Web Tools and Techniques for E-Learning

Jack Massa

This paper is intended as an introduction to web-based training for technical communicators who are new to e-learning or want to move into e-learning. We focus on asynchronous, self-paced WBT programs rather than instructor-led or synchronous programs. The paper discusses what makes WBT effective and techniques for achieving success. It also reviews some authoring tools, with a focus on two prominent and affordable tools, Dreamweaver and Flash.

WHAT MAKES E-LEARNING EFFECTIVE?

Successful web-based training (WBT) is built through careful analysis, user-centered design, and iterative development and testing. These same principles apply to the design of any user interface or online information product, and are already familiar to many technical communicators. We'll discuss them briefly here as they apply to WBT.

Analysis

The first question to ask when considering any training project is: What problem are you trying to solve? Typically, this is a performance problem in a work situation. For example, customer service representatives need to handle calls quickly and with a high degree of customer satisfaction. If you work for a product vendor, the problem may be product-centric: network administrators must install and configure our software without having to call support.

Whatever the problem, a complete understanding of it involves understanding the audience, the tasks they must perform, and the definition of success for these tasks. Investigation of these issues yields audience and task analysis documents and helps determine:

1. Is training the proper solution to the problem *and*
2. What kind of training intervention is needed?[1]

Design

In *Designing Web-Based Training*, William Horton offers this encapsulation of "50,000 years of instructional design:"

1. Show them.
2. Tell them.
3. Let them try.
4. Repeat.[2]

While tongue-in-cheek, this description does present key guidelines for designing web-based training. Effective WBT teaches by both describing and demonstrating. It provides learners with opportunities to practice what they learn, and it encourages repetition of both the presentation and the practice.

E-learning authority Michael W. Allen lists three priorities for the effectiveness of any training:

1. Ensure that learners are highly motivated to learn.
2. Guide learners to appropriate content.
3. Provide meaningful and memorable learning experiences.[3]

The Techniques section of this paper will look at ways to implement these design principles in WBT projects.

Development

Finally, WBT projects are best developed through a process of rapid-prototyping and testing. This allows you to test all aspects of the design with representative learners and to refine the product on an ongoing basis. Allen refers to this process as "successive approximation." He suggests dispensing with design specs and paper storyboards in favor of building and testing working prototypes.[4]

TECHNIQUES

Tell Them First - Overview and Concept Pages

One venerable formula for instructional design reads as follows: "Tell me what you're going to tell me, tell me, then tell me what you just told me." This sequence of preview, presentation, and review helps reinforce learning and retention.

WBT often needs pages that simply tell facts or present information. These include overview pages that introduce each lesson and list the contents. They may also include pages that explain or illustrate concepts, in a way similar to online documentation.

These techniques help make overview and concept pages effective:

- Keep text short. Write as concisely as possible. If more text is absolutely needed, present additional pages.
- Illustrate ideas with pictures and diagrams. Learners are typically conditioned by the Web to expect graphics. If at all possible, hire a graphic designer for your project. If there is no budget for this, make optimal use of available artwork and clip art. When introducing a concept, strive to present it in both text and pictorial form.
- For Overview pages, provide links to all topics. This is recommended in addition to the typical Next and Previous buttons, to let the learner choose the path through the content.

Show Them - Demonstrations

Demonstration pages allow learning by observation. For example, learners can watch an animation representing a task being performed in a software interface, or read and listen to an example conversation between a customer service representative and a customer.

Here are some guidelines for demonstration pages:

- Keep them simple. If the demonstration is long or complex, divide it over several pages.
- Test the timing of animations on various computers, browsers, and with as many learners as possible. This is particularly important if the animation includes text for learners to read, since people read at various speeds.

- If using audio, provide a mute button for learners who may not be able to listen in their environment. Be sure to provide a text version for learners who may not have audio available.

Let Them Try - Interactivity

A hallmark of successful WBT is a high degree of interactivity. Adult learners usually learn best by doing, and interactive exercises take advantage of this.

Question-and-answer quizzes placed at the end of a lesson represent the most elementary form of interactivity. These may include multiple-choice, fill-in-the-blank, drag-and-drop questions or other methods. Done well, quizzes can reinforce the learning of concepts. However, they are not useful for teaching skills, are not particularly engaging, and are ignored by many learners.

Effective WBT uses other methods to foster interaction. These may include:

- **Layering information on concept pages**. For example, an illustration of a machine, a computer network, or a software diagram can include hotspots on specific components. When a learner clicks or points to the component, additional information pops up in a dynamic layer.
- **Software simulations**. For software application training, WBT often includes a simulated interface that allows the learner to enter data and perform transactions. This works best when the simulation requires the learner to solve a real-world problem and displays realistic results.
- **Problem-solving scenarios**. For soft-skills training, scenarios present learners with problem situations and various options for solving them. Each option may lead to a different path for the learner to explore. While the development overhead of creating multiple paths can be a burden, the value in terms of learner success often makes this approach worthwhile.

Engage the Learner - Stories

Recall the first and third of Michael W. Allen's priorities for effective training:

- Ensure that learners are highly motivated to learn.
- Provide meaningful and memorable learning experiences.

The best recent examples of WBT I've seen answer these challenges by engaging the learner in stories. Stories engage the learner's emotions and make the learning experience memorable. Also, because the characters and situations relate to the learner's real-world challenges and problems, stories enhance motivation.

Here are some examples of WBTs that effectively use stories.

- To introduce the various steps needed to implement a software system, a WBT employed the simple metaphor of a roadmap. The learner started each lesson by clicking a sign along the road.
- To teach the use of a communication application, a story was created involving a space station. Surgeons on earth needed to treat a critically-ill patient on the station through telemedicine. But first the learner had to set up the communication apparatus.
- To train technical support staff, a WBT was given the personality of a superhero. In each lesson, the learner identified with the superhero, coming to the rescue of different computer users.

- To teach corporate sales representatives to use the company e-mail system, a WBT was designed in the form of an interactive game. The learner was placed in the role of a spy receiving mysterious e-mails and had to learn the e-mail system to complete her mission.

While some of these projects enjoyed big budgets and expensive production values, these are not necessary and are not the crucial point. With a little imagination and creativity, even very low-cost WBT projects can involve learners through stories.

Provide Maximum Learner Control

Allen's second priority is to "Guide learners to appropriate content." This means making the WBT content easy to find. It does *not* mean providing only one path through the content. Horton recommends providing WBT courses with multiple access methods, which may include a table of contents or menu, an index, a course map, and even a search function.[5]

Whether you find it feasible to implement all of these options, it is important to give the learner navigational control. Let the learner choose which topics to learn and in what order.

Each page should provide Next and Previous links, as well as links to the start of the current lesson and the course home page or main menu. Each lesson overview page should allow the learner to jump to any topic in the lesson.

If possible, allow learners to bookmark their place so they can return at a late time.

TOOLS

Despite a shakeout in the market in the past few years, many authoring tools are available that will help you create effective WBT. These range from general-purpose tools designed for presentations and web-authoring to dedicated e-learning applications. The tools cover a broad range in terms of price, sophistication, and how easy they are to learn.

Macromedia Dreamweaver

Dreamweaver is an excellent starter tool for WBT and general web-page authoring. Essentially, Dreamweaver automates the authoring of web content by generating HTML and JavaScript code through a WYSIWYG interface. Among its many capabilities, Dreamweaver makes it easy to create animations and show/hide layers for WBT pages.

If you have Dreamweaver, you can download free extensions for e-learning. Once installed, these added modules become available through the Dreamweaver interface. They allow you to create quizzes, including drag-and-drop exercises, and also to track learner results and output them to standards-compliant learning management systems (LMS).

Dreamweaver is fairly quick to learn and develop on. However, it is not as powerful for WBT as many other tools. It's capabilities are limited to what can be supported through the of native HTML and JavaScript functionality of web browsers.

Macromedia Flash

Though it was initially designed for creating small and fast-loading web animations, Flash has evolved into an industrial-strength tool for web applications. More and more, it is becoming a development tool of choice for WBT.

With Flash, you can do everything you can do in Dreamweaver plus much more. While Dreamweaver's output is limited to what a native web browser can support, Flash's output is practically unlimited. This is because Flash creates its own application file (SWF file) that runs inside the browser window but does not depend on the browser's native capabilities.

Flash is excellent for creating animations, exercises, and simulations of all kinds. It supports rich media, including audio and video. Added to this, its modest price makes it a very attractive tool for even low-budget WBT projects.

The only drawback of Flash as opposed to some other tools is its complexity. The learning curve can be steep. And, while many effects can be attained through the user interface, you need to write code to unleash Flash's full power. Flash's Actionscript language is very similar to JavaScript.

Other Tools

Depending on your project and budget, here are some other tools you might want to investigate:

- DemoShield by InstallShield Corporation
- RoboDemo by e-Help
- Microsoft PowerPoint
- Macromedia Director
- Macromedia AuthorWare
- Macromedia Breeze

CONCLUSION

For organizations of all sizes, putting training on the Web has many well-known advantages in terms of accessibility, efficiency, and cost. The key to realizing these advantages is effective design that motivates learners and gives them accessible and memorable learning experiences. With their existing skill sets and the tools and techniques discussed in this paper, technical communicators have the opportunity to create effective e-learning.

Notes

1 For an accessible introduction to these issues, see Piskurich, Chapters 1-4.
2 Horton, page 15.
3 Allen, pages 59-60
4 Allen, Chapter 4
5 Horton, Page 121-125

Sources

Michael W. Allen, *Michael Allen's Guide to E-Learning*. John Wily & Sons, 2003.
William Horton, *Designing Web-Based Training*, John Wily & Sons, 2000.
George M. Piskurich, *Rapid Instructional Design*, Jossey-Bass/Pfeiffer, 2000.

Jack Massa is a freelance writer and online information developer based in Atlanta. Since 1984, Jack has written and edited marketing copy, training, documentation, and articles covering all kinds of technology. He's designed and developed web sites, help systems, multimedia projects, and web-based training; taught workshops at industry conferences; and managed teams for large and small companies. Jack founded Guidance Communications, Inc. in 1996. In addition to STC, Jack is a member of the Metro-Atlanta Chamber of Commerce, the Technology Association of Georgia, and the Atlanta Chapter of ASTD. He is the web master for the Atlanta-based Freelance Forum and the manager of the STC Atlanta Online Communication SIG.