Computers as Theatre
Brenda Laurel

Post-Virtual Reality
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 creates 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?”). 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

5An 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.
Computers as Theatre

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 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 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 Böcker 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 enactable or enacting an 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 preactivity 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 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).

Images and program are copyright © 1990 by Apple Computer, Inc.
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 intrinsic 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, 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. 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

---

6For an analysis and thorough bibliography of Coleridge’s criticism, see Allen and Clark (1962), pp. 221–239.
Computers as Theatre

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 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 metalevel 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, physi-
typically 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].

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. 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-

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

8The 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.

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/ Galatea 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. 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 catharsis 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.

---

9Here 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.