. These actions and representations may change as a result of ongoing action (for instance, as the result of capturing or inferring people's preferences). A program also defines the environment for action and the other objects that inhabit that environment, including their representations and capabilities. Actually, the elements of action and environment and their representations are usually the result of more than one program—with the Macintosh, for instance, certain aspects of the "interface" are embedded in the operating system and Finder code. Of course, the potential of a program is also shaped by the hardware for which it is written—what kind of math it can perform, for instance, and the qualities of its graphical display.

In theatrical terms, a program (or a cluster of interacting programs) is analogous to a script, including its stage directions. A script is constrained by the physical realities of the kind of theatre in which it is to be performed and the capabilities of the stage machinery and actors. Program code is equivalent to the *words* of a script (including the theatre's own brand of jargon—for example, "move stage left" or "counter-cross"). In his investigations of artificial intelligence, Julian Hilton adds another dimension to this analogy:

> The text [of a play] therefore, is a combination of explicit and implicit notational systems which have as their initial purpose the enablement of an event in which performers and audience can share as partners. While obviously the notion of a computer was alien to Shakespeare, that of his theatre as a complex space-time machine was certainly not . . . [Hilton, 1991]

umbers in certain ways is
t some numbers to crunch
ı fine or gross detail. Thus
eds to be reconceived as
ı, rather than what a pro-
lefinition lands us back in
ı and computer agents. It
ed before: "program."
ns that defines the poten-
n-computer activity and
and representations may
ion (for instance, as the
ple's preferences) A pro-
for action and the other
nt, including their repre-
y, the elements of action
ntations are usually the
with the Macintosh, for
erface" are embedded in
de. Of course, the poten-
he hardware for which it
n perform, for instance,
lay.
r a cluster of interacting
ncluding its stage direc-
physical realities of the
formed and the capabil-
ctors. Program code is
including the theatre's
, "move stage left" or
of artificial intelligence,
to this analogy:

bination of explicit and
e as their initial purpose
erformers and audience
the notion of a computer
ıeatre as a complex
[Hilton, 1991]

Functionality is equivalent to the script parsed not by words but by *actions* An apparent difference between programs and theatrical scripts is that programs are not intrinsically linear in form, while scripts generally are. At the highest level, this nonlinearity means that programs can cause different things to happen depending upon the actions of their users The way in which computer functionality differs from dramatic action is that some portion of it is shaped by a person as the action unfolds; that is, "authorship" is collaborative in real time (this aspect will be further explored below in the discussion of plot). In summary, then, *functionality consists of the actions that are performed by people and computers working in concert, and programs are the means for creating the potential for those actions*

An "application" is generally described as a program designed to deliver a particular functionality to "end-users," which is distinct from the type of programs that are not directly accessible to people, such as those which live deep in the bowels of missile silos and operating systems. Informal taxonomies of applications exist; for example, applications for word processing and spreadsheets belong to the larger class of productivity applications; drawing, painting, and music programs are often classified as "creativity" applications; and adventure, action, and strategy games are "entertainment" applications *The most important way in which applications, like plays, are individuated from one another is by the particular actions that they represent*. Applications are analogous to individual plays; the larger categories are analogous to genres and forms of plays (tragic, comic, didactic, etc.).

We have used the word "representation" throughout Chapter 1 to distinguish the shadowy realms of art and human-computer activity from phenomenal reality. Webster's defines a representation as "an artistic likeness or image" (and also, incidentally, as "a dramatic production or performance"). The Greek word for artistic representation is *mimesis*. Both plays and human-computer activities are *mimetic* in nature; that is, they exhibit the characteristics of artistic representations. A mimesis is a made thing, not an accidental or arbitrary one: using a pebble to represent a man is not mimetic; making a doll to represent him is We often use the word "representation" followed by "of" and then the name of some object—a character is a representation of a person, a landscape painting is a representation of a place. But in art as in human-computer activities, the object of a mimesis (i.e , that which it is intended to represent) may be a real thing or a virtual one; that is, a thing that exists nowhere other than the imagination A play may be a mimesis of events (literally, a series of actions) that are taken from history or that are entirely "made up." *Mimetic representations do not necessarily have real-world referents.*

In computerese, two kinds of representations are acknowledged: internal and external representations In the Macintosh Finder, for instance, a page icon is the external representation of a document. Both the document and the icon have internal representations that consist of the code that defines them—how they look and behave However, in keeping with the principle that "the representation is all there is," an internal representation has no value by itself—just as the script for a performance is never seen (and hopefully never thought about) by an audience. As a program, an internal representation is merely the potential for what may be manifest in the external representation—that which has sensory and functional properties As it is used in this book, the term "representation" subsumes both aspects.

We have said that human-computer activities can be defined as representations of actions with agents of both human and computer origin. The word "agents" has a particular meaning in computerese, which is a derivation of the more general sense of the word A computer-based "agent" is defined as a bundle of functionality that performs some task for a person, either in real time or asynchronously An example is the mail-sorting agents developed by Thomas Malone in the Object Lens project [Crowston and Malone, 1988]. Agents may be represented anthropomorphically —that is, as characters—but they need not be The Aristotelian definition

present a man is not mimet-
ı is. We often use the word
and then the name of some
:ation of a person, a land-
of a place. But in art as in
ject of a mimesis (i.e., that
nay be a real thing or a vir-
ts nowhere other than the
nesis of events (literally, a
n history or that are entire-
ions do not necessarily have

of representations are
ıal representations. In the
page icon is the external
th the document and the
s that consist of the code
and behave. However, in
representation is all there
no value by itself—just as
ever seen (and hopefully
e. As a program, an inter-
otential for what may be
ion—that which has sen-
t is used in this book, the
h aspects.
nputer activities can be
ons with agents of both
ord "agents" has a partic-
ch is a derivation of the
omputer-based "agent" is
that performs some task
ynchronously. An exam-
ped by Thomas Malone
ton and Malone, 1988].
morphically—that is, as
e Aristotelian definition

of an agent is the root of both of these permutations: *An agent is one who initiates and performs actions*. So in any human-computer activity, there is at least one agent—the human who turns on the machine—and if the machine does anything after it boots, then there are at least two. This book uses the more general definition because, as we will see later in this chapter, computer-based agency is present in all human-computer activities, whether or not it is coalesced into coherent agent-like "entities" in the representation.

Given these definitions, we can now take a run at the four causes as applied to human-computer activity:

- *Formal cause:* The formal cause of a particular human-computer activity is the form of what it is trying to be. Human-computer interaction generally lacks the kind of well-known formal categories offered by drama (comedy, tragedy, etc.).[2] What we can say, however, is that the *form* of human-computer activity is a representation of action with agents that may be either human, computer-based, or a combination of both. We will discover more of the characteristics of that form as we identify its structural elements and the relations among them.
- *Material cause:* The material cause of a human-computer activity, and also of a play, is the enactment—that which unfolds before a person's senses. As plays employ the sights and sounds produced by actors moving about in scenic environments, computers may employ graphics, sound and music, text characters, and even tactile and kinesthetic effects. In the discussion of structural elements below, we will see how these sensory materials are shaped into more sophisticated constructs.

---

[2] Although application categories like "word processing" or "productivity" are sometimes invoked by designers as if they were formal criteria, I would argue that they are rather part of the end cause, since their definitions are essentially functional rather than formal. As most computer-using writers know, it is still impossible to derive the "canonical" form of a word processor from all of the instances that exist on the market; we can speak only about a word processor's expected or necessary functionality.

- *Efficient cause:* The efficient cause of a human-computer activity is the skills and tools of its maker(s). Since a given application is probably based, at least in part, on chunks of program code that have been created by other people for other purposes, the computer equivalent of a playwright is usually a group of people. Both theatre and human-computer activity design are collaborative disciplines; both depend upon a variety of artistic and technical contributions. In both domains, the quality and nature of these contributions are strongly influenced by the available tools. Theatrical artists today are increasingly relying on computer-based tools for such tasks as lighting and scene design, lighting execution, moving scenery, designing costumes, storing and simulating dance notation and period movements, and, of course, writing scripts. Theatrical folk express the same frustrations with their tools as graphic designers and other artists who are working in the computer medium itself.

- *End cause*: The end cause of a human-computer activity is what it is intended to *do* in the world once it is completed. Thus the end cause obviously involves functionality: Word processors had better spit out documents. But experience is an equally important aspect of the end cause; that is, what a person thinks and feels about the activity is part of its reason for being the way it is. At the very least, a person must understand the activity well enough to do something. At best, he or she should be engaged, pleased, or even delighted by the experience. This aspect of the end cause seems trivial to many; it is too often handed off as an afterthought to harried interface designers who follow programmers around with virtual brooms and pails. How much better it is to place the notion of pleasurable experience where it can achieve the best results—as part of the necessary nature of human-computer activity.

e of a human-computer
s maker(s). Since a given
least in part, on chunks
created by other people
er equivalent of a play-
ople Both theatre and
are collaborative disci-
ty of artistic and techni-
s, the quality and nature
influenced by the avail-
are increasingly relying
h tasks as lighting and
moving scenery, design-
ing dance notation and
urse, writing scripts.
frustrations with their
er artists who are work-

an-computer activity is
ld once it is completed
nvolves functionality:
t documents. But expe-
ct of the end cause; that
bout the activity is part
is. At the very least, a
ity well enough to do
d be engaged, pleased,
. This aspect of the end
oo often handed off as
designers who follow
rooms and pails. How
. of pleasurable experi-
results—as part of the
er activity.

The Six Elements and Causal Relations Among Them

One of Aristotle's fundamental ideas about drama (as well as other forms of literature) is that a finished play is an *organic whole*. He used the term *organic* to evoke an analogy with living things. Insofar as a whole organism is more than the sum of its parts, all of the parts are necessary for life, and the parts have certain necessary relationships to one another. He identified six qualitative elements of drama and suggested the relationships among them in terms of formal and material causality.[3]

I present Aristotle's model here for two reasons. First, I am continually amazed by the elegance and robustness of the categories and their causal relations. Following the causal relations through as one creates or analyzes a drama seems to automagically reveal the ways in which things should work or exactly how they have gone awry. Second, Aristotle's model creates a disciplined way of thinking about the design of a play in both constructing and debugging activities. Because of its fundamental similarities to drama, human-computer activity can be described with a similar model, with equal utility in both design and analysis.

Table 2.1 lists the elements of qualitative structure in hierarchical order. Here is the trick to understanding the hierarchy: Each element is the formal cause of all those below it, and each element is the material cause of all those above it. As you move up the list of elements from the bottom, you can see how each level is a successive refinement—a *shaping*—of the materials offered by the previous level. The following sections expand upon the definitions of each of the elements in ascending order.

[3] The explicit notion of the workings of formal and material causality in the hierarchy of structural elements is, although not apocryphal, certainly neo-Aristotelian. See Smiley [1971].

Element	In Drama	In Human-Computer Activity
Action	The whole action being represented. The action is theoretically the same in every performance.	The whole action, as it is collaboratively shaped by system and user. The action may vary in each interactive session.
Character	Bundles of predispositions and traits, inferred from agents' patterns of choice.	The same as in drama, but including agents of both human and computer origin.
Thought	Inferred internal processes leading to choice: cognition, emotion, and reason.	The same as in drama, but including processes of both human and computer origin.
Language	The selection and arrangement of words; the use of language.	The selection and arrangement of signs, including verbal, visual, auditory, and other nonverbal phenomena when used semiotically.
Melody (Pattern)	Everything that is heard, but especially the melody of speech.	The pleasurable perception of pattern in sensory phenomena.
Spectacle (Enactment)	Everything that is seen.	The sensory dimensions of the action being represented: visual, auditory, kinesthetic and tactile, and potentially all others.

Table 2.1 **The six qualitative elements of structure in drama and in human-computer activity.**

Enactment

Aristotle described the fundamental material element of drama as "spectacle"—all that is seen. In the *Poetics*, he also

In Human-Computer Activity

The whole action, as it is collaboratively shaped by system and user. The action may vary in each interactive session.

The same as in drama, but including agents of both human and computer origin.

The same as in drama, but including processes of both human and computer origin.

The selection and arrangement of signs, including verbal, visual, auditory, and other nonverbal phenomena when used semiotically.

The pleasurable perception of pattern in sensory phenomena.

The sensory dimensions of the action being represented: visual, auditory, kinesthetic and tactile, and potentially all others.

›f structure in drama and in

ıtal material element of
ən. In the *Poetics*, he also

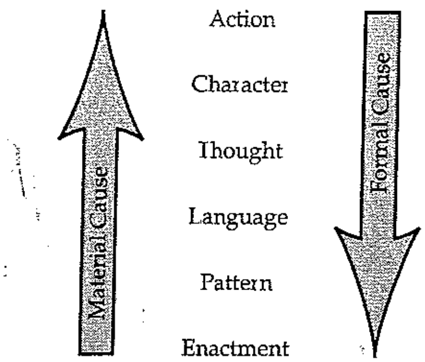


Figure 2.2 **Causal relations among elements of quantitative structure.**

referred to this element as "performance," which provides some basis for expanding the definition to include other senses as well. Some scholars place the auditory sense in the second level because of its association with music and melody, but, as will be seen in the next section, it is more likely that the notion of melody pertains to the *patterning* of sound rather than to the auditory channel itself.

One difference, probably temporary, between drama and human-computer activity is the senses that are addressed in the enactment.[4] Traditionally, plays are available only to the eyes and ears; we cannot touch, smell, or taste them. There are interesting exceptions. In the 1920s, for instance, director David Belasco experimented with using odors as part of the performance of realistic plays; it is said that he abandoned this approach when he observed that the smell of bacon frying utterly distracted the audience from the action on stage. In the mid-1960s, Morton Heilig invented a stand-alone

[4]Aristotle defined the enactment in terms of the audience rather than the actors. Although actors employ movement (kinesthetics) in their performance of the characters, that movement is perceived visually—the audience has no direct kinesthetic experience. Likewise, although things may move about on a computer screen, a human user may or may not be having a kinesthetic experience.

arcade machine called Sensorama, which provided stereoscopic filmic images, kinesthetic feedback, and environmental smells—for example, on a motorcycle ride through New York City, the audience could smell car exhaust fumes and pizza. Sensorama's problem was not that it addressed the wrong senses; it simply happened at a time when the business community couldn't figure out what to do with it—pinball parlors were monolithic, and it would be several years before *Pong* kicked off the arcade game industry.

At the same time that Heilig was thinking about multisensory arcade games and movie theatres, the development of new genres of participatory theatre accelerated. Such artists as Judith Melina and Julian Beck of the Living Theatre, Robert Wilson, Peter Brook, Jerzy Grotowski, and John Cage experimented with performances that began to dissolve the boundaries between actors and audience by placing both in the same space. Wilson, Cage, Josef Svoboda, and others produced works that integrated filmic and photographic images, musical instruments, and machines in novel ways.

In the 1980s, these trends toward increasing the sensory dimensions of audience participation gave rise to works where the audience could touch the actors and scenery and move about freely in the performance space. For example, in *Tina and Tony's Wedding*, a contemporary "interactive" play, the audience is invited to follow the actors around from room to room (kinesthetic), to touch props and sit on the furniture (tactile and kinesthetic), and to share in a wedding banquet (taste and smell). Another notable example is Chris Hardman's Antenna Theatre, where audience members move around a set prompted by taped dialogue and narration heard through personal headphones. A spate of site-specific interactive plays and "mystery weekends" in the late 1980s enjoyed a fair amount of commercial success. Contemporary performance art shares many of the same origins.

It is interesting that the development of this theatrical genre has been concurrent with the blossoming of computer games as a popular form of entertainment, and I speculate that computer games have in some ways served as a model

ı, which provided stereo-
ɛedback, and environmen-
orcycle ride through New
ɜll car exhaust fumes and
not that it addressed the
at a time when the busi-
t what to do with it—pin-
it would be several years
me industry.
'as thinking about multi-
heatres, the development
ıeatre accelerated. Such
eck of the Living Theatre,
rotowski, and John Cage
ıat began to dissolve the
ience by placing both in
Svoboda, and others pro-
nd photographic images,
ı novel ways.
d increasing the sensory
ion gave rise to works
actors and scenery and
e space. For example, in
ɔrary "interactive" play,
ctors around from room
and sit on the furniture
in a wedding banquet
ɔle example is Chris
ıdience members move
ialogue and narration
A spate of site-specific
ɜnds" in the late 1980s
success. Contemporary
ne origins.
nent of this theatrical
ɔssoming of computer
ment, and I speculate
ıys served as a model

for it. In fact, it is in the areas that dramatic entertainment and human-computer activity are beginning to converge that pan-sensory representation is being most actively explored. When we examine that convergence, we can see ways in which human-computer activity has evolved, at least in part, as drama's attempt to increase its sensory bandwidth, creating the technological siblings of the kind of participatory theatre described above.

The notion of "interactive movies," which has gained popularity in the late 1980s, has its roots in both cinema and computer games, two forms that combine theatre and technology. Earlier works were relatively isolated. These include the productions of Laterna Magica in Czechoslovakia and an "interactive movie" that was shown in the Czech Pavilion at the 1967 World Expo in Montreal, Canada, in which the audience was allowed to influence the course of the action by selecting from among several alternatives at a few key points in the film (however, it is rumored that all roads led to Rome—that is, all paths through the movie led to the same ending). The idea of interactive movies has been rekindled and transformed into a bona fide trend by advances in multimedia technology. Likewise, there were early experiments in interactive television in the mid-1970s (such as the failed Warner QUBE system). Interactive TV had to await similar technological advances before finally becoming a 1990s buzz-word.

In drama, the use of technology to create representations goes at least as far back as the *mechane* of the ancient Greeks. Cinema as a distinct form diverged from drama as the result of the impact of a new performance technology on form, structure, and style. In complementary fashion, computer games can be seen to have evolved from the impact of dramatic ideas on the technology of interactive computing and graphical displays. Computer games incorporate notions about character and action, suspense and empathy, and other aspects of dramatic representation.[5] Almost from the begin-

[5]Within the art of computer games, there are various forms, including action games, strategy games, adventure games, and so on.

ning, they have involved the visual, auditory, and kinesthetic senses (you need only watch a game player with a joystick to see the extent to which movement is involved, both as a cause and effect of the representation)

At the blending point of cinema and computer games are such new forms as super-arcade games like *Battle Tech* and sensory-rich amusement park installations like *Star Tours*. These types of systems involve the tactile and kinesthetic senses; some are investigating the inclusion of the other senses as well through both performance technology and direct stimulation to the nervous system [Rosen and Gosser, 1987] "Virtual reality" systems, which are discussed in Chapter 6, increase intensity through techniques described as *sensory immersion*—instead of looking at a screen, for instance, a person is surrounded by stereoscopic sounds and visual images delivered through earphones and "eye-phones." Through the use of special input devices like specially instrumented gloves and suits, people may move about and interact directly with objects in a virtual world. Interestingly, the first virtual reality systems and applications were developed for nonentertainment purposes like computer-aided design, scientific visualization, and training. Home computers and home game systems are not far behind these expensive, special-purpose systems in their ability to deliver multisensory representations.

The element of enactment is composed of all of the sensory phenomena that are part of the representation. Because of the evolutionary processes described above, it seems appropriate to say that enactment can potentially involve all of the senses. These sensory phenomena are the basic material of both drama and human-computer activity; they are the clay that is progressively shaped by the creator, whether playwright or designer.

Pattern

The perception of patterns in sensory phenomena is a source of pleasure for humans. Aristotle described the second element of drama as "melody," a kind of pattern in the realm of sound. In the *Poetics* he says that "melody is the greatest of the pleasurable accessories of tragedy" [*Poetics*, 1450b, 15–17]. The orthodox view is that "spectacle" is the visual dimension and "melody" is the auditory one, but this view is problematic in the context of formal and material causality. If the material cause of all sounds (music) were things that could be perceived by the eye (spectacle), then things like the vibration of vocal cords and the melodies of off-stage musicians would be excluded. On the contrary, all that is seen in a play is not shaped solely by the criterion of producing sounds or music (although this may have been more strictly true in the performance style of the ancient Greeks than it is today). The formal-material relationship does not work within the context of these narrow definitions of music and spectacle.

In the previous section, we have already expanded spectacle into all sensory elements of the enactment. The notion of melody as the arrangement of sounds into a pleasing pattern can be extended analogically to the arrangement of visual images, tactile or kinesthetic sensations, and probably smells and tastes as well (as a good chef can demonstrate). In fact, the idea that a pleasurable pattern can be achieved through the arrangement of visual or other sensory materials can be derived from other aspects of the *Poetics*, so its absence here is something of a mystery. Looking "up" the hierarchy, it could be that Aristotle did not see the visual as a potentially semiotic or linguistic medium, and hence narrowed the causal channel to lead exclusively to spoken language. Whatever the explanation, the orthodox view of Aristotle's definitions of spectacle and melody leaves out too much material. As scholars are wont to do, I will blame the vagaries of translation, figurative language, and mutations introduced by centuries of interpretation for this apparent lapse and proceed to advocate my own view.

The element of pattern thus refers to patterns in the sensory phenomena of the enactment. These patterns exert a formal influence on the enactment, just as semiotic usage formally influences patterns. A key point that Aristotle made is that patterns are pleasurable to perceive in and of themselves, whether or not they are further formulated into semiotic devices or language; he spoke of them, not only as the material for language, but also as "pleasurable accessories." Hence the use of pattern as a source of pleasure is a characteristic of dramatic representations, and one which can comfortably be extended to the realm of human-computer experience.

Language

The element of *language* (usually translated as diction) in drama is defined by Aristotle as "the expression of their [the characters'] thought in words" [*Poetics*, 1450b, 12–15]. Hence the use of spoken language as a system of signs is distinguished from other theatrical signs like the use of gesture, color, scenic elements, or paralinguistic elements (patterns of inflection and other vocal qualities). In the orthodox view, diction refers only to words—their choice and arrangement. That definition presents some interesting problems in the world of human-computer activities, many of which involve no words at all (e.g., most skill-and-action computer games, as well as graphical adventure games and graphical simulations). Are there elements in such nonverbal works that can be defined as *language?*

When a play is performed for a deaf audience and signing is used, few would argue that those visual signs function as language. The element of language in this case is expressed in a way that takes into account the sensory modalities available to the audience.[6] A designer may choose, for whatever reason, to build a human-computer system that neither senses nor responds to words, and which uses no words in the representation. Hardware configurations without keyboards, speech recognition, or text display capabilities may be unable to work with words.

[6]It is interesting to note in this context that American Sign Language (ASL) is in fact a "natural language" in its own right, and not a direct gestural map of English or any other spoken language. If a language can be constructed from gesture, then it follows that spoken words are not essential elements of language.

e of them, not only as the
"pleasurable accessories."
ce of pleasure is a charac-
and one which can com-
lm of human-computer

translated as diction) in
e expression of their [the
ics, 1450b, 12–15]. Hence
ystem of signs is distin-
like the use of gesture,
stic elements (patterns of
) In the orthodox view,
choice and arrangement.
resting problems in the
many of which involve
action computer games,
s and graphical simula-
nverbal works that can

leaf audience and sign-
se visual signs function
guage in this case is
account the sensory
A designer may choose,
-computer system that
ls, and which uses no
e configurations with-
ext display capabilities

rican Sign Language (ASL)
t, and not a direct gestural
If a language can be con-
en words are not essential

In human-computer activities, graphical signs and symbols, nonverbal sounds, or animation sequences may be used in the place of words as the means for explicit communication between computers and people. Such nonverbal signs may be said to function as language when they are the principal medium for the expression of thought. Accordingly, the selection and arrangement of those signs may be evaluated in terms of the same criteria as Aristotle specified for diction—for example, the effective expression of thought and appropriateness to character.

Thought

The element of *thought* in drama may be defined as the processes leading to a character's choices and actions—for example, to emotion, cognition, reason, and intention. Understood in this way, the element of thought "resides" within characters, although it can be described and analyzed in aggregate form (the element of *thought* in a given play may be described as concerned with certain specific ethical questions, for example). Although it may be explicitly expressed in the form of dialogue, thought is *inferred*, by both the audience and the other characters (agents), from a character's choices and actions. In his application of a theatrical analogy to the domain of artificial intelligence, Julian Hilton puts it this way: "What the audience does is supply the inferencing engine which drives the plot, obeying Shakespeare's injunction to eke out the imperfections of the play (its incompleteness) with its mind." [Hilton, 1991]

If we extend this definition of thought to include human-computer activities, it leads to a familiar conundrum: Can computers think? There is an easy answer. Computer-based agents, like dramatic characters, do not have to *think* (in fact, there are many ways in which they cannot); they simply have to *provide a representation from which thought may be inferred*.

When a folder on my Macintosh desktop opens to divulge its contents in response to my double-click, the representation succeeds in getting me to infer that that's exactly what happened—that is, the "system" understood my input, inferred *my* purpose, and did what I wanted. Was the system (or the folder) "thinking" about things this way? The answer, I think, is that it doesn't matter. The real issue is that the representation succeeded in getting me to make the right inferences about its "thoughts." It also succeeded in representing to me that it made the right inferences about mine!

Thought is the formal cause of language; it shapes what an agent communicates through the selection and arrangement of signs, and thus also has a formal influence on pattern and enactment. The traditional explanation of how language serves as material for thought is based on the overly limiting assumption that agents employ language, or the language-like manipulation of symbols, in the process of thinking. This assumption leads to the idea that characters in a play use the language of the play quite literally as the material for their thoughts.

I favor a somewhat broader interpretation of material causality: *The thought of a play can appropriately deal only with what is already manifest at the levels of enactment, pattern, and language.* Most of us have seen plays in which characters get ideas "out of the blue"—suddenly remembering the location of a long-lost will, for instance, or using a fact to solve a mystery that has been withheld from the audience thus far. The above theory would suggest that the interjection of such thoughts is unsatisfying (and mars the play) because they are not drawn from the proper material. Plays, like human-computer activities, are closed universes in the sense that they delimit the set of potential actions. As we will see in the discussion of action below, it is key to the success of a dramatic representation that all of the materials that are formulated into action are drawn from the circumscribed potential of the particular dramatic world. Whenever this principle is violated, the organic unity of the work is diminished, and the scheme of probability that holds the work together is disrupted.

This principle can be demonstrated to apply to the realm of human-computer activity as well. One example is the case in which the computer (a computer-based agent) introduces new materials at the level of thought—"out of the blue." Suppose a new word processor is programmed to be constantly checking for spelling errors and to automatically correct them as soon as they are identified. If the potential for this behavior is not represented to you in some way, it will be completely disruptive when it occurs, and it will probably cause you to make seriously erroneous inferences, to perhaps think "something is wrong with my fingers, my keyboard, or my computer." The computer "knows" why it did what it did ("thought" exists) but you do not; correct inferences cannot be made.[7] A text message, for instance, or an animation of a dictionary with its pages turning (language), could represent the action as it is occurring.

Other kinds of failures in human-computer activity can also be seen as failures on the level of thought. One of my favorite examples is a parser used in several text adventure games. This particular parser did not "know" all of the words that were used in the text representation of the story. So a person might read the sentence, "Hargax slashed the dragon with his broadsword." The person might then type, "take the broadsword," and the "game" might respond, "I DON'T KNOW THE WORD 'BROADSWORD'." The inference that one would make is that the game "agent" is severely brain-damaged, since the agent that produces language and the agent that comprehends it are assumed to be one in the same. This is the converse of the problem described in the last paragraph; rather than "knowing" more than it represented, the agent represented more than it "knew." Both kinds of errors are attributable to a glitch in the formal-material relationship between language and thought.

[7] In human factors discourse, this type of failure is attributed to a failure to establish the correct conceptual model of a given system [see Rubinstein and Hersh, 1984, Chapter 5]. The dramatic perspective differs slightly from this view by suggesting that proper treatment of the element of thought can provide a good "conceptual model" for the entire medium. It also avoids the potential misuse of conceptual models as personal constructs that "explain" what is "behind" the representation—that is, how the computer or program actually "works."

Character and Agency

Aristotle maintained that the *object* of (i.e., what is being imitated by) a drama is action, not persons: "We maintain that Tragedy is primarily an imitation of action, and that it is mainly for the sake of the action that it imitates the personal agents" (*Poetics*, 1450b, 1–5). In drama, *character* may be defined as bundles of traits, predispositions, and choices that, when taken together, form coherent entities. Those entities are the agents of the action represented in the plot. This definition emphasizes the primacy of action.

In order to apply the same definition to human-computer activities, we must demonstrate first that agents are in fact part of such representations, and second, that there are functional and structural similarities between such agents and dramatic characters.

In a purely Aristotelian sense, an agent is one who takes action. Interestingly, Aristotle admits of the possibility of a play without characters, but a play without action cannot exist [*Poetics*, 1450a, 22–25]. This suggests that agency as part of a representation need not be strictly embodied in "characters" as we normally think of them—that is, as representations of humans. Using the broadest definition, all computer programs that perform actions that are perceived by people can be said to exhibit agency in some form. The real argument is whether that agency is a "free-floating" aspect of what is going on, or whether it is captured in "entities"—coalesced notions of the sources of agency.

The answer, I believe, is that even when representations do not explicitly include such entities, their existence is implied. At the grossest level, people simply attribute agency to the computer itself ("I did this, and then the computer did that"). They also attribute agency to application programs ("My word processor trashed my file"). They often distinguish between the agency of system software and applications ("Multifinder crashed Excel"). They attribute agency to smaller program elements and/or their representations ("The spelling checker in my word processor found an error").

(i.e., what is being imi-
›ns: "We maintain that
action, and that it is
it imitates the personal
ma, *character* may be
tions, and choices that,
entities. Those entities
l in the plot. This defi-
n.
›n to human-computer
that agents are in fact
nd, that there are func-
veen such agents and

agent is one who takes
of the possibility of a
without action cannot
sts that agency as part
embodied in "charac-
-that is, as representa-
efinition, all computer
e perceived by people
form. The real argu-
ee-floating" aspect of
ired in "entities"—coa-

when representations
ies, their existence is
imply attribute agency
then the computer did
to application pro-
my file"). They often
system software and
xcel"). They attribute
id/or their representa-
rd processor found an

In social and legal terms, an agent is one who is empowered to act on behalf of another. This definition has been used as part of the definition of agents in the mimetic world. It implies that, beyond simply performing actions, computer-based agents perform a special kind of actions—namely, actions undertaken on behalf of people. It therefore also implies that some sort of implicit or explicit communication must occur between person and system in order for the person's needs and goals to be inferred. I think that this definition is both too narrow and too altruistic. There may be contexts in which it is useful to create a computer-based agent whose "goals" are orthogonal or even inimical to those of human agents—for instance, in simulations of combat or other situations that involve conflicting forces. Agents may also work in an utterly self-directed manner, offering the results of their work up to people after the fact.

Other criteria that have been applied to interface agents (such as anthropomorphism) will be treated in the section on agents in Chapter 5. For now, we will use the broader definition of agents to apply to human-computer activity: entities that can initiate and perform actions. Like dramatic characters, they consist of bundles of traits or predispositions to act in certain ways.

Traits circumscribe the actions (or kinds of actions) that an agent has the capability to perform, thereby defining the agent's potential. There are two kinds of traits: traits that determine how an agent can act (internal traits) and traits that represent those internal predispositions (external traits). People must be given cues by the external representation of an agent that allow them to infer its internal traits. Why? Because traits function as a kind of *cognitive shorthand* that allows people to predict and comprehend agents' actions [see Laurel, 1990]. Inferred internal traits are a component of both dramatic probability (an element of plot, as described in Chapter 3) and "ease of use" (especially in terms of the minimization of human errors) in human-computer systems. Part of the art of creating both dramatic characters and computer-based agents is the art of selecting and representing external traits that accurately reflect the agent's potential for action.

Aristotle outlined four criteria for dramatic characters that can also be applied to computer-based agents [*Poetics*, 1454a, 15–40] The first criterion is that characters be "good" (sometimes translated as "virtuous"). Using the Aristotelian definition of "virtue," good characters are those who successfully fulfill their function—that is, those who successfully formulate thought into action. Good characters *do* (action) what they *intend* to do (thought). They also do what their creator intends them to do in the context of the whole action. The second criterion is that characters be "appropriate" to the actions they perform; that is, that there is a good match between a character's traits and what they do. The third criterion is the idea that characters be "like" reality in the sense that there are causal connections between their thoughts, traits, and actions. This criterion is closely related to dramatic probability. The fourth criterion is that characters be "consistent" throughout the whole action; that is, that a character's traits should not change arbitrarily. The mapping of these criteria to computer-based agents is quite straightforward.

Finally, we need to summarize the formal and material relationships between character and the elements above and below it in the hierarchy. Formal causality suggests that it is action, and action alone, that *shapes* character; that is, a character's traits are dictated by the exigencies of the plot. To include traits in the representation that are not manifest in action violates this principle. Material causality suggests that the stuff of which a character is made must be present on the level of thought and, by implication, language and enactment as well. A good example is the interface agent, Phil, who appears in an Apple promotional video entitled "The Knowledge Navigator" (© 1988 by Apple Computer, Inc.). In the original version, Phil was portrayed by an actor in video format. He appeared to be human, alive, and responsive at all times. But because he behaved and spoke quite simply and performed relatively simple tasks, many viewers of the video complained that he was a stupid character. His physical traits (high-resolution, real-time human portrayal) did not match his language capabilities, his thoughts, or his actions (simple

r dramatic characters
based agents [*Poetics,*
: characters be "good"
Using the Aristotelian
are those who success-
e who successfully for-
acters *do* (action) what
do what their creator
the whole action The
"appropriate" to the
iere is a good match
iey do. The third crite-
e" reality in the sense
ween their thoughts,
ely related to dramatic
characters be "consis-
t is, that a character's
mapping of these cri-
straightforward.
formal and material
e elements above and
lity suggests that it is
racter; that is, a char-
encies of the plot. To
it are not manifest in
ausality suggests that
iust be present on the
guage and enactment
ace agent, Phil, who
video entitled "The
le Computer, Inc.). In
by an actor in video
and responsive at all
oke quite simply and
viewers of the video
er. His physical traits
rayal) did not match
or his actions (simple

tasks performed in a rather unimaginative manner). In a later version, Phil's representation was changed to a simple line-drawn cartoon character with very limited animation. People seemed to find the new version of Phil much more likable. The simpler character was more consistent and more appropriate to the action.

The Whole Action

Representations are normally thought of as having objects, even though those objects need not be things that can or do exist in the real world. Likewise, plays are often said to represent their characters; that is, *Hamlet* is a representation of the king of Denmark, and so on. In the Aristotelian view, the object of a dramatic representation is not character but action; *Hamlet* represents the action of a man attempting to discover and punish his father's murderer. The characters are there because they are required in order to represent the action, and not the other way around. An action is made up of incidents that are causally and structurally related to one another. The individual incidents that make up *Hamlet*—Hamlet fights with Laertes, for instance—are only meaningful insofar as they are woven into the action of the mimetic whole. The form of a play is manifest in the pattern created by the arrangement of incidents within the whole action.

Another definitional property of plot is that the whole action must have a beginning, a middle, and an end. The value of beginnings and endings is most clearly demonstrated by the lack of them. The feeling produced by walking into the middle of a play or movie or being forced to leave the theatre before the end is generally unpleasant. Viewers are rarely happy when, at the end of a particularly suspenseful television program, "to be continued" appears on the screen. My favorite Macintosh example is an error message that I sometimes encounter while running Multifinder: "Excel (or some other application) has unexpectedly quit." "Well," I usually reply, "the capricious little bastard!" Providing graceful beginnings and endings for human-computer activities is most often a nontrivial problem—how to "jump-start" a database engine, for example, or how to complete a network communications session. Two rules of thumb for good beginnings is that the potential for action in that particular universe is effectively laid out, and that the first incidents in the action set up promising lines of probability for future actions. A good ending provides not only completion of the action being represented but also the kind of emotional closure that is implied by the notion of catharsis, as discussed in Chapter 3.

A final criterion that Aristotle applied to plot is the notion of magnitude:

> To be beautiful, a living creature, and every whole made up of parts, but also be of a certain definite magnitude. Beauty is a matter of size and order . . . Just in the same way, then, as a beautiful whole made up of parts, or a beautiful living creature, must be of some size, but a size to be taken in by the eye, so a story or Plot must be of some length, but of a length to be taken in by the memory [*Poetics*, 1450b, 34–40]

The action must not be so long that you forget the beginning before you get to the end, since you must be able to perceive it as a whole in order to fully enjoy it. This criterion is most immediately observable in computer games, which may require you to be hunched over a keyboard for days on end if you are to perceive the whole at one sitting, a feat of which only teenagers are capable. Similar errors in magnitude are likely to occur in other forms, such as virtual reality systems, where the raw capabilities of a system to deliver material of seemingly infinite duration is not yet tempered by a sensitivity to the limits of human memory and attention span, or to the relationship of beauty and pleasure to duration in time-based arts.

Problems in magnitude can also plague other, more "practical" applications as well. If achievable actions with distinct beginnings and ends cannot occur within the limits of memory or attention, then the activity becomes an endless chore. On the contrary, if the granularity of actions is too small and those actions cannot be grouped into more meaningful, coherent units (such as a word processor that only lets you type or a spreadsheet that only lets you add up columns of numbers), then the activity becomes an endless stream of meaningless chores. These problems are related to the *shape* of the action as well as its magnitude, the first subject to be treated in Chapter 3.

The notion of beauty that drives Aristotle's criterion of magnitude is the idea that made things, like plays, can be organic wholes—that the beauty of their form and structure can approach that of natural organisms in the way the parts fit perfectly together. In this context, he expresses the criterion for inclusion of any given incident in the plot or whole action:

> An imitation of an action must represent one action, a complete whole, with its several incidents so closely connected that the transposal or withdrawal of any one of them will disjoin and dislocate the whole. For that which makes no perceptible difference by its presence or absence is no real part of the whole [*Poetics*, 1451a, 30–35]

If we aim to design human-computer activities that are—dare we say—*beautiful*, this criterion must be used in deciding, for instance, what a person should be required to do, or what a computer-based agent should be represented as doing, in the course of the action.

In this chapter, we have described the essential causes of human-computer activity—that is, the forces that shape it—and its qualitative elements. In the next chapter, we will consider the orchestration of action more closely, both in terms of its structure and its powers to evoke emotional and intellectual response.

Characteristics of Good Constraints

May's analysis suggests that constraints—limitations on the scope and nature of invention—are essential to creativity. Certainly, some constraints on the choices and actions that may be expressed by people are technically essential to any human-computer activity. The question is how those constraints should be determined and expressed. The standard techniques for introducing constraints—instructions, error messages, or dialogue boxes, for instance—are almost always destructive of our *engagement* in the activity by forcing us to "pop out" of the mimetic context into a metacontext of interface operations.

Constraints can be either *explicit* or *implicit*. Explicit constraints, as in the case of menus or command languages, are undisguised and directly available. When we are in doubt about the "legality" of certain choices or actions, we should be able to find the rules and protocols of a system straightforwardly expressed, either in the manual, or in an on-line "help" facility. Implicit constraints, on the other hand, may be inferred from the behavior of the system. We can identify implicit constraints when a system fails to allow us to make certain kinds of choices. There is no way, for example, to negotiate with the enemy in most combat-based computer action games. In most word processors, there is no facility for "drawing" or "painting" images in a document, and the absence of drawing tools makes it less likely that we will think of doing so. Some constraints have both implicit and explicit qualities. In Microsoft Excel, for example, menu-based operations are not selectable and the document cannot be closed until the current item has been properly entered on the spreadsheet. If we attempt such an "illegal" maneuver, nothing happens at all. We may infer from this "nonbehavior" that we must do something else or do something differently.

Explicit constraints can be used without damage to engagement if they are presented before the action begins. A good example is the determination and expression of rules in child's play, which occurs before play actually begins and cre-

aints

:raints—limitations on the
ire essential to creativity.
: choices and actions that
:chnically essential to any
estion is how those con-
expressed. The standard
aints—instructions, error
stance—are almost always
: activity by forcing us to
to a metacontext of inter-

t or *implicit*. Explicit con-
command languages, are
When we are in doubt
:es or actions, we should
ls of a system straightfor-
ianual, or in an on-line
n the other hand, may be
system. We can identify
ails to allow us to make
no way, for example, to
combat-based computer
rs, there is no facility for
n a document, and the
less likely that we will
have both implicit and
el, for example, menu-
nd the document cannot
een properly entered on
an "illegal" maneuver,
r from this "nonbehav-
or do something differ-

d without damage to
ore the action begins. A
d expression of rules in
actually begins and cre-

ates a contract binding the participants to behave within certain constraints. Once the action has started, however, explicit constraints often prove disruptive—an argument about the rules can ruin a perfectly good session of cowboys and Indians ("Wait a minute—who says Indians can only be killed with silver bullets?")[5] Implicit constraints are preferable during the course of the action, simply because the means for expressing them are usually less intrusive than those used for explicit constraints.

Constraints may also be characterized as *extrinsic* or *intrinsic* to the mimetic action. Extrinsic constraints have to do, not with the mimetic context, but with the context of the person as operator of the system. Avoiding the "reset" and "escape" keys during play of a game has nothing to do with the game world and everything to do with the behavior of the computer. Playing an improvised scene without the use of language has nothing to do with the dramatic action of the scene but is an extrinsic constraint designed to improve the actors' gestural acuity—a different context than the mimetic one. Extrinsic constraints have been used successfully in a variety of sports and other disciplines to distract the part of consciousness that can interfere with performance [see Gallwey, 1976]. The technique is generally inappropriate in human-computer activity, however, because it sets up a secondary context that demands part of a person's attention.

Extrinsic constraints can be made to appear intrinsic when they are expressed in terms of the mimetic context. If the "escape" key is defined as a self-destruct mechanism, for

[5]An interesting exception is the ongoing process of rule-making and enforcement that is sometimes an element in children's play—a sort of *metagame* that provides its own distinct pleasures. A similar metagame occurs in the theatre when stagehands and "real people" wander in and out of the action, as in some of the plays of Christopher Durang and Thornton Wilder, or in certain productions of Brecht. Seen in this way, the metagame is also mimetic, and the actors are merely performing the roles of "real people" as well as portraying other dramatic characters. Because it is mimetic, this is a "false" context shift, much like a play within a play, or a dream in which one has false awakenings. Such metagames or metaplays do not violate engagement, but enhance it through the same means as the mimetic "core" activity.

instance, the constraint against pressing it in the course of flying a mimetic spaceship is intrinsic to the action We need not shift gears to consider the effect of the key upon the computer or the game program. Expressing constraints in this manner preserves the contextual aspect of engagement.

Another good example of well-contextualized extrinsic constraints is the "borders" option in Microsoft Word 4.0. Once we have reached the screen in which borders can be created, we see a graphical, direct-manipulation interface We can move lines of various types around in a representation of a document until the desired border style has been achieved Unfortunately, getting to this neat little border construction screen is problematic at best We must select "paragraph" from the "format" menu and then click on a "borders" button on the paragraph screen, a logical hierarchy that is forced upon the activity. Directness is obliterated by the operational overhead created by this scheme. A second problem is that buttons defining border styles coexist with the direct-manipulation display on the borders screen and have doppelgänger functionality. Nested within the button problem is the additional difficulty that the top three buttons invoke actions, while the fourth is also used inconsistently to indicate a state or mode that we may have entered by combining other elements via direct manipulation. No wonder people have difficulty with it (My husband, a senior engineer at Apple Computer, hasn't figured it out.)

The mimetic context itself can be teased apart, especially in task-oriented activities. Bødker [1989] uses the interface for creating footnotes in Microsoft Word as an example of how a task that is part of the general context of document creation is nevertheless extrinsic to the subcontext of writing the text. The WYSIWYG-style interface for authoring does not work for footnote creation; the flow of the authoring activity is disrupted by the dialogue box that requires the author to specify the form of a footnote before actually writing it. The point here is that an activity that may be part of the mimetic whole can be seen to inject extrinsic constraints if it is staged or represented poorly. Bødker's example can be seen as a failure in

ng it in the course of fly-
) the action. We need not
e key upon the computer
nstraints in this manner
agement.
contextualized extrinsic
in Microsoft Word 4.0.
which borders can be cre-
ipulation interface. We
nd in a representation of
style has been achieved
ttle border construction
ust select "paragraph"
k on a "borders" button
ierarchy that is forced
rated by the operational
second problem is that
with the direct-manipu-
and have doppelgänger
on problem is the addi-
buttons invoke actions,
tently to indicate a state
by combining other ele-
onder people have diffi-
ior engineer at Apple

teased apart, especially
39] uses the interface for
as an example of how a
of document creation is
ext of writing the text.
thoring does not work
uthoring activity is dis-
es the author to specify
y writing it. The point
rt of the mimetic whole
nts if it is staged or rep-
n be seen as a failure in

the arrangement of incidents, with the by-product of disrupting engagement.

Constraints should be applied without shrinking our perceived range of freedom of action: Constraints should limit, not what we can do, but what we are likely to think of doing. Such *implicit* constraints, when successful, eliminate the need for explicit limitations on our behavior. Context is the most effective medium for establishing implicit constraints. The ability to recognize and comply with implicit, context-based constraints is a common human skill, exercised automatically in most situations and not requiring concentrated effort or explicit attention. It is the same skill that we use to determine what to say and how to act when we interact with a group of unfamiliar people—at a party, for instance. The limitations on behavior are not likely to be explicitly known or consciously mulled over; they arise naturally from our growing knowledge of the context. The situational aspects of the current context and the way in which they have evolved over the course of the action establish dramatic probability that influences our actions and expectations. In summary, then, constraints that are implicit and intrinsic to the mimetic context are least destructive of engagement and other qualities of experience, although explicit and extrinsic constraints can be successfully employed if they frame rather than intrude upon the action.

Establishing Constraints Through Character and Action

Since human-computer activities are dramatic in nature, it is reasonable to look for guidance in the development of constraints to other dramatic forms: theatrical performance and improvisation. In the theatre, the actor is constrained in the performance of a character primarily by the script, and secondarily by the director, the accoutrements of the theatre (including scenic elements, properties, and costumes), and the performances of fellow actors. The actor must work within exacting constraints, which dictate the character's every word, choice, and action. In spite of these narrow limits, the actor still has ample latitude for individual creativity. In the words of acting teacher Michael Chekhov:

> Every role offers an actor the opportunity to improvise, to collaborate and truly co-create with the author and director. This suggestion, of course, does not imply improvising new lines or substituting business for that outlined by the director. On the contrary. The given lines and the business are the firm bases upon which the actor must and can develop his improvisations. *How* he speaks the lines and *how* he fulfills the business are the open gates to a vast field of improvisation. The "hows" of his lines and business are the ways in which he can express himself freely [Chekhov, 1953].

The value of limitations in focusing creative activity is recognized in the theory and practice of theatrical improvisation. Constraints on the choices and actions of actors improvising characters are probably most explicit in the tradition of *commedia dell'arte*. Stock characters and fixed scenarios provide *formal* constraints on the action in that they affect the actor's choices through formal causality. Conventionalized costumes for each character, a standard collection of scenic elements and properties, and a repertoire of *lazzi* (standard bits of business) provide *material* constraints on the action. Likewise, people who are engaged in computer-based mimetic activities are subject to formal and material constraints.

Constraints expressed on the level of character may function as either material or formal constraints, depending upon how they affect the action. Traits and predispositions provide materials from which action is formulated. They also give form to thought, language, and enactment.

Specific objectives or motivations on the part of the human agent(s) constrain the action in both games and task-oriented applications. Highest-level objectives (or, in the lingo of method acting, "super-objectives") are usually known explicitly before the action begins. Computer games provide obvious examples. In *Star Raiders*, for instance, the objective of the human character is to destroy all the Zylons in several quadrants of the galaxy. In *Zork*, the objective is to gather all the treasures in the maze and return them to the trophy room. In

Parker Brothers' *The Empire Strikes Back*, the objective is to destroy as many of the Imperial Walkers as possible before they reach the power plant and blow up the planet. However, as science fiction author Harlan Ellison observed in an unpublished review of the game, it is not possible to meet that goal because the bad guys just keep getting better—an affliction shared by many video games. "The lesson," moans Ellison, "is the lesson of Sisyphus. You cannot win. You can only waste your life struggling and struggling, getting as good as you can be, with no hope of triumph." One might speculate that this incredibly frustrating feature of game design contributed to the decline of the video-game genre in 1983 and 1984.

In task-oriented applications, the choice of the application itself indicates an awareness of super-objectives: Word processors are used to create documents, drawing programs are used to create graphical compositions, etc. However, as applications become more integrated and flexible (or are replaced by "environments" as conceptual units in the human-computer universe), people's goals cannot be so readily inferred by simply noticing what applications they launch. Increasingly, systems will need to employ either explicit conversations with people to determine task objectives or implicit user-modeling techniques to infer objectives from behavior, as discussed below.

The way in which a computer-based system responds to a person can help to narrow down and flesh out the person's objectives, and it can also lead to fairly predictable kinds of action. People who play computer games find that their objectives are rapidly elaborated as the action progresses, through the workings of dramatic probability. As the Zylons close in on a friendly starbase in a game of *Star Raiders*, for instance, we discover that we must develop a strategy for preventing the starbase from being surrounded or captured in order to fulfill our super-objective; otherwise, the action will be prematurely terminated. Hence a whole series of fairly predictable, causally related choices on the part of the human character is stimulated by the single super-objective that has been expressed as a formal constraint at the beginning of the game.

When a person's super-objective is not clear-cut from the point of view of the computer (as in the case of integrated applications or environments), techniques could be devised for inferring it, without imposing explicit limitations. For instance, the system might notice what tools we select, how often, and in what combinations. This noticing behavior could enable the formulation of a hypothesis about our immediate objective, leading to the automatic tailoring of the environment and tools. As other tool-sets are used, the system could formulate a more global hypothesis about the whole activity. If the activity is such that inferences of this type have a low probability of being accurate or a high probability of being annoying or confusing, explicit dialogue could be employed. Explicit dialogues with the "system" about our intentions (as in the Bødker example earlier in this chapter) can be tedious and disruptive. Conversely, casting the conversational partner as an agent *character* (as opposed to the amorphous "system") can provide contextual smoothing. For instance, in a research project at Apple Computer, Allen Cypher developed a program that can sense repetitive activity (Figure 4.1). When the program notices that a person is doing something over and over (such as adding numbers or animating an object frame-by-frame), an agent named "Eager" appears and offers up a plan for completing the activity. This highly animated dialogue, coupled with the program's power to clarify the person's objectives and help to achieve them, provides a very attractive means for introducing formal constraints [Cypher, 1990 and 1991].

In activities where we interact with more contentful worlds (such as simulation environments), a system might utilize templates to ascertain our objectives. A system could notice what we are doing, select a template that most nearly matches our apparent motivation, and adjust the system's contributions to the action accordingly. For instance, a person interacting with a simulation of a space station might be trying to redesign it or trying to learn how to operate its controls, or perhaps to experience the environment under various conditions. There is the beginning of a "plot" implicit in each of these goals, and the system could assist in bringing that plot to life. The system's reasoning might go something like this: "If he is doing *x*, then he probably wants *y*. Therefore incidents *a*, *b*, and *c* are likely to cause him to make choices *d*, *e*, and *f*." A template would contain the candidate objective and a set of incidents that would be likely to elicit certain responses based on that objective. The system might then use the person's actual responses as a measure of the accuracy of its initial inference and switch templates if necessary. When it had established a person's objective with a high degree of confidence, the system might kick off a specific scenario by enacting a predesigned *inciting incident*. Furthermore, information about individual people could augment such templates, tailoring the action to such traits as a person's job and skills as well. Such templates would function as recipes for the formulation of action and could be used to both predict and constrain a person's behavior.

Material constraints may be provided implicitly through exposition presented during the action. People discover "physical" and behavioral aspects of a mimetic world, characters, and past events in this manner. To ensure that people become familiar with such elements early on, the designer of a simulation-based activity may wish to delay active human participation until the bulk of the exposition has been presented. The "attract mode" of many arcade games performs this expository function. The notion of "guided tours" employed by Apple Computer attempts to exhibit the properties and behaviors of key objects in narrated simulations, a kind of pre-activity exposition. In the Guides project at Apple, the represented character actions of slumping, doing other things, and falling asleep indicate through enactment a guide's reduced level of interest in the piece of information that is currently displayed. This expository behavior implies that the guide will have little to suggest in the way of related items.

The kinds of actions that a person can take in representational worlds are also constrained by the capabilities of the input and output devices used in the system. By constraining what—or whether—people may see, hear, and say, the system

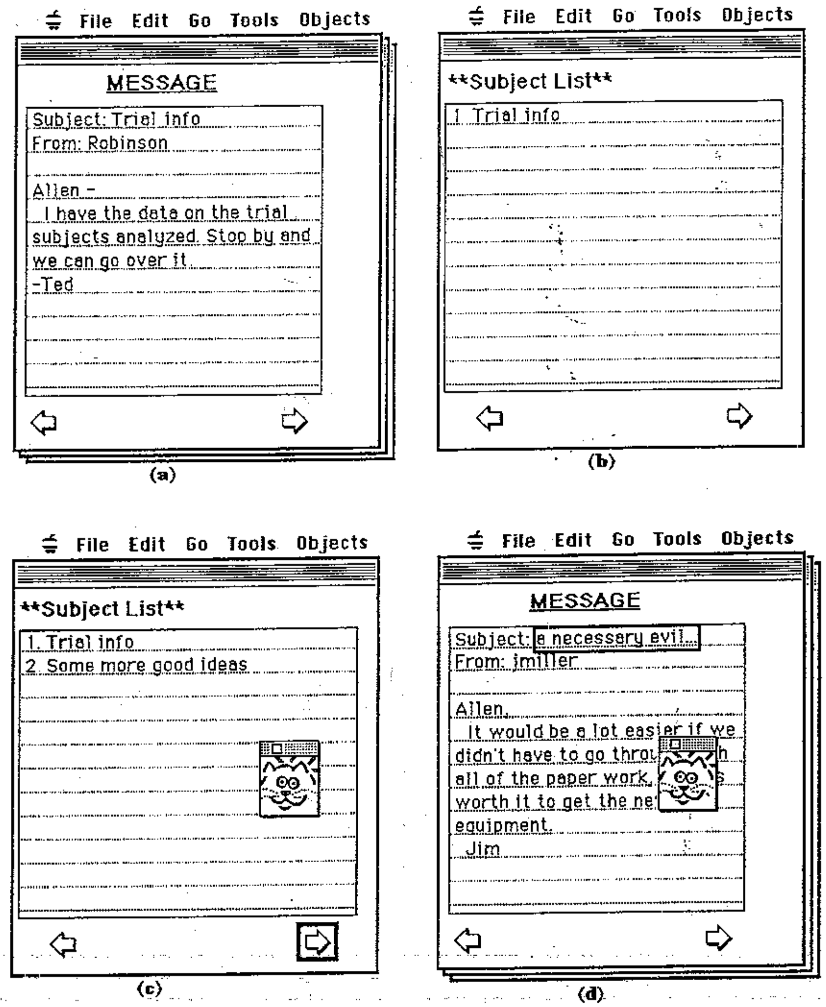
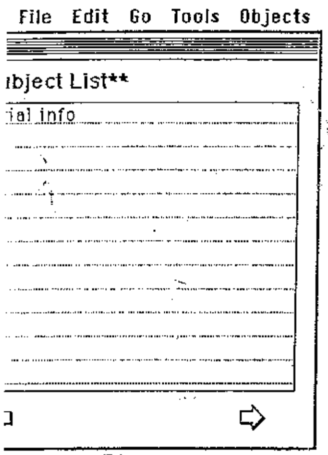
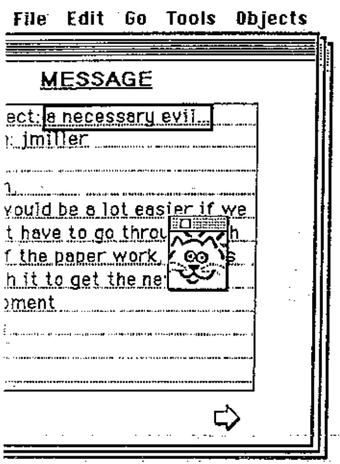


Figure 4.1 "Eager" is an agent-like entity that notices patterns of action and tries to create programs to continue those patterns Here, the user has a stack of message cards (a) and she wants to make a list of the subjects of the messages. She copies the first subject and pastes it into a new "Subject List" card (b) Then she goes to the second message, copies its subject, and adds it to the list. At this point, the Eager icon pops up (c), since Eager has detected a pattern in the user's actions. Eager also highlights the right-arrow

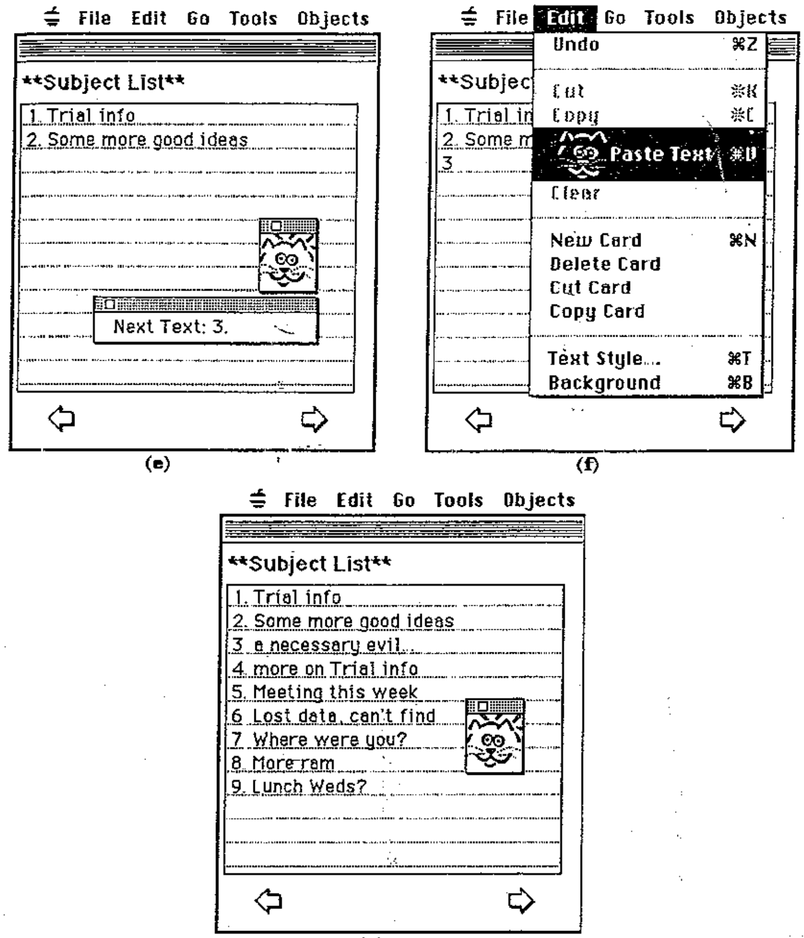


(b)



(d)

ı that notices patterns of
continue those patterns.
rds (a) and she wants to
. She copies the first sub-
" card (b). Then she goes
and adds it to the list. At
nce Eager has detected a
ighlights the right-arrow



(e) (f) (g)

button in green (c), since it anticipates that the user will click here next. Eager continues anticipating that the user will navigate to the third message, select (d) and copy its subject, go to the Subject List, click at the start of the third line (e), type "3" (f), and then paste in the subject (g). The user is now confident that Eager knows what to do, so she clicks on the Eager icon and it completes the task automatically (h).

may implicitly constrain their thoughts, choices, and actions. In systems that employ simple language parsers, for instance, words that are unknown to the system cannot effect any change in the world; choices and actions that are represented by unknown words cannot be performed.

It is difficult to avoid such a disruptive effect when people are allowed or encouraged to make choices that they cannot effectively express to the system. For instance, the text adventure games developed by Infocom are presented entirely in a verbal mode. People are encouraged to use natural language to express their choices, and so they expect words to work. They have no clue to tell them which words are unknown to the system *except the experience of failure*. On the other hand, given the text-based nature of the game and the equipment that it is usually run on, people are never encouraged to attempt to express themselves through gestures or physical actions. The absence of visual and kinesthetic modes in the system is accepted as a given, and the resulting constraints are unobtrusive. Such constraints are extrinsic to the action but may be utilized effectively if they are presented simply and explicitly, or if they are integrated into the mimetic context (for example, "this ship is not equipped for voice communication").

Generally, the more modes that are present in the interface (verbal, visual, auditory, etc.), the more complex the system must be in order to handle the reception and interpretation of a wide variety of inputs and to formulate and orchestrate its responses. Constraining people through limitations on input and output capabilities becomes less effective as the number of modes in the interface increases; separate sets of constraints for each mode serve to confuse and frustrate people. In a multimodal interface environment, intrinsic formal and material constraints are therefore preferable to those based on the technical characteristics of the interface.

Engagement: The First-Person Imperative

In the foregoing discussion, *engagement* was held up as a desirable—even essential—human response to computer-mediated activities. Engagement has cognitive components,

ights, choices, and actions
guage parsers, for instance,
system cannot effect any
ctions that are represented
rmed.
ruptive effect when people
e choices that they cannot
r instance, the text adven-
are presented entirely in a
l to use natural language to
xpect words to work. They
ds are unknown to the sys-
the other hand, given the
e equipment that it is usu-
aged to attempt to express
sical actions. The absence
he system is accepted as a
are unobtrusive. Such con-
t may be utilized effective-
l explicitly, or if they are
(for example, "this ship is
on")
are present in the interface
nore complex the system
tion and interpretation of
nulate and orchestrate its
ugh limitations on input
s effective as the number
parate sets of constraints
rustrate people. In a mul-
nsic formal and material
those based on the tech-

rson Imperative

ment was held up as a
response to computer-
s cognitive components,

but it is primarily understood as an emotion. Why should we demand that all human-computer activities elicit this particular emotional response? What is its nature, and what is its value? What can designers do to guarantee that it occurs?

Engagement, as I use the concept in this book, is similar in many ways to the theatrical notion of the "willing suspension of disbelief," a concept introduced by the early nineteenth-century critic and poet Samuel Taylor Coleridge.[6] It is the state of mind that we must attain in order to enjoy a representation of an action. Coleridge believed that any idiot could see that a play on a stage was not real life. (Plato would have disagreed with him, as do those in whom fear is induced by any new representational medium, but that is another story.) Coleridge noticed that, in order to enjoy a play, we must temporarily suspend (or attenuate) our knowledge that it is "pretend." We do this "willingly" in order to experience other emotional responses as a result of viewing the action. When the heroine is threatened, we feel a kind of fear for and with her that is recognizable as fear but different from the fear we would feel if we were tied to the railroad tracks ourselves. *Pretending that the action is real* affords us the thrill of fear; *knowing that the action is pretend* saves us from the pain of fear. Furthermore, our fear is flavored by the delicious expectation that the young lady will be saved in a heroic manner—an emotional response that derives from knowledge about the *form* of melodrama.

The phenomenon that Coleridge described can be seen to occur almost identically in drama and computer games, where we feel for and with the characters (including *ourselves* as characters) in very similar ways. Yes, someone might cry, but manuscripts and spreadsheets aren't pretend! Here we must separate the activity from its artifacts. The *representation* of a manuscript or spreadsheet as we manipulate it on the screen is in fact pretend, as compared to physical artifacts like data files (in the computer's memory or on a storage medium) and hard copy. The artifacts are real (as are actors, lighting

[6]For an analysis and thorough bibliography of Coleridge's criticism, see Allen and Clark [1962], pp. 221–239.

instruments, and reels of motion-picture film), but the rules involved in working with the *representations* (plays or human-computer activities) must subsume the knowledge, at some level, that they are representations. Why? First, because the fact that they are representations is the key to understanding what we can do with them. Second, because their special status as representations affects our emotions about them, enabling experiences that are, in the main, much more pleasurable than those we feel in real life. The distinguishing characteristic of the emotions we feel in a representational context is that there is *no threat of pain or harm in the real world.*

The key to applying the notion of "willing suspension of disbelief" to representational activities that have real-world artifacts is to ensure that the likelihood of *unintentional* effects on those artifacts approaches zero. The other day I experienced a power failure while I was working on this manuscript. I had learned to save my work often, but losing just a few paragraphs evoked plenty of unpleasant real-world emotion. Quite simply, *my system should never have let that happen.* My first word processor, although it lacked nearly all of the features that I appreciate in the one I use today, had a fail-safe feature that took the opportunity to automatically save an active file whenever there was a pause in the input stream—on the average, about every seven seconds. For people who use systems without such a feature, a power outage can be a context shift of the worst possible kind. Such interruptions to the flow of representational activity must be avoided if the powers of representational media are to be preserved. Saving my work has receded from an obsession to a kind of tic, but it shouldn't be there nipping at my subconscious at all.

Furthermore, engagement entails a kind of playfulness—the ability to fool around, to spin out "what if" scenarios. Such "playful" behavior is easy to see in the way that people use spreadsheets and word processors. In my house-buying example in the previous chapter, I played around with different scenarios for making trade-offs in my purchase decision. The key quality that a system must possess in order to

ɔicture film), but the rules
ɜentations (plays or human-
e the knowledge, at some
ʒ Why? First, because the
: the key to understanding
l, because their special sta-
ɪr emotions about them,
ɪe main, much more plea-
e The distinguishing char-
a representational context
n in the real world
of "willing suspension of
ities that have real-world
ɔod of *unintentional* effects
. The other day I experi-
was working on this
ny work often, but losing
/ of unpleasant real-world
ɔuld never have let that hap-
ɪgh it lacked nearly all of
ɔne I use today, had a fail-
y to automatically save an
: a pause in the input
y seven seconds. For peo-
a feature, a power outage
possible kind. Such inter-
ational activity must be
ional media are to be pre-
d from an obsession to a
e nipping at my subcon-

ntails a kind of play-
:o spin out "what if" sce-
ɪsy to see in the way that
processors. In my house-
ɔter, I played around with
offs in my purchase deci-
must possess in order to

foster this kind of engagement is reversibility—that is, the ability to take something back. What if I failed to save a copy of my spreadsheet before I monkeyed around with a scenario that turned out to be disastrous? What if that scenario altered a significant amount of my data? The theory of hypertext suggests one solution, where various stages of a "document" (or, more correctly, an activity) can be saved and linked to the current version. This solution is unsatisfactory in that it is likely (at least in contemporary hypertext systems) to create a bewildering proliferation of documents. I don't really want to page back through versions of my work; I want to turn back the clock. The dimension of change is best represented through time, not fixed states. A simple chrono-scrollbar would suffice. Yes, the implementation is hard, but the hardest part is probably visualizing the appropriate representation in the first place.

I notice how word processing has changed my writing style. Now I am able to move chunks of text (roughly corresponding to ideas or elements in an argument) around within a document. I can more easily experiment with the visual components of the information I am creating by changing fonts and paragraph styles. But there is nothing sadder or more disruptive than seeing the message, `"Can't Undo."` With a typewriter, I still had the hard copy and a handy bottle of correcting fluid. Here again, the notion of document creation as an activity unfolding through time is superior to a notion of independent operations on an artifact of which one must remember to take snapshots.

Engagement is what happens when we are able to give ourselves over to a representational action, comfortably and unambiguously. It involves a kind of complicity. We agree to think and feel in terms of both the content and conventions of a mimetic context. In return, we gain a plethora of new possibilities for action and a kind of emotional guarantee. One reason why people are amenable to constraints is the desire to gain these benefits.

Engagement is only possible when we can rely on the system to maintain the representational context. A person should never be forced to interact with the system *qua* system; indeed, any awareness of the system as a distinct, "real" entity would explode the mimetic illusion, just as a clear view of the stage manager calling cues would disrupt the "willing suspension of disbelief" for the audience of a traditional play. Engagement means that a person can experience a mimetic world directly, without mediation or distraction. Harking back to the slogan, "the representation is all there is," we can see that interface designers are often engaged in the wrong activity—that is, representing what the *computer* is doing. The proper object of an "interface" representation is what the *person* is doing with the computer—the action. Thinking about things this way automatically avoids the trapdoors into meta-level transactions with "the system."

Characteristics of First-Person Experience

Another way to describe a person's involvement in the representational context of human-computer activity is as a *first-person* experience [see Laurel, 1986b]. In grammar, the personness of pronouns reflects where one stands *in relation to* others and the world. Most movies and novels, for example, are third-person experiences; the viewer or reader is "outside" the action and would describe what goes on using third-person pronouns: "First he did this, then they did that." Most instructional documents are second-person affairs: "Insert Tab A into Slot B"; "Honor your father and your mother." Operating a computer program is all too often a second-person experience: A person makes imperative statements (or pleas) to the system, and the *system* takes action, completely usurping the role of agency.

Agency is a key component of first-person experience. Although we may describe experiences in which we are not an agent using first-person pronouns (I saw this, I smelled that), the ability to *do* something sooner or later emerges as a criterion. On the one hand, doing very simple things can be an expression of agency—looking around, for instance, or reaching out and touching something (such simple types of agency are often responsible for the "breakthrough" experiences reported by many people who have used contemporary virtual-reality systems). On the other hand, doing something relatively complex in an indirect or mediated way may not have a first-person feel. In the early days of computing, programmers would submit a program and data on punched cards and come back to pick up the results a day or two later. Although they were telling the computer what to do quite exactly, during the hours of waiting for the computer to "crunch" those programmers were not experiencing a feeling of agency. Today, imploring a system to do something in a highly constrained, formal language can engender a similar feeling that somebody (or something) else is in control.

This is not to say that people cannot experience agency when there are computer-based agents in the representational environment. Agents that are well characterized and amenable to dialogue and collaboration can give a person the sense of being one of several agents in a complex action. An agent can be a mentor or a dictator, a liberator or a jailor. The difference is in the person's experience of *agency*—the power to take action—whether the context includes other agents or not.

First-person sensory qualities are as important as the sense of agency in creating satisfying human-computer experiences. Quite simply, the experience of first-person participation tends to be related to the number, variety, and integration of sensory modalities involved in the representation. The underlying principle here is *mimetic*; that is, a human-computer experience is more nearly "first-person" when the activity it represents unfolds in the appropriate sensory modalities. The intuitive correctness of this notion is witnessed by the direction of technical evolution in the areas of simulators and games—toward higher resolution graphics and faster animation, greater sound capabilities, motion platforms, and mimetic input devices like force-feedback controllers. In task-oriented applications, new technologies are allowing researchers to replace indirect or symbolic representations and manipulations with direct, concrete ones—for example, physically pointing or speaking as opposed to typing, spatial and graphical representation of data as opposed to textual representation, etc. (see Color Plates II and III). Likewise, the evolution of natural-language interfaces is beginning to replace the elaborate conventions of menu-based and command-based systems with systems that employ language in ways that are mimetic of real-world activities like conversation and question-and-answer dialogues [see, for instance, Schmandt, 1985].[7]

Sensory first-personness is not limited to the system's "output"; it must include the modalities that people can employ when they take action in mimetic worlds. Since it is all one representation, the desire for symmetry between "input" and "output" modalities is strong. Engagement is disrupted when my machine talks to me (especially if it asks me a question) and I can't *talk* back.[8] Furthermore, the real-world relationships among modalities affect our expectations in representational worlds that include them; for instance, greater force applied to the throwing of an object should make it appear to go farther, surfaces that look bumpy should feel bumpy, and balloons make noise when they pop.

When we sit back and contemplate the complexity involved in creating first-person experiences, we are tempted to see them as a luxury, not a necessity. But we mustn't fall prey to the notion that more is always better, or that our task is the seemingly impossible one of emulating the sensory and experiential bandwidth of the real world. Artistic selectivity is the countervailing force—capturing what is essential in the most effective and economic way. A good line-drawn animation can sometimes do a better job of capturing the move-

[7]This paragraph is adapted from "Interface as Mimesis" [Laurel, 1986b].

[8]The Guides project provides a counter-example. The several guides do in fact speak at various points in the program. The desire for I/O symmetry is mitigated by context: The guides are cast as storytellers, embodying a conventional relationship in which one person talks and others listen without interruption. Even so, the product would undoubtedly be improved by the addition of voice input. But if and when it is implemented, then the content and conversational style of that input will need to measure up to those of the computer-enacted agents—a tall order.

sed to typing, spatial and opposed to textual repre- d III). Likewise, the evolu- s beginning to replace the sed and command-based language in ways that are e conversation and ques- for instance, Schmandt,

: limited to the system's dalities that people can imetic worlds. Since it is for symmetry between trong. Engagement is dis- e (especially if it asks me rthermore, the real-world t our expectations in rep- em; for instance, greater n object should make it look bumpy should feel en they pop.

nplate the complexity eriences, we are tempted sity. But we mustn't fall s better, or that our task nulating the sensory and rld. Artistic selectivity is what is essential in the good line-drawn anima- of capturing the move-

; Mimesis" [Laurel, 1986b].

iple. The several guides do in The desire for I/O symmetry as storytellers, embodying a rson talks and others listen uct would undoubtedly be : if and when it is implement- le of that input will need to agents—a tall order.

ments of a cat than a motion picture, and no photograph will ever capture the essence of light in quite the same way as the paintings of Monet. The point is that first-person sensory and cognitive elements are essential to human-computer activity. There is a huge difference between an elegant, selective multisensory representation and a representation that squashes sensory variety into a dense but monolithic glob of text.

Multisensory experience offers advantages that go beyond engagement, as media theorist Tom Bender describes:

> The kinds of information we receive from our surroundings are quite varied, and have different effects upon us. We obtain raw, direct information in the process of interacting with the situations we encounter. Rarely intensive, direct experience has the advantage of coming through the totality of our internal processes—conscious, unconscious, visceral and mental—and is most completely tested and evaluated by our nature. Processed, digested, abstracted second-hand knowledge is often more generalized and concentrated but usually affects us only intellectually—lacking the balance and completeness of experienced situations. . . . Information communicated as facts loses all its contexts and relationships, while information communicated as art or as experience maintains and nourishes its connections [Bender, 1973].

Bender's observations have been supported quite persuasively by the "multimedia revolution" in computer-based educational activities. Likewise, educational simulations (as opposed to tutorial or drill-and-practice forms) excel in that they present *experience* as opposed to *information*. Learning through direct experience has, in many contexts, been demonstrated to be more effective and enjoyable than learning through "information communicated as facts." Direct, multisensory representations have the capacity to engage people intellectually as well as emotionally, to enhance the contextual aspects of information, and to encourage integrated, holistic responses. This broad view of information subsumes artistic applications, as well as traditional knowledge representation. What Bender calls "direct experience," plus the experience of personal agency, are key elements of human-computer activity.

Empathy and Catharsis

In drama, we experience empathy with the characters; that is, we experience *vicariously* what the characters in the action seem to be feeling. Empathy is subject to the same emotional safety net as engagement—we experience the characters' emotions as if they were our own, but not quite; the elements of "real" fear and pain are absent. When we are agents in a mimetic action, our emotions about our *own* experiences partake of the same special grace. When I took my five-year-old daughter on the *Star Tours* ride at Disneyland (a wild ride combining flight simulator technology with *Star Wars* content), she turned to me in mid-shriek and shouted, "If this was *real*, I'd be *scared*!"

Even in task-oriented applications, there is more to the experience than getting something done in the real world, and this is the heart of the dramatic theory of human-computer interaction. Our focus is not primarily on how to accomplish real-world objectives but rather how to accomplish them in a way that is both pleasing and amenable to artistic formulation—that is, in a way in which the designer may shape our experience so that it is enjoyable, invigorating, and whole.

When we participate as agents, the shape of the whole action becomes available to us in new ways. We experience it not only as observers or critics but also as comakers and participants. Systems that incorporate this sensibility into their basic structure open up to us a whole new dimension of dramatic pleasure. This is the stuff of dream and desire, of life going *right*. It is the vision that fuels our love affairs with art, computers, and any other means that can enhance and transform our experience.

The experience of pleasure in a whole action is also influenced by how that action is defined or bounded. In the domain of document creation, for instance, my pleasure and satisfaction has been enormously increased by developments in word processing and printing technology that allow me to engage in more of the *whole* action, from inception to final result. In the days of typewriters, one created documents that would be completely transformed in appearance (one hoped) through the process of publication Through the addition of document design to the application of word processing, and with the assistance of a laser printer, I can now influence the final appearance of a publication through my own (design and formatting) actions, and I can bask in the sense that the thing is really *done* by seeing it in something that closely approximates its published form We will develop this example further in Chapter 6.

The most complex and rewarding result of dramatic action is *catharsis*, defined by Aristotle as the pleasurable release of emotion. That's not to say that all emotions aroused by a play are necessarily pleasant ones. Pity, fear, and terror are mainstays of noncomic forms. It is not the emotion itself, but its release that is deemed "pleasurable." Furthermore, emotions aroused by a play differ in context and expectation from those experienced in real life. When we are viewing a play or film or even riding a roller coaster, we expect emotions to be aroused and to have the opportunity to release them. Aristotle's point is that emotional arousal and release is intrinsically pleasurable in the special context of representations; indeed, that is one of their primary values to us.

In Chapter 1 we discussed a Brechtian view of catharsis that suggests that emotional closure necessarily takes place beyond the temporal "ending" of a play. Brecht's hypothesis was based on a view that requires the integration of the experience of a play into our ongoing life. Brecht's ideas have been interpreted primarily in a political and social light. Julian Hilton offers a more semiotically inclined view of the same phenomenon:

> The totality of the performed event functions as a means of reflective support to the audience, which by no means stops when the performance itself stops. Indeed, in the case of fundamental mythologic structures, such as the Pygmalion/Galathea mythology to which I referred above, their power derives doubly from their synecdochic property of representing in parable form a common human truth and from their persistence in real time operations of the imagination—that is, the imagination uses such myths in a way similar to programming macros or subroutines. The attraction of reflective support is

that it accepts and draws interest from the potential for contradictory resolution of any problem and turns the contract of error from a negative one (a loss of truth or of totality in content) into the leading edge of investigation [Hilton, 1991].

Catharsis depends upon the way that probability and causality have been orchestrated in the construction of the whole; it also depends upon our uninterrupted experience of engagement with the representation. More than that, it is the pleasure that results from the completion of a form. The final form of a thing may be suspected from the beginning or unforeseen until the very end; it may undergo many or few transformations. It may be happy or sad, because the "success" of the outcome in terms of the representational content is not nearly so potent as the feeling of completion that is implicit in the final apprehension of the shape of a whole of which one has been a co-creator. The theory of catharsis dictates that no matter how monumental or trivial, concrete or abstract, the representation affords the occasion for the complete expression of those emotions that have been aroused in the course of the action. In plain terms, it means that we must design clear and graceful ways for things to end.

Of all forms of human-computer activity, computer games are both the worst and best at providing catharsis. They are the best when a player or a computer-based opponent wins, and they are the worst when no one wins, but the action is truncated because it could not continue.[9] In task-oriented environments, the trick is to define the "whole" activity as something that can provide satisfaction and closure when it is achieved. This depends in part on being able to determine what a person is trying to do and striving to enable them to do *all* of it. In simulation-based activities, the need for cathar-

[9]Here again, it seems that the designers at Lucasfilm are in the forefront. Ron Gilbert counsels game designers to avoid situations in which a player must "die in order to learn what not to do next time." [Gilbert, 1989] In a presentation at SIGGRAPH '90, LucasArts Entertainment's research director Doug Crockford showed a re-edited version of *Star Wars* in which Luke Skywalker was killed in his first battle with Darth Vader. The story was over in less than three minutes.

om the potential for
blem and turns the contract
of truth or of totality in
vestigation [Hilton, 1991].

way that probability and
n the construction of the
uninterrupted experience of
n. More than that, it is the
letion of a form. The final
d from the beginning or
ay undergo many or few
or sad, because the "suc-
e representational content
ing of completion that is
f the shape of a whole of
ne theory of catharsis dic-
ntal or trivial, concrete or
the occasion for the com-
that have been aroused in
ms, it means that we must
hings to end.
activity, computer games
viding catharsis. They are
ter-based opponent wins,
ne wins, but the action is
ntinue.[9] In task-oriented
e the "whole" activity as
ion and closure when it is
being able to determine
triving to enable them to
vities, the need for cathar-

Lucasfilm are in the forefront.
oid situations in which a player
next time." [Gilbert, 1989] In a
Entertainment's research direc-
version of *Star Wars* in which
le with Darth Vader. The story

sis strongly implies that what goes on be structured as a whole action with a dramatic "shape." If I am flying a simulated jet fighter, then either I will land successfully or be blown out of the sky, hopefully after some action of a duration that is sufficient to provide pleasure has had a chance to unfold. Flight simulators shouldn't stop in the middle, even if the training goal is simply to help a pilot learn to accomplish some midflight task. Catharsis can be accomplished, as we have seen, through a proper understanding of the nature of the whole action and the deployment of dramatic probability. If the end of an activity is the result of a causally related and well-crafted series of events, then the experience of catharsis is the natural result of the moment at which probability becomes necessity.

This chapter has analyzed various ways in which dramatic ideas and techniques can be employed to influence the way human-computer activities *feel* to people who take part in them. Hopefully, it has illustrated some of the benefits of a dramatic approach in terms of engagement and emotion. The chapter has emphasized the need to delineate and represent human-computer activities as organic wholes with dramatic structural characteristics. It has also suggested means whereby people experience agency and involvement naturally and effortlessly. The next chapter explores structural techniques more deeply, returning to Aristotle's six elements, and suggesting principles and rules of thumb for designing each of them in the computer domain.