

10 Tips for Using Flash in e-Learning

By AI Lemieux, Senior Designer, SyberWorks, Inc.

In our last article, we looked at using graphics in e-Learning. Now, we'll examine another important tool for creating successful course interactions – <u>Adobe Flash™</u>. Flash is a vector-based animation and interactivity program that allows you to create everything from simple animations to complex interactive applications. With its built-in programming language – ActionScript – Flash is fully scriptable and can communicate with several backend database languages. Since the first Flash product appeared (from Macromedia), the e-Learning community has worked with Flash developers and engineers to make Flash content more and more accessible in e-Learning environments.

With its feature-rich list of capabilities, built-in components, and e-Learning output templates, Flash makes a lot of sense for maximizing course content. In this article, we'll highlight some of the important features of Flash and how they can make your courses shine.

1. Animations for Step-by-Step Procedures

In *10 Tips for Using Graphics in e-Learning*, we saw the importance of using graphics in courses, to strengthen how well they meet learning objectives. We talked about different delivery methods for conveying subject matter, and one of those methods was animation. Flash uses timeline-based animation, which makes it quite easy to build animated movies. Graphical elements may be created directly in Flash, but a majority of developers use other tools that they are more comfortable with – especially *Adobe Creative Suite*. With Flash CS3, you can now import native <u>Adobe Illustrator</u> and <u>Adobe Photoshop</u> files. You can even convert "layered" files from these programs into Movie Clips and animate the individual layers using keyframes in the timeline.

In e-Learning materials, step-by-step procedures can be particularly tricky. Think of the last piece of furniture that you had to put together. How useful were its instructions? One of the things IKEA does well is its non-text instructions for assembling its furniture. These instructions are pure imagery. Take that concept a step further, and you have animation. An animated step-by-step instruction is *much* more powerful than static imagery. During an animation, you can highlight specific areas, use animated arrows, and express a learning objective much more naturally and effectively.

When the <u>American Contract Bridge League</u> (ACBL) asked SyberWorks to produce a series of courses for its Bridge-playing members, we used Flash to create animated card-playing sequences, and to illustrate specific turns in games. The card images were prepared in Adobe Illustrator, imported into Flash, and converted into Movie Clips. Using scripts provided by ACBL, we created animations for each play, to illustrate the rules that it represents. Other graphical and text elements were added in Flash to point out parts of the animation. These courses are now delivered to hundreds of Bridge players across the country.



411 Waverley Oaks Road Building 3, Suite 319 Waltham, MA 02452 www.syberworks.com 1-888-642-7078



2. Easy Audio Integration

Adding audio to an HTML-only course presents difficulties for course developers. First, multiple audio formats are available on the web today: WAV, AIF, MP3, M4a, etc. Even worse is the vast array of audio-player applications and their file-type associations, both for Mac and Windows platforms. Having users download specific players is a distraction from the course material, and a potential source of additional problems.

Flash, however, supports several audio formats and plays them all using the Flash Player, which has a 98% market penetration. The Flash Player is available in all major browsers and platforms, and is even becoming available in more Internet appliances. You can easily import almost any audio file, add it to the Flash timeline, publish the Flash movie, deliver the course, and be fairly certain that users will be able to hear the audio without having to download an extra player.

Another problem with audio is that, if a sound file is not set up to "stream," there is a chance that some of the sound may drop out. Flash has built-in streaming capabilities, so you can be assured that your audio content will be delivered to users without drop-outs and gaps.

At SyberWorks, we use Flash for in-course audio content. Audio is recorded in our audio lab and saved in a common format. The file is then imported into Flash. ActionScript is used in conjunction with onscreen buttons, to allow users to control audio playback. This interactive audio file can then be attached to any course, using the SyberWorks Web Author application.

3. Course Navigation

How smart is your course navigation? By "smart," I mean can it communicate with your LMS? Most incourse navigation is pretty boring and perfunctory. If you spice up yours with Flash, your courses will become more useable.

Since Flash can work with <u>SCORM</u> and <u>AICC</u> courses, on-screen buttons can be scripted via ActionScript to send calls to the LMS for certain conditions. A button at the end of a lesson might send a finish statement to the LMS, signifying lesson completion and triggering further actions. Another button might automatically submit the results of a quiz.

Better learning experiences are possible, since the entire interface is completely customizable, and buttons are very easy to create in Flash. If you want, you can even use its built-in library of buttons to create your course navigation.

At SyberWorks, we retro-fitted some existing e-Learning courses (developed for restaurant employees) for our LMS, complete with all of the calls necessary for course status and completion. These Flashbased courses had a cohesive look and feel that was much more compelling than a plain-text HTML web experience.





4. Creating Simulations

Some software tools, such as <u>Camtasia</u> and <u>Captivate</u>, are designed solely for creating software simulations. They have advantages and disadvantages, and SyberWorks does use them, along with other simulation software packages. But what if the software you need to create a simulation for isn't available on your platform, or only works behind a client's firewall?

When the <u>Palm Pilot</u> came out, I had the opportunity to create a software simulation that showed new owners how to use the device. Since the Palm software only resides on the Palm Pilot, I couldn't use a PC-based software-simulation tool to capture screens. So I used my scanner and Photoshop, imported the screens and artwork into Flash, and created the simulation.

At SyberWorks, we are currently working with an Internet company whose proprietary software (for security reasons) only works on computers behind their firewall. Since we can't install or use their software in-house, we had to go to the client's location to take screen shots. Those images were then imported into Flash, captions and highlights were added, and the course material came to life.

5. Video

Video is becoming more prevalent on the web these days, with the rise of <u>YouTube</u> and other videostreaming services. Video can significantly enhance course materials, but (as with audio) it can be difficult to manage, due to multiple video formats and players. On the PC, AVI is the dominant format and <u>Windows Media Player</u> is the usual player. On the Mac, MOV is the format of choice and the <u>Quicktime Player</u> from <u>Apple</u> is the dominant player. However, there are other video formats and players, including <u>Real media</u> / Real Player and others.

When Flash MX 2004 was released, video was a major component. And with Flash CS3, video has become even easier to incorporate. You can deploy video on your own server or on a dedicated <u>Flash</u> <u>Streaming Video Server</u>, for extra bandwidth. Flash has its own video format, called Flash Video (FLV). Flash videos can be played by any Flash movie (SWF) file, so no additional player is necessary. In fact, a majority of the movies found on YouTube were done in the Flash video format.

A short demonstration video, not longer than 1 minute, can be embedded in a Flash movie. Anything longer than that can be linked to a Flash movie file and set to progressively download from your server. If you are expecting a lot of traffic and have longer movies, you should use a Flash Video Streaming server. There are many of these FMS services out there, and SyberWorks uses <u>UpStream Networks</u> to host our larger videos.

With Flash and ActionScript, it is also possible to do more than just play videos. Cue Points can key off of specific video frames, to control other events. And the new captioning feature can add closed captioning for video segments. At SyberWorks, we created a video of a speaker presenting a slideshow. Cue Points in the video triggered relevant text to appear, and made it seem that the viewer was actually on-site, watching the original presentation.



411 Waverley Oaks Road Building 3, Suite 319 Waltham, MA 02452 www.syberworks.com 1-888-642-7078



But keep in mind that video is processor-intensive. Consider your audience and the minimum system requirements necessary to view streaming Flash videos. Dropped frames and inconsistent playback can occur on slower machines.

6. Drag-and-Drop Interactions

A simple multiple-choice quiz can be effective in testing situations, but drag-and-drop interactions can add many useful features of an interactive process. With drag-and-drop interactions, you can simulate experiences that mimic the real world.

Using Flash and ActionScript, drag-and-drop interactions can be created from scratch, using **hitTest** properties and **if** conditionals to test whether an object has been dropped on a specific target. If the object has landed on its correct target, the application shows a correct response or visual/audio feedback. If the object is not dropped on its correct target, an incorrect response or visual/audio feedback can occur.

In Flash, you can access built-in drag-and-drop (and other) interactions, by going to the **Window** menu, then to **Common Libraries**, and then to **Learning Interactions**. The interactions are ActionScript 2.0 only, but have all of the features necessary to create many desired interactions. You can then use the Flash **Component Inspector** to configure interactions, change their text, and alter the conditions of their tests.

At SyberWorks, we created drag-and-drop interactions to meet the many custom-content needs of our clients. Doing a custom application gives us complete control over both the look and feel of course environments and the parameters of their interactions. Audio also helps to give user feedback for every action, and a final screen tells users how effectively they completed the exercises.

7. Components

Flash comes with ready-made user-interface components that you can add to courses and configure through the **Component Inspector** and the **Parameters** panel. These components include: Radio Buttons, Checkboxes, Text Fields, clickable Selection Buttons, List Boxes, and Combo Boxes. Components in Flash CS3 are easier than ever to customize; simply double-click on a component and edit its "skin."

ActionScript can call on a specific object and test whether its condition matches a Boolean value, as in a Radio Button or Checkbox. You can even use ActionScript to test for specific conditions and return certain values, to create your own quizzes and test interactions.

You can also dynamically update the text for components from an external text file. So application development is much easier now, using Flash Components and ActionScript.





8. Dynamic Text

As mentioned above, Flash lets you pull text into Dynamic Text fields. You can import variables from a simple text file, or load text from an HTML or XML file. Flash CS3 has expanded its capabilities to handle XML files, with E4X.

You can also use a loader object to bring external text into text fields. This loader object has specific conditions that when met, will display text when the loading is completed (or display an error message if a problem occurred). Dynamic text can be formatted through an external CSS file, or using simple HTML tags. But any text field that will be accepting dynamic text *should have all font characters embedded or set to a system font.*

The major advantage of using dynamic text is flexibility. When the content of your application changes, it's much easier to change an external text file than to change text embedded in a Flash file. Text that is imported dynamically is cached by the Flash Player, so be sure to clear your cache when you update the content of your app.

9. Built-In Quiz Templates

In addition to its built-in learning interactions, Flash also offers Quiz templates. Simply go to **File**, then to **New**, and in the **Templates** section, select the **Quiz** category.

There are three Quiz styles to choose from. At the left of the stage, a Control component provides instructions and is governed by the Component Inspector. You can make adjustments to the entire Quiz and choose result options. Each frame in the Quiz template has its own set of learning interactions for you to choose from. So when you need a multiple-choice question, you can copy and paste the frame for multiple-choice questions and modify it as needed.

It's very easy to set up a Quiz from this template. And by default, the **Publish** settings are set to **SCORM 1.2**, for sending the template's Quiz results to an LMS. We'll talk more about that in the next section.

10. Built In SCORM/AICC Support

Flash supports the SCORM 1.2 and 2004 standards, as well as AICC. Any application that you develop in Flash (that needs to communicate with the SyberWorks LMS) must be "published" using these templates. Go to the **File** menu, and then to **Publish Settings**. Click the **HTML** tab and choose the appropriate template for the type of tracking you want.

In an HTML file that has been published by Flash, you will find a large JavaScript file, containing all of the functions and variables necessary to populate the Application Programming Interface for the LMS. There are also functions for connecting Sharable Content Objects with the LMS.





But keep in mind that Flash does *not* generate a manifest file for its published content. So you may need to use an external manifest application to collect all of the components in your application, for upload to an LMS. (A manifest file is a library of all of the assets required for a course, including SCO's, HTML, SWF, and other files.) The SyberWorks LMS supports all SCORM 1.2- and AICC- compliant courses, and our customer-service representatives can help you configure your manifest files, for hosting on our LMS.

Summary

If you want to give your courses a lot more zing, Flash is a perfect tool for building richer learning experiences. Custom Flash development may take longer than traditional course development, but the extra time it takes is more than offset by improved, more interesting, learning experiences for students.

And with Flash, you *never* need to worry about students being able to view your course materials, including audio and video. Built-in components, learning interactions, and quizzes, plus support for SCORM and AICC standards, make Flash an attractive choice for creating compelling e-Learning courseware.

About the Author:

Al Lemieux is a Senior Designer at SyberWorks, Inc. SyberWorks is a custom e-Learning solutions company that specializes in Learning Management Systems, e-Learning solutions, and custom online course development. Al is also a frequent contributor to the <u>Online Training Content Journal</u>.

The Online Training Content Journal

The Online Training Content Journal blog looks at best practices, techniques, and trends in online training development and e-Learning instructional design.



411 Waverley Oaks Road Building 3, Suite 319 Waltham, MA 02452 www.syberworks.com 1-888-642-7078