THIS WEEK: Development Techniques

Build a SCORM 1.2 Conformant Course with Lectora 2007

By Nicholas Bird

The world of e-Learning content development is still a complex and inconsistent place. In many ways, content is still the "Wild West" of learning. There is a variety of tools available for companies to use to develop e-Learning. Many of these tools can be used to produce simple to complex e-Learning courses. These tools are valuable to companies because they are designed to simplify the process of packaging SCORM content for use and tracking in the Learning Management System (LMS) of your choice.

In my experience with developing content for various corporations, the tools are often not well understood and the people responsible for using them are offered limited training. While the tools are easy to use, they all have features and nuances that the training never seems to cover. I often joke with people that the most important questions you will have about a tool will never occur to you during training. This means that the communities supporting the tools, and the experiences people share, are the most valuable assets available to a content developer.

In this article, I will discuss the common practice of developing content with the intent of getting it to seamlessly integrate with a LMS. The value of this article will be in the tips and tricks for making SCORM 1.2 conformant content that you create with Trivantis' Lectora Professional Publishing Suite 2007 SP5 (5100) work properly. Specifically, I will focus on content that includes a scored assessment, and I will address communicating achievement of the required min-

Lectora is a very popular tool for developing e-Learning content. Because many developers need to create courses that will integrate seamlessly with a LMS, this week's article focuses on some tips and tricks for making SCORM-conformant content with Lectora. If you are working in Lectora for the first time, you definitely should take advantage of the information here!

A publication of

THE eLEARNING GUILD

Extra Insights
Page 8

imum score to the LMS. This should prepare you to build similarly structured content that will work in your LMS.

(*Editor's Note:* If you are not familiar with Lectora, you may want to read Len Perkins' article, "Using Lectora to Repurpose e-Learning Content – Can It Really Deliver?" published August 7, 2006 in *Learning Solutions.* Perkins' article provides a more introductory view of working with Lectora.)

Templates and content structure in Lectora

In my experience, one of the elements you must master for success with Lectora is the proper use of course templates. The version of Lectora Professional Publishing Suite described here provides the option to "Save Title as Template" on the File menu. This is important, as it allows you to build a shell that is common to all of your courses, and to save it for use as a starting point in future content development. This should be very exciting for anyone interested in rapid e-Learning development, or in content development standardization.
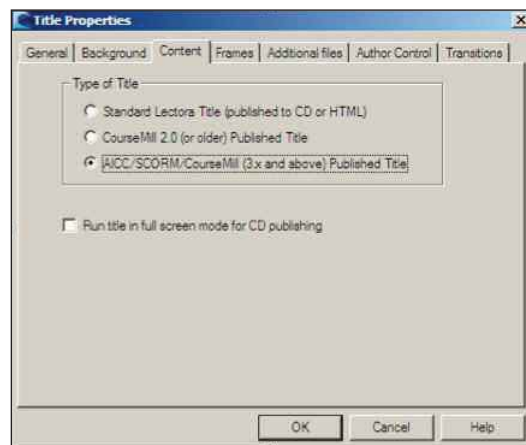
Lectora offers a wide variety of options for creating custom e-Learning content. In this article I will limit discussion of specific presentation layer objects, such as buttons, pages, and actions, to instances where the

use of the objects is critical to the performance of a course loaded into a SCORM-conformant LMS.

A basic template would contain these basic elements, and I will describe them as you would use them to build a SCORM-conformant course:

• A Title (Course container) and an Assignable Unit (Lesson)
• A Chapter and one or more Pages (Module)
• A Test

I'll also describe what it takes to communicate with the LMS, and the publishing process.

Figure 1 *Title Properties Window*

Learning Solutions e-Magazine™ is designed to serve as a catalyst for innovation and as a vehicle for the dissemination of new and practical strategies, techniques, and best practices for e-Learning design, development and management professionals. It is not intended to be THE definitive authority ... rather, it is intended to be a medium through which e-Learning professionals can share their knowledge, expertise, and experience. As in any profession, there are many different ways to accomplish a specific objective. **Learning Solutions** will share many different perspectives and does not position any one as "the right way," but rather we position each article as "one of the right ways" for accomplishing an objective. We assume that readers will evaluate the merits of each article and use the ideas they contain in a manner appropriate for their specific situation.

The articles in **Learning Solutions** are all written by people who are actively engaged in this profession – not by journalists or freelance writers. Submissions are always welcome, as are suggestions for future topics. To learn more about how to submit articles and/or ideas, please visit our Web site at www.eLearningGuild.com.

The Title and Assignable Unit

In Lectora, the Title is the container that holds all the course objects. A Title can contain multiple Assignable Units. Assignable Units will be explained in more detail shortly, but basically they are lesson-size learning objects.

The Title has a variety of properties on various tabs, (see Figure 1 on page 2), but there is one critical value. To add an Assignable Unit to a Title, the **Type of Title** radio button on the **Content** tab in the **Title Properties** window must be set to: **AICC/SCORM/Course-Mill (3.x and above) Published Title**, as shown in Figure 1. Lectora automatically creates an Assignable Unit when you do this.

Title: Child objects

The Title has child objects defined for the entire Title. You can build these objects into a Lectora template to avoid further modification. A typical example would be creating actions and a text block that defines variables used to display a page counter. You can use the value of the variables set by the actions to assign a modifiable page count value to a Text Block object. The Assignable Unit can then inherit these objects, and use them to display a page counter such as "Page 1 of 25" on the screen.

The Assignable Unit

In order to be SCORM-conformant, the course must contain an Assignable Unit. The Assignable Unit defines

the maximum score on the Test, and the mastery score the learner must achieve in order to receive completion credit. You can track each Assignable Unit in a course.

For the example in this article, the course will contain one Assignable Unit with one graded test. The required values set for the Assignable Unit can then become part of your template. This makes it possible to enforce consistency between courses, and to ensure proper configuration of items built using the template. In my experience developing content in Lectora, I have found it desirable to structure content this way to simplify the build and tracking of tests. You can also build untracked knowledge checks into the courses as needed to help the learner master content and prepare for successful completion of the test at the end of the course.

It is important to understand that the Assignable Unit needs the values shown in Figure 2 on page 3 for Maximum Score and Mastery Score. The Maximum Score is the best score the learner can receive on the test for this Assignable Unit. The Mastery Score is the score the learner must get in order to receive credit for completing the course in the LMS. The defaults, 100 and 75, respectively, do not appear in the Properties Window when it opens, and the developer can leave those fields blank (see the field labels in Figure 2). However, I personally prefer to explicitly define these values (by typing them in) to avoid confusion. All of the other values shown in Figure 2 are the defaults that appear when the Window opens. In this case, there is no time limit for the Test.

Assignable Unit: Objects

A series of objects can be added directly under the Assignable Unit object to define global features such as user interface images, navigation buttons, a progress bar, and the table of contents. (See Figure 3.) You should build these objects into the template and not modify them. If you built the template properly, these objects will be sized, positioned, and formatted as needed in the template. You can also add pages as children to the Assignable Unit if desired.

Chapters and Pages

In Lectora you can use Chapters to organize the content of the course. Every learning organization generally will use their own language for groups of pages. Others might refer to a Chapter as a Module or Lesson. Each chapter will consist primarily of Pages and the objects for those Pages. Chapters can also contain Actions if variable manipulation is required. For example, if you have a Chapter that does not connect linearly to the content, you may want to hide the page count. An example of this could be Course Help. You would organize such a feature in a Chapter, but it would only be accessible through a link from within the course.

For the purposes of building a template in Lectora, you would want to provide a sample Chapter that shows the developer the base formatting. This may include any Actions for a typical Chapter, as well as critical Pages within a Chapter that have specific formatting. An example would be the first and last Pages of a Chapter. If they are the same in all content Chapters, it is beneficial to build their layouts into the template.

If the content has a specific Help Chapter, or other content that is consistent across all courses, these Chapters should be built completely and be part of the template. Typically the course will have a Help Chapter for items such as navigation, support, assessments, and so on. The more work the template can eliminate for the developer, the better.

Pages

The Page in Lectora is the presentation object. The Page allows you to present content to the learner in the form of text and graphics, audio, video, and animation, to name a few of the more common options. As with the Chapter, the Page itself requires little formatting, once set up. As a development team, you will determine whether Pages appear in the Table of Contents, the naming convention for each Page, what objects the Page does and does not inherit, any transitions for the Page as the course moves from page to page, and the use of metadata for the Page.

The children of a Page object include all the items positioned on that Page. Typical objects would include text blocks, images, audio, and video.

Page properties

The name of each Page in Lectora is important. You should give each Page a descriptive and unique name. No two pages should have an identical name, as duplications can generate warnings during the publishing process.

Using the Inherit tab (see Figure 2), the developer determines which objects the Page will include or exclude. What is available to that Page is dependent upon what the parent object has inherited. For example, suppose you have built your navigation buttons as children to the Assignable Unit. If you want them available on a page in a Chapter within the Assignable Unit, the Chapter cannot exclude the inheritance of those buttons. If the Chapter has excluded the buttons, the Page does not have the ability to include or exclude those buttons.

Metadata can be added for every resource object (that is, animated image, audio, document, image, video) or page object. For SCORM-conformant courses in Lectora, metadata publishes as standard XML metadata into the accompanying *imsmanifest.xml* file. Lectora has very useful help documentation for creating metadata for an object. You can manage metadata from the Property tab for an object.

While the tools are easy to use, they all have features and nuances that the training never seems to cover. ... [T]he most important questions you will have about a tool will never occur to you during training. This means that the communities supporting the tools, and the experiences people share, are the most valuable assets available to a content developer.

Figure 3 *Assignable Unit Objects*

Objects in Lectora

Lectora allows the developer to incorporate a wealth of objects into the content of the course. This section will briefly identify the objects that Lectora 2007 SP5 supports. There are two classes of objects (more or less), and I will refer to these as Lectora objects and Imported objects.

What does that mean? There are objects that are already fully built within Lectora (see Table 1). There are also files that are added to the course in Lectora but which require externally imported files to complete the object build (see Table 2).

Tests

While Lectora offers both Tests and Surveys, the current example only includes a Test. The simplest way to describe the difference between a Test and a Survey is that a Test has scoring options on the Results tab of the Properties window.

There are nine tabs on the Test Properties window, but for our purposes I am only going to focus on four of the tabs. These are control properties that support the SCORM conformance of the course.

Test Content Tab

The Content tab in Figure 4 offers options to the course developer that determine how the test will function. For the purposes of this example, I chose to show the learner feedback for each question, and to require that the student answer all of the questions on the test.
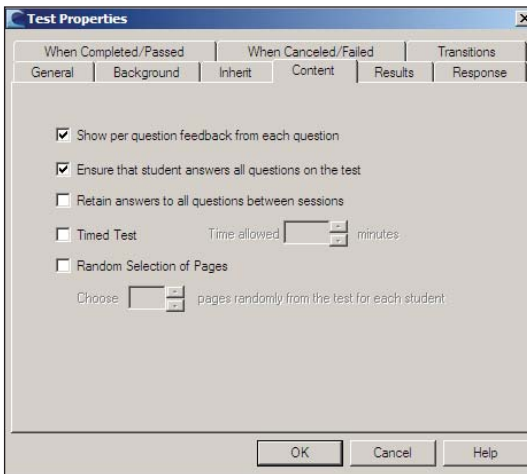
Figure 4 *Test Properties — Content Tab*
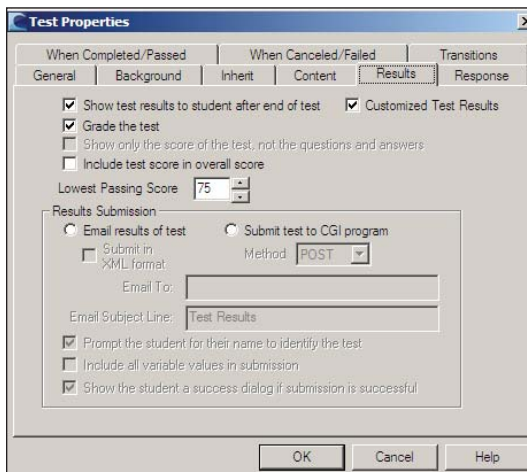
Figure 5 *Test Properties — Results Tab*

Table 1 *Lectora Objects*

Object Type	Description
Equation	Modifiable like a Text Block, and published as an Image
Form Object	Lectora help defines a Form as providing a means of communicating a collection of data outside the Lectora title.
Menu	You can build a custom menu to provide the learner with specific functionality, such as a non-linear navigation menu.
Progress Bar	You can use this feature to display a customizable progress bar on the screen.
Reference List	Gathers, and displays in a list, all of the references captured for the objects in the Title.
Table of Contents	Displays a list of the contents of the course usable for navigation.
Text Block	Used for adding textual content to a Page object.

Table 2 *Imported Objects*

Object Type	Supported Formats
Animation	GIF, SWF, SPL
Audio	WAV, MIDI, AU, AIFF, MP3, WMA, Real Networks Streaming, Windows Media Streaming
Button	GIF, BMP, PNG, JPG, JPEG, TIFF, WMF, EMF
Document	TXT, RTF
External HTML Object	Java Applet, Header Scripting, Shockwave Movie, Flash Movie, Other, ASP scripting, JSP scripting, Meta tags, Top-of-file scripting, PHP scripting, Cascading Style Sheet, ColdFusion, ASP.NET (.aspx)
Image	GIF, BMP, PNG, JPG, JPEG, TIFF, WMF, EMF, Clip Art
IPIX Image	IPX, IPS
Video	AVI, MOV, MPG, RM, WMV, ASF, Real Networks Streaming, Windows Media Streaming

Test Results Tab

The Results tab in Figure 5 on page 5 is important to ensure the test communicates properly with the LMS. (You do not need the Results Submission section of the Results tab for SCORM communication of the test results.) SCORM-conformant content uses SCORM communication to pass values to the LMS. In this example, we chose to show test results to the learner at the end of the test. To meet customer requirements for presentation, we customized the results. We selected the option to Grade the test. Each test question has a weighted value, which the system tracks to determine if the minimum passing score of 75% is achieved.

When Completed/Passed Tab and When Cancelled/Failed Tab

The developer has the ability to control the path of the learner through the course, depending on whether the learner has passed or failed the test. In my examples in Figures 6 and 7, you can see that we chose to allow the learner the opportunity to retest as needed. The flow of the course was controlled by creating a Chapter for pass and fail scenarios. Each Chapter had unique content based on their result. Achieving at least 75% on the test takes the learner to a page within the success chapter. That page informs the learner of their successful completion, and provides instructions for exiting the course. More importantly, we added two actions that modified the necessary course variables to the success page as children.

Communicating with the LMS

In this course, we needed to capture and communicate to the LMS two critical pieces of information. The first variable tracked was AICC_Score. As defined in Lectora's help, "AICC_Score has the AICC score for the title that is the sum of all of test scores in the course. If you modify the value of this variable within the content of a course that contains more than one automatically graded test, the modification may/will be lost whenever the student completes an automatically graded test." We chose to create an action on the success page to force the value of this variable to be set as shown in Figure 8. This ensured that we send only the final score from their attempts to the LMS.

The second item we chose to track was AICC_Lesson_Status. Again as defined in Lectora's help, "AICC_Lesson_Status contains the current status of the course within the AICC LMS. Lectora sets the value of this variable to "incomplete," and leaves the completion status of the course up to the LMS to determine, based on the current *AICC_Score* for the student and the value the instructor used for the Mastery Score when

the course was published." We chose to set parameters around this variable as shown in Figure 9 on page 7 to control when the value is set to complete.
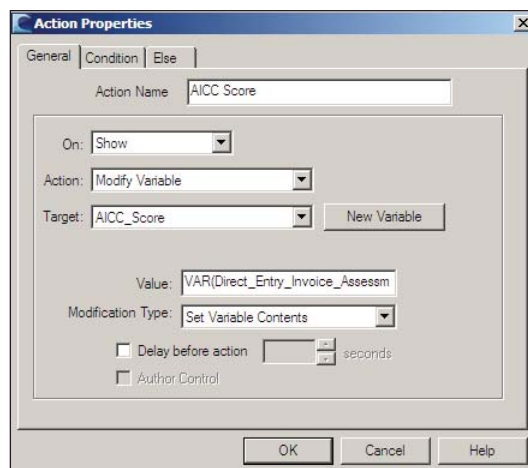
For this course the Condition shown in Figure 10 on page 7 demonstrates how the LMS would not display a Completed status until the learner received at least a 75% score on the course test. The variable in
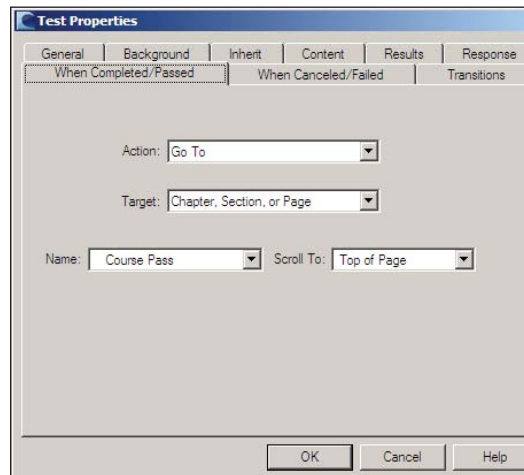
Figure 6 *Test Properties — When Completed/Passed Tab*
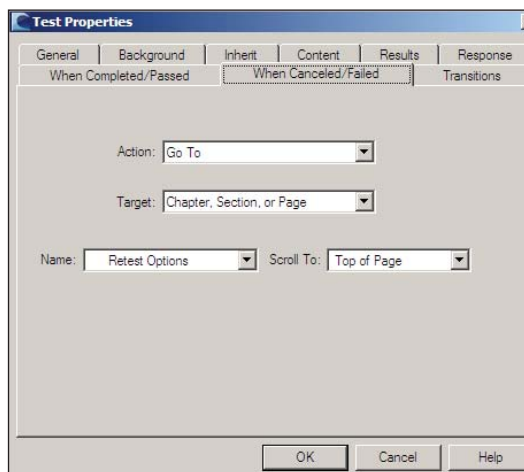
Figure 7 *Test Properties — When Canceled/Failed Tab*

Figure 8 *AICC_Score Action*

Lectora ties to the value set in Figure 8. During our negative testing we found this was the most consistent way to ensure our required functionality held with the LMS tracking the courses. Not to assert it was a Lectora limitation, as it could have been some other customization we made, a condition of the LMS we were integrated with, and so on.

Publishing

The final part of producing SCORM content in Lectora is publishing the course. The Publish menu offers a variety of options, but the relevant choice for this discussion is Publish > Publish to SCORM/Web-Based. There are two steps to the publishing process.

The first step is a high-level error check that ensures the content is consistent to the point it will not immediately fail the publishing process. All errors and warnings should be resolved to ensure a published course will function properly in a LMS.

Errors are the more severe of the two. An error might display if you build a course with an Assignable Unit but without a scored assessment. By definition, a course with an Assignable Unit, intended to be SCORM conformant, needs either a scored test or a manual modification to the AICC_Score or AICC_Lesson_Status variables, so the LMS can track the score or the completion.

Lectora will not allow you to publish with any errors present. A typical warning will appear if you have two Pages with the same name. Again, these should be resolved before publishing.

Once the first publishing step has completed without issue, a Publish button appears. Clicking the Publish button opens the Publish SCORM Location window shown in Figure 11.

Not all the values shown are required, and many are the default values. For our purposes, we chose to make our content SCORM 1.2 Conformant, to have the course send Test/Survey Question Interactions to the LMS, and to prompt to restore the last viewed location within the LMS. Each company will decide what data to track and functionality to enable. The version of SCORM is important, to ensure the content will function properly with your LMS.

The other critical tab for our development was the Options tab shown in Figure 12 on page 8. Again, we left many of the defaults as is, but it is important when you publish to ensure the Publish All Pages/Resources in the Title radio button is selected. Store the Published Title in a Zip file is also helpful, as this is the most common method for sharing a SCORM-conformant package with an LMS administrator for upload.
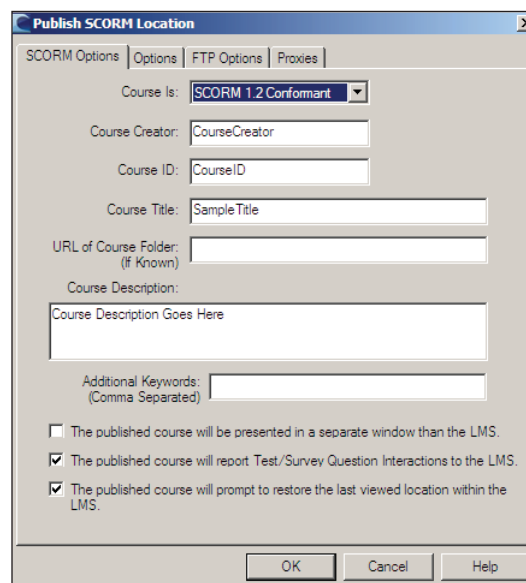
Conclusion

While we did not disclose the nuts and bolts of our development in this article, I hope that the information presented is helpful and potentially useful for other
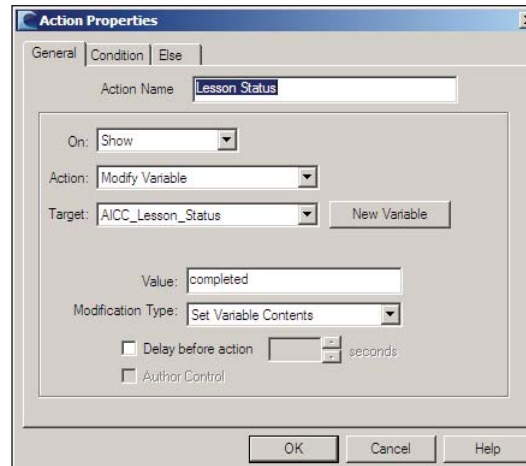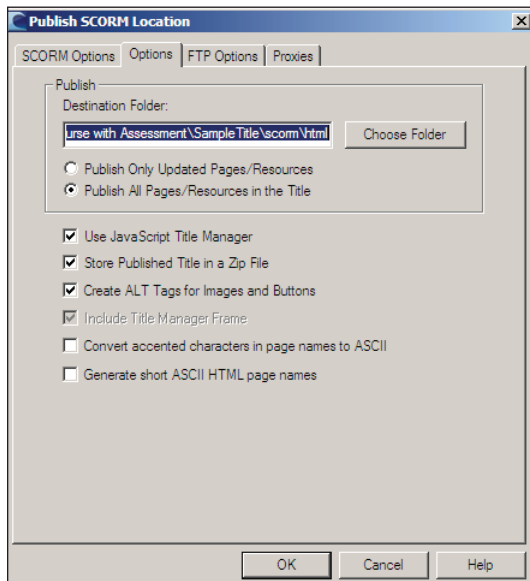

Figure 9
AICC_Lesson_Status
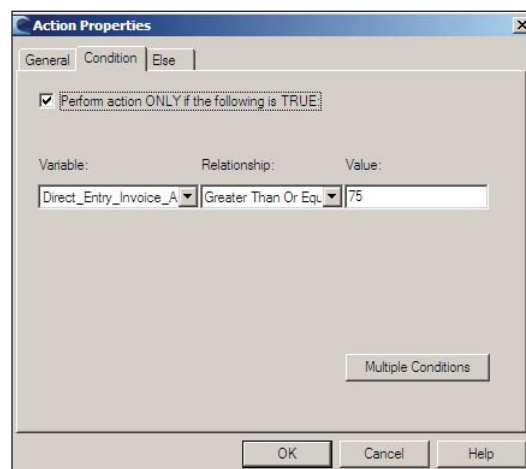
Figure 10
AICC_Lesson_ Status Condition

Figure 11 *Publish SCORM Location*

people working in Lectora for the first time. Lectora is a very easy tool to learn and build with. You can make many of the features complex, but with just the information in this article you should understand the fundamentals of SCORM development in Lectora, and be capable of putting simple, SCORM conformant content together. Using some of the power features of Lectora, such as the Title templates, can allow you to produce these simple SCORM-conformant courses in a very rapid and repeatable way.

Figure 12 *Publish SCORM Location — Options Tab*

Author Contact

Nicholas Bird is a Project Manager with RWD Technologies, Inc. (www.rwd.com) in Baltimore, MD. Nicholas has been involved in custom content development and e-Learning since 1999, when he began working with Lockheed Martin developing custom ILT, and synchronous online delivery courses for training Fortune 500 employees on skills like HTML, Website Design, JavaScript, and VBScript. He also spent time as a trainer and Project Manager of e-Learning Technologies at QVC, Inc., where he was responsible for all aspects of developing various training solutions for internal employees and partner companies.

Nicholas is in his fifth year at RWD, and has been involved in the design and writing of many e-Learning courses for a variety of Fortune 100 manufacturing, telecommunications, pharmaceutical, agricultural, and information technology companies. Nicholas really enjoys the creative process, and helping clients and

coworkers find innovative solutions to their training needs. Nicholas has a M.B.A focusing on Technology Management from the University of Phoenix. You can contact Nicholas at nbird@rwd.com.

Discuss these articles in the "Talk Back to the Authors" Forum of Community Connections (http: // www.elearning-guild. com/community_connections/forum/categories. cfm? catid= 17&entercat =y). You can address your comments to the author(s) of each week's article, or you can make a general comment to other readers.

Additional information on the topics covered in these articles is also listed in the Guild Resource Directory.

In the Archives

This is the third article that Nicholas Bird has written for Learning Solutions. The eLearning Guild has previously published articles whose topic relates to this week's article. These are available to members in the Learning Solutions Archive online. Members must log in to download them. Here are the authors, the article topics, and the publication dates. *(Not a Guild Member? Join today for immediate access to all 275 articles!)*

Previous Articles by Nicholas Bird

"Preparing for and Developing Interactive Content in Captivate 3.0" (March 24, 2008)

"Using RWD uPerform to Rapidly Design and Develop e-Learning" (May 14, 2007)

Article on related topic

Len Perkins, "Using Lectora to Repurpose e-Learning Content – Can It Really Deliver?" (August 7, 2006)

Best of the Blogs

"How to keep your conversations from turning into broadcasts": http://www.chrisbrogan.com/keep-your-conversations-from-turning-into-broadcasts/ (October 31, 2008)

"Web 2.0 Learning" (JetBlue – includes example): http://elearningtech.blogspot.com/2008/10/web-20-learning.html (October 29, 2008)

"Corporate raiders and e-Learning: Fending off a potential threat": http://elearnqueen.blogspot.com/2008/10/corporate-raiders-and-e-learning.html (October 27, 2008)

"TWearning? 47 Awesome Twitter Tools": http://elearndev.blogspot.com/2008/10/twearning-47-awesome-twitter-tools.html (October 23, 2008)

DO YOU HAVE AN INTERESTING STRATEGY OR TECHNIQUE TO SHARE?

Get It Published in...

The eLearning Guild's

LEARNING SOLUTIONS

Practical Applications of Technology for Learning

e-Magazine SM

This publication is by the people, for the people.

That means it's written by YOU the readers and members of ***The eLearning Guild***! We encourage you to submit articles for publication in **Learning Solutions e-Magazine**.

Even if you have not been published before, we encourage you to submit a query if you have a great idea, technique, case study, or practice to share with your peers in the e-Learning community. If your topic idea for an article is selected by the editors, you will be asked to submit a complete article on that topic. Don't worry if you have limited experience writing for publication. Our team of editors will work with you to polish your article and get it ready for publication in **Learning Solutions**.

By sharing your expertise with the readers of **Learning Solutions**, you not only add to the collective knowledge of the e-Learning community, you also gain the re-cognition of your peers in the industry and your organization.

How to Submit a Query

If you have an idea for an article, send a plain-text e-mail to our editor, Bill Brandon, at bbrandon@eLearningGuild.com, with the following information in the body of the e-mail:

- **A draft of the first paragraph**, written to grab the reader's attention and identify the problem or issue that will be addressed.
- **A short outline of your main points** addressing the problem or resolving the issue. This could be another paragraph or it could be a bulleted list.
- **One paragraph on your background or current position** that makes you the one to tell this story.
- **A working title** for the article.
- **Your contact information:** name, job title, company, phone, e-mail. This information is to be for the writer of the article. We are unable to accept queries from agents, public relations firms, or other third parties.

All of this information should fit on one page. If the topic fits our editorial plan, Bill will contact you to schedule the manuscript deadline and the publication date, and to work out any other details.

Refer to www.eLearningGuild.com for Author Guidelines.