ADL Joint Co-Lab: Contract Number N61339-04-2-0002-P00003

Prototype Final Report: Guidance and Prototypes for the Design and Development of Learner Assessments

Written by: William H. Blackmon and Nina Pasini Diebler

September 29, 2006

Submitted by Carnegie Mellon
Anne R. Humphreys
Office of the Provost
Email: ah34@andrew.cmu.edu
Telephone: (412) 268-3420

Description of Problem

SCORM (Sharable Content Object Reference Model) has never directly addressed how to create an assessment, nor when and how a SCO should be considered an assessment. This has created confusion in the instructional design and content development communities. Developers wonder if assessments can be delivered in SCORM-conformant content and, if they can, then how they should be structured to collect the assessment data and metrics. Designers require this data for the validity and reliability reporting that is common to the summative and formative evaluation processes, especially in the Department of Defense (DoD).

Without guidance on how to create assessments within SCORM content, many organizations have resorted to delivering assessments using proprietary assessment solutions provided by their learning management systems (LMSs). This is not the ideal solution because assessments and assessment metrics defined within a given LMS solution are not interoperable or reusable in other LMSs or other learning experiences.

Prior to the release of SCORM 2004, the amount of data that could be collected and stored about a learner's progress within individual SCOs was limited. LMSs were only required to support a subset of the defined CMI data elements. For example, one of the data elements that LMSs were not required to support was "cmi.interactions". The cmi.interactions data element provides a detailed model for designers and developers to collect metrics about learner responses or performance within a SCO; particularly data related to performance on assessments such as correct response, learner's response, duration taken to respond, and weight of the particular item relative to the overall assessment score.

SCORM 2004 now requires any SCORM-conformant LMS to support all of the CMI data elements defined in the SCORM Run-Time Environment (RTE). Most data elements require little explanation to allow a content developer or instructional designer to use them in SCORM content; for example, the description of a typical CMI data element is usually about one half to one page in the SCORM RTE book. The cmi.interactions element is exceedingly complex; its description is approximately 46 pages in the SCORM RTE book because of detailed data collection this data element enables.

Overall Approach

By properly using cmi.interactions, instructional designers can now collect metrics for formative and summative evaluation reporting, and link cmi.interactions and other CMI data elements to sequencing rules to create remediation or adaptive learning strategies that provide more customized learning experiences. This may significantly improve learner performance by allowing designers to better match performance on individual learning objectives with remediation and feedback strategies. With guidance on when and how to use cmi.interactions in assessments, designers will be able to create robust assessments using SCORM 2004 that meet both the delivery and data collection requirements of their clients and ensure learner mastery of instructional material.

The primary objectives of this project are to understand and document the requirements for creating assessments within SCORM-conformant content, document best practices for the design and development of assessments within SCORM-conformant content, and develop a set of sample SCOs that can be used as assessment templates to provide guidance to instructional designers and content developers in the ADL SCORM community.

Overall Project Work Completed

During the course of this project, work was completed in five key areas:

Survey of Assessment Requirements in the DoD. We posted a web-based survey and received approximately 100 completed surveys that identified the following issues as being of primary importance:

- security of assessments;
- creating and using question banks;
- collection of learner performance data;
- randomization of questions.

The vast majority of assessments use basic multiple-choice, true-false, and matching questions, which are all easy to automatically grade by computers. A smaller number of assessments utilize short answer questions, which typically require a human reviewer.

Documentation of Best Practices for Instructional Designers. We created the "Best Practices for the Design and Development of SCORM Assessments." Sixteen pages of this guide are intended for Instructional Designers and Subject Matter Experts who are enlisted to create courses. This part of the guide covers:

- general assessment guidelines;
- learning strategies;
- structuring SCORM assessments;
- creating test banks; and
- identifying data collection requirements.

The programming details imposed by SCORM are not mentioned in this part of the guide.

Documentation of Best Practices for Programmers. The second part of the Best Practices guide is focused on programmers who will have to implement the assessment strategies determined by the instructional designer. This part of the guide is a shortened version of the information provided in two quick-tips:

- "Understanding cmi.interactions" which provides an overview of the SCORM data model elements for collecting information about assessments; and
- "Securing Assessments" which provides a simple way to provide a measure of security for your assessments when they are delivered through a SCORM-conformant LMS.

Implementation of Prototypes SCOs. We created several prototype SCOs. For instructional designers, these SCOs show working examples of assessments in SCORM. For programmers, these SCOs show working examples of an assessment making the appropriate GetValue and SetValue calls and using the SCORM data model to record these values. We created a JavaScript library, called the SCORM Learner Assessment Generator (SLAG), to facilitate the programmer creating individual assessments. SLAG separates:

- the content of individual questions;
- the description of individual assessments;
- the presentation of questions as HTML;
- the grading of assessments; and
- the SCORM calls.

This separation makes it easier to update SLAG for changes to the description of questions (e.g., to use the QTI Lite specification), changes to SCORM, etc.

Implementation of Demonstration Courses and Updates to the LSAL Templates. We created several demonstration courses:

- the SCORM Assessment Demo Course, which shows all of the different types of assessments discussed in the Best Practices Guide;
- the Standalone Assessment Demo Course, which allows a programmer to interact with the SLAG without having to run the course through a LMS.
- Updated LSAL Templates, which include changes based on updates to SCORM and our updated best practices in the two years since the templates were first released;
- Updated LSAL Templates with SLAG, which use SLAG for presenting assessments used in the templates; and
- Example Course using LSAL Template 6, which is a demonstration of assessments used in a course that uses a pre-test to allow a learner to test out of materials that he has already masatered.

Issues, Problems, and Unexpected Lessons Learned

While creating the survey used to collect data from various DoD agencies, we discovered that SurveyMonkey.com offers an excellent and free tool to quickly create a survey, release it to the public, and analyze the responses. We were surprised at how well this service worked.

The SLAG tool is written in JavaScript, which is a particularly difficult language to create programs in, mainly due to lack of IDE support. Testing must be done on web browsers and if there is any problem with the JavaScript, the typical result is that the program will simply not run and will not provide the usual diagnostic information available with most other programming languages. However, JavaScript was clearly the only choice for writing SLAG as the tool had to run on the learner's browser and JavaScript is the only language that we can expect to be installed on every browser as it is a requirement for SCORM to function at all.

After our presentation at Implementation Fest, we received feedback from several members of the community. Their initial reaction was that both the Guide and the SLAG tool would be welcome additions to their arsenal of tools used by their instructional designers and their SMEs who have been forced to develop content quickly. They also said they hoped to see a simple tool that would allow their Instructional Designers and SMEs to create the XML files used by SLAG to represent questions and assessments.

Period of Performance and Cost

The period of performance for the Learner Assessment project was August 4, 2005 to September 30, 2006. The value of the award was $100,000.