Advanced Distributed Learning, Intelligent Tutoring Systems, and SCORM 2.0

J. Wesley Regian, Concurrent Technologies Corporation

Xiangen Hu, Workforce ADL Co-lab

Abstract

We argue that the existing "object" definition of SCO in the current implementation of SCORM limits the ability of developers to implement advanced approaches to automated instruction that are known to be remarkably effective in achieving certain types of learning outcomes. These advanced approaches to automated instruction are often associated with so-called Intelligent Tutoring Systems, or ITSs. As reflected in the literature, ITSs make use of various Artificial Intelligence (AI) knowledge representation and programming techniques in order to efficiently bring learners from novice to competency or even expertise in complex domains. Which of the various AI techniques are required for a given knowledge or performance domain depends on characteristics of the domain. Importantly, however, none of the (so-called) ITS features can be effectively implemented (if at all) within the current SCORM framework. Our proposal is to bring back some fundamentals of traditional software "objects" in SCO, and also to require certain enabling features in compliant LMS. We believe that with enhancements in the standards for SCO and LMS, basic and advanced features of ITS can be implemented in (future) SCORM compliant instructional systems.

Problem Definition

Given that the ultimate goal of e-learning is **learning**, it is important to consider the degree to which SCORM supports the techniques that have been proven to support efficient learning. Sequencing rules are necessary but not sufficient to enable effective e-learning.

One of the major issues of the current SCORM is its "packaging" goal. The definition of SCO was not intended to be an "object" that delivers "learning", but instead an "object" intended to be transferred from on system to another. Such definition brought a fatal consequence: it prevents SCORM from supporting highly effective learning techniques such as those that are associated with advanced instructional techniques such as Intelligent Tutoring Systems (ITSs).

The shortcomings of SCORM for delivering effective learning techniques have been discussed before - and remedies have been proposed. For example, SCORM 2004 sequencing and navigation notions proposed a partial solution to the issues. Specifically, it allowed selection of content delivery as a function of learner performance. Our point here is that cognitively principled automated instruction involves significantly more features than SCORM 2004 S&N can handle.

Components and Features of ITS

In the literature on Intelligent Tutoring Systems, the described software-based instructional systems typically apply one or more of the following techniques.

1	**Generative**	The capability to generate appropriate instructional interactions at run-time based on student performance.
2	**Mixed Initiative**	The capability to initiate interactions with the student as well as to interpret and respond usefully to student-initiated interactions. Natural language dialogue is sometimes taken as the focus of this feature.
3	**Interactive**	The provision of appropriately contextualized, domain-relevant, and engaging learning activities.
4	**Student Modeling**	The capability to assess the current state of the student's knowledge and the implied capability to do something instructionally useful based on the assessment.
5	**Expert Modeling**	The capability to model expert performance and the implied capability to do something instructionally useful based on the assessment.
6	**Instructional Modeling**	The capability to make pedagogical inferences and decisions based on the changing state of the student model, based on the prescriptions of an expert model, or both.
7	**Self Improving**	The capability to monitor, evaluate, and improve its own teaching performance as a function of experience.
8	**Simulation**	Simulations of systems, devices, or processes used to contextualize instruction about the simulated systems, devices, or processes.

So What?

We are not claiming that using ITS or Artificial Intelligence (AI) techniques automatically makes for better instruction. We have no doubt that one could implement every AI method currently known, and still build monumentally bad instruction. Instead, our point is this. It is universally conceded that different instructional approaches (pedagogies) are optimal for different learning outcomes (categories of knowledge and/or skill). Simply stated, instruction should be tailored according to what you want the student to be able to do. We who study learning are familiar with a variety of pedagogies and learning outcomes, and with a variety of useful approaches to the taxonomic characterization of both pedagogies and learning outcomes. We even have rough agreement on general guidelines for deciding what

pedagogy to apply given a specified desired learning outcome. The problem we wish to highlight here is that the current SCORM does not allow us to implement the full range of pedagogies that are required to achieve the full range of learning outcomes.

Bottom line, there exists an engineering discipline that can consistently produce human performance (by cognitive-science based training) that is remarkable – essentially an instructional engineering discipline. Following appropriate methods, we can reliably train human beings that are 1 standard deviation superior to the average performer in any task. Such a person performs at the 84rd percentile - including tasks that include perceptual, declarative, cognitive, and motor determinants of performance. Again, the problem is that SCORM does not currently enable these methods. We believe it is time to consider the next version of SCORM that makes the implementation and re-use of these methods easy.

Use Cases

ITSs have been shown to be extremely effective for knowledge intensive domains, complex device maintenance, and high performance motor skill tasks.

Studies of authoring tool efficiency have shown very large gains from tools that allow re-use of courseware components. Re-use of domain knowledge representations may well exceed gains from simple courseware re-use.

Proposed Solution

It is premature to propose specific standards for implementing ITS capabilities within SCORM. Instead we propose a hypothetical feature-property matrix between ITS features/components and the required specifications in object oriented programming – where the cells in the matrix will be filled in by a panel of experts. The initial task would be to understand the nature of each cell by answering some questions: Can this cell be addressed by enhancements to the SCO definition, LMS requirements, or must both SCO and LMS be specified to address the cell. It is likely that cells in the matrix are related such that supporting any one ITS feature requires consideration of several object properties. We would like to bring together a panel of experts to begin specifying the standards within SCORM for supporting ITS.

	Class	Object	Instance	Message Passing	Inheritance	Abstraction	Encapsulation	Polymorphism	Decoupling
Generative									
Mixed Initiative									
Interactive									
Student Modeling									
Expert Modeling									
Instructional Modeling									
Self Improving									
Simulation									