Gaps in SCORM Implementation and Practice Using an Online Simulation

Patrick Shane Gallagher, Ph. D.
ADL/Si-International
Alexandria, VA
patrick.gallagher@si-intl.com

ABSTRACT

In assessing SCORM 2004 for its affordances facilitating the implementation of specific requirements representing a simulation-based model optimized for interoperability and reusability several implications have come to light ranging from gaps in the technical architecture to standard implementation practice to instructional designers and programmers perspectives and understanding. They were identified technically within the RTE and Sequencing as well as in the common implementation practice of designing SCOs purely for content presentation. Findings also point to the need for persistent arbitrary SCO to SCO communication and the ability to conceptualize, design, and implement reusable functional SCOs to fully implement a simulation as an interoperable model within a SCORM environment. Also implied, are gaps in instructional design practice for SCORM-based solutions as well as gaps in the understanding of IT engineers and practitioners in relation to learning theories and practices. In respect to SCORM 2004 and simulations in general as a valuable reusable pedagogical model, the underlying behaviorist pedagogy inherent in SCORM's design needs to be revisited and in so doing the academic community needs to become more involved in its evolution.

These findings were derived from a gap analysis using a specific set of requirements derived from an existing online simulation learning environment as the criterion and the Run-time Environment (RTE) and Sequencing of the SCORM 2004 technical architecture as the condition. Results were based on an analysis of quantitative and qualitative data collected from 26 members of the SCORM community employed in industry, government, standards/specifications entities, and academia.

Participants were asked to provide levels of agreement to indicator statements of the relevance of the SCORM 2004 targets to the SIMREF at both the individual and set levels. They were also asked to describe alternate standards, specifications, technologies, and capabilities necessary to fulfill the requirements.

ABOUT THE AUTHOR

Patrick Shane Gallagher, Ph. D. is currently the technical lead for the Advanced Distributed Learning (ADL) initiative and a program manager for SI International. He was the Chief Knowledge Engineer/Instructional Technologist for the Performance Improvement Operation within the Analysis, Simulations, Systems Engineering, and Training (ASSET) Business Unit of Science Applications International Corporation (SAIC). Dr. Gallagher has a successful track record in leading, designing, and implementing enterprise learning and development solutions specializing in the convergence of enterprise learning and knowledge technologies. He led the knowledge management support for the NASA Johnson Space Center office of the CKO and was also the analysis team lead and knowledge architect for the Joint Knowledge Development and Distribution Capability (JKDDC) project for the Department of Defense (DoD) and the JKDDC Joint Management Office. Dr. Gallagher has extensive experience in designing and developing learning and knowledge architectures and systems and has led an internal research and development project for SAIC in the design and development of new content object models to support smart enterprise systems. Dr. Gallagher has accrued customer recognition and awards for thought leadership, innovation, and design and continues to pursue models for convergence in the areas of e-learning, learning technology standards, and knowledge management.

Gaps in SCORM Implementation and Practice Using an Online Simulation

Patrick Shane Gallagher, Ph. D.
ADL/Si-International
Alexandria, VA
patrick.gallagher@si-intl.com

BACKGROUND

Major trends have emerged in the implementation of online learning. These trends center on the convergence of technologies and the blending of and transitioning from old to new instructional and instructional design paradigms. They also exist beyond traditional organizational boundaries crossing the line between private and public, corporate and the military.

These changes have brought with them many pedagogical opportunities, considerations, and challenges and have created the need for instructional designers to be aware of the design implications associated with new and emerging online learning systems (Shank 2001). In this environment, instructional design is being re-evaluated, and new models of design are being sought (Sims 1997) resulting in serious challenges in the field of instructional design. One prominent challenge is in the selection and application of instructional strategies to achieve higher order learning outcomes. As epistemologies have shifted from behaviorism to cognitivism and constructivism, learning object content models the building blocks typically used for online instruction begin to fall short as they typically support lower level outcomes such as declarative knowledge acquisition supporting a behaviorist or objectivist view of learning.

Shifting epistemologies and the quest for more meaningful online learning have defined learner-centric design as an overarching tenet. Supporting this shift, the convergences of Information and Communication Technology or ICT-based knowledge management and e-learning systems are providing more learner control. New types of interactions and learning experiences will have to be considered and developed according to capabilities offered by the technology. This will require new approaches and techniques to bring technology use to its full potential (Gallagher 2002). Although there are several approaches and models currently being considered and/or used successfully in an online environment, in the corporate and government training arena and especially within the Department of Defense (DoD) (Menaker, Coleman, Collins, & Murawski, 2006), the approach gaining attention is that of the simulation.

Simulation Overview

In education, simulations have come to encompass children's simulation-games, curricula based on student modeling, lab simulations for science study to commercial and expensive flight simulators for teaching airline pilots how to fly. They have also come to encompass large networked simulations for military battlefield training, virtual reality, microworlds, and goal-based scenarios. In other words, the definition is at once all-encompassing or specific depending on who is creating the definition. According to Alessi, an educational simulation is a program that incorporates a learner-manipulated model accompanied with a learning objective that includes understanding the model (Alessi 2000).

Educational simulations are considered important tools to support learning both in the literature and by scientists and practitioners. Yet, there exists confusion over scope and definition usually due to terminology. The same type of simulation often is described by many terms. For example, *microworld, management flight simulator, business simulator, business game, management simulator,* and *learning environment* are all terms that sometimes describe the same kind of simulation. Also, two simulations having the same name may be very distinct in functionality and type (Maier and Grobler 2000).

Diversity in terms illustrates the diversity in purposes surrounding the development and deployment of simulations in the learning context. Such purposes include learning to be a better manager, learning how to perform and function with a team (e.g. medical or flight), understanding systems through exploration (e.g. virtual labs or models) and virtually any discipline where application and higher order learning are important. Simulations can allow the engineer/scientist to modify a system and then test that against a known set of inputs or provide a system that can be used to support various modeling and simulation domains. Simulations can facilitate training by immersing a learner in a virtual environment that is too costly or dangerous to allow in reality such as toxic environments or high-fidelity flight simulators.

Effective e-learning uses a variety of and a partnership of tools. These tools should be used to represent meaningful problems, situations and contexts (Norton 2003). As learning and activity are considered inseparable and are embodied in tool usage, learning objects and resources should support the complex interactions required for meaningful learning. “Meaningful learning results from the recognition of a problem, the intention to solve it, the conceptual understanding of the system in which the problem occurs, the generation and evaluation of alternative solutions based on alternative perspectives, and reflection on the activities that resulted in its solution (Jonassen and Churchill 2004)”. The rich environment presented by a well-designed simulation allows for immersive learning, social negotiation, tool usage, and problem solving and is a useful method for creating effective engaging e- learning.

Training consists of learning and assessment activities for the acquisition of specific knowledge and skills and is based on many methods or pedagogical techniques. Individual training has traditionally been based upon a one-way transmission model of instructor (computer for online environments) to learner with the underlying assumption that the learner will gain knowledge and skills through this limited type of activity. However, learning to apply skills and knowledge requires much greater interaction; therefore “learning by doing” results in much more meaningful and effective learning (Swinski and Williams 2004). Unfortunately, effective, immersive, and authentic training and learning environment development is expensive and sometimes logistically impossible. Simulation technology provides a possible framework within which such immersive training might be conducted.

Simulations can also provide an authentic and effective assessment environment. By actually performing within a simulated activity, learners can be assessed on how well they can apply and understand what they have learned. Formatively, simulations can be used to help learners reflect on and shape their knowledge and skills. Summatively, simulations can be used as spaces to exhibit performances of understanding. For problem-based competencies, simulations make an excellent assessment tool to certify whether someone can problem-solve or perform analysis activities. An example of a summative reflective assessment is that of an after-action review of an exercise to highlight what was don right as well as identify areas of improvement (Aldrich 2006).

Pedagogical Models and Simulations

The definition of pedagogical models differs depending on the context in which they are discussed. In the context of learning theories, Driscoll discusses pedagogical models alongside conceptual and mental models as a part of schema theory. In this context, they are models built upon students’ models of the world in order to help in understanding. In this sense, pedagogical models are a tool to provide “…strategies for helping learners make predictions from and debut their current models of understanding (Driscoll 2000).” Grimmitt states that the selection of curriculum content and the choice of methodology (or methodologies) selected for the ability to bring about learning outcomes as components of designing constitutes a pedagogical model. He also states that a pedagogical model should deploy specific pedagogical procedures or strategies which determine how learners will experience, engage with, and respond to the content (Grimmitt 2000).

However, a recent trend is for designers of online learning to look at reusable models or designs of learning embodying specific instructional theories and related strategies as separated from specific learning resources (Oliver and McLoughlin 2003). Research is also focusing on the application of model-based development or engineering to instruction. Sallaberry, Nodenot, Laforcade, and Marquesuzaa (2005) are using the Unified Modeling Language (UML) to develop pedagogical models based upon problem-based learning (PBL) as a basis for a global reusable information system to support learning. These models or designs are thought of as components of reuse incorporating other reusable resources such as learning objects (Oliver and McLoughlin 2003; Gallagher 2005).

The predominant approach to object-based online learning is focused on a content-based pedagogical model or as content-centered approaches to learning (Oliver and McLoughlin 2003). This model essentially provides content presentation as the means to transmit knowledge from the content to the learner. Content-centered models have evolved because content is relatively easy to author and manage through information systems (IT) such as content management systems (CMS) and learning content management systems (LCMS). These systems work well with a tangible chunk of content that can be easily described as an object with specific defining attributes (Watson and Watson 2007). Other examples of models defining these content chunks are with S1000D and the Darwin Information Typing Architecture (DITA).

Contrasting the content-centered model is that of goal-based models. These include models built upon inquiry-based learning, problem-based learning, case-based learning, and other models where learners participate in active learning experiences (Oliver and McLoughlin 2003). These models place more emphasis on learning activity designs instead of content transference (Koper 2003). Examples of these models are the Open University's Educational Modeling Language and the IMS Learning Design (Koper 2003; Olivier and Liber 2003).

Simulations are also considered unique instructional strategies that are consistent and repeatable in an instructional context (Norton and Sprague 2001). Saunders (1997) described simulations as a cyclic learning process, and Saleh (2005) states that simulations remain one of the most efficient models of teaching. A pedagogical model is considered as having curriculum content and the choice of methodology (or methodologies) thought capable of bringing about learning outcomes through deploying specific pedagogical procedures or strategies (Grimmitt 2000). A pedagogical model is also considered to be a model to help students understand and elicit their models of the world (Driscoll 2000). In the context of instruction and in light of the previous descriptions and definitions, an instructional simulation can be considered a pedagogical model.

When targeted towards learning, well-designed simulations can have a high level of learning transference ideal in education and training. Transference is considered the ability of a learner to apply what has been learned in a learning situation quickly and effectively to other real-life situations (Driscoll 2000). This characteristic enhances the desirability of not only using but reusing simulations on a broad scale. However, as simulations are usually very contextual in both design and implementation, such reuse would not only require reusable designs and models but the use of and interface with interoperability standards and specifications for learning technology such as the Shareable Content Object Reference Model (SCORM).

Connecting Simulations and Standards

Currently, simulation interoperability standards exist mostly in the form of the High Level Architecture (HLA) developed by the Defense Modeling and Simulation Organization (Defense Modeling and Simulation Office [DMSO] 2006) and approved as an open standard by the IEEE in 2000 and its predecessor the Distributed Interactive Simulation (DIS) an IEEE standard maintained by SISO (Simulations Interoperability Standards Organization). These standards are intended to facilitate interoperability and reusability among distributed simulations and their components within the DoD and is integral to the modeling and simulation community. However, these simulations currently facilitate collective training and exercises usually on large scales and do not have any discrete provisions for the tracking or supporting of individual training and education activities thus keeping the two worlds separate.

Most online simulations designed for individual use exploit standards and specifications supporting web browsing as developed through the World Wide Consortium (W3C). Typically browsers access web pages as a client with the web pages being served to the client (web browser) by a server. When using a standard web browser for web page access the web browser is referred to as a thin client. Non-browser applications residing on the client side but still exploiting web standards are referred to as thick clients.

Access occurs either through a thick client with proprietary functionality and communication protocols (Miller and Childs 2004) or through other client-server based architectures. In these architectures the actual simulation engine is on the server side with the client used only for communication with the simulation through a user interface in a thin client (i.e. - web browser). There is movement toward the use of purely thin client-based simulations employing mobile code specification, standards, and technologies (Swinski and Williams 2004).

Currently, efforts have been underway to develop interoperability standards between simulations, simulation engines, and SCORM supporting individual training and tracking using a LMS. For example, SISO has been working with industry, AICC, and ADL to develop specifications for simulation interoperability standards for SCORM to be added to the existing IEEE Learning Technology Standards. This would allow external simulation environments to track, assess, and provide data on an individual that could subsequently be stored and managed through an individual training event on a LMS. At this time, however, these specifications are still in the preliminary stage of standardization by bodies such as the IEEE.

Research Focus and Scope

SCORM is an established framework with ubiquitous conformant content that, however, does not easily allow learning to occur beyond the simple acquisition of declarative knowledge and is thought by some to fall very short in terms of cognitive and psychomotor skill acquisition (Jonassen and Churchill 2004). To begin to utilize other pedagogical models such as simulations within this framework, these models need to be analyzed to determine whether they can be integrated into the existing SCORM or whether the existing SCORM needs to be extended to enable this type of training. As a beginning, this study analyzed an online simulation to establish a set of requirements to assess SCORM for its abilities in implementing those requirements while maintaining such innate SCORM tenets as interoperability and reuse.

The focus of this research was to assess SCORM for its affordances facilitating the implementation of specific requirements representing a simulation-based model optimized for interoperability and reusability. This special set of requirements was called the Simulation Requirements Framework or SIMREF and represented an assessment criterion. The study addressed the overarching research question: Assuming the condition of the Run-time Environment (RTE) and Sequencing capabilities of the technical architecture of SCORM 2004 and a criterion of the SIMREF, are there gaps in the capabilities SCORM 2004 provides to facilitate a simulation-based pedagogical model optimized for interoperability and reusability? To answer this question, SCORM's technical architecture was assessed for its strengths and weaknesses in meeting the requirements of the SIMREF. To clarify the differences between architecture and implementation, implementation was addressed as well. Specifically, the research was concerned with the following questions addressing both the SCORM technical architecture and its implementation:

- Are functional or typed SCOs necessary to fulfill specific requirements of the SIMREF? If so, which ones?
- Using a thin client (non-server based) object based delivery mechanism, is it possible to fulfill the requirements of the SIMREF using SCORM 2004 without any extensions?
- If extensions other than SCO to SCO data sharing are needed for SCORM 2004 to fulfill the requirements of the SIMREF what would they be?
- Using a standard browser-based delivery mechanism, is SCORM 2004 sequencing adequate for fulfilling all of the requirements of the SIMREF? If not, what sequencing specific extensions are required?
- Using a standard browser-based delivery mechanism, is complex arbitrary data sharing between SCO's necessary to fulfill specific requirements of the SIMREF?
- Is it necessary to use customized LMS functionality and communications to fulfill specific requirements of the SIMREF?
- As gaps are identified in fulfilling the requirements of the SIMREF, do relationships exist between them?

Although this research was concerned with understanding SCORM for its abilities to implement a simulation as a type of pedagogical model, it was scoped to specifically focus on SCORM 2004 and the requirements derived from a specific simulation. An implementation of a simulation (one of many advanced pedagogical models facilitating meaningful online learning) may be somewhat representative of the implementation of simulations as a model and of other advanced pedagogical models. This analysis will give insight into SCORM in terms of strengths and weaknesses in respect to its ability to facilitate simulations. In so doing, a set of requirements representing a specific online simulation has been developed as the SIMREF. In the gap analysis methodology, the SIMREF represented the criterion and SCORM 2004 represented the condition.

The approach used in developing the SIMREF was a use-case approach commonly found in software development. In following this approach, an available instructional online simulation was chosen and a use-case scope diagram was developed (Cockburn 2001) based upon the inherent functionality of the simulation as it is commonly deployed as part of a learning environment. After formative evaluation activities, the requirements were scoped down to those specifically affected by a SCORM implementation. Although collaboration would be desirable to include as a requirement, it was decided to focus on individual users (learners) due to the inherent known issue of SCORM's inability to support collaboration at this time.

METHODOLOGY

This study made use of a gap analysis methodology by assessing the condition or existing state known as SCORM 2004 against a developed criterion known as the SIMREF to identify the gaps between the states, their causes, and their symptoms. Survey methods were used as the primary data collection strategy.

Instrument development consisted of the development of a set of real-world requirements derived from PharmaSim a simulation existing as a primary component of an online learning environment designed to teach marketing principles and is used within various schools of management (James, Kinnear, & Deigahn, 1999). and exhibiting specific characteristics as defined by the end node in each branch of Maier and Grobler's taxonomy (Maier and Grobler 2000) (Figure 1). These requirements were know in the study as the SIMREF.

Next, to target the specific areas of SCORM under scrutiny (i.e. RTE and Sequencing), these requirements were adjusted or slightly modified to maintain the following overarching tenets: maximum reuse across multiple environments, interoperability, and durability.

To ensure that the requirements would target the necessary scope, a set of developmental parameters were constructed and included as part of the final survey and were intended to guide the thinking of the survey respondents as they completed the survey. The parameters included: 1) Development will use multiple SCOs not a single large SCO; 2) development should use SCOs that are based upon functionality or type instead of just instructional content (A functional SCO is a SCO that provides a specific function or set of functions not necessarily intended to deliver conventional instructional content - i.e. a role assignment function or a scenario choice function.); 3) all functional SCOs will be delivered as components of the course content package; 4) SCOs should be considered to have specific functionality so that the set of SCOs making up the content package will work together as a system; 5) a simulation engine will be embedded within a SCO and delivered as part of the course content package; 6) SCOs will not be required to communicate with an external system; and 7) network accessibility is not a factor.

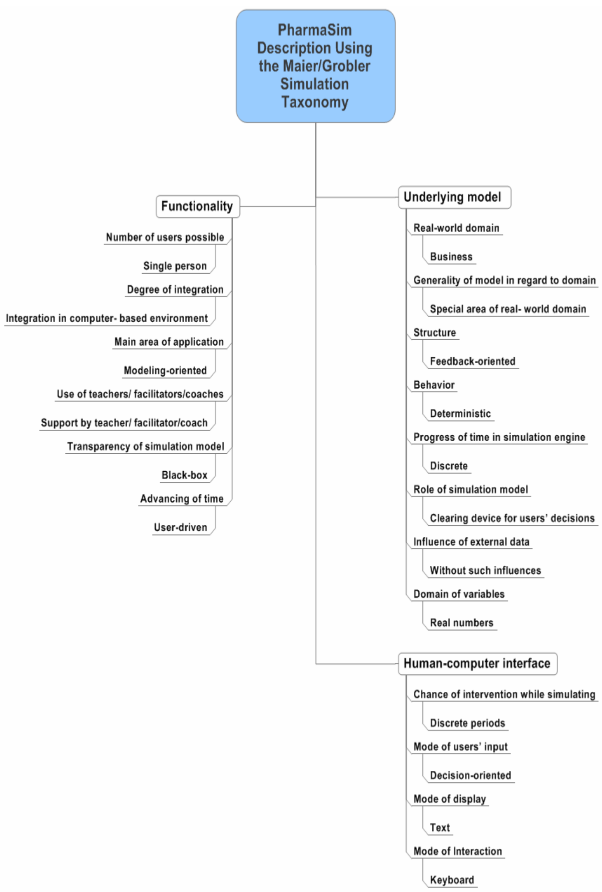

Figure 1 PharmaSim Characteristics from Taxonomy

The requirements were then trimmed and adjusted once again to tailor the assessment to requirements coupling with the run-time environment, sequencing functionality of SCORM 2004, and SCO implementation. This final set of functional requirements was documented into a simulations requirements framework or SIMREF. In terms of a gap analysis, the SIMREF represents the criterion or a desired state and was transformed into explanatory variables to facilitate data collection and analysis.

To address the research questions, a 50 item survey was developed called the Sim SCORM 2004 Survey. Based on the research questions, six indicators (survey items) and two open-ended questions were constructed. The indicators, in the form of agreement statements, were based upon the relevance of the indicator to each requirement as perceived by each respondent. In total, six agreement statements were constructed eliciting relevance levels as items on a traditional five point Likert scale with a rating of 1 equaling strongly disagree and a rating of 5 equaling strongly agree. The final form of the survey used the six Likert items matrixed against each of the eight requirements of the SIMREF resulting in a total of 48 Likert items plus two open-ended questions not tied to a specific requirement. Figure 2 illustrates an example of the survey layout.

Requirement 1: user selects their role and chooses a scenario within the simulation-based course.

Sub-level Components:

- The learner has logged on to the LMS and has chosen the simulation-based marketing course; user creates profile by choosing one of three roles and one of five scenarios.
- User role in the simulation will be progressively upgraded by default (i.e. assistant brand manager to brand manager) as determined by simulation performance, unless user disables default option.
- After making selections, user submits form.
- After submitting form, welcome page and case (scenario overview) are displayed and simulation play begins.
- It is assumed that this requirement is fulfilled by at least one dedicated SCO

Circle the number to the right of each statement that indicates your level of agreement.	1=strongly disagree 2=disagree 3=neutral 4=agree 5=strongly agree
a. SCO's are very relevant to fulfilling this requirement.	1 2 3 4 5
b. The use of one or more purely functional SCO is very relevant to fulfilling this requirement.	1 2 3 4 5
c. Updating or modifying the SCORM sequencing functionality is very relevant to fulfilling this requirement.	1 2 3 4 5
d. SCO to SCO data access or sharing of data between SCO's is very relevant to fulfilling this requirement.	1 2 3 4 5
e. This requirement can only be fulfilled by extending the SCORM 2004 in a manner other than shared data access between SCO's.	1 2 3 4 5
f. This requirement can only be met by the use of a LMS provided thick client providing communication with external systems and different LMS specific functionality.	1 2 3 4 5

Figure 2 Survey Example

The sampling method consisted of an oversampling approach resulting in a sample that was developed through self selection, snowball, and comprehensive techniques and came from two sources: all members of the TWG or their representatives contacted by ADL and attendees to ADL's Implementation Fest 2007. Both sources used a snowball approach to increase participation facilitating oversampling.

The size of the population of those developing and implementing e-learning solutions overall is not large and the subpopulation of those specifically implementing SCORM is even smaller and is somewhat more specialized. For example, there were 648 registered users on adlcommunity.net (adlCommunity 2007), there were approximately 200 organizations recognized as SCORM adopters by ADL (ADLNet 2006), and there were approximately 77 points of contact (POCs) in the ADL Technology Working Group (TWG). Also, the number of attendees to the 2006 Implementation Fest totaled 350 with those having titles indicating organizational roles of a developer nature numbering between 200 and 230. The number of attendees to the 2007 Implementation Fest totaled 331 (no role data was available).

Using these lists as a guide, those indicators were used to characterize the population size of those working for a recognized SCORM adopter organization from 200 to 2000 assuming a minimum of one per organization and a maximum of 10 per organization. The reality is that the size most likely lies somewhere in between and the actual population size was most likely closer to the lower number of 200.

For the purposes of this study, a moderate approach assumed a size of 250 with a target sample size of ≥ 25 or 10% of the population. This target gave a potential minimum number of participants per variable of 25. The actual number of valid responses was 26 exceeding the target sample size. These 26 respondents were experienced SCORM developers employed in industry, government, standards/specifications entities, and academia.

DATA ANALYSIS

The data collected in both quantitative and qualitative forms and was organized by the research questions. For each "yes or no" question, quantitative data means were analyzed to obtain the answer. The Likert scale of agreement to statements of relevance was used in the following way: values of 1 represented no relevance, values of 2 represented little relevance, values of 3 represented values of neutral relevance, values of 4 represented some relevance, and values of 5 represented high relevance. As variable means were produced, they were then looked at in terms of three categories: ≤ 2.49 equals negative relevance, 2.50 - 3.49 equals neutral relevance, and ≥ 3.50 equals positive relevance. For questions asking "which ones" or "what type," both quantitative analysis and qualitative analysis were used. Quantitative analysis pointed to which variable met specific conditions and qualitative analysis was employed to understand more about the condition. Qualitative data collected through open-ended items 1 and 2 were also analyzed thematically, comparatively, and contextually with the data presented in multiple formats including lists, tables, and quotes. The findings by research question are summarized below in Table 1.

Table 1 Summary of Research Findings

Research Question	Findings
1. Are functional or typed SCOs necessary to fulfill specific requirements of the SIMREF? If so, which ones?	Functional or typed SCOs will be required to meet all of the requirements of the SIMREF.
2. Using a thin client (non-server based) object based delivery mechanism, is it possible to fulfill the requirements of the SIMREF using SCORM 2004 without any extensions?	It is not possible to meet the SIMREF requirements without extensions to SCORM 2004.
3. If extensions other than SCO to SCO data sharing are needed for SCORM 2004 to fulfill the requirements of the SIMREF what would they be?	Extensions that may be required other than SCO to SCO data sharing include the DITA spec, HLA standards, and SOAP specification, and capabilities for supporting the management and tracking of data values and global variables.
4. Using a standard browser-based delivery mechanism, is SCORM 2004 sequencing adequate for fulfilling all of the requirements of the SIMREF? If not, what sequencing specific extensions are required?	SCORM 2004 sequencing is adequate for meeting all of the requirements of the SIMREF. Potential extensions if needed would be support for the passing, storing, and retrieval of data values, support to track and provide global variables, and support for if-then logic.
5. Using a standard browser-based delivery mechanism, is complex arbitrary data sharing between SCOs necessary to fulfill specific requirements of the SIMREF?	Complex arbitrary data sharing between SCOs will be required to meet the requirements 1, 4, 5, 6, and 7 of the SIMREF.
6. Is it necessary to use customized LMS functionality and communications to fulfill specific requirements of the SIMREF?	It is not necessary to use customized LMS functionality and communications to meet the requirements of the SIMREF.
7. As gaps are identified in fulfilling the requirements of the SIMREF, do relationships exist between them?	The null was rejected that the correlation coefficient between LMSCLIENT and SCORMEXT is 0. The correlation coefficient is .588 $p < .05$ indicating a weak positive correlation between LMSCLIENT and SCORMEXT. However, no other relationships exist between variables, experience groups, or between experience groups and the specific variables SCO2SCO 1, SCO2SCO4, SCO2SCO5, SCO2SCO6, or SCO2SCO7.

RESULTS AND CONCLUSIONS

The findings from the data analyses indicated that according to the SCORM development community gaps do exist in the implementation of the SIMREF with respect to SCORM 2004 technical architecture as well as in common implementation practice. These gaps occurred within the communication affordances in the RTE and in the data value/variable management and if-then logic within Sequencing. Gaps are also present in the common implementation practice of using SCOs purely for content presentation. Also perceived by the community are potential gaps in the collection of standards and specifications that define SCORM 2004 in this particular case.

Based on the findings of this study the following conclusions can be stated:

- It would not be possible to meet the requirements of the SIMREF in respect to SCORM 2004 without extensions. Specifically, it will be necessary to extend SCORM 2004 RTE to include arbitrary complex data sharing between SCOs. Potentially, it may be beneficial to extend

SCORM Sequencing to better support the management and tracking of data values and global variables as well as the inclusion of if-then logic.

- There are standards, specifications, and other technologies that could potentially be used to extend SCORM 2004 to allow the SIMREF to be met. These potential standards and specifications include SSP, DITA, HLA, and SOAP. Other technologies that may have potential to support the SIMREF were various web development technologies including Director, Flash, MySQL, and PHP.
- The common practice of only developing SCOs as vehicles to present content will not suffice in this case. Functional or typed SCOs will be required to meet all of the requirements of the SIMREF. Such SCOs may not actually present any content at all but may contain only programming code or functions.
- Although a common practice in integrating simulations with SCORM is to develop and implement a LMS-specific thick client providing specific communication functionality to a LMS, this technique would not be necessary to meet the SIMREF requirements.

The SIMREF contained eight requirements describing functionality necessary to support a simulation-based learning environment. The functionality represented by these requirements supports learner introduction and initial setup; tracking learner profile changes, status and progress; furnishing and receiving simulation input and output data to other systems; providing simulation state feedback to the learner; providing contextual decision-making information to the learner; providing contextual decision coaching to the learner; and providing end-of-period reflection input and storage capability per learner.

Providing contextual and decision dependent functionality requires the broadcasting of status data by some systems and the ability to make sense and act on that data by others. In this case, a specific system would be contained "black-box fashion" within a SCO as a functional SCO. Implicit within the implementation of these requirements is the need to communicate data between SCOs. Also implicit is the potential need for SCO's to persist (co-exist during runtime) - currently not allowed in SCORM.

The argument could me made that SCORM (all versions) allows this communication now through the Run-time Environment (RTE) using the API and the CMI data model. While this may be true to some extent, the CMI data model is a pre-defined somewhat limited model designed to communicate event data to a LMS about events occurring within a SCO. For example, it can communicate a learner's score compared to a preset mastery level indicating whether or not a learner has "passed" the SCO or it could communicate whether or not a learner has "finished" the SCO. It can also communicate other types of SCO related event data including the learner's location within the SCO (i.e. bookmark using the *cmi.location*[1] object). The *cmi.location* object has historically been used for multiple communication purposes and has been suggested as a communication solution to the SIMREF from one participant.

Another capability that could be considered in this context may be the CMI data model's ability to communicate a stream of interaction data using the *cmi.n.interaction* data object. A specific class of interaction called a performance interaction (*cmi.n.interaction.performance*) can track and communicate up to 125 specified and ordered Boolean events. This has the potential of assessing and scoring a learner in a simulation contained within a specific SCO. However, with only 125 Boolean pre-assigned and pre-ordered events, this method may not be robust enough for communicating rich state data snapshots produced by a simulation engine. Other potential CMI objects for storing and retrieving state data are the cmi.launch_data and cim.suspend_data objects. However these and the previous CMI objects, besides being limited in capacity, produce data that can only be read by the SCO producing it. In other words, there is no SCO to SCO communication.

As one of the goals of implementing the SIMREF is not only interoperability but reusability, the above solution may have another serious flaw. In using the CMI model for communication, data would have to be pre-defined either as strings or as arrays of Boolean data hard coded as read-only data within the content package and/or stored by the Run-Time Environment (RTE). Even if the data could be communicated to other SCOs, this would create a tightly coupled situation severely reducing reusability.

SCO to SCO data sharing is discussed as complex arbitrary data sharing in the IMS Shareable State Persistence (SSP) Data Model version 1.0. It is presented as a SCORM extension and describes how the SSP Information Model and its abstract application

[1] The CMI data model uses dot notation indicating objects, identifiers, children, and/or type – i.e. interaction.n.performance where "interaction" is the object, "n" is the identifier, and "performance" is the type.

programming interface (API) are bound to[2] the SCORM Run-Time API using dot-notation. It is complex because it allows data sharing between complex interactive content as in a simulation. It is arbitrary because it allows content objects (i.e. SCOs in the SCORM lexicon) to request allocation (from the runtime service) of an arbitrary number of independent data "buckets" and access those buckets. In this specification, additional data sharing support include the accessibility of persistent data buckets by other content objects and storage requirements of the content object's data buckets that can be explicitly specified as discoverable properties not requiring the content object to be launched (IMS GLC, 2006).

In other words, complex arbitrary data sharing would allow a SCO to define its data storage requirements, store its data in a persistent manner, and allow other SCOs to access and use the stored data as needed. This would accomplish SCO to SCO data sharing and greatly facilitate reuse by encouraging the development of functional SCOs as components in a loosely coupled manner much like that of a Service Oriented Architecture (SOA).

The need for SCO to SCO data sharing has been confirmed with the results of this study (SCO to SCO data sharing being deemed relevant in five of the requirements). Although the relevance rating of SCO to SCO data sharing to each requirement could logically be affected by the SCORM experience of the respondent, this could not be confirmed. A much larger amount of data would most likely be required for these types of relationships to emerge.

Although the findings determined that extending SCORM Sequencing would not be necessary to fulfilling the SIMREF, qualitative data suggested differing levels of agreement and offered specific suggestions. These suggestions encompassed the support for the management and tracking of data values and global variables as well as the inclusion of if-then logic.

The SCORM Sequencing and Navigation book discusses the inclusion of global objective variables with both Boolean or numerical data value storage and tracking capability. It also discusses the if-then model used to determine sequencing rules. However, as a programming language it is very limited. Conditions are relegated to True or False with the exception of the Objective Measure (-1 - +1 values) and types are limited. Resulting actions are also limited in type allowing essentially only navigation decisions. Also, conditions are evaluated from either pre-set "flags" hard coded in the content package or by values contained within the Objective variable. The perception of coding and implementing these sequencing rules may be that they are too low-level much like the difference between an assembly language and higher-level languages in computer programming.

SCOs are commonly developed as vehicles to present content and usually consist of the display and manipulation of text and graphics. They are typically not thought of as performing a specific reusable function or offering a capability for other SCOs to make use of. The findings suggested that all eight requirements of the SIMREF would need a purely functional SCO for implementation. Designing functional SCOs would require a change in how SCORM is typically implemented. This implementation would mean that a SCO contains code for acting on incoming data and sending it back out much like a service in a SOA and may also have a user interface (UI) for directly or indirectly interfacing with the user (learner) as well.

An example of this might be Requirement 3 of the SIMREF, "Data flows as input and output from an embedded simulation." In this example, the SCO contains a simulation engine processing "what if" data and actual decision data from the user (learner). "What if" impact data is available as output to other systems that may display the impact data to the user to evaluate potential decisions or perform other functions. Decision data, impacting the state of the simulation, is available as final results of the decision to other systems for evaluation and display to the user. Therefore, the simulation engine only acts on or processes data based upon user input and internal code and algorithms communicating with other systems for other processing including displaying reports, evaluating remaining budget, viewing simulation status, or coaching.

The other requirements of the SIMREF also imply specific functionality including activities such as role and scenario choice, scenario or backstory presentation, and collecting and tracking of learner reflections. As a set, all eight requirements function together to complete the functionality of the simulation learning environment.

The value of having SCOs perform specific functional behaviors is that these SCOs can function independently of each other creating a loosely coupled environment. Also, the context is in the collection of the SCOs and how they behave together not in the individual SCOs. It is in the collection of these types of SCOs that can come together to define a pedagogical model. Designing is this manner allows SCOs to have

[2] The term "bound to" refers to the mapping, synchronizing, and transporting of data. It is also considered a definition of behavior that can be applied to a data element.

smaller granularity and greater abstraction. Combined with loose coupling, these tenets give the SCOs high reusability for inclusion in different learning environments. Different learning environments could be based on pedagogical models such as another simulation or other pedagogical models giving the designer the ability to design using models of learning.

For example, if designing an exploratory troubleshooting learning environment as described by Jonassen and Churchill (2004), potentially the same types of SIMREF functionality could be applied or reused. The troubleshooting learning environment consists of a case library of previously solved problems, a troubleshooter that enables the learner to practice troubleshooting, and a conceptual model of the system being troubleshot. “Learning objects could be articulated for each of those - conceptual model objects, troubleshooter objects, and case library objects” (Jonassen & Churchill, 2004, p. 39). The preceding quote illustrates that functional or typed learning objects would be needed to fulfill the troubleshooting learning environment.

In keeping with the troubleshooting learning environment scenario, through repurposing or direct reuse, SIMREF functionality could be applied. Areas of application could be to the Conceptual Model accessibility, functions of the Troubleshooter such as “action,” “results,” and “interpretation” as well as Case Library support. A high level diagram of the Jonassen and Churchill model is illustrated in Figure 3.

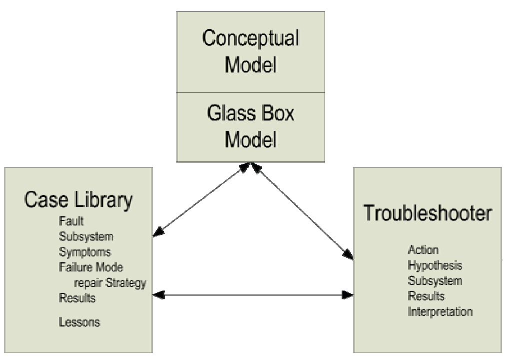

Figure 3 Troubleshooting Learning Environment Model (Jonassen & Churchill, 2004)

To allow discovery enabling reusability of functional or typed SCOs, the application of either extensions to the IMS metadata model if used or in the current applied metadata model would be required. These extensions would support the typing or describing SCOs based upon specific functionality supporting search and discovery during the authoring process or during delivery.

Besides describing functionality, typing could also facilitate the support, use, and reuse of specific types of learning models. For example, SCOs could be developed supporting the use of simulations, or other models including problem-based learning or case-based learning. As a designer is applying a model to a learning solution, SCOs could be discovered based upon pedagogical model type and applied either as is or through repurposing. This would impact design, authoring, and reuse, however, as SCOs supporting one type of pedagogical model or applied instructional theory may require different levels of abstraction and granularity thereby constraining them to possibly only one model. An example of this would be designing SCOs for Component Display Theory where small granularity size and less abstraction might be ideal. This could be in contrast to designing for Instructional Transaction Theory where specific instructional strategies and knowledge objects may require a radically different rationale for the granularity and abstraction which also could be in direct contrast to designing for problem-based learning, situated learning, generative learning, and other models.

In the context of pedagogical models and applied instructional theory, it is not surprising that granularity and reusability of learning objects may be seen as orthogonal. Currently, ADL does not endorse the use of SCO typing as it is seen to lower reusability. However, in essence, it allows the units of reusability to include other things besides SCOs into the realm of pedagogical models implemented at the activity level of the CAM.

Also, in practice even traditional content-based SCOs are not typically designed to offer much reusability. A lack of “designing for reuse” is most likely due to the influence of traditional instructional design’s approaches, strategies, and goals. Instructional designers are trained to approach design in terms of a complete solution for meeting an identified set of learning or performance goals and/or objectives as a single context. Also, instructional designers may fall into the trap of allowing the affordances of most online learning authoring tools and object models to dictate design - commonly a one-way transmission model of learning supporting declarative knowledge acquisition. When translated into a SCORM-based course, designers still think of the course or module they design as a cohesive unit with content breaking down into smaller units of disaggregation – i.e. courses, modules, units, lessons, and topics. This breakout is typically described by the SCORM CAM as either clusters of activities, activities, and SCOs. Unfortunately, the above breakdown is a common practice as described by the Learning Systems Architecture Lab in their SCORM Best Practices Guide for Content Developers (Learning Systems Architecture Lab [LSAL] 2004). However ADL does not advocate this tradition as it is recognized to severely limit design and reusability.

To combat this tradition of non-reusability, designers will have to begin designing for reuse. This will include systems thinking at the macro of "course" level as well as at the micro or SCO (learning object) level. They will need to begin thinking not only about instructional purpose but the functionality supporting instructional purpose and how it supports the pedagogical models they are using. Also, understanding how enterprise IT (information technology) learning tools such as LMSs function in high level terms and what their goals and purposes are may help instructional designers better understand not only their emerging toolset but why reuse needs to occur.

A common practice in integrating simulations with SCORM is to develop and implement a thick client that provides specific communication functionality to a LMS but was determined not to be necessary to meet the SIMREF. This technique is similar to that described in the SITA implementation (Haynes, Marshall, Manikinda, & Maloor, 2004) as it uses a simulation engine external to the SCORM environment and the thick client is treated as a SCO being the liason between the simulation and the LMS. This solution breaks down in several ways. First, a thick client is more than a standard web-browser potentially creating interoperability and bandwidth issues for the user. Second, as in the SITA approach, it treats the simulation as an entity external to the learning environment, and third, accessibility is severely limited as the simulation engine exists on a remote server.

In contrast to the above techniques and to SITA, a purely SCO-based solution as outlined in the SIMREF contains a simulation engine existing as a SCO with other simulation support functionality existing as other SCOs. The content package containing these SCOs would be downloaded at the point of use creating the potential for off-line use with LMS synchronization occurring at a later time.

Implications and Recommendations

From the conclusions reached and the ensuing discussion, there are several implications from this study that could result in recommendations for the SCORM 2004 specification and implementation practices. First, in order to accommodate a simulation encapsulated in a content package, SCORM functionality should be extended in facilitating inter-SCO communication. The logical technology to accomplish this type of communication is IMS Shareable State Persistence (SSP). Although other standards and specifications exist, SCORM 2004 would benefit from including the SSP specification as permanent component.

SCORM Sequencing may benefit from tools supplying high level programming language capabilities for authoring or developing sequencing logic for SCOs. The IMS Simple Sequencing Specification itself may benefit from extensions to its if-then logic and the inclusion of a more robust set of actions.

Designing for reusability could occur at macro and micro levels when using SCORM. These levels include the SCO or learning object, SCORM activities, and other representations of pedagogical models. In the relationships between these components or levels, this approach could be used to determine their necessary level of abstraction and granularity size. This potentially may enable the development, use, and reuse of advanced pedagogical models.

The way SCORM is implemented currently should change to include the addition of functional and typed SCOs. This implementation would be facilitated by the addition of SSP to SCORM. By implementing this change to current practice, designers and developers would need to change the way they approach and think about design and development to include both macro and micro systems approaches and an understanding about enterprise learning technologies and what is gained by designing for reuse. This change implies that the educational programs for instructional design and instructional technology may need to change to accommodate systems thinking, reusability design tenets, and enterprise approaches to e-learning and knowledge management. Also implied is the need for a closer marriage of information technology and instructional technology in the preparation of instructional designers and IT developers working in the instructional technology field. This marriage should focus developers to work closely with designers to understand and translate learning designs into functionality.

Typically, graduates from instructional design and instructional technology graduate programs do not possess understanding in design perspectives encompassing reusability. They also do not have an understanding of design in an enterprise environment and/or the underpinning technologies of learning objects, learning object content models, and enterprise learning systems. Also, typically, graduates from computer science (CS) or information technology (IT) undergraduate and graduate programs do not have an understanding of how CS and/or IT supports and facilitates learning. This condition is illustrated by the response patterns within the sample of this study. For those with experience primarily in IT, responses were closely aligned across all requirements which tended to be the opposite of responses by those whose primary experience was that of instructional systems design. Also, the answers from the open-ended items concerning alternate solutions only came from those with IT experience.

A lack of understanding typically present in instructional designers may be due to the focus on learning theory, instructional design processes, and instructional strategies and not on the technological and enterprise landscape in which these theories and processes will be applied. This occurs assuming that as they build an instructional plan, a programmer will then implement their plan in some environment possibly not knowing anything about what that environment may consist of or how it will impact or be impacted by other environments.

A lack of understanding by those in CS or IT may be due to the focus on the gathering of requirements and building of systems based upon those requirements – not from the generation or theoretical understanding of the gathered requirements. When working together to design and implement enterprise learning technologies, requirements generation actually occurs during the design by instructional designers with the implementation of those requirements occurring during system development by system and software engineers. This dichotomy leaves a gap in the understanding necessary for optimum design and development of enterprise learning technology systems. As instructional designers work closely with developers, an understanding needs to occur about the role each has to play in designing and implementing reusable, functional SCOs and pedagogical models. To bridge this gap, curriculum development and evaluation of a blended field of instructional design/technology and computer science/information technology should occur. Graduates from the curriculum could function as instructional architectures or instructional engineers designing and applying research derived models to solve enterprise learning problems.

Most specifications and standards including those from AICC, IMS, DMSO, the IEEE LTSC, ARIDNE as well as SCORM, have been designed by IT developers who may not be native to the field of education and training and may not have a deep understanding of learning theory and practices. In fact, most of those that are and have been heavily involved with the design and evolution of SCORM come from the electrical engineering, computer science, and other technical fields.

The design of the CMI data model (incorporated into SCORM from AICC) reflects a behaviorist model in the learner interactions supported and types of data stored and tracked. The limited CMI data types include (among others) completion of a SCO presented to the learner, objectives, scores, and interactions. Multiple interactions exist within the CMI including multiple-choice, short and long text fill-in and even performance types. However, the interactions and the data types in the model represent only what can be collected through a learner's response to a given stimulus and is usually quantitative in form. In practice, this collection usually occurs through the presentation of information, the presentation of questions on that information, and the responses of the learner to the questions. Responses are gathered using the interactions and are evaluated for score or pass/fail which is sent via the Run-time Environment (RTE) to be stored in the LMS. This occurs at the SCO level and learning is only tracked by the LMS by the completion or passing of a SCO. In this fashion, there are some comparisons to programmed instruction or even mastery learning.

Prior to SCORM 2004, SCOs were commonly available to a learner to take in whatever order they wanted. This was seen as an attempt to support a more exploratory learning environment with learner self-direction. With SCORM 2004, SCOs can also be sequenced based upon pre-defined rules which are based upon the attainment of learning objectives defined globally within a SCORM course. The sequences of SCOs are set up in a score threshold or pass/fail navigation model which still supports mainly an objectivist view of learning.

The limitations of the CMI do not govern what occurs instructionally within a SCO, but it limits what can be communicated by a SCO to the outside world. For example, a single SCO could have a complex simulation with a 3D user interface (UI). A learner could interact with the SCO at length but the only data communicated by the SCO would be limited to the CMI data model. It would not be possible for the rich data set that would be produced in this case to be utilized as evidence of competency attainment or understanding. This limitation in combination with the lack of SCO persistence, a limited model of sequencing, and the individualized nature of SCORM reflects an inability to support constructivist learning tenets such as alternative assessments and activity-based learning.

In all fairness to SCORM, however, these limitations are embodied in most all online learning environments utilizing a learning object content model for individualized self-paced instruction and may not be only due to the model itself but also to common instructional design practices. By not having a thorough understanding of what is capable within a LOCM such as SCORM, instructional designers fall back on easy to design and easy to program models that end up as what is commonly referred to as "page turners."

These conditions may explain why the inherit pedagogy supported by SCORM is one primarily based upon information or content transmission. It may also explain the differences in the responses from study participants depending on where their experience lies. If applied learning theorists and those versed in fields such as instructional design, instructional technology, and educational psychology had been more involved in its inception, models supporting more meaningful learning experiences may more easily be supported and applied by SCORM. Consequentially, specifications and standards comprising SCORM and their implementation practices reflect an underlying pedagogy that is based in behaviorism, does not agree with contemporary theories of learning and practice, and will not support more constructivist models learning - i.e. simulations.

As SCORM moves into its next evolutionary state through the formation of its new steward tentatively named Learning Education Training Standards Interoperability (LETSI), it is time to actively reach out to the academic community for support, critique, and inclusion. This overture would help ensure that future iterations of SCORM not only support but embody current understandings about learning and pedagogy. However, in so doing, the academic community also needs to see the need and value of designing and implementing in an efficient and cost saving manner. The academic community also needs to understand how to incorporate reusability and interoperability in the artifacts of instructional design and not dismiss these tenets as not relevant or completely orthogonal to good instruction and meaningful learning.

REFERENCES

adlCommunity. (2007, 7/17/07). "Main Section for adlCommunity." Retrieved 7/17/07, 2007, from http://adlcommunity.net/user/view.php?id=5&course=14.

ADLNet. (2006). "DoD Publications." Retrieved 05-03-06, 2006, from https://adlregistry.dtic.mil/2/index4.htm.

Aldrich, C. (2006). Beyond Books**:** 66.

Alessi, S. (2000). Building Versus Using Simulations. Integrated and Holistic Perspecitves on Learning, Instruction and Technology. J. M. Specotr and T. M. Anderson. Netherlands, Kluwer Academic Publishers**:** 175-196.

Cockburn, A. (2001). Writing Effective Use Cases. Boston, Addison-Wesley.

Defense Modeling and Simulation Office [DMSO]. (2006, 5-08-2006). "Defense Modeling and Simulation Office (DMSO Website." Retrieved 6-9-06, 2006, from https://www.dmso.mil/public/.

Driscoll, M. P. (2000). Psychology of learning for instruction. Boston, Allyn and Bacon.

Gallagher, P. S. (2002). Distributed Learning: Best Practices in Industry, Academia, and Governmental Agencies. McLean, SAIC**:** 49.

Gallagher, P. S. (2005). Design for Knowledge Management. Innovations in E-Learning Symposium 2005, Fairfax.

Grimmitt, M. (2000). "Pedagogies of Religious Education: Opening up a new field of research in RE." Retrieved 18 September, 2007, from http://www.studyoverseas.com/re/re.htm.

IMS Global Learning Consortium, I. (2006). IMS Shareable State Persistence Information Model. Retrieved 6/20/07, 2007, from http://www.imsglobal.org/ssp.

Haynes, J., Marshall, S., Manikinda, V., & Maloor, P. (2004, December). *Enriching ADL through integration of HLA Simulation with SCORM Instruction with SITA (Simulation-based Intelligent Training and Assessment).* Paper presented at the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2004.

James, S. W., Kinnear, T. C., & Deighan, M. (1999). *PharmaSim: A brand management simulation administrator's guide v 2.0*. Charlottesville: Interpretive Simulations.

Jonassen, D. and D. Churchill (2004). "Is There a Learning Orientation in Learning Objects?" International Journal on E-Learning **3**(2): 32-41.

Koper, R. (2003). Combining Reusable Learning Resources and Services with Pedagogical Purposeful Units of Learning. Reusing Online Resources: a sustainable approach to e-learning. A. Littlejohn. London, Kogan Page**:** 46-59.

Learning Systems Architecture Lab [LSAL] (2004). SCORM Best Practices Guide for Content Developers. Pittsburgh, Carnegie Mellon University.

Maier, F. H. and A. Grobler (2000). "What are we talking about?—A taxonomy of computer simulations to support learning." System Dynamics Review 16(2): 135-148.

Menaker, E., S. Coleman, et al. (2006). Harnessing Experiential Learning Theory to Achieve Warfighting Excellence. Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2006. Orlando, Interservice/Industry Training, Simulation, and Education Conference.

Miller., M. and R. Childs (2004). Practical modeling & simulation for online courses: Proposal for A low level architecture. World Conference on

Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA) 2004, Lugano, Switzerland, AACE.

Norton, P. (2003). COPLS: An Alternative to Traditional Online Course Management Tools. Society for Information Technology and Teacher Education International Conference 2003, Albuquerque, New Mexico, USA, AACE.

Norton, P. and D. Sprague (2001). Technology for Teaching. Bostin, Allyn and Bacon.

Oliver, R. and C. McLoughlin (2003). Pedagogical Designs for Scalable and Sustainable Online Learning. Reusing Online Resources: a sustainable approach to e-learning. A. Littlejohn. London, Kogan Page.

Olivier, B. and O. Liber (2003). Learning Content Interoperability Standards. Reusing Online Resources: a sustainable approach to e-learning. A. Littlejohn. London, Kogan Page.

Sallaberry, C., T. Nodenot, et al. (2005). Model Driven Development of Cooperative Problem-Based Learning Situations Implementing Tools for Teachers and Learners from Pedagogical Models. HICSS '05: Proceedings of the 38th Annual Hawaii International Conference on System Sciences. Hawaii, IEEE**:** 40c - 40c

Shank, P. (2001). "Out with the Old: Is it Time to Rethink Instructional Design?" Online Learning: 1-4.

Sims, R. C. H. (1997). " Interactive Learning as an "Emerging" Technology: A Reassessment of Interactive and Instructional Design Strategies." Australian Journal of Educational Technology 13 (1): 68-84.

Swinski, J. and M. Williams (2004). Web Browser-based Architecture for Reusable Simulation Objects (RSOs) in PC Simulations. Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2004. Orlando, I/ITSEC.

Watson, W. R. and S. L. Watson (2007). "An Argument for Clarity: What are Learning Management Systems, What are They Not, and What Should They Become?" TechTrends 51(2): 28-34.