CHAPTER 18

Design Considerations for Multimedia Programs

Craig Locatis

Technology-based training delivery systems, such as CD-ROM-based and web-based multimedia, offer unique environments for developing training and performance support programs, and each has distinct design requirements. While these requirements are in addition to the design requirements for typical classroom instruction, they do not obviate the need for traditional instructional design models and strategies. Human beings do not evolve as quickly as technology, and their fundamental learning processes remain essentially unchanged. Indeed, technology-based delivery systems make instructional design knowledge more important than ever, because without such guidance, developers may employ the technology superficially, exercising its bells and whistles to create programs loaded with pyrotechnics but little educational value. Multimedia effects can be used in ways that make little, if any, contribution to learning or—worse—in ways that actually intrude on the learning process. Distance learning experiences, without the benefit of a live instructor (or even other trainees), usually suffer more from poor instructional design than do traditional classroom courses. Instructional design principles provide strategies for ensuring that technology-based delivery systems will achieve desired learning outcomes. Developers need to know how these strategies can be applied in multimedia and distance learning environments. They also need to be aware of some additional design and development issues that arise when the technology-based systems are used.

This chapter begins with a definition of multimedia and a brief description of multimedia trends, to establish a context for understanding its remaining sections. This background information is followed by a discussion of how the use of multimedia affects the entire instructional development process and a discussion about how instructional design theories relate to the application of multimedia technology. Additional multimedia design and development issues are then delineated—issues relating specifically to the unique features of the technology.

Multimedia Technology and Technology Trends

Technically, the term "multimedia" refers to a program that uses two or more of the following types of information: text, graphics, pictures, animation, audio, and video. Today, the term refers to programs that are computer-based and interactive. Interactive programs are nonlinear and alter the content presented and/or the order in which it is displayed, based on user input. The foremost interactive multimedia device currently is the computer, and when people talk about interactive multimedia programs today, they usually are referring to programs that are computer-based (Ambron, 1988). Consequently, this chapter uses the term multimedia to describe any instructional delivery system that is computer-based and interactive, including delivery by CD-ROM or by the Internet. Typically, multimedia programs are also self-paced. In addition, when distance learning or performance support is involved, there usually is some presumption that the programs are provided via networks. Hypermedia is a closely related term referring to multimedia content that is composed of interlinked units of information that users can browse and explore (Shneiderman & Kearsley, 1989). The pre-eminent example of networked multimedia and hypermedia is the World Wide Web. Early multimedia programs were not as integrated as they are today, in part because computers lacked sufficient power. Multimedia content was recorded as analog television and presented via special peripheral devices that the computers controlled. Students would interact with text on computers that would direct the devices to display still images or motion episodes on separate television monitors.

Four major trends have contributed to making the computer a multimedia appliance: conversion, convergence, capacity and costs, and commonness.

Conversion. Conversion means being able to convert analog content to digital, but it also means that devices that were previously analog are becoming digital. Compact audio discs, digital still and video cameras, and digital telephones are, perhaps, the most prevalent examples, but digital radio and television broadcasts also are possible.

Convergence. The conversion to digital has helped fuel a second trend, convergence, or the coming together of heretofore separate technologies. It is now possible to view digital movies or place phone calls with computers because computing, telephony, and television are coming together. Convergence not only applies to hardware but to software. At one time, desktop publishing and word processing software were very different. Eventually, the layout, graphic, and WYSIWYG (what you see is what you get) features of the former became integrated with the font, spell check, and grammatical features of the latter, so that functionally the two types of software tools are almost indistinguishable. Many of the applications built with software tools are converging also. Authoring tools that formerly created computer-based instruction for delivery on standalone computers or CD-ROM can now generate instruction for delivery on the Internet. While some of the programs generated provide electronic performance support systems (EPSS) and others provide more formal instruction, many programs incorporate both. In these cases, it is sometimes difficult to determine just where the online support ends and training begins, since users can move seamlessly from the support system to more directed instruction.

Capacity and Costs. The third trend driving multimedia computing concerns capacity and costs. The former keeps going up, while the latter keeps declining. New capabilities engender new demand and economies of scale lower production costs.

Commonness. Cost and capacity factors have contributed to the forth trend—commonness. It used to be that computers were the exclusive domain of a privileged, knowledgeable few. Now, computers are ubiquitous, in the workplace and at home. And, as computers have become more common, they have become easier to use and readily available as a means of providing multimedia instruction.

Multimedia in Instructional Development

Early in every training development project, the developer-client team faces the task of choosing an appropriate delivery system. Since many factors, such as content, learner attributes, cultural context, and costs, must be considered and balanced, models have been developed to facilitate this complex decision (e.g., Pallesen et al., 1999). Once a decision has been made to use multimedia in some form, additional decisions still have to be made about hardware and software and the various features that will be required. Computers may need a certain amount of memory and storage to accommodate the large multimedia files, and they may need to accommodate external storage media such as CD-ROM or digital videodisc.

Machines may also may need to be networked, and the network may require certain transmission speeds and other quality factors. Multimedia developers face a unique challenge. Since technology changes rapidly, there is always an excuse not to use it; something better and cheaper is always on the horizon. But such reasoning can lead to postponing adoption decisions forever. If the potential benefits of technology are to be used to advantage, adoption decisions will have to be made, and the timing of these decisions becomes paramount. Adopt too early and the technology may be too expensive, unstable, nonstandardized, or non-interoperable with other technologies. Adopt too late and the older technology may prove to be more expensive and cumbersome than newer technology or, worse, no longer supported (Locatis & Al-Nuaim, 1999).

Technology's role in instructional development extends beyond the choice of delivery platforms and affects the development process itself. Most development models have an initial analysis phase, where performance problems are appraised to determine whether they would be most appropriately resolved through training, performance support, or some other intervention. Often, the learner population and the context in which training and/or performance support may be provided is assessed. In the design phase, the goals and objectives of the intervention are established, and strategies are planned for achieving them. Detailed media choices are made, and storyboards, scripts, and screen layouts are devised. These strategies are often refined and realized in the form of a training and/or performance improvement product in a development phase. Finally, most models have an evaluation phase, during which the interventions developed are tried out and their effectiveness determined. Multimedia affects this development process in at least three ways.

1. **Technology Assessment.** Multimedia adds a requirement to do some technology assessment in the early analysis stage. If multimedia are likely to be part of the solution to a training or performance problem, then some of the up-front analyses of these problems should include an appraisal of technology options as well. What technologies are being used currently in the workplace and how comfortable are people with them? How receptive are the organization and the various groups within it to new technology? Sometimes, no additional technology investment may be required when existing infrastructure is leveraged. Other times, modest technology upgrades, such as sound cards or higher resolution displays for existing desktop computers, may be needed. On the other hand, major upgrades to faster computers or higher speed networks might be necessary, or older technology must be replaced completely. Even when appropriate technology is in place or management is willing to invest in new technology, the attitudes and readiness of employees may argue against it—at least not without some additional traditional classroom instruction or personal mentoring. Much

depends on previous experiences with technology and the confidence with which people use it.

2. **Development Stages.** Multimedia affects the development process by making it possible to overlap and shorten the development stages. Much of the up-front analysis, planning, and design in instructional development is done on the computer and, when multimedia programs are to be created, it may unnecessary to create screen layouts, tests, and other instructional assets on paper that later will be programmed. Instead, easy-to-use authoring tools and markup languages can be employed to generate immediate approximations of the final product. Such rapid prototyping allows early review of the content and design alternatives and speeds development. The prototypes are concrete objects, not theoretical plans, having specific features that developers and their clients can experience and discuss. Moreover, these mockups can be tested with small numbers of users and the feedback used to develop more refined, working prototypes.
3. **Pilot Testing.** The use of multimedia usually accentuates the importance of pilot testing in development projects, especially when the development projects are large and the investments in deploying technology widely are substantial. Pilot testing one or a few completed, working prototypes can help determine whether these program components are effective enough to justify further development and whether the technologies employed will prove cost effective if deployed on a wider scale. Unlike some traditional pilot tests in instructional development, multimedia testing uses program components, not the entire program. For example, if an instructional program is to consist of 10 multimedia modules, the pilot test may involve one or two of the first prototypes. If an EPSS is to support a dozen different task areas, one or two areas might be rapidly prototyped, further developed until they are functionally complete, and pilot tested. After pilot testing, additional instructional modules may be developed or further functionality might be added to the EPSS.

Instructional Theory and Multimedia Development

One of the greatest threats to the integrity of multimedia projects is for developers to become so enamored with the capabilities of the technology that they attempt to use them all, even if they add expense and have marginal utility. Instructional design theories and models provide guidance for designing multimedia programs that will achieve a range of learning outcomes. The key is to determine what outcomes one wants to achieve and then to identify the appropriate instructional theories for attaining them. Is the objective to obtain a certain knowledge or skill, to solve problems, to apply procedures, or learn about oneself? Is it to motivate, to foster reflection, or to encourage individuals to learn together? Since most multimedia programs have multiple goals and objectives, several approaches may be relevant. While the theories sometimes seem contradictory, they usually are not. Instead, each theory tends to emphasize particular aspects of the learning process. The theories that figure more prominently for one given set of learning outcomes may be totally irrelevant or only tangentially applicable to a different set of objectives.

At least three reasons support the use of instructional design models and strategies in the design of multimedia programs for learning and performance support. First, these models and strategies have stood the test of time. Indeed, some are based on teaching traditions dating back to ancient Greece. Second, they have empirical support. While in some cases this support is based on field experience, most of the models are supported by substantial scientific research. Finally, the models offer concrete, practical guidelines for bringing about different educational outcomes. When a model is relevant to given instructional goals, it usually offers a blueprint for constructing learning experiences that will facilitate their attainment.

In sum, the theories are authoritative sources for making informed design decisions. They provide rational frameworks for thinking about and discussing design options that can be more productive and have greater payoff than exchanges between developers and clients about their personal preferences and predilections. Although it is not possible in a single chapter to describe how each model and strategy in this book can be applied in designing multimedia programs, the models can be discussed categorically, depending on whether they have a behavioral, cognitive, or humanistic/social/affective orientation.

Behavioral Models and Multimedia Design

Behavioral models emphasize the responses learners make during instruction. The responses are made to some stimulus, but other than acknowledging that some cuing or prompting may be necessary to evoke responses, behaviorists have little to say about how knowledge or information should be presented and portrayed externally, not to mention the thinking that transpires internally inside the learner's head. The models do, however, have something to say about sequencing instruction and how learners should progress.

The primary tenet of behavioral theories is that responses that are appropriate to attaining learning and performance outcomes should be reinforced and those that are inappropriate should not. One implication for interactive multimedia design is that learners should be actively responding to the program content, not passively viewing it. This principle avoids the unappealing and ineffective "page turning" approach to multimedia, apparent now in much web-based material. Moreover, opportunities for interaction should contribute directly to attaining a program's goals. That is, practice should be on the criterion behavior or some approximation of it. A multimedia program that teaches about workplace diversity, for example, may quiz learners on demographic facts. A more appropriate form of interaction might be to react to workplace scenarios, identifying the most appropriate action a manager should take. Another implication is that learners need immediate knowledge of results after they respond and, possibly, remediation. Feedback might be provided in games, for example, by having a counter score each response. The games might have text or sounds encouraging students to try again when their responses are incorrect and that route students to remedial learning events when they fail to attain a certain performance threshold. In one multimedia simulation, the victim of a gunshot wound dies if the student/doctor does not order the right tests and make the correct diagnosis within the proper time frame.

Behavioral theories also hold that certain skills are prerequisite to others and that learning should progress in small steps. Adding and subtracting single digits, for example, is preparatory to adding and subtracting multiple digits where carrying and borrowing may be involved. Consequently, behaviorists also prescribe mastery of earlier skills before allowing students to progress to later ones. The approach has intuitive appeal and works well in cases where prerequisites can be identified. The problem is that it is not always clear that prerequisites for certain skills exist, or what constitutes mastery. Swimmers, for example, do not have to master dog paddling before progressing to the crawl stroke, and when physicians learn medical diagnosis they need not have mastered history-taking skills before learning what laboratory tests to order. Even when the identity of prerequisite learning tasks is clear, there may be a range of sequencing options. In addition or subtraction, for example, it is possible to teach the steps from first to last or to teach the last step first, then the next to the last step, and so on.

Prerequisites and mastery learning, advocated by most behavioral models, raise questions about learner control of instruction and generally argue against it. Most multimedia developers, however, are loath to handcuff their users electronically and usually provide a range of tools for exploring their programs. Research on aptitudes, however, indicates that students with well-developed self-learning skills can thrive in open environments (Snow & Lohman, 1984). Older students or students who already have substantial knowledge of the subject matter are more likely to have these skills than novice learners. Two design options are popular when learners are novices. One is to provide advice about how to proceed through a program, based on students' previous performance. Another is to provide programs having mechanisms for activating or deactivating learner control (cf., Shin et. al., 1994). Some may allow instructors to set the learner control function, while others may allow students freedom at first, but impose more direction when performance is low.

Behavioral models dictate that instruction should proceed in small steps, but the need for very small steps has largely been discredited. Small steps are not only boring; they lead to instruction that is inefficient. Most multimedia developers break instruction down into logical steps that reflect the subject matter and the tasks that are performed in the real world, rather than divide content into arbitrary subskills. If there is any question about step size, they are prone to opt for large steps initially, dividing instruction into smaller steps only if tryouts indicate learners are having difficulty. While this approach ensures development of lean programs, concerns arise that the programs provide insufficient instruction when used in conjunction with learner control (Schnackenberg & Sullivan, 2000). Some developers also opt for creating adaptive programs that dynamically adjust the size of step and difficulty of the information presented based on previous performance. These programs are more sophisticated than those imposing more direction when performance is low, because the programs have algorithms that constantly compute estimates of each learner's knowledge level during the course of interaction (Tennyson et al., 1984).

In summary, behavioral models may be best applied to multimedia design when specific, concrete behaviors are to be learned (when there is a right way to perform a task). If program control is desirable, and prerequisite sequences are well defined, behavioral approaches will work well. In any multimedia program, frequent, active responding and immediate feedback are recommended. Indeed, these are distinctive features of multimedia delivery systems; not to use them is to waste the system capabilities.

Cognitive Models and Multimedia Design

Cognitive models fall into two camps: those that endorse expository teaching and those that favor letting students discover and construct knowledge on their own. Most cognitive models focus on the stimulus (how knowledge is presented to the learner) as well as the thinking that goes on inside a learner's head. Where the two cognitive camps diverge is on how much guidance should be provided. Those advocating expository approaches are similar to behaviorists in favoring program control. Unlike the behaviorists, however, cognitivists are more tolerant of allowing learners to progress without mastering prerequisites. Students may only partially comprehend component concepts and skills initially, and it is only when they have been exposed to all components that the pieces "fall into place" and understanding occurs. Constructivists argue that since all learning is constructed in the head of the individual learner, the learners should be free to explore rich environments and make sense for themselves from a variety of data sources and experiences. Thus, they favor extremes of learner control.

The most pervasive influence of cognitive models on multimedia development is, perhaps, in the area of knowledge representation. Both camps want to ensure that knowledge is appropriately portrayed and that conceptual relationships and problemsolving strategies become internalized appropriately by students. Expository models particularly stress content organization though tables of contents, concept maps, graphical displays, and other mechanisms that make explicit subject relationships. Information visualization strategies from screen layout to the use of color in multimedia programs are of special concern (cf., Lynch & Horton, 1999; Shneiderman, 1998; Tufte, 1990).

One popular multimedia design strategy is to employ metaphors to depict content. A program teaching accounting may use a spreadsheet metaphor, while one teaching research skills might situate the instruction in a virtual library setting. Another strategy is to use analogies. While metaphors establish more global context for representing knowledge, analogies provide more specific ones. A heart, for example, may be likened to a pump, a brain to a computer. Metaphors and analogies are often employed in noninstructional multimedia applications. An airline's website, for example, may use a ticket counter metaphor that users can interact with to obtain information about fares, destinations, and schedules. When metaphors and analogies work, they are spectacular, but when they fail, results can be abysmal. Thus, they should be chosen carefully.

Cognitive theories have also influenced multimedia design in the development of intelligent tutoring programs and expert systems. While adaptive instructional programs dynamically adjust instruction, the adjustments are based on statistical probabilities calculated from previous performance in the program. Intelligent tutoring programs can be much more sophisticated, presenting problems intentionally designed to detect reasoning errors and repair them. Some compare student performance to an underlying expert model and intervene when a student's performance begins to diverge. Others have underlying knowledge bases of rules experts employ to solve problems that they explicitly teach, and they may have the capability to solve the problems they present to students themselves because they are integrated with expert systems (Wenger, 1987). There are also experts systems that exist independently of tutoring programs; these are not used so much to detect reasoning errors as to prevent them. The systems can be viewed as a special type of EPSS that renders advice based on information users provide. The systems are intended to reduce the amount of information people need to keep in their heads and to suggest solutions that might be overlooked (Shortliffe, 1990). In medicine, for example, it is impossible to keep track of all drug interactions, and some interactions produce side effects mimicking diseases and syndromes. Expert systems can inform physicians of interactions in the medications they may consider prescribing or, if a patient presents with certain symptoms that can be ascribed to drug interactions or a disease, they may suggest drug interaction as a possible cause to help guard against possible misdiagnosis.

Cognitive theory's influence on interface design has been to encourage development of simpler interfaces and to reduce the amount of cognitive overhead students need to interact with programs (Norman, 1993). Anyone who has stared at a computer screen understands that the notion of metaphor extends beyond multimedia applications to the computer's operating system. Users no longer need to know commands but can interact with a "desktop," and they use programs with graphical user interfaces that enable them to place "files" into "folders" or to "bookmark" their favorite websites.

Constructivist cognitive theories have added a dimension of openness to multimedia development (Hannafin et al., 1997). Of special importance is the development of case studies and simulations to allow learners to discover underlying concepts and problemsolving strategies. Sometimes the cases and simulations have fidelity with the real world, and sometimes they do not. A multimedia chemistry program may offer a virtual laboratory with devices and substances that the users can manipulate to conduct experiments much as they would in a real lab. A program teaching map reading and computational skills, on the other hand, might place learners in a completely hypothetical situation, such as being in a mythical land, and require them to use maps, compasses, and other devices to locate treasure. Constructivist theories support the use of databases and online information sources to solve problems. The Internet and World Wide Web constitute a vast information space for addressing research questions or for browsing and exploration that can be employed to teach the metacognitive skills of learning how to learn on one's own.

A designer of multimedia programs can use a variety of tools and strategies derived from cognitive instructional models. If using expository teaching methods, one can structure procedural or rule-based content using algorithms (Scandura); present information and concepts using an advance organizer (Ausubel); insert mnemonics to aid recall of facts, lists, and associations; or structure lessons around the nine events of instruction (Gagné and Briggs). Merrill's component display theory, with its detailed prescriptions for each type of content and performance outcome, is particularly well suited for multimedia instructional design. If inquiry, discovery, or constructivist methods are more appropriate, multimedia environments allow extensive access to different resources (databases, simulations, expert opinions, picture files, etc.) and learner control to access those resources as desired.

Humanistic/Social/Affective Models and Multimedia Design

Humanistic, social, and affective theories address emotional and cultural factors in learning, educational outcomes related to learning how to work in groups, and the development of attitudes toward both subject matter and learning. They also are concerned with motivation. Although they are diverse, they have impacted multimedia design. Cultural factors are an important factor in multimedia development. For example, care is taken to avoid stereotypes. If programs require characters, they usually are representative of the target population for which programs are intended. A number of motivational strategies—such as employing cues to gain and direct attention, offering encouragement, and providing problems that are challenging but still doable to instill confidence and feelings of accomplishment are used routinely in multimedia programs. Keller's ARCS model, for example, is just as applicable to multimedia training as it is to traditional classroom training.

Of all the theoretical perspectives, those in the humanistic, social, and affective realm probably have more to say about how multimedia programs are used than about how they are designed. The ways multimedia programs are used can dramatically affect learning outcomes in ways developers might not have anticipated. For example, skilled teachers can use the most culturally pugnacious and biased websites of extremists groups as springboards for learning about fear and hate and for encouraging critical thinking.

The adult learning model (Knowles) and the popular education model (Vella) can certainly incorporate multimedia instruction. Since the adult learning focus is on the independent, self-directed learner, the availability of multimedia resources on the World Wide Web can greatly facilitate the learner's ability to locate relevant information and instruction. However, adult educators advocate personal attention by an instructor/facilitator, so that multimedia approaches alone would not be entirely consistent with the model. Popular education, which focuses on face-to-face group activity, could employ multimedia instruction as an adjunct to classroom work, rather than as the main teaching mode. However, technology advances, such as desk-to-desk computer videoconferencing, may soon make humanistic models more possible to implement via technology-based delivery systems.

Most multimedia programs are intended for individual use, but research shows additional benefits accrue by having students use them in groups (Hooper, 1992). Peer discussion fosters further elaboration of the content and allows more advanced learners to help their cohorts. Cooperative learning approaches identify specific strategies for fostering such outcomes while discouraging competition and ensuring that all students contribute. While most group learning with computers has occurred in classroom settings, online communication tools are enabling students to learn collaboratively at a distance (Hiltz, 1994). While some tools, such as email and discussion groups, allow for asynchronous communication, others, such as chat, videoconferencing, and streaming video applications allow for one-to-one, one-to-many, and many-to-many communication (Locatis & Weisberg, 1997). Both synchronous and asynchronous modes of distance learning have recently been used in higher education and in business, and cooperative learning is possible even in these virtual classrooms. A challenge for multimedia developers will be to create environments where multimedia resources and online collaboration tools are integrated to allow knowledge sharing and advancement (Scardamalia et al., 1992).

Multimedia Design and Development Issues

All the design and development decisions in traditional instruction development and performance improvement projects should also be addressed in multimedia development projects. The performance and training needs must be identified, the characteristics of the performers must be assessed, and the organizational contexts and cultural environments need to be appraised. Decisions about objectives, content sequencing, instructional methods and strategies, and media choices must be made. Resolving these issues correctly becomes even more critical in multimedia projects, because greater startup and development costs are often involved. But the prospect of using multimedia often raises additional concerns, beyond those evident in traditional instructional design or performance improvement projects.

Asset Use and Allocation

It is one thing to have the capability to present content using a range of media and another actually to use those capabilities. First, every modality employed adds to development costs, and some modalities are more expensive than others. A program having video, animation, and graphics will generally be more expensive to create than one having just graphics and text. Moreover, use of some media boost delivery system requirements. Audio and video may require more powerful computers with CD-ROM or DVD drives and sound cards or, if delivered via the Internet, a certain level of network performance and capacity. Second, use of some modalities or too many modalities may be counterproductive. For example, while it may seem clever to have continuously running animations on a text page, they distract from reading. A better option may be to allow users to activate and control them on their own. If the animations are gratuitous and do not add substantive content, they will have little benefit, even when interaction is added. Novice developers often use varied backgrounds and colors, moving text, and other gimmicks that usually only contribute to eye strain. Ironically, abuses such as these are more likely because authoring tools have made it easier to create so many varied multimedia assets. Having a clear purpose and strategies based on an instructional design model helps developers select and use assets wisely.

Content Granularity

Many multimedia programs are excessively long and linear. Most users work with computers interactively and exercise a high degree of control over the work that they perform with them. Long audio and video sequences or reams of text can turn an active technology into a passive one. Selecting appropriate granularity of multimedia content is a difficult problem. Old adages such as "Limit information to what can be displayed on a single screen" are not very useful. Not everyone has the same size screen, and most users can adjust its resolution anyway. As computer use has increased, more users know that they can scroll down displays and that there may be more information than what fills a single screen. Moreover, when text is involved, many users may print it for use offline. They may prefer to print the entire resource, or sections of it, rather than individual pages. Finally, convergence has made it possible to broadcast audio and video as streams over the Internet. These programs may be archived and offered on demand or live, but are usually just as passive as regular radio and television programs. The granularity problem depends on how the multimedia information is to be used and what users may expect. Users may tune into a live video broadcast of a symposium on the Internet expecting to see it in its entirety. Users accessing an archive of the conference, however, may prefer to choose among different presentations that were made during the symposium instead of the whole program. Again, thorough analysis and design using a coherent model helps determine appropriate granularity.

Meaningful Interaction

Learning and performance improve when users can interact with content meaningfully. Of the many levels and types of interaction, the simplest is paging forward and backward through successive displays. For an EPSS, such page turning may suffice, since most users are simply looking up information for immediate application. When learning is required, higher levels of interaction are warranted. Some of the types of interaction are the following:

- Problems and selectable options for solving them. Simple links that provide feedback may be used, or learners may be routed to different parts of a program.
- Constructed response items. Users must compose their answers in writing, and they are given feedback and routed through the program based on their responses.

- Monitoring the time participants take to respond and scoring their performance.
- Complex computer simulations. These may involve problemsolving and time-sensitive performance, such as solving trauma cases in emergency rooms. In this example, the computer may alter the condition of the simulated patient automatically, introducing new complications, if certain interventions fail to be made, and each option students choose may be used to calculate problemsolving performance.

Developers and clients must trade off the level of interactivity and the corresponding development cost. To evaluate any input students compose themselves is much more difficult than to assess their selections from a list of options. Adding time as a variable and/or scoring algorithms often requires formidable programming. Using a well-chosen instructional design model based on research and experience often helps in determining when sophisticated interactions are worth the additional time and expense—when and to what extent they add learning and performance value.

Human Interaction. One type of interaction that often is overlooked in multimedia program development and use is interaction among students or between students and teachers. The presumption that students should individually use multimedia programs has a long history. As more tools evolve that allow collaboration online, multimedia developers need to define ways to incorporate them into their programs, since student-to-student and student-to-teacher interactions are by nature much more open and freewheeling than any that can be realized between a computer and a single user. While the asynchronous communication tools such as email predominate today, synchronous tools such as videoconferencing are becoming more feasible at the desktop. Human interaction also can be designed into applications involving live streaming audio and video broadcast of lectures, seminars, and other events over the World Wide Web. In addition to providing a link on a web page to the broadcast, links also can be provided that allow users to send email or chat messages to the program presenters. If planned appropriately, audience participation can be encouraged, and the broadcasts may assume some of the character of radio and television talk shows. These capabilities will allow technology-based delivery systems to provide closer adherence to humanistic and social instructional models.

User Interfaces

Human factors and human-computer interaction are important considerations in multimedia program development. Computer displays should not have the density of print media, and care must be taken to ensure fonts and images are legible and displays are clear and uncluttered. An overarching goal is to make the interface transparent, so that users can concentrate on learning the content from an instructional program or obtaining support of an EPSS without being preoccupied with the computer as an intervening tool. One way to accomplish this is by devising "affordances" or creating interactive multimedia objects with functionalities that are intuitive and obvious. Forward and backward arrows for paging and printer icons for printing are popular examples. Another way to accomplish this is by making parts of the interface similar to the computer operating system or other application tools users are likely to understand already. For example, if the computer operating system and most other applications for the computer use drop-down menus, then this method of making menu selections should be considered. One benefit of offering resources on the web is that browsers provide a standard interface for accessing content. Finally, the overall design should be consistent. Screen layouts should follow a standard format and navigation tools should be uniformly placed so users know exactly what to expect as they interact with the program.

User Navigation

While navigation is part of the interface, it deserves special consideration. With linear media like video, users are guided through the program. While books are nonlinear, their size and organization are usually apparent at a glance. Not so with computer programs, and disorientation is common. While care should be exercised to provide navigation features that are consistent with program- or learner-control strategies developers choose to adopt, it is often advisable to provide tools allowing users to exit the program, go to the start of the program, or access "landing pads" in the program (such as indexes and tables of contents) that let them jump to other parts. In addition, users should not have to traverse more than three menu layers to access content. Search features also may be important, especially if users are expected to look up information, as in an EPSS. When multimedia is designed for the web, links outside the program to other resources and websites need careful consideration. Linking to existing databases and other related online resources is a unique advantage, enabling developers to leverage resources and not have to develop them themselves. However, placement is crucial. The overall goal is to encourage users to stick with the program and the website, so users should not be encouraged to leave. Users may become perplexed if they select a link to an outside resource that they expect to be internal. External links should be grouped in one section of the program and appropriately labeled, rather than riddled throughout. Since external resources are beyond developer control and may be moved or deleted by those in charge of these sites so as to make the links inoperative, this strategy also has the advantage of making the links easier to maintain.

Asset Management

One of the most labor-intensive efforts in multimedia development is the identification, creation, and/or capturing of resources. When existing photographs, artwork, or other resources are identified for inclusion in programs, permissions, clearances, or copyrights may have to be obtained. Many graphics, sound, and video resources also may have to be developed from scratch. While these resources may already be in digital format, they often are not. Photos may have to be scanned and audio and digital recordings digitized. The higher the resolution of a digitized image or the fidelity of a digitized audio source, the larger its file size. More storage space will be consumed when it is stored on a CD-ROM and more bandwidth will be consumed if the file is delivered over the Internet. Consequently, many developers may create or capture them in lowest resolution acceptable. A better strategy, however, is "capture large; deliver small." Since so much effort is involved, digitize resources at the highest resolution, archive them, and then export them at a lower, acceptable quality. The reason is that technology improves rapidly. As the resolution of computer displays improves and bandwidth increases, standards of acceptable quality will rise also. When programs are updated, it will be easier to re-export the high-resolution resources at the new standard of acceptability than to have to recapture them. Even if a program is not updated, many of the multimedia resources in it might be re-used in future projects, and it may be worthwhile to catalog individual assets and place them in a database so they can be retrieved and repurposed.

References

Ambron, S. (1988). New visions of reality: Multimedia and education. In S. Ambron & C. Hooper (Eds.), *Interactive multimedia*. Redmond, WA: Microsoft Press.

Hannafin, M., Hill, J., & Land, S. (1997). Student centered learning and interactive multimedia: Status, issues, and implications. *Contemporary Education, 68*(2), 94–99.

Hiltz, S. (1994). *The virtual classroom: Learning without limits via computer networks*. Norwood, NJ: Ablex Publishing.

Hooper, S. (1992). Cooperative learning and computer-based instruction. *Educational Technology Research and Development, 40*(3), 21–38.

Locatis, C., & Al-Nuaim, H. (1999). Interactive technology and authoring tools: An historical review and analysis. *Educational Technology Research and Development, 47*(3), 63–75.

Locatis, C., & Weisberg, M. (1997). Distributed learning and the Internet. *Contemporary Education, 68*(2), 100–103.

Lynch, P., & Horton, S. (1999). *Web style guide: Basic design principles for creating web sites*. New Haven, CT: Yale University Press.

Norman, D. (1993). *Things that make us smart*. Reading, MA: Addison-Wesley.

Pallesen, P.J., Haley, P., Jones, E.S., Moore, B., Widlake, D.E., & Medsker, K.L. (1999). Electronic delivery systems: A selection model. *Performance Improvement Quarterly 12*(4), 7–27.

Scardamalia, M., Berieter, C., Brett, C., Burtis, P., Calhoun, C., & Smith L.N. (1992). Educational applications of a networked database. *Interactive Learning Environments, 2*(1), 45–71.

Schnackenberg, H., & Sullivan, H. (2000). Learner control over full and lean computer-based instruction under differing ability levels. *Educational Technology Research and Development, 48*(2), 19–35.

Shin, E., Schallert, D., & Savenye, W. (1994). The effects of learner control, advisement, and prior knowledge on young students' learning in a hypertext environment. *Educational Technology Research and Development, 42*(1), 33–46.

Shneiderman, B. (1998). *Designing the user interface*. Reading, MA: Addison-Wesley.

Shneiderman, B., & Kearsley, G. (1989). *Hypertext hands-on!* Reading, MA: Addison-Wesley.

Shortliffe, E. (1990). Clinical decision support systems. In E. Shortliffe & L. Perreault (Eds.), *Medical informatics: Computer applications in health care.* Reading, MA: Addison-Wesley.

Snow, R., & Lohman, D. (1984). Toward a theory of cognitive aptitude for learning from instruction. *Journal of Educational Psychology, 72*(3), 347–376.

Tennyson, R., Christensen, D., & Park, O. (1984). The Minnesota Adaptive Instructional System: An intelligent CBI system. *Journal of Computer-Based Instruction, 11*(1), 2–13.

Tufte, E. (1990). *Envisioning information.* Cheshire, CT: Graphics Press.

Wenger, E. (1987). *Artificial intelligence and tutoring systems: Computational and cognitive approaches to the communication of knowledge.* Los Altos, CA: Morgan Kaufman Publishers.

About the Author

Craig Locatis is an educational research specialist and project officer at the National Library of Medicine (NLM), where he has worked on research and development projects involving new technology for 20 years. Projects have ranged from developing an interface for one of the first online library catalogs to creating prototype multimedia applications. Dr. Locatis was one of the founders of the Library's Learning Center for Interactive Technology, a national center for demonstrating interactive health care programs. He also was responsible for a project connecting the national medical libraries of eight republics in the former Soviet Union to the Internet. His current assignment is with NLM's Office of High Performance Computing and Communication, where he manages Next Generation Internet contracts and experiments with applications involving broadband networks. Dr. Locatis reviews for several professional journals and has published widely. While at the NLM, he has taught or guest lectured at universities in the metropolitan Washington, DC area and at Vilnius University in Lithuania.