

# Showing Instead of Telling

Ron Baecker

Bell University Laboratories Professor of Human-Computer Interaction  
Knowledge Media Design Institute and Department of Computer Science

University of Toronto

40 St. George Street #7228, Toronto Ontario M5S 2E4 Canada

+1 416-978-6983

rmb@kmdi.toronto.edu

## ABSTRACT

We propose the use of dynamic visual representations to augment traditional static text as documentation. In other words, we suggest using live demonstrations or moving pictures to show people how to do things, and not just using written or spoken words to tell them what to do. We present and illustrate examples of five methods and technologies for creating such dynamic documentation, and summarize what is known about their effectiveness.

## Categories and Subject Descriptors

H.5. [Information Interfaces and Presentation], H.5.2. [User Interfaces]: Training, help, and documentation.

## General Terms

Documentation, Design, Human Factors

## Keywords

Software visualization, movies, animation, digital video, screen linking, streaming, animated icons, screen capture.

## 1. INTRODUCTION

There continue to be increasing challenges for the field of documentation. The emergence of the Web as a universal digital library has meant that documentation must be designed for the Web as well as for traditional formats and media such as paper. The development of new handheld and wireless technologies has resulted in additional documentation needs and in challenging new small-screen formats for documentation presentation and access. Finally, the phenomenon termed “bloatware” by some [McGrenere and Moore, 2000] and “feature creep” by others [Hsi and Potts, 2000] continues to enlarge the set of functionality that must be documented and the scope of the documentation problem.

A good example may be found in Microsoft MS Word. Hsi and Potts [2000] have documented the increase in functionality of MS Word from 311 functions in Word 2.0 to 614 functions in Word 95 to 955 functions in Word 97. This same phenomenon is depicted graphically in terms of the morphology of the Insert Menu only in Figure 1.

[Proceedings of ACM SIGDOC 2002, 10 - 16.](#)

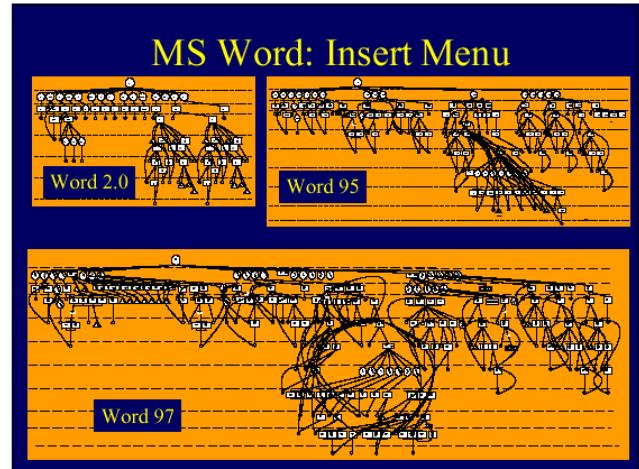


Figure 1. Growth in complexity of the MS Word Insert Menu (Hsi and Potts, 2000).

Responses to these challenges may be seen in the Proceedings of the 2001 International Conference on Systems Documentation.

Some contributors proposed novel uses of technology, such as database technology [Russell, 2001] or XML-based architectures [Priestley, 2001] for storing and managing documentation. Other papers discussed documentation methodologies and philosophies, including semiotic engineering [Silveira, Siechenius de Souza, and Barbosa, 2001] and a “rhetoric of objects” [Price, 2001].

There were discussions of the long-desired goal of intelligent help [Ames, 2001], and of cross-cultural issues that arise in the design of multi-lingual Web sites [He, 2001]. The Conference reflected awareness of new issues and challenges raised by Open Source software [Berglund and Priestley, 2001], as well as the realization that creating documentation is computer-supported collaborative work in which culture and artifacts mediate work process [Spinuzzi, 2001].

Finally, there was one paper [Jackson, 2001] dealing with using multimedia to enhance documentation of hardware service procedures.

Documentation, as well as online help, technical support, and training, may be viewed as media for answering questions that users have about computer software, systems, and technology. Many questions can be characterized by the following taxonomy [Baecker and Small, 1990; Baecker, Small, and Mander, 1991]:

- **Identification:** What is this?
- **Transition:** Where have I just come from?
- **Orientation:** Where am I?
- **Choice:** What can I do now?
- **Demonstration:** What can I do with this?
- **Explanation:** How do I do this?
- **Feedback:** What is happening?
- **History:** What have I done?
- **Interpretation:** Why did that happen?
- **Guidance:** What should I do now?

In almost all cases, answers to these questions are currently delivered as text, sometimes augmented by still-graphic images. Unfortunately, despite all the skill and care applied to the creation of these texts, many users don't read manuals and often find "online help" unhelpful.

This paper asserts that answers can and should often be delivered visually, in other words, users should be *shown how* instead of being *told what* to do.

Showing people how to use software is a form of *software visualization* [Stasko, et al., 1998]. Software visualization has been defined [Price, Baecker, and Small, 1993] as "the use of the crafts of typography, graphic design, animation, and cinematography with modern human-computer interaction and computer graphics technology to facilitate both the human understanding and effective use of computer software."

To date, the field has focused primarily on enhancing *programmers'* and *students'* understanding of how software operates internally (see, for example, Baecker, 1982, 1998). Software visualization can also show *computer users* what to do, augmenting text and still-graphic forms of documentation and on-line help with visual explanations.

Five methods and technologies to illustrate this approach will be presented in this paper. Most will be demonstrated at the conference. The first two deal with technologies that allow live demonstrations and explanations to be delivered in real-time to the desktop of a user who wants to be shown how to do something. The other three deal with technologies for creating and delivering to a user's desktop upon demand previously created visual demonstrations and explanations.

These five methods are:

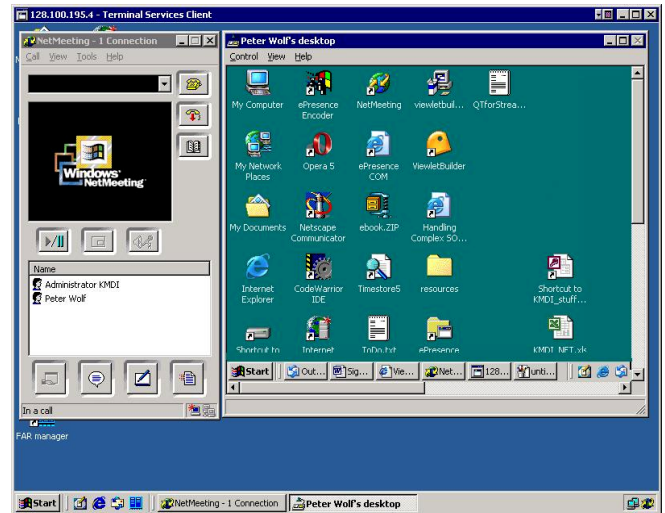
1. *Screen linking*, allowing someone who knows how to do something to control the screen of a user in real time, and to give a live demonstration of a technique
2. *Visual streaming* of audio, video, slides, and data, allowing the real-time delivery of visual demonstrations and explanations to a user
3. *Animated icons*, for encapsulating in minimal real-estate dynamic demonstrations and explanations, especially those clarifying the meaning and purpose of static icons
4. *Screen capture* presentation tools, allowing the easy creation of simple animated visual demonstrations and explanations
5. *Structured video* systems for creating visual demonstrations and explanations, which are then accessed by users on the Web and delivered over the Internet.

After introducing these five methods, we shall review the state of experimental evidence supporting their utility, summarize and integrate what we have presented, and discuss directions for future research.

## 2. SCREEN LINKING

Screen linking software, pioneered in Timbuktu from Netopia (see [www.netopia.com](http://www.netopia.com), previously Farallon Computers) was developed for allowing remote control of PC desktops. The key idea is that a *host* system gives permission for an *observer* system to link to the host's screen. This allows the observer to watch what is going on and even to control what is going on (see, for example, Figure 2, which shows the screen linking capability of Microsoft Windows NetMeeting).

One prime use of such technology is technical support, allowing a support representative to see at a distance exactly what problem a user is having and what the user may be doing to create the situation.



**Figure 2. Screen linking in Microsoft Windows NetMeeting**  
The large window on the right shows to the remote viewer what is on the desktop of another PC user.

Screen linking can be used as a form of live visual documentation to answer questions of Choice, Demonstration, Explanation, Feedback, Interpretation, and Guidance. Users can ask questions and engage in follow-up dialogue via text chat or Internet telephony.

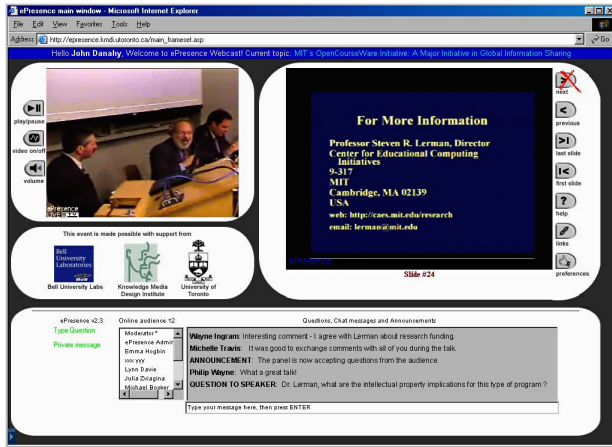
## 3. VISUAL STREAMING

NetMeeting provides other capabilities besides screen linking. The host and the observer (henceforth, "the *collaborators*") can speak together using Internet telephony and see each other using digital videoconferencing. In fact, the collaborators are not limited to two individuals. The number of NetMeeting participants is typically limited by network capacity; given the current Internet, this is usually 5-10 people.

The scalability limitations of videoconferencing may be overcome by going to a webcasting technology. *Webcasting* is the Internet broadcasting of streaming audio possibly accompanied by streaming video so that it can be viewed via a Web browser on a personal computer. Webcasting, because the

audio and video are going outbound only, can transmit to hundreds or even thousands of viewers.

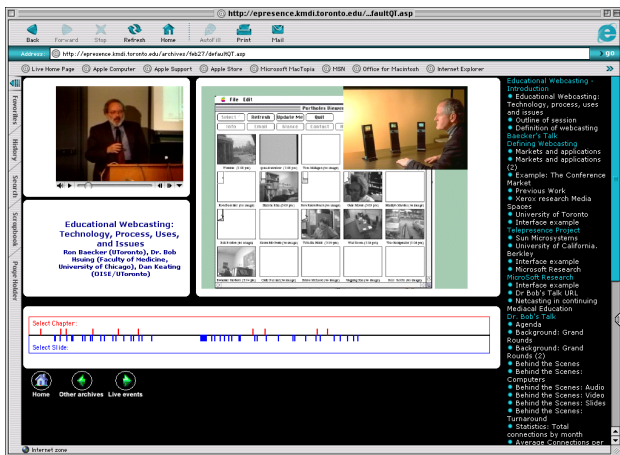
An interesting interactive webcasting environment is the ePresence system from our laboratory [Baecker 2002a,b]. Its functionality currently includes support for video, audio, slide, and data broadcasting; slide review; integrated moderated chat; question submission; and the semi-automated creation of structured, navigable event archives.



**Figure 3. The ePresence interface for live webcasting**

The video window and its controls are in the upper left, the slide window or data window and its controls are in the upper right, and the chat system is at the bottom.

The real-time webcasting interface is illustrated by the screen snapshot shown in Figure 3. Slide controls allow a remote viewer to independently browse through the slides in the window in the upper right. This area (significantly enlarged) can also be used to display a remote screen for us in a live demonstration of a remote computer system, similar to the screen linking capability described above. The chat system supports public chat, private messages, and questions to the speaker via a moderator.



**Figure 4. The ePresence interface for video archives**

An outline of the structure of a talk appears in the upper right, an interactive timeline below the slide window. Both are used for navigating nonlinearly through a structured video archive.

The archives interface (Figure 4) currently allows retrospective navigation and browsing through a webcast using an outline of the logical structure of the talk and its slides. The structure is input by the chat moderator during the talk and may be updated afterwards. The archive can also be navigated via an interactive timeline.

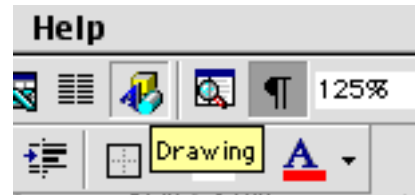
Visual streaming can be used to give a live demonstration of a system's functionality or its interface to a distributed audience located anywhere in the world. It is appropriate for answering questions of Choice, Demonstration, Explanation, Feedback, Interpretation, and Guidance. This live visual documentation can be recorded and made accessible through structured video archives, which need not be viewed linearly, but can be navigated, browsed, and even searched.

#### 4. ANIMATED ICONS

Modern graphical user interfaces make extensive use of icons (see, for example, Horton, 1994), symbols intended to represent functions or files of data. Icons representing files typically only distinguish between different data types, such as Word documents or Excel spreadsheets. The actual designation of each file is given in accompanying text. Icons representing functions, on the other hand, are often intended to convey completely to a user the function he or she is activating and its purpose.

The strengths of icons are that they are compact and usually quickly recognizable. They are thought to be "universal" because a specific natural language is "not needed". The major weakness of icons is that they are sometimes non-obvious. It is also hard to scale a set of icons to handle hundreds or thousands of objects.

Two methods designed to assist with icon identification are bubble help and tooltips. Bubble help brings words identifying or clarifying the purpose of an icon into a small thought bubble that pops up beside the icon. Tooltips, a far more common technique, does the same in a small rectangular window (Figure 5).

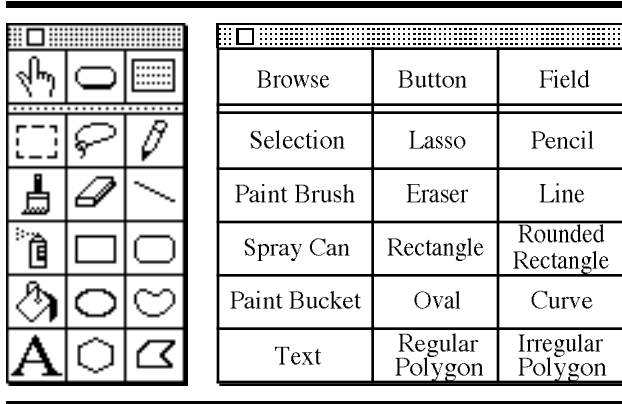


**Figure 5. A tooltip for Microsoft MS Word**

The word "drawing" appears just below and to the right of the perhaps mysterious icon it is intended to explain.

Our research on animated icons (Baecker, Small, Mander, 1991) attempted to provide more certain recognition of the meaning of an icon by using animation to provide a dynamic visual representation of each function, in other words, to answer Identification, Choice, and Demonstration questions. Our pilot project used the HyperCard tool icons shown in Figure 6.

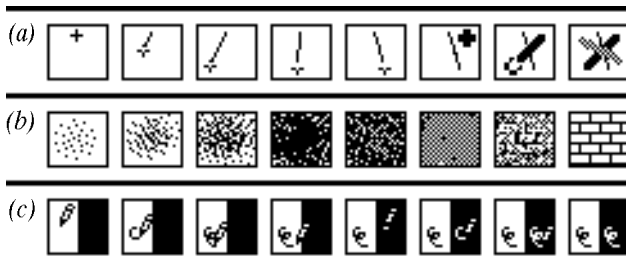
An experienced Macintosh-knowledgeable animator was hired to create an animation of each of the icons (Figure 7). The animations were typically constrained to fit into the space occupied by each icon.



**Figure 6. The Hypercard tool palette**

The icons appear on the left above, and descriptive titles appear on the right. Users saw only the icons.

We used a quasi-experimental design to evaluate the animated icons. We recruited 8 subjects with varying degrees of familiarity with paint tools. They were asked to explain the static icons, then asked to explain the icons again after viewing the animations. The animations proved helpful in explaining the Selection, Lasso, Paint, Curve, and both Polygon tools, cases where some users had trouble when presented only the static icons.



**Figure 7. Keyframes from prototype animated icons**

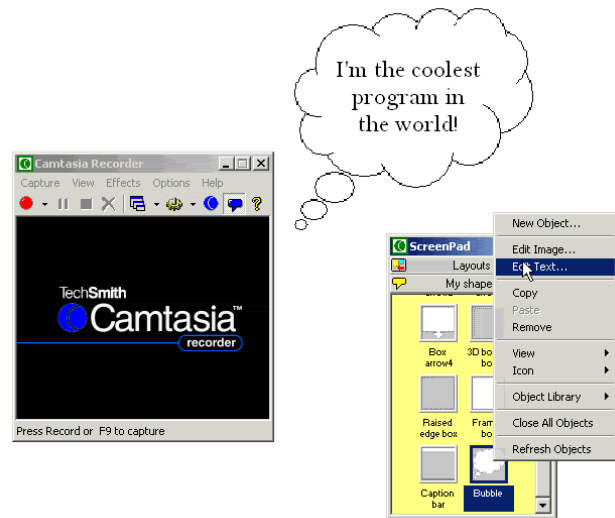
To “see” the movie, imagine motion through the keyframes. The actual images were approximately the size shown here.

We also experimented with non-speech audio [Buxton, 1989] cues, so-called program auralization [DiGiano and Baecker, 1992], and found the sound compelling. However, no formal test of the use of sound was carried out.

## 5. SCREEN CAPTURE PRESENTATIONS

Animated icons are a promising visual technique for illustrating a limited set of functions. A significantly more general technique is afforded by the screen capture capabilities that exist in a number of products, for example Snagit and Camtasia from Techsmith Corporation [[www.techsmith.com](http://www.techsmith.com)], and Viewlet Builder from Qarbon Inc. [[www.qarbon.com](http://www.qarbon.com)].

Screen capture tools allow a visual documentation specialist to select a region of a screen, to mimic a user’s interaction with a program, and to capture an animated portrayal of that interaction (Figure 8). These animations can then be overlaid with titles, annotations, sound, and even music to provide a visual demonstration of the interaction. This is particularly useful in answering Choice, Demonstration, Explanation, History, and Guidance questions.



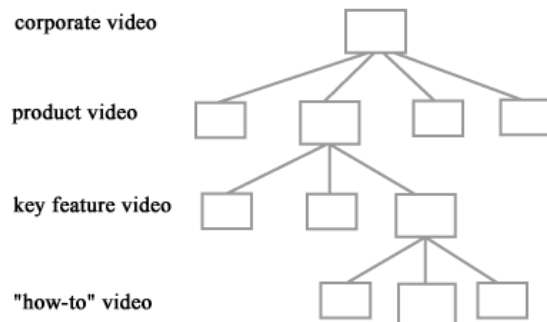
**Figure 8. Screen captures from Techsmith’s Camtasia.**

An overlay has been added in a thought bubble to the capture on the left. A second screen capture has been added, including a moving cursor and pop-up menu on the right.

## 6. STRUCTURED INTERNET VIDEO

A more ambitious approach is to create structured digital video presentations that demonstrate software and show users how to accomplish desired tasks. More generally, they can be used to answer Choice, Demonstration, Explanation, History, and Guidance questions. These video presentations can then be integrated into the documentation, training, support, and sales sections of a company’s Web site. They can be accessed from the Web by users and streamed over the Internet “just in time.”

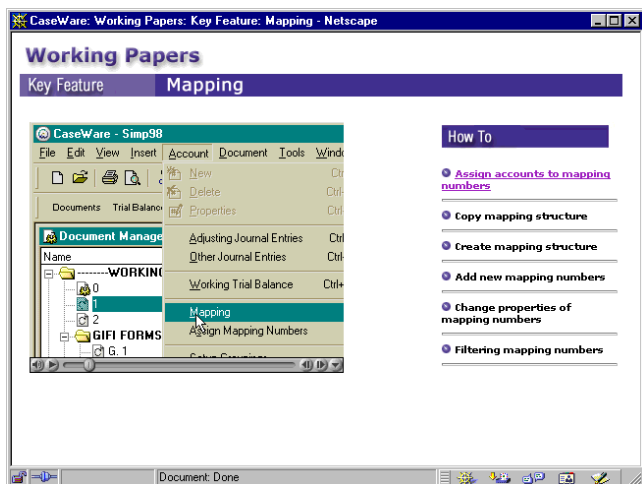
We developed Expresto Creator, novel software to assist in creating such videos [Baecker, et al., 1996; Baecker and Smith, 2002]. Creator is an easy-to-learn and easy-to-use system that represents video as hierarchically structured documents including outlines, scripts, audio tracks, still images, and video sequences. Because of the ability to represent movies hierarchically in terms of an outline, authors of visual documentation can develop ideas both top-down and bottom-up, can modify the structure with ease, and can work at various levels of detail. Creator is an outline processor for movies.



**Figure 9: Hierarchical structure of CaseWare videos**

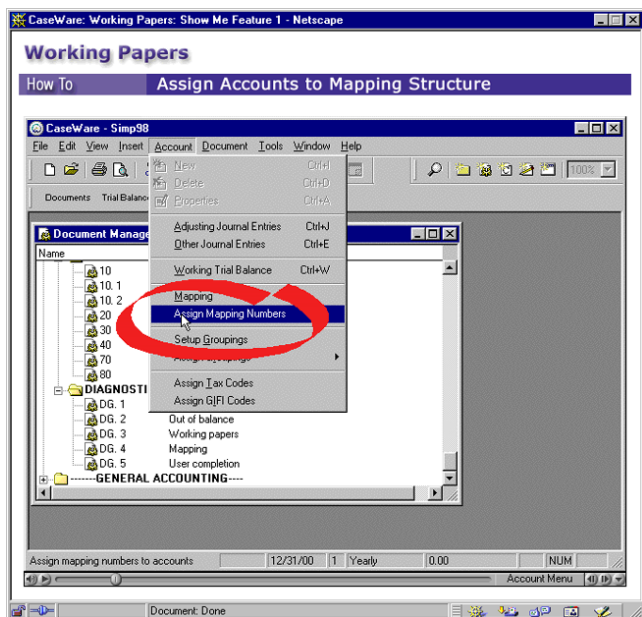
The levels introduce the company, its products, their key features, and explanations of how to accomplish tasks.

To illustrate how such technology may be used, we briefly present our work with CaseWare International, a firm that develops software for auditors. The ultimate goal was a series of short videos based on the hierarchical structure that appears in Figure 9. The bottom three levels of the hierarchy are useful as visual documentation and in support and training.



**Figure 10: A key feature video about “Working Papers”**  
The movie appears on the left; a list of “show me how” movies is on the right.

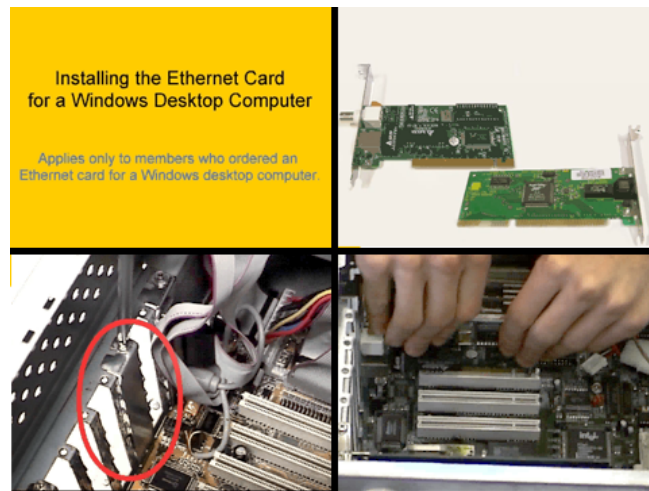
Figure 10 shows a typical display presented to a user viewing a movie about the “mapping” key feature of CaseWare’s “Working Papers” software. The user can then select a movie demonstrating and explaining how to do a task that makes use of the key feature (Figure 11).



**Figure 11: A Show Me How To Movie**  
This movie explains how to accomplish a task with a key feature of the “Working Papers” software.

The rapid growth of demand for high-speed Internet access in the home and the scarcity of trained installation personnel

have encouraged many subscribers to do the work themselves, which in turn led to our next example. Based on a usability analysis of typical problems encountered in this process, we produced a multimedia CD showing users what to do. Some frames appear in Figure 12.



**Figure 12: Another Show Me How To movie with hardware.**  
Sample frames showing how to install an Ethernet card.

In some cases, creators of video documents choose to disseminate them via a video cassette or a CD-ROM. Yet, increasingly, the Web and the Internet provide a more interesting medium for publishing and dissemination.

Creator’s hierarchical structure provides interesting opportunities for web publishing of video documents. The author is easily able to output the movie as a web document. This document appears on the Web in a way that displays the hierarchical structure that was part of the author’s original concept. Users can view the entire document, or can use this structure to navigate within the document and to access with ease portions of it.

This concept can be illustrated by a project we did for a firm called CanadaLeg, developers of the iWalkFree, a simple and elegant aid to walking which replaces crutches but can be used hands-free. The result was a 9-minute movie explaining how to assemble, fit, adjust, and use the device.

Publishing the movie in this form on the Web (Figure 13) allows a viewer to see and understand the structure of the production, and to access and view small chunks (scenes), larger chunks (acts), or the entire production. Access and viewing can either be by streaming or by downloading, based on tradeoffs between storage and bandwidth.

Because Creator encourages authors of visual demonstrations and explanations to enter a script while creating the video, users may easily search for portions of the video that use particular words or phrases in the movie structure or script.

## 7. BEHAVIOURAL FOUNDATIONS

A recent comprehensive review of the instructional effectiveness of video media [Wetzel, Radtke, & Stern, 1994] suggests that video’s strengths arise in situations where ... “[it] might provide additional visual forms of information to that available in descriptions given in text... and when learning procedural sequences might be benefited by conveying motion video compared to static or verbal

descriptions” (p. 210). It is also in the spirit of the approach of minimalist instruction [Carroll, 1990], in that it emphasizes training on real tasks, getting started fast, and reducing the amount of reading. Finally, appeal and enhanced motivation are clear strengths of the use of multimedia.

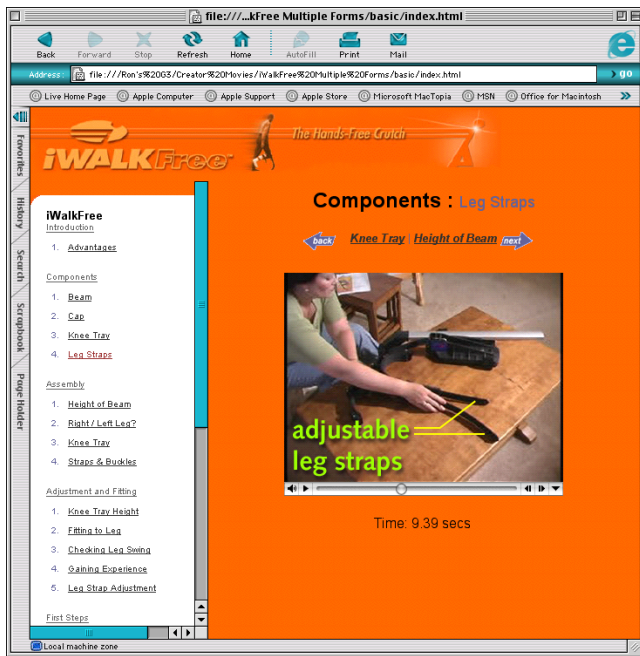


Figure 13. The published version of the iWalkFree movie

The outline is on the left. The movie is paused in the Leg Straps scene of the Components act. A click on one of the arrows moves the viewer to the preceding or following scene.

The literature [Wetzel, Radtke, and Stern, 1994] also points to a number of success factors for visualization. It is important not to use graphics in isolation, but to combine animation or video with text. In general, it is better if significant quantities of text are spoken rather than written. Not surprisingly, the proper design of visual material and the careful timing of spoken material are essential for success.

Dual coding theory [Paivva, 1990; see also Mayer and Sims, 1994] supports the synergistic use of text and image by asserting that multiple representations of a problem help learners build connections among the different representations and thereby to understand new concepts.

The literature also suggested that video media are best for applications involving visual/spatial domains, complex procedures, and problem solving. Although Wetzel, Radtke, and Stern [1994] make no mention of software visualization, it is clear that software explanations fit these criteria. There is also evidence that visual presentations are best for students with low prior knowledge and high spatial ability

Yet the evidence is not conclusive. There are many studies that “demonstrate” the effectiveness of dynamic presentations, and many that “demonstrate” just the opposite. One reason is that it is hard to do controlled experiments with “equivalent” presentations in text and video. Text is typically scanned and rescanned; traditional video has suffered from strictly linear presentations. Many studies are flawed because they do not allow switching back and forth between learning and practice.

There is strong evidence that the impact of visuals can decay significantly over time. Results are also highly dependent on the quality of the visualizations. For example, two studies that purport to show that animation and video are not valuable are Stasko, Badre, and Lewis [1993] and Bayles [2002]. Yet the former is flawed because the animation tested was not of excellent quality, and the latter was flawed because it tested a pointless use of animation in a Web site.

## 8. SUMMARY AND CONCLUSIONS

We have proposed the use of dynamic visual representations to augment traditional static text as documentation, showing people how to do things, and not just telling them what to do. We have presented and illustrated examples of five methods and technologies for creating such dynamic documentation, and summarized how little is known about their effectiveness.

Clearly these techniques are promising, but not proven. There is a need for more experimental evidence shedding light on their utility, as well as market response to the technologies and techniques. Yet there are substantial obstacles standing in the way of this happening. We have seen in working with our own clients that overloaded documentation staff members rarely have the time to try out new techniques. No matter how compelling are the visualizations, documentation specialists will still be required to document everything in text. Creating visualizations also requires new skills that they typically do not possess, so there are also training implications.

Yet, since many users neither read manuals nor “online help”, it is important to try to advance the technology, design discipline, and work practice of software visualization. We hope that this paper contributes towards this goal.

## 9. ACKNOWLEDGMENTS

Our work on visual streaming was supported by the Bell University Laboratories of the University of Toronto, and also by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the IBM Centre for Advanced Studies. Project co-director Dr. Gale Moore, Chief Architect Peter Wolf, Maciek Kowzloski (Videotelephony Inc.), David Torre, Anne Postic, Kelly Rankin, and Ben Mazzotta played key roles.

The work on animated icons was carried out at the Apple Computer Human Interface Group. Ian Small, Eric Martin, and Richard Mander played key roles, with the active encouragement of Joy Mountford.

Expresto Creator (see [www.expresto.com](http://www.expresto.com)) was developed primarily by Eric Smith and Dave Goulden, with lots of help from Agnes Ouellette, Anne Postic, John Hancock, Jonathan Hung, Kelly Rankin, Ilona Posner, Ali Mazalek, Mike Ananny, and others.

Expresto Creator is likely to be released open source sometime in the fall of 2002. For information, please email [creatorinfo@kmdi.toronto.edu](mailto:creatorinfo@kmdi.toronto.edu).

## 10. REFERENCES

- [1] Ames, A. (2001). Just What They Need, Just When They Need It: An Introduction to Embedded Assistance, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 111-115.
- [2] Baecker, R. (1982). *Sorting Out Sorting*, 30 minute colour sound film. available to order on video from <http://www.utoronto.ca/~ic/media/vidcol/science.html>

- [3] Baecker, R. (1998). *Sorting Out Sorting*: A Case Study of Software Visualization for Teaching Computer Science. In Stasko, J., Domingue, J., Brown, M., and Price, B. (Eds.), *Software Visualization: Programming as a Multimedia Experience*. MIT Press, 369-381.
- [4] Baecker, R. (2002a). Highly Interactive Webcasting with Structured Archives. Poster presentation at ACM CSCW'2002, New Orleans, LA., 16-20 Nov. 2002.
- [5] Baecker, R. (2002b). Design Principles and System Architecture for Rich Media Interactive Webcasting with Structured Archives, submitted for review.
- [6] Baecker, R., Booth, K., Jovicic, S., McGrenere, J., and Moore, G. (2000). Bridging the Gap Between What Users Know and What They Need to Know, *Proc. ACM CUU2000 Conference on Universal Usability*, 17-23.
- [7] Baecker, R., Rosenthal, A., Friedlander, N., Smith, E., and Cohen, A. (1996). A Multimedia System for Authoring Motion Pictures, *Proc. ACM Multimedia'96*, 31-42. (Reprinted in Jaffay, K. and Zhang, H.J. *Readings in Multimedia Computing and Networking*, Morgan Kaufmann, 2002, 836-847.)
- [8] Baecker, R. and Small, I. (1990). Animation at the Interface In *The Art of Human-Computer Interface Design* (B. Laurel, Editor), Addison-Wesley, 251-267.
- [9] Baecker, R., Small, I. S., and Mander, R. (1991). Bringing Icons to Life, *Proc. CHI '91*, 1-6.
- [10] Baecker, R.M. and Smith, E. (2002). Hierarchical Structure in the Digital Video Lifecycle. Draft manuscript.
- [11] Bayles, M. (2002). Designing Online Banner Advertisements: Should we Animate?, *Proc. CHI'2002*, 363-366.
- [12] Berglund, E. and Priestley, M. (2001). Open-Source Documentation: In Search of User-driven, Just-in-time Writing, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 132-141.
- [13] Buxton, W. (1989). Introduction to this Special Issue on Non-Speech Audio. *Human-Computer Interaction* 4(1).
- [14] Carroll, J.M. (1990). *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill*. MIT Press.
- [15] DiGiano, C. and Baecker, R. (1992). Program Auralization: Sound Enhancements to the Programming Environment, *Proc. Graphics Interface '92*, 44-52.
- [16] Finn, K., Sellen, A., and Wilbur, S. (Eds.) (1977), *Video-Mediated Communication*. Erlbaum.
- [17] He, S. (2001). Interplay of Language and Culture in Global E-commerce: A Comparison of Five Companies' Multilingual Websites, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 83-88.
- [18] Hsi, I., and Potts, C. (2000). Studying the Evolution and Enhancement of Software Features. *Proc. International Conference on Software Maintenance*, 143-151.
- [19] Horton, W. (1994). *The Icon Book: Visual Symbols for Computer Systems and Documentation*, Wiley.
- [20] Jackson, S. (2001). Editing Computer Hardware Procedures for Multimedia Presentation, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 68-72.
- [21] Mayer, E. and Sims, V. (1994). For Whom Is a Picture Worth a Thousand Words? Extensions of a Dual-coding Theory of Multimedia Learning, *Journal of Educational Psychology* 86(3), 389-410.
- [22] McGrenere, J. and Moore, G. (2000). Are We All In the Same "Bloat"? *Proc. Graphics Interface 2000*, 187-196.
- [23] Paivio, A. (1990). *Mental Representation: A Dual Coding Approach*, Oxford University Press.
- [24] Price, J. (2001). A Rhetoric of Objects, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 147-151.
- [25] Price, B., Baecker, R., and Small, I. (1993). A Principled Taxonomy of Software Visualization, *Journal of Visual Languages and Computing*, 4(3), Sept. 1993, 211-266.
- [26] Priestley, M. (2001). DITA XML: A Reuse by Reference Architecture for Technical Documentation, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 152-156.
- [27] Russell, J. (2001). "Yes, But Does It Scale?" — Practical Considerations for Database-driven Information Systems, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 12-21.
- [28] Silveira, M., Siechenius de Souza, C., and Barbosa, D. (2001). Semiotic Engineering Contributions for Designing Online Help Systems, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 31-38.
- [29] Spinuzzi, C. (2001). Software Development as Mediated Activity: Applying Three Analytical Frameworks for Studying Compound Mediation, *Proc. 19<sup>th</sup> Annual Conf. on Systems Documentation*, 58-67.
- [30] Stasko, J., Badre, A., and Lewis, C. (1993). Do Algorithm Animations Assist Learning? An Empirical Study and Analysis, *Proc. INTERCHI'93*, 61-66.
- [31] Stasko, J., Domingue, J., Brown, M., and Price, B. (Eds.) (1998). *Software Visualization: Programming as a Multimedia Experience*. MIT Press.
- [32] Wetzel, C., Radtke, P., & Stern, H. (1994). *Instructional Effectiveness of Video Media*, Erlbaum.