

EMERGING TECHNOLOGIES

Learning Objects: Scorn or SCORM?

Robert Godwin-Jones

[Virginia Commonwealth University](#)

In the communities dealing with the design and implementation of on-line teaching and training (higher education, government, industry), **SCORM** is a hot topic these days. The "Sharable Content Object Reference Model" is a standard for the packaging and deployment of Web-based "learning objects," defined by Bob Banks as "a relatively small, reusable resource, through which a coherent, identifiable piece of learning can be achieved." Part of the debate surrounding SCORM derives from its origin, the United States Department of Defense, but for the most part relates to questions about its technical implementation and, even more, about its pedagogical implications, in which SCORM sometimes becomes the whipping boy for those who object to the idea of learning objects themselves. Despite concerns, wide-spread support for SCORM has developed in the past year among content developers and software companies, and with the recent release of a new version of SCORM, it seems an opportune moment to examine its use and importance for language professionals.

Standards Development and the Path to SCORM

The SCORM standard is developed by the Advanced Distributed Learning (**ADL**) consortium, but the individual components come from a variety of sources. One of the main contributors is the **IMS (Instructional Management System) project**, which has worked on standards development in several areas since 1995. Its major contribution to SCORM is the set of meta-data used. SCORM implements the "Learning Object Meta-data" (**LOM**) specification, which is based on IMS work and specifications developed by the European group **ARIADNE**. The LOM was approved as an **IEEE** standard in July, 2003, and describes what content is (title, description, keywords), who owns it, what it costs (if anything), the technical requirements for its use, and educational objectives. Although the meta-data could be transmitted in a variety of ways, the usual method is XML, which allows data to be easily machine-read and exchanged. In fact, the emergence of XML is a key enabler for the recent upsurge in interest and use of standardized ways of operating and interoperating on the Internet.

The IMS Project also contributed to SCORM its process for packaging and transmitting learning content. All SCORM-compliant content must contain a "manifest" file (in XML) which lists all the resources (files) used and their relationship to each other (structure). Resources for a SCORM learning object, called a "**SCO**" for "Sharable Content Object", are normally placed in a folder which is then saved as a zip archive. This zip file can be easily stored, shared or imported into SCORM-compatible software, usually a learning management system (LMS) such as **Blackboard** or **WebCT**.

The third key element of SCORM is the mechanism for communicating between a SCO and the LMS. The Application Program Interface, or API, to allow this interaction was derived from work done by **AICC**, the Aircraft Industry Computer Committee, one of the first groups to work on interoperability and standards. Since airplanes may last considerably longer than software tools or even computing platforms, it has been important to the aircraft industry to future-proof training and maintenance materials, so that the content will continue to be accessible should current hardware and/or software become obsolete. The API SCORM implements uses Javascript to communicate between a SCO and an LMS. In order for that communication to be able to take place, the LMS must contain an "API adapter" which is capable of receiving through Javascript information sent by the SCO and which in turn sends information back.

An LMS might provide to the SCO information such as the user's name (allowing the SCO to personalize messages), the course ID, or how far the user progressed in the lesson in the previous session; in turn, the SCO might send the LMS the status of the user's completion of the lesson, the score received on an

assessment, and the level of mastery achieved. This information may be sent to and stored in the electronic gradebook of the LMS, so that grades for assessments included in SCORM content might appear alongside grades generated by in-house assessments of the LMS. Most LMS now support SCORM, including not only major vendors, but also products such as [Angel](#), [CyBE0](#), or [WebMentor](#) and open-source projects such as [Moodle](#) or [dokeos](#). How SCORM is integrated into the LMS varies, however, along with the user interface. SCORM support is built into the current version of WebCT while Blackboard requires a plug-in. Typically, SCORM content is imported into the LMS as a zip file, then unpackaged on the server, and made available as a linked resource, often opening in a new external window, usually in a frames environment.

SCO's can contain any variety of electronic content, such as text, graphics, animation, audio, or video, but typically will present content in a structured way with built-in assessments in the form of questions in multiple choice or other formats. SCORM content is typically designed to be used as self-paced instruction, based on assessed content mastery. SCORM-compliant content can be created with existing authoring tools, adding manually the necessary Javascript code and XML files (for meta-data and manifests). However, there are a number of content creation tools which are SCORM compatible. [RELOAD](#), for example, includes both an editor and a SCORM player, is free, and is available for Windows, Macintosh, and Linux. The [Simple Content API](#) is a free converter of documents to SCORM from Rustici Software. Microsoft has a free suite of tools, called the [LRN toolkit](#). There is a SCORM [extension](#) to Dreamweaver (from Macromedia), as well as a [tool](#) for Flash.

Issues in SCORM Implementation

The intense interest in SCORM is a reflection of the concern over steep costs and propriety formats of LMS. If content is created within an LMS, it is not easily exported to other systems or shared with colleagues (even sometimes those using the same system). Learning content which is SCORM-compliant can be easily imported and exported and used across a variety of LMS. This was the principal goal the ADL had in mind in developing SCORM, to allow government and military training materials to be used in a variety of software products. SCO's are supposed to be reusable and sharable, designed so they could be used in a variety of learning scenarios. That means that for SCORM, learning content must be broken down into minimal units which can then be put together to construct lessons or on-line courses. The SCO's are meant to be used in the context of an LMS, which is responsible for managing how the user interacts with them.

One of the issues in SCORM's implementation has been how to structure the order and presentation of SCO's to the user. While [SCORM 1.2](#) (first released in 2001) includes a specification for prerequisites as a means to define the order in which SCO's are presented, this has not been a feature often used by developers, in large part because it has not been widely implemented by LMS. The latest version, [SCORM 2004](#) (formerly 1.3), released in January, 2004, attempts to address this issue by adding a set of standard sequencing rules (based on [IMS Simple Sequencing](#)). This allows quite sophisticated logic to be added to SCO's, to enable features like remediation, pre- and post-assessments, and conditional branching. In other words, SCORM is moving closer to what has traditionally been done in computer-based instruction (CBI) or with Intelligent Tutoring Systems (ITS).

While this brings a powerful new capability to SCORM, it also represents an additional layer of complexity for developers. This adds to long-standing concerns that the complexity of SCORM poses a steep barrier to small commercial or individual developers. While more tools for SCORM creation are becoming available, their use still requires a good understanding of how SCORM is implemented. This is true not just of the programming/scripting of SCO's but also of their instructional design. In fact, in the area of instructional design, there has been some serious concern with SCORM expressed. Some critics, such as [Stephen Downes](#), feel that the reusability which SCORM celebrates, is incompatible with instructional design given the crucial importance of context in any learning environment. In some cases,

this merges with [concerns](#) of a philosophical or political nature, that SCORM, as a product of the US Department of Defense, is imbued with a certain ideology and culture.

There are also technical concerns. Most often SCO's consist of a single Web page. In part this is due to the fact that the API uses Javascript to communicate between the SCO and LMS. This works fine for a single page but unless that page is in a frames environment (or uses cookies to save variable values), Javascript information is lost when the page is unloaded. The client-side nature of SCORM also has repercussions in terms of assessments. If assessments are constructed, as they normally are in SCO's, in Javascript, then answer information may not be secure. This has led to a number of obfuscation techniques to hide the assessment data, none of which is totally satisfactory. Another issue in the use of SCORM has to do with the presentation of content. Although the idea is that SCO's might be pulled together from a variety of sources to build a course, the look and feel of the individual units is likely to be quite different. There are no standards for the style, structure, or navigation used in SCO's, although a compelling [solution](#) has been proposed, in which styles would be applied when the content is delivered to the user, rather than hard-coded by the developer. Finally, another technical issue involves [cross-domain scripting](#). Unless a SCO is delivered from the same server as the LMS, built-in browser security safeguards will cause a scripting error. Workarounds have been found, but this remains a concern.

Outlook for Language Learning

As the SCORM standard evolves over time, it is likely that the technical issues will be resolved and new features implemented. SCORM 2004 adds powerful options for sequencing content, although support of the new version in most LMS may take some time. One of the challenges with SCORM 2004 will be for tool developers to design content creation tools which take advantage of the new features, but remain within the technical competence of typical instructors. It is not likely that the average language instructor will take advantage of the SCORM features if content development remains as technically challenging as it has been. Nor is it likely that many institutions will have the resources to offer an array of programmers and instructional designers to help instructors create SCORM-compliant on-line learning resources. In fact, it not surprising that the sustained effort required to create and maintain a complex technical standard such as SCORM is funded by a government agency, rather than through universities. Grant monies typically fund short-term research and development efforts, not the kind of long-term project SCORM represents.

The funding source has also had an impact on the nature of the project. SCORM is a standard useful for self-paced individualized instruction, an educational environment favored by industry, government, and the military. It does not address group or collaborative learning, important elements in language learning. There have been [suggestions](#) to add collaborative tools to the SCORM standard, but none of the proposed extensions have yet been implemented. In order to provide the greatest level of portability and interoperability, the SCORM standard is based on client-side, rather than server-based interactions and that in itself makes integration of collaborative tools problematic. Of course, SCORM content is designed to be used within an LMS, which will itself likely supply a set of collaborative tools. However, the SCO will not be directly linked to those tools.

[David Wiley](#) points out that modern learning theories emphasize the importance of collaborative learning; he characterizes some aspects of the use of learning objects "oppressive," because the learner's access to the learning environment is restricted to a fixed, determined path. He offers an interesting scenario of using SCO's in combination with "blog-based learning." Learners might be alerted to new learning objects in repositories (perhaps through subscription to [RSS](#) syndication) then participate in blog discussions of the content. He also sees learning objects as potential resources, among others, in a rich problem-solving learning environment, in which they could be drawn upon as needed as learners seek to reach solutions. One of the recent ADL proposals points in part in this direction. [CORDRA](#), the Content Object Repository Discovery and Resolution Architecture, is an effort by ADL to define a standard way to create

repositories of learning objects. It is envisioned that such repositories could contain the "auxiliary resources" included in the SCORM 2004 specifications. These resources could be used in interactive lessons, as materials for reference, remediation, or advanced learning, automatically triggered by determination of mastery level achieved, or simply made available as additional, supplementary learning resources in a more open-ended system.

SCORM is being more widely known and used internationally, as the ADL-sponsored [International Plugfest](#) this year in Zurich demonstrated. However, language professionals have not yet embraced this new standard en masse. In part, this may be a concern based on previous experiences with standards and language learning, which as Lars Borin [points out](#), do not always address the needs of language learning and teaching. However, there are some signs of interest, such as the Defense Language Institute's [LangNet](#), which uses tools (from [Metasoft](#)) that are SCORM-compliant to organize and manage learning objects. The Canadian LLearnproject (Language Learning Environments and Resource Network) also uses SCORM. Mike Bush, of Brigham Young University (BYU), has been interested in SCORM for some time and has been a driving force behind the [ID2SCORM](#) conferences, which aim to connect instructional design and SCORM. He is co-author of a forthcoming article in the *CALICO Review* on the important role SCORM is playing in the development of Arabic and Swahili learning materials at BYU. Of interest as well is the Video Asset Description ([VAD](#)) project out of BYU, which extends the LOM model by including MPEG-7 elements for working with DVD video.

While SCORM integration may make sense in specific environments, the kind of controlled environment it represents may not be ideal for all aspects of language learning. However, it is likely to play an increasingly important role in projects involving self-paced language study. For less-frequently taught languages, for example, SCORM offers an interesting option for creating an individually tailored instructional environment, by combining SCO's using the new sequencing capacity of SCORM 2004. SCORM makes most sense for lower-level learning tasks rather than for higher-level learning outcomes (synthesis/integration, critical evaluation, real world simulations). At an upper level of instruction, SCO's could perhaps be used most profitably in an open, constructivist learning environment.

RESOURCE LIST

Learning Objects

- [Knowledge Objects, Knowledge Objects & Mental Models](#) by David Merrill (PDF)
- [Learning Solutions - Learning Objects: Behind the Buzz](#) from "Chief Learning Officer"
- [Three Objections to Learning Objects](#) by Norm Friesen
- [Learning Theory and Learning Objects](#) by Bob Banks (PDF)
- [Design, Standards and Reusability](#) by Stephen Downes
- [Learning Objects: Difficulties and Opportunies](#) by David Wiley
- [Elusive Vision: Challenges Impeding the Learning Object Economy](#) White paper from Macromedia (PDF)
- [e-LEARNING INTEROPERABILITY STANDARDS](#) from Sun (PDF)

Standards

- [Learning Technology Standards: An Overview](#) good summary from cetis (center for educational technology interoperability standards)
- [Who's involved?](#) list from cetis of organizations involved with developing educational technology standards
- [Who's Doing What?](#) chart of directions currently being pursued by various standards organizations for educational uses (from cetis)
- [LOM Learning Object Metadata](#) - used in SCORM

- [IEEE LOM Standard Not Yet Ready For "Prime Time"](#) by IEEE LOM Standard Not Yet Ready For "Prime Time"
- [Aviation Industry CBT Committee \(AICC\)](#) standard-setting body on which SCORM is based
- [IMS Global Learning Consortium](#) standard-setting group for meta-data
- [IEEE Learning Technology Standards Committee \(LTSC\)](#) standard-setting body for learning objects
- [ARIADNE](#) Alliance of Remote Instructional Authoring and Distribution Networks for Europe

SCORM

- [SCORM Home Page](#) From ADL (Advanced Distributed Learning)
- [SCORM Downloads and Specifications](#) from ADL
- [SCORM Newsgroups](#) sponsored by ADL
- [SCORM demos](#) from the Academic ADL Co-Lab
- [SCORM Overview](#) from the UK Learning Lab
- [SCORM](#) from xml cover pages
- [SCORM "cliff notes" version](#) from rha associates
- [Two-Minute SCORM Overview for Developers](#) from Rustici Software (PDF)
- [CanCore and SCORM](#) comparison of Canadian standard for metadata, derived from LOM
- [SCORM: Clarity or Calamity?](#) from Online Learning Magazine
- [E-learning, single sourcing and SCORM](#) PDF file
- [A feature or a bug; SCORM and cross domain scripting](#) about Javascript issue in SCORM usage (from cetis)
- [Efficient Development through SCORM Standards](#) presentation by Paul F. Merrill, Michael D. Bush, and Thor Anderson (Powerpoint)
- [Experts question SCORM's pedagogic value](#) from cetis
- [Dynamic Appearance Model Analysis and Alternatives](#) from the Canadian Department of National Defence
- [Supporting Collaborative Learning Activities with SCORM](#) by Albert Ip & Ric Canale

Repositories

- [ADL to make a 'repository SCORM'](#) from cetis
- [ARIADNE Knowledge Pool](#) European-wide distributed repository
- [Repositories](#) list from the ADL Academic Co-Lab
- [Fellows ODLP](#) on-line learning project from the EU

Tools

- [dokeos](#) open source learning management system
- [SCORM import module](#) for use in dokeos
- [Simple Content API](#) free converter of documents to SCORM (from Rustici Software)
- [CyBEO](#) Learning Management System supporting SCORM (in French)
- [Angel](#) SCORM-compliant LMS from CyberLearninglabs
- [LRN](#) Microsoft's elearning toolkit (supports SCORM)
- [WebMentor](#) SCORM-compliant LMS from Avilar
- [RELOAD](#) a free player and mini-LMS for SCORM content (editor also available), one of few for Mac & Linux
- [RELOAD editor adds Content Packaging and SCORM authoring](#) from cetis
- [SCORM Runtime Wrapper](#) for use in Dreamweaver (from Macromedia)
- [Metasoft](#) suite of SCORM-compliant set of content management tools (from Digital Concepts)
- [Question Tools](#) SCORM-compliant Quiz creator
- [SoftChalk LessonBuilder](#) SCORM-compliant Web lesson creator

Language Learning

- [Intelligent on-line Language Learning](#) from Mind2Learn
- [Where will the Standards for Intelligent Computer-Assisted Language Learning Come From](#) by Lars Borin
- [Online Instruction for the 21st Century](#) Schedule for 2003 BYU Conference on implementing SCORM (PDF)
- [SCORM-compliant Quartet](#) English language learning software (from The Q Group)
- [VAD](#) Video Asset description project out of BYU