Learning Technology

publication of

IEEE Computer Society

Technical Committee on Learning Technology (LTTC)

Volume 7 Issue 1 | ISSN 1438-0625 | January 2005

Special Issue on

SCORM 2004 Sequencing & Navigation

Guest Editors:

Dr. Eric Roberts
Chief Scientist for Learning,
Advanced Distributed Learning Initiative
Ejrphd@aol.com

Dr. Michael W. Freeman
Deputy Director,
ADL Initiative
freemanm@adlnet.org

Dr David Wiley
Assistant Professor,
Utah State University
david.wiley@gmail.com

Dr Demetrios Sampson
University of Piraeus and CERTH
Greece
sampson@unipi.gr

Guest Editorial

The release of SCORM 2004 (Sharable Content Object Reference Model) represents a moment of stability, according to official statements from the Advanced Distributed Learning Initiative (http://www.adlnet.org). ADL derived its initial SCORM 2004 "Sequencing and the Navigation" book on the IMS GLS Simple Sequencing specification, version 1.0, and then developed refinements that permit its integration with the other parts of SCORM. This version of the document suite realizes the functionality originally imagined as a necessary baseline capability. Changes are expected to be minimal, repairing small pieces when bugs are identified, decreasing in frequency and scope over time.

This is true only from an engineering perspective, however. It is true that the functionality is stable. That is an important point for those charged with investing in instructional materials. Yet the functionality being described in SCORM only defines communications between instructional content and learning management systems. What creative instructional designers can accomplish for the benefit of learners -- using such communication functionality has only just begun to be demonstrated – as the papers collected in this edition of the Newsletter show.

These papers were received in response to a formal call put out by ADL for a Workshop on SCORM 2004 Sequencing & Navigation. The first such workshop ever held, the intent was to speak directly to critics who decry how SCORM limits the conduct of instructional design by showcasing clever uses of SCORM 2004 Sequencing and Navigation capabilities. Even using the relatively limited functionality of SCORM v1.2, some of the paper authors show how they have exploited the technology to solve problems never imagined by SCORM architect and developers. The Arora paper, from the Dhirubhai Ambani Institute of Information and Communication Technology, discusses a novel application to address a need for disseminating scientific information. JGP Consulting in the UK describe the use of Topic Maps to provide learners with contextual information in order to traverse content in an intentional, purposive manner. Papers from U.S.-based Boeing and Intelligent Automation, Inc. provide examples of how traditional, didactic, SCORM-conforming instructional content can be developed to work with complex, dynamic simulations that use the High-Level Architecture. Both are exemplary designs that realize significant instructional goals. Researchers at Tamkang University in Taiwan discuss one interesting way of providing learners with the benefits of SCORM individualized instruction as well as peer-to-peer interactions in a collaborative learning environment. The Ostyn paper offers a degree of abstraction in curriculum planning through the use of content-independent templates for developing instruction as well as assessments. Finally, researchers at the National University of Ireland and the University of Memphis, in Tennessee, consider how Intelligent Tutoring System functionality can be approached using the Sequencing and Navigation capabilities availed in SCORM 2004.

Perhaps not commonly known, it is interesting to note that the affordances of such instructional technologies were in the minds of ADL and SCORM creators from the beginning of the initiative in 1997. It is the case that industry-provided "use cases" drove the development of SCORM. Training scenarios of the type most often found in U. S. Department of Defense technical training were used to ground and orient the development of the reference model. The use cases provided a touchstone to determine if the problems that were intended to be solved actually were being solved. At the same time, SCORM developers included certain constructs that were intended to be used in ways un-imagined in the use cases but frequently discussed by instructional technologists. One of these is a rudimentary student model –one of the defining elements of any intelligent tutoring system.

So why were these constructs and technologies not included in the use cases?

Looking back over the history of ADL and the development of SCORM, it appears that one of the keys to the success of the effort has been the negotiated, consensual determination of the model. The mandate to ADL to "advance the state of the art" in the development of instruction for the Pentagon was contradicted to some degree by the additional mandate to include all industry stakeholders in determining how this was to be accomplished. Some have said that the compromise resulted in an overwhelming momentum purchased, perhaps, at the expense of a less-than-wholly-elegant architecture. That momentum is attested to by the array of organizations and institutions represented in these papers.

To be sure, many of the papers included here do reflect an authoritative, directed orientation to the conduct of instruction. DoD training is usually focused on imparting and certifying specified levels of competence at demonstrable, job-performance skills as opposed to completely individualized goals of self-actualization that are more common in institutions of higher education. But SCORM 2004 is stable now. That is no longer a constraint. We can use it how we want.

The challenge is issued: how clever can you be? What can you imagine? What can you make?

We look forward to the next Special Issue on Sequencing and Navigation to celebrate your answers.

Dr. Eric Roberts
Chief Scientist for Learning,
Advanced Distributed Learning Initiative
Ejrphd@aol.com

Dr. Michael W. Freeman
Deputy Director, ADL Initiative
freemanm@adlnet.org

Dr. David Wiley
Assistant Professor,
Utah State University
david.wiley@gmail.com

Dr. Demetrios Sampson
University of Piraeus and CERTH,
Greece
sampson@unipi.gr

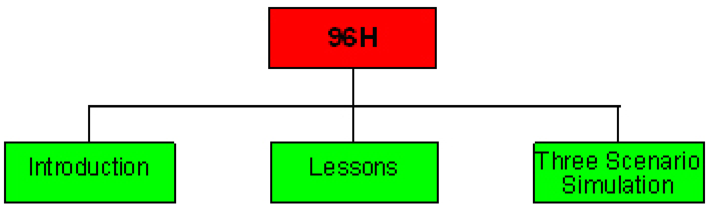
Converting the Common Ground Station Operator's Course from SCORM 1.2 to 2004

In November, 2001, Imedia.it, Inc. was tasked with developing an Interactive Multimedia Instruction (IMI) product to provide training on the basic operation of the Common Ground Station (CGS). The CGS is a system which acquires, processes, displays, and disseminates data from multiple sensors including Moving Target Indicator (MTI) radars; Unmanned Aerial Vehicles (UAV); Imagery Intelligence (IMINT), Signal Intelligence (SIGINT), and Electronic Intelligence (ELINT) platforms. The purpose of the course was to prepare Army Reserve and National Guard soldiers for reclassification into the 96H MOS and replaces 70 hours of classroom instruction from the active component CGS Operator's Course currently taught at Fort Huachuca, Arizona. Initially this was to assist reserve soldiers in becoming MOS qualified without having to attend the full 19 week course. Although that option has not been fully developed at this time, the active component school at Fort Huachuca is now considering adding it to their Program of Instruction (POI) as an additional resource for student review and practice. The course is currently in validation at Fort Huachuca; SME reviews have been completed, and group trials are currently underway.

The enabling learning objectives for the CGS course come from 23 lessons out of the active component POI. These lessons were specified by the Distance Learning office and the Army Reserve design team at Fort Huachuca as providing a foundation which would enable soldiers to conduct basic operations in the CGS. The IMI package contains these 23 knowledge-based lessons as well as 3 assessment scenarios. Each of the assessment scenarios is comprised of three vignettes.

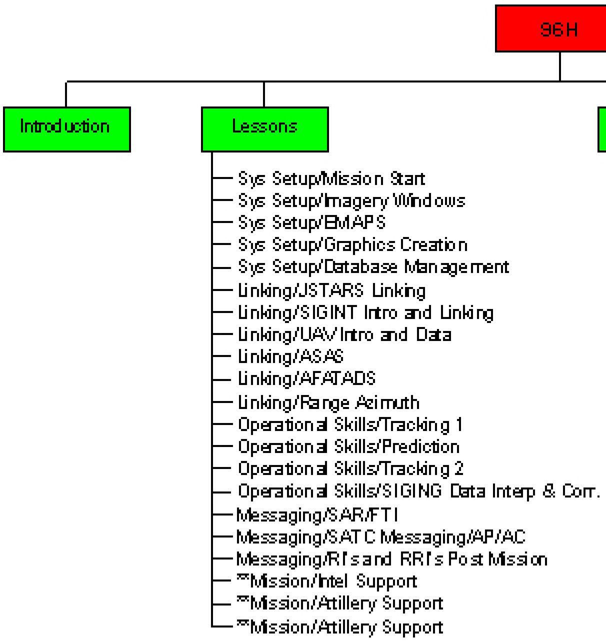
The assessment scenarios provide realistic situations for the student to practice with the CGS software. Using animated graphics, the student is put into a setting where his performance on his mission can be tracked and assessed. The missions are carried out using the CGS e-Trainer, which provides an accurate emulation of the software capabilities of the CGS system. This emulator, programmed in JAVA, allows the students to perform the prescribed tasks just as they would in a CGS. The emulator tracks their performance on these tasks and records them to a SQL database for tracking purposes. The performance results (the AAR) are then displayed to the student in an ASP page. All errors are linked back to the 23 lessons for remediation purposes.

The Statement of Work (SOW) specified that the course be SCORM 1.2 compliant. The resulting course structure is divided into 24 independent SCOs and one dependent SCO. Each of the knowledge-based lessons is a separate SCO, and the three assessment scenarios are combined in a SCO. Incorporating the assessment in a single SCO allows for control navigation within the assessment module. In order to provide the remediation from the AAR requested by the client, a copy of the lessons is embedded in the scenario SCO. Book marking is done on two levels. The LMS tracks that the student is in the assessment module. Tracking within the assessment module is done at the vignette level and stored in the SQL database. The introduction is a dependent SCO. The Activity tree for the SCORM 1.2 compliant course is shown below.



When the course was restructured to be SCORM 2004 compliant, the objective was to take advantage of the sequencing and navigation control offered. The LMS would be used to control all navigation and book marking. Additionally, it would be used to direct remediation. The assessment module was broken into nine chunks by defining each vignette as a SCO. The vignettes were then combined into three aggregations. To break the module down it was necessary to rework the transition pages so there was a smooth flow back to the LMS at the end of the vignette. Here the next step was either advancement to the next vignette or remediation. Additionally, it was necessary to see that the beginning for next vignette was not awkward.

At this time some questions remain. First, the original remediation directed the learner to a specific point within the lesson. Our new application merely directs the learner to the beginning of the lesson. We are looking into how we might still direct the learner to a specific point in the lesson. Secondly, the simulation module utilizes a database to facilitate transfer of the learner performance data from the Java code to the LMS. We are continuing to examine how we can drop use of the database. The new Activity Tree for the SCORM 2004 compliant manifest is shown below.



Don Holmes
Imediait
don.holmes@imediait.com

Jaime Henderson
Imediait

Jeff Choat
Imediait

Conversion of Air Force Modeling and Sim course to SCORM 2004

Background

The course domain is an introduction to Modeling and Simulation (M&S) for Air Force (AF) personnel. We originally planned to convert the course to a SCORM v1.2 conformant course. At the time the conversion effort began, the subject course was being delivered through Meridian Learning Management System (LMS). This LMS offered the ability to deliver SCORM v1.2 courses, but since SCORM 2004 had just recently been released, the LMS did not yet support SCORM 2004. This was the primary basis for creating a SCORM v1.2 course. Later on the decision was made to convert the course to SCORM 2004, due to the ability to sequence and navigate the content to produce a more robust and effective course.

Tools

ReLoad: The free, open-source Reload Editor was a good tool for our purposes. It is a tool designed to facilitate the packaging of existing content into SCORM conformant content packages. Two different versions are available: 1) the original version which creates SCORM 1.2 packages and 2) the updated version by the Alexandria ADL Co-Laboratory which creates SCORM 2004 packages.

Word: We built a design map, or course outline in Word, which really helped to clarify and define the organization. For example, see part of outline below:

L02: Lesson 2: Categories and Processess (aggregation)
    L02-C1: Lesson Intro (SCO) (child of aggregate)
    L02-C2: Legacy and Next Gen systems (SCO) (child of aggregate)

There was discussion about whether to use the reload tool metadata generator. We weren't sure if all data elements would be supported for SCORM 2004 yet, so we decided to enter them manually using textpad.

CMU Templates: The Carnegie Mellon University templates were valuable in relating the design of the course to SCORM functions, allowing easier transformation of ISD requirements to SCORM implementation. For example, for each aggregation, there was a section for: metadata (this could be filled in during process, so that information was already there when metadata files were built); objective information; identifiers for children (names of the children of the aggregate); aggregation structure (we could show a diagram of the aggregate structure); and sequencing behaviors. These templates are helpful, but it would be great to have a tool to go from the CMU templates info, straight into reload tool package. Right now the CMU templates metadata are not in the same order as the application profiles require.

Sample RTE: We used the sample Run Time Environment (RTE) to test the content, due to scarcity of SCORM 2004 conformant LMS. This sample RTE platform gave us a chance to view the content after it was developed. The look and feel of the particular SCOs and learning objectives could be discussed. The navigation of the SCOs could be seen and debugged. Since the content was being converted to SCORM conformant content, there was much discussion about how the content may have to be changed to accommodate navigation rules without compromising Air Force policy.

SCORM Test Suite: Once the content ran successfully in the sample RTE, then we tested it using the SCORM Test Suite Version 1.3.1.

Initial Considerations

Size – The AF M&S Introductory Course ("course") is quite large in storage size due to the extensive use of multi-media audio and video. As a consequence, we will need to ensure we are efficient in our SCORM packaging (i.e. only including files in the package that are used by the lessons.) Also we consulted Air Force Policy regarding SCO size and found that Aggregation should be at lesson level, and SCO should be at learning objective level.

Context – The course currently makes use of a main menu, which will not be available in the SCORM version of this course. So we will likely need to make some adjustments to the content. For example, the course introduction as a SCO, being launched by the Sample RTE shows a reference to a non-existent Main Menu button.

Asset Definition – There are a lot of assets in this course. Ideally, each of the assets used by the course will be referenced explicitly in the manifest file (rather than just existing in the Package Interchange File (PIF) – the compressed zip file). This will require a lot of Resource entries in the manifest and some decisions as to what these entries will look like. For instance, should all of the navigation gif image files (back, forward, help, etc.) exist as individual assets? Or should they be packed together as one asset?

LMS Communication – Do we wish to have any learner tracking activities with this course? For example, do we want to require the course introduction to be viewed prior to taking any lessons? Do we want the LMS to enforce any time restrictions associated with any lessons? Do we want any scores to be communicated back to the LMS?

Meta-data – It seems to make the most sense to begin constructing meta-data after the assets have been defined (i.e. the manifest's resources have been completely defined). This approach should prevent the possibility of wasting time constructing meta-data for an asset, which might be rolled into another (as in the navigation images example above).

Other considerations: We decided it would be best to include each quiz question with the associated learning objective versus making them their own SCOs. We discussed how content may have to be changed due to navigation of LMS, and obtained input from Air Force on allowed changes to content. We discussed about how pretest and post test might be tracked by LMS; the instructional design choice was for these tests to be tracked, but not to allow the student to test out of any lessons.

Snippets from our design and development notes

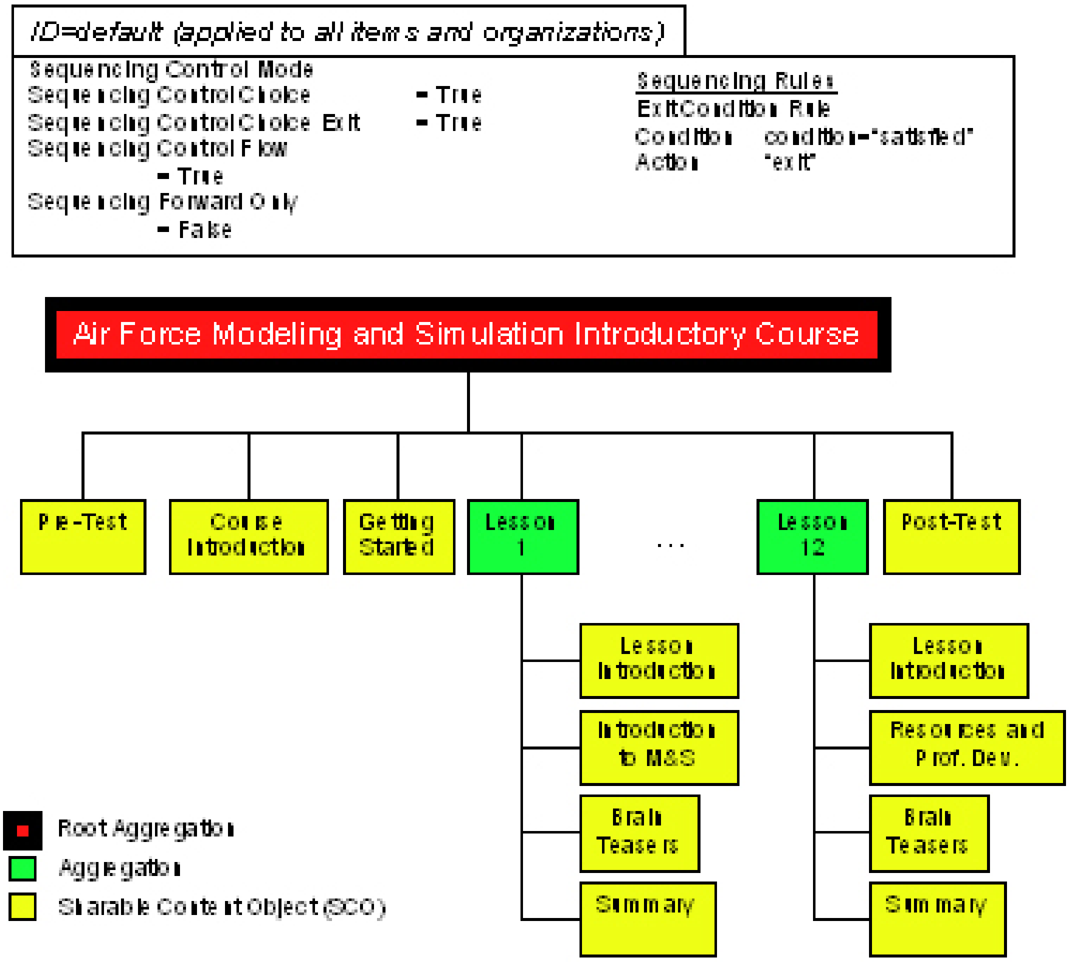
Building the SCOs: The newly created index file is responsible for creating the frameset of the lesson and has been updated to reference global media files, such as the global navigation frame source. Once the lesson directory is prepared, copy the directory to a new directory – this will be your first Sharable Content Object or Sharable Content Asset of the lesson. I made my directory names the same as the identifiers used in the specification templates (e.g. "L05-C-1" is the directory where the first child of the lesson 5 aggregation can be found). Delete any files from this directory that are not relevant to the SCO (i.e. html files, graphic files, flash animations, other directories etc.). If the SCO does not reference them, delete them – we don't want to waste space.

Update the index file to reference your first html file – the point where the SCO begins. Edit this html file. All references to shared media files should already be updated due to the preparation step; however, if any references appear incorrect, now is another time to change them. Also, when building a SCO, global search and replace is very helpful in this process (i.e. find and replace in all files instances of src="gui" and replace with src="../gui"). I used TextPad extensively, www.textpad.com. If you use search and replace, be sure to pay attention to both double quoted strings and single quoted strings – "pic1.gif" is not the same as 'pic1.gif'.

Building package with sequencing rules: In reload tool, right click on the lesson node, "Lesson 5: Organization". Select "Advanced Sequencing: Add Sequencing". A new sequencing node appears under the lesson. Right click on this new sequencing node and select "Advanced Sequencing: Add Control Mode". Set all the values (choice, choiceExit, etc.) to true except for the value of forwardOnly, set it to false. Right click on the original sequencing node again and select "Advanced Sequencing: Add Sequencing Rules". A new sequencing rules node appears. Right click on it and select "Advanced Sequencing: Add ExitCondition Rules". A new set of nodes appear. Set the rule action node's action to "exit". Right click on the "ExitCondition Rules" node and select "Advanced Sequencing: Add Rule Conditions". A new set of nodes appear. Set the Rule Condition's operator to "noOp" and its condition to "satisfied".

Issues: Evidently the Sample RTE does not offer any reporting capability to retrieve the data model element cmi.interaction at this time. This is something under consideration for future Sample RTE releases. This is important because, for this course, we wanted the LMS to track such things as which question on a test multiple students got wrong, so we could see a trend.

Activity Tree showing application of sequencing rules



Conclusions: Use tools whenever possible! Monitor adlnet.org for updates to document, test suite, and tool releases. Run content on multiple LMS if possible.

Damon Regan
Joint ADL Co-Lab

Susan Marshall
Joint ADL Co-Lab
susan.marshall1@us.army.mil

Patricia Mulligan
Air Force Agency for Modeling and Simulation

The Boeing Fighter Training Center I/ITSEC 2004 Demonstration: Integration of SCORM 2004 S&N and a Weapon System Trainer

Introduction

This paper describes how we integrated a commercial off-the-shelf (COTS) learning management system (LMS), SCORM-Compliant training that used SCORM 2004 Sequencing and Navigation, a full-fidelity DIS-Compliant flight simulator device, and automated performance assessment for the Boeing booth at the Interservice/Industry Training, Simulation and Education Conference (I/ITSEC) December 2004.

The goal of the demonstration was to showcase Boeing technology for Net-Centric Learning Management by letting people experience the concept first-hand. The demonstration featured technology that used DIS to automatically assess student performance in an F/A-18 weapon system trainer (flight simulator) and report the assessment to a COTS LMS. To verify the ability to perform complex performance measurements, we chose a mission that consisted of several complex and semi-complex maneuvers including taking off, climbing, turning, selecting weapon, operating the mission computer, and launching a missile.

The demonstration was held at a very large trade show that had a wide variety of attendees from various military, academia, government, and industry careers plus was open to the public. In order to be enjoyable to everyone who experienced the "Boeing Fighter Training Center," we had to create several missions of various degrees of difficulty and properly prepare each "student" to fly their specific mission in a way that was stimulating to each of them. Since the trade show was held in conjunction with a very busy conference, a very tight time schedule constraint was imposed for the training.

This paper discusses the SCORM-compliant, web-based training that was created to determine which mission the student should attempt and prepare the students to fly that mission. It includes a brief discussion of the SCORM 2004 sequencing and navigation rules, CMI data model elements, and API functions used. Due to space limitations, this paper includes a very brief discussion of using the SCORM to integrate the simulator performance assessment with the LMS. This paper does not discuss the simulator performance assessment technology or the evaluation criteria used. Those subjects will be discussed at the WoSS&N workshop and in forthcoming papers and presentations by Boeing.

The Problem

Provide stimulating, engaging, quality training consisting of basic information, intermediate knowledge, and mission information custom-tailored to students with a wide variety of proficiency within a very tight training time schedule.

Solution

ADL's vision is to "Provide access to the highest quality education and training, tailored to individual needs, delivered cost effectively, anywhere and anytime." SCORM 2004 was designed to provide the ability to deliver custom-tailored, quality training integrated with an LMS. Therefore, the solution was to create SCORM 2004 Compliant content that provided an assessment, proper sequencing and automated navigation through the variety of content.

Details, Design Decisions, and Lessons Learned

Three simulator missions and sets of assessment criteria were created for novice, intermediate, and expert students which we titled "Wingman," "Flight Lead," and "Mission Commander." The same skill categories were used for the training which had to both familiarize the student with the cockpit (Cockpit Fam) and had to introduce the particular mission and key grading criteria (Mission Brief). Therefore, the training consisted of a pre-assessment (Proficiency Eval) and three different paths through two topics; followed by a common simulator mission generator and performance assessment interface. Multiple entry points were defined for remediation to specific topics within the familiarization and mission briefs based on simulator performance.

The activity tree for this lesson can be expressed two different ways as shown in Figure 1 below. Option A required fewer sequencing and navigation rules but the course outline as presented to the student was not as intuitive as Option B.

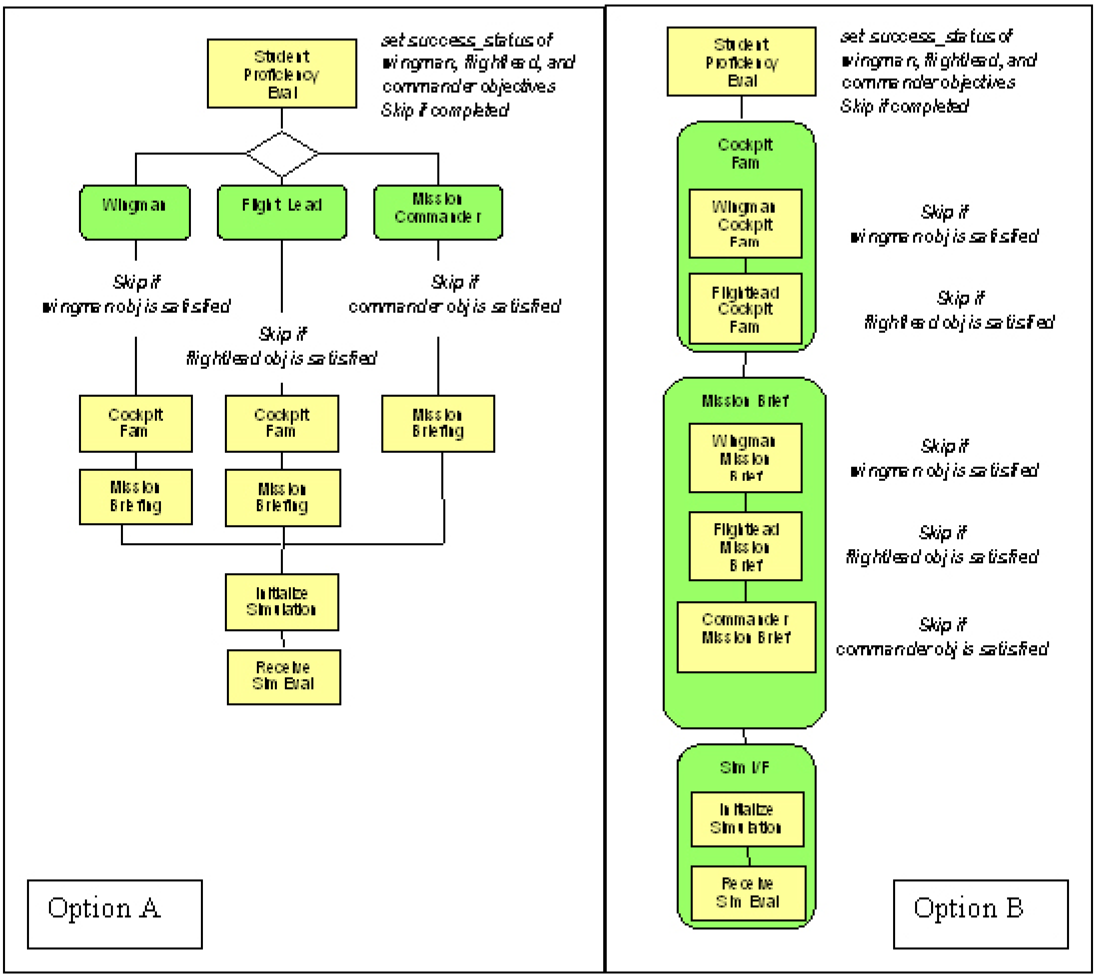


Figure 1. Boeing Fighter Training Center WBT Activity Tree

Proficiency Evaluation

The performance evaluation unit was used to determine the level of training and mission to present to the student. The key decisions were first, how familiar was the student with the F/A-18 cockpit, its controls, and the terminology used; and second, how skilled is the student in flying an aircraft and operating a weapon system. The evaluation sets the ***success_status*** of the *wingman*, *flightlead*, and *commander* objectives.

Although this unit would ideally be a performance assessment, the short seat time did not allow time for novices to fumble around. Instead, a questionnaire was designed to assess whether the student was familiar with a) how to fly an aircraft; b) where the controls on an F/A-18 are; c) how to engage a threat and launch a missile. The first questionnaire, consisting of several questions, had a complex scoring algorithm that rated the student on a normalized score. However, the questions and answers were often misinterpreted and many people were uncomfortable with the choices. Additionally, the logic used in scoring was very difficult to understand. The sequencing rules that sequenced based on that score was also hard to understand. Therefore the, questionnaire was just as time-consuming as the performance evaluation and often provided a faulty recommendation.

The questionnaire ultimately used at I/ITSEC 2004 consisted of one question that asked the student to describe himself as a novice, an intermediate who knew how to fly aircraft other than a fighter aircraft, a skilled fighter pilot who was unfamiliar with the F/A-18, or an experience F/A-18 pilot. Each answer in the multiple-choice question was tied directly to one of three skill ratings. Until amended to include simulator experience, the questionnaire assessed a large number of highly skilled students as “novices.” While the amended question provided a decent categorization of skills, there were still a number of pilots that should have been in the intermediate level classified as novices.

better proficiency evaluation would have been provided by a simple interactive knowledge-based assessment with direct ties to objectives such as “knows location of throttle”, “knows purpose of throttle”, “knows location of TDC”, and “knows how to operate TDC”. To implement using simple sequencing, each answer sets the ***success_status*** for all three “objectives;” failing one, and passing two of them. Although this closely matches the common use case of scoring a pre-test and skipping units of instruction if topics are mastered, it is disconcerting to view the novice as having “mastered” the intermediate and expert curriculum. A possible solution is to add another sequencing variable of “*assigned*” in addition to satisfied and *completed*. Whenever branching is based on skill level or career path, this should be considered as a review records showing that the student mastered material that had not been assigned can lead to a great deal of problems. The student did not successfully master or complete the units of instruction nor the objectives they represent. Rather, based on the student’s current skill level or job assignment, those units (objectives) are not (yet) relevant.

Cockpit Familiarization and Mission Brief Units

The cockpit familiarization and mission brief units consist of a single dynamic web page from the Boeing Virtual Training Environment that is able to load an interactive, 3D simulation of the F/A-18 cockpit and a lesson script that controls the text and audio displayed and also controls the 3D simulation. Each node in the activity tree for those units loads the same resource but assigns a different primary objective and has a unique set of parameters that describe which lesson script to load providing streamlined testing of the SCORM API calls. In the interest of keeping the tight training time schedule, all user interactivity with the 3D cockpit was disabled. Although we have demonstrated that this provides less training effectiveness, it ensured that every student spent the same amount time in the lesson. Each node presented several learning objectives that would need to have individual remediation, yet the nodes were not broken into multiple SCOs since that would require the 3D cockpit to be unloaded and reloaded between each SCO transition. SCORM Navigation was used in every SCO, providing for automatic navigation when each SCO is complete also ensuring that the training time schedule was followed.

Simulation Interface and Remediation

Remediation is accomplished by reading a file generated by the simulation and setting the ***score.scaled***, ***score.raw***, ***score.min***, ***score.max***, ***success_status***, ***completion_status***, and description of several learning objectives. Although remediation involves the same resource and 3D simulation, additional information is used to provide streamlined, specific remediation. This can be accomplished by having the SCO examine the ***success_status*** of the learning objectives as set by the performance evaluation in the simulator. It can also be accomplished by having specific remediation nodes that have additional parameters and are skipped if mastered. Neither remediation method is shown in the activity tree.

Brandt Dargue
The Boeing Company,
St. Louis, Missouri
Brandt.W.Dargue@Boeing.com

Meeting Armor Requirements with SCORM Reuse within the Scope of Content Repositories

SCORM Principles Meet Real Requirements

The US Army Armor School has real, immediate requirements to reaggregate content, which can be met using existing SCORM standards and the projection of developing SCORM principles. These principles can be grouped as follows:

- SCORM proper: packaging and metadata standards to support reuse
- CORDRA: processes for searching and finding courseware across repositories
- Repository Communication & Delivery: processes for SCORM/CORDRA conformant LMSs and repositories to communicate, share, recombine, and deliver recombined content to students

Of these, the first exists in its 2004 version. The second is under development. The third describes the functional domain that will allow SCORM and CORDRA to reach their full vision. Although the ADL is placing this development following the CORDRA search development, the immediate Armor School needs do not require this global search ability. What the Armor School does require is the ability to reuse SCOs by reaggregating them for multiple user audiences.

The development of a prototype, proof-of-principle solution for the Repository Communication & Delivery domain will not only meet the Armor School requirements, it will also serve as an early model that can be used by the Army, ADL, and other SCORM adopters.

Requirements and Reaggregation Solution Model

Requirement: Create and update DL content that is delivered to different military occupational specialty (MOS) audiences containing a significant overlap of “core common” tasks and a large number of tasks which are almost identical except that the different audiences will execute them with different equipment (“core parallel”).

Requirement: Retain already developed, effective DL courseware for courses which will be discontinued in their current form, but whose content will still remain a training requirement to be taught in other venues not yet determined.

Requirement: Leverage the investment in DL content by allowing key portions of it to be reused in the instructor-led mode by resident and DL instructors and leaders in the field.

Reaggregation Solution Model: Create all future courseware not as a stand-alone SCORM-conformant course, but as “a bucket of SCOs” where each task is packaged as a SCO and designed to be intelligible individually or packaged as part of one of several pre-identified modes. SCOs will include their own help files, so will not be dependent on “course overview” SCOs. These SCOs will use both SCORM 1.2 and SCORM 2004 models.

Reaggregation Solution Model: As resources permit, some pre-identified SCOs may be developed at the sub-task level with core components and MOS-specific components to be deployed as MOS-specific tasks built up from common SCORM-conformant building blocks. We intend to build on the work of Dr. Adelaide Cherry, who has developed the concept of the “encapsulating module.”

Reaggregation Solution Model: The above SCOs will be managed in and delivered from within a prototype SCO Repository. This repository will follow the CORDRA concept of a number of distributed repositories whose content can be searched, accessed, configured, and delivered by distanced management tools to distanced users, all via a variety of SCORM-conformant technologies, not requiring a specific common proprietary platform.

This model will not merely revolutionize the way we design courseware, but it will also revolutionize the way that resident training is designed at the Armor School.

Courseware Affected

Courses which require the reaggregagation of core common and "core parallel" tasks include Advanced Noncommissioned Officers Course, Basic Noncommissioned Officers Course, Cavalry Scout, and Tank Crewman. Courses which will benefit from having their content rejuvenated by being made available for follow-on reaggregation include Scout Leader's Course, Advanced Noncommissioned Officers Course, M1A2 SEP Tank Commander Certification, and M1A1 Master Gunner.

As resources permit, we will explore the use of advanced encapsulating modules to deliver sub-tasks "scaffolded" for multiple skill levels and layered for multiple echelons (different tasks simultaneously). We are particularly interested in partners or ADL sponsorship to more deeply explore and document this aspect.

Repository Model

Our model envisions an instructor or content developer working within a SCORM/CORDRA-conformant LMS who wishes to create a curriculum by assembling a series of SCOs. This could be a schoolhouse content developer who needs to create a course that teaches a variety of tasks using existing SCOs, it could be a DL instructor who needs to develop a lesson for student remediation or a weekend synchronous collaborative session, or it could be an NCO assembling some exercises to lead his soldiers through during "sergeant's time."

First, he or she uses the LMS to search for a SCO addressing a specific task or topic. The SCO is found by identifying elements of its metadata, and it is located on a proponent's repository server. Instead of copying the SCO from the repository server to the local LMS server, a reference pointer to the SCO is made in the LMS's database. After finding several SCOs, including instructional blocks, exercise blocks, and test/evaluation blocks, the instructor/developer uses the LMS's tools to organize the SCOs into a SCORM-conformant learning experience by saving them as a Content Organization Template. This template is simply the form of an IMS manifest existing in an LMS database, either imported from a SCORM PIF, or made in the LMS, and able to be exported as the manifest of a new SCORM package if required.

When the instructor or content developer has completed assembling the courseware, it is represented to the student as a courseware table of contents. The links on the table of contents screen may point to a SCO housed locally on the LMS or to a SCO housed in a remote repository server. A user clicking on the remote SCO's launch link will launch the SCO from the repository server. The user's interaction data with the SCO is stored by the SCO utilizing the SCORM API provided by the repository server, which then relays the interaction information to the LMS for permanent storage.

Developers of the SCORM anticipated the launching of remote SCOs by allowing fully qualified URLs (i.e. href="http://www.myrepository.com/mysco.htm") to be used for a SCO's launch link within a package manifest. However, they left it up to SCORM adopters and developers to figure how to make launching a remote SCO actually work in future rounds of SCORM development, now SCORM/CORDRA development. These issues deal with built-in browser security restrictions that do not allow client-side code that is launched from one server to manipulate or access client-side code that is launched from another server, especially across browser frames. For example, if a user clicks a launch link on a page provided by a LMS and launches a SCO residing on a remote repository server, browser security restrictions do not allow that SCO to access the LMS provided SCORM API. As well, without special security exceptions, the remote SCO cannot access a server-side "catcher" script residing on the LMS server. There are a few approaches to addressing this, and this is a crucial technical step that must be addressed before the SCORM/CORDRA model can eventually work as envisioned. This Armor School-Industry model will provide a prototype solution for this, which we will invite other SCORM adopters to consider.

Summary

The key pieces of the Armor School/Industry model which will be developed are:

1. Courseware consisting of multiple task-level SCOs, designed to be recombined in several pre-defined configurations.

2. SCORM-conformant tools to allow instructors and content developers to find SCOs, assemble them into courseware by organizing, reorganizing, and saving content organization templates, and "publishing" the resulting course and making it available to students.

Combining the CORDRA concept with the idea of remotely launching SCO's from proponent content repositories provides a solution that will allow proponent organizations to maintain positive control over the content they develop and distribute while at the same time eliminating SCO version control issues.

The basic technology already exists to allow LMS's to remotely search for and launch SCOs from an unlimited number of proponent content repositories, and merely needs to be developed into SCORM/CORDRA conformant components. By looking past the developing CORDRA work to the mechanisms by which content repository servers relay user interaction information back to the LMS, we will not only meet current Armor School requirements, but smooth the way for this inevitable development, and allow wider Army and ADL audiences to begin refining this solution to their immediate needs.

Nathan Ashlock
Alion Science and Technology

Wanda Majors
U.S. Army Armor Center, Ft. Knox

David Nilsen
Alion Science and Technology
david.nilsen@knox.army.mil

George Paschetto
Alion Science and Technology

Building Intelligent Learning Managment Systems to mimic the Teacher Student relationship

Abstract

This paper investigates strategies for building intelligent learning management systems, which can mimic the teacher student relationship and proposes extensions to the SCORM version 1.3.1 to enable these strategies. The analysis is partly based on a number of research papers published on the analysis of learning in a classroom environment against individual tutoring. The proposed extensions would enable support for Learning Management Systems (LMS) initiating conversation with the learner. This proposal involves the addition of a number of data elements to the SCORM data model.

Introduction

The Department of Defence(DoD) and the White House Office of Science and Technology Policy (OSTP) launched the Advanced Distributed Learning (ADL) initiative in November 1997 [1]. The mission of the ADL is to provide access to the highest quality education and training, tailored to individual needs, delivered cost effectively anytime anywhere. Since then, the ADL have developed the Sharable Content Object Reference Model (SCORM) to build very descriptive learning objects. The major vendors of Learning Management Systems (LMS), all support SCORM learning objects, which implies that SCORM compliant content is fully deliverable without modification on many LMS's. The SCORM model is broken up into three different sections: Content Aggregation Model (CAM) [3], Sequencing and Navigation (SN) model [2] and the SCORM Run Time Environment (RTE) [4]. The CAM describes the learning content using XML tags. Each SCORM compliant learning object has the following fields to describe learning content:

```
<manifest>
<organizations>
<resources>
<(sub)Manifest(s)>
```

These fields have several metadata elements to fully encode the attributes. The SCORM RTE describes the LMS's requirements for interacting with the learning objects. The SN defines the various methods of delivering courses to clients.

Extending the Manifest

A number of studies were carried out on teaching environments and the effects on the participating students [5] and have shown in that in a typical classroom environment on average every student asks about 0.1 questions every hour. The speed with which different students can progress through instruction varies by factors of three to seven [6]. With individual tutoring, students may ask or answer on average 120 questions per hour [6]. The achievement of individually tutored students may exceed that of classroom students by as much as two standard deviations – an improvement which is equivalent to raising the performance of 50th percentile students to that of 98th percentile students [7].

Clearly, individual tutoring is the way forward for education, however, it would not be feasible to have the same number of teachers as students. Building learning objects is the first step in accomplishing automated one-to-one tutoring environment. An analogy can be drawn from the state of the art as a visually and hearing impaired teacher (LMS) with limited ability to turn a page, watch the time and remember data. To change our teacher into an intelligent successfull teacher the current version of SCORM and the current API available has to be changed. With the current state of SCORM, there is no interaction initiated by the LMS, which takes advantage of the vast metadata that is available from the learning object. Extending the manifest to include a personal element with various attributes, such as, name, age, address, ability, learning preference and associated metadata would enable the LMS to initiate conversation with the client on a 'personal' level.

```
<manifest>
<organizations>
<resources>
<(sub)Manifest(s)>
<Personal>
```

The `<Personal>` field would initially be empty in every learning object. Once the learning experience has commenced the personal field would be dynamically filled from locally stored information.

The current API available to the learning object vendors offers only a limited number of functions. For example, *Initialize(parameter)* and *Terminate(parameter)* session methods for initializing and ending communication with the LMS; Data Transfer methods *GetValue()*, *SetValue()* and *Commit()*; Support Methods *GetLastError()*, *GetErrorString()* and *GetDiagnostic()*. As the LMS monitors and tracks the users learning experience, manages learning content, learner progress and learner interactions, extending the current API available the LMS could offer more personalised help if the client is having difficulties with the learning objects. The "teacher" is effectively still 'visually impaired' and would need to be prompted for help, however, the teacher could be programmed for a number of special cases such as the start of communication or if a learning object 'times out'.

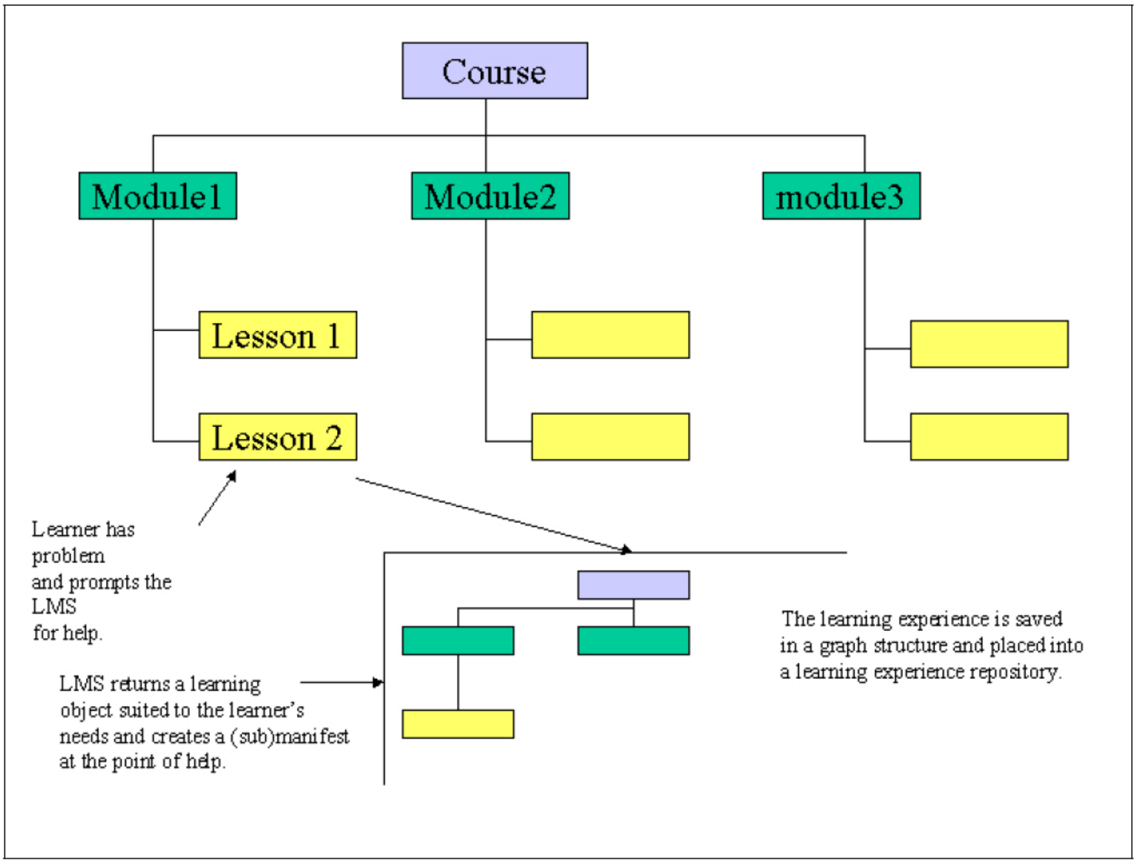


Figure 1: Activity Diagram incorporating new LMS

Effects of adding help on the activity tree

An Activity Tree is a general term that represents an instance of hierarchical learning activities and the corresponding sequencing information for the interoperable application of specified sequencing behaviours. The activity tree is not a static structure and is free to change dynamically with the needs of the learning object vender. As can be seen from the tree shown in Figure 1, the root of the activity tree represents a course, which has tree modules and each of the modules have two lessons. The learner has difficulties with lesson two in module one and prompts the LMS for help. The LMS uses the knowledge of all the meta-data about the learner and the learning object to search for a new learning object specific to the learners needs. The LMS returns the new learning object represented as a sub-manifest in the activity tree. Once the learning experience has concluded the entire learning experience is represented as a graph structure and is saved in a learning experience repository. Using structure matching techniques and searching for isomorphisms between graphs, the LMS could call on past experiences when dealing with a client.

Conclusion

In conclusion, extending the manifest and building learning experience repositories is the next step in achieving an automated one-to-one tutoring experience. The LMS would have the same ability as an intelligent, experienced visually impaired teacher, being able to make suggestions based on a vast amount of meta-data available for both the learning object and the client and also based on past experiences with similar clients. Future models could include a visual aid for the LMS in the form of a camera, which would enable the LMS to monitor the client’s body language. Neural Networks could be trained to recognize when a client is stressed, confused, frustrated or simply not relaxed, and automatically suggest assistance for the current learning experience.

References

[1] ADL (july 22, 2004). *Sharable Content Object Reference Model 2nd Edition Overview*, Available at ADLNet.org

[2] ADL(july 22, 1984). *SCORM Sequencing and Navigation Model Version 1.3.1*, Available at ADLNet.org.

[3] ADL(july 22, 1984). *SCORM Content Aggregation Model Version 1.3.1*, Available at ADLNet.org.

[4] ADL(july 22, 1984). *SCORM Run Time Environment Version 1.3.1*, Available at ADLNet. org.

[5] Gettinger M (1984). *Individual differences in time needed for learning: A review of the literature*, Educational Psychologist,19, 15-29.

[6] Graesser, A.C.,& Perso, N.K.(1994). *Question asking during tutoring*., American Research Journal, 31, 104-137.

[7] Bloom, B,S(1984) *The two sigma problem: The search for methods of group instruction as effective as one-to-one tutoring*, Educational Researcher,13, 4-16.

Keith Maycock
Department of Computer Science,
National University of Ireland,
Maynooth, Co. Kildare, Ireland

John G. Keating
Department of Computer Science,
National University of Ireland,
Maynooth, Co. Kildare, Ireland
john.keating@may.ie

The different ways to implement the RTE and SN

Introduction

The designer of the systems based on SCORM, has problems when they have to design the SN and the RTE. Conceptually, SN and RTE work together; SN sends queries about data model to RTE asking for the status of the SCOs and depending of the type of the status, SN does one or other sequence. This interdependency between SN and RTE involves simultaneity at run-time.

Another factor that we have to take into account is that the programmers that code RTE and SN, must work in a collaborative way, at the same time and in the same modules. This type of job has a lot of difficulties because it requires the usage of resources that cannot be available at the same time. Another possibility is that only one programmer codes the SN and the RTE, but this possibility is very complicated because it is just too much for one programmer.

In this paper we propose two ideas of design in which the RTE and the SN are implemented as separate modules. Logically, there is a interdependency between them that we keep.

Our proposals

We can think of the implementation of the communication between the RTE and SN with an automaton with states whose number depends on the activity tree and the sequence rules.

The idea is to insert some element or some module that let us do the separation. In order to get this goal, we have two ways:

- We can do this in a dynamical way, that is, we can transform the activity tree and the sequence into a graph plus its states plus data model. The module of SN generates a class with the necessary code for the graph plus its states plus data model and the module of RTE uses the Computational Reflective [1] to execute the generated class by the SN at run time. The advantage of this method is that the process is dynamic and the changes in the graph are caught by the RTE module at run-time.

  We add to the architecture a repository of LOs packed in SCORM. These LOs can be located anywhere and any system compatible with SCORM must use them without problems.

  The most appropriated language for the described way is Phyton [2]. Phyton is a reflective language. Java [3] can also be used because it has implemented classes to do this, but the implementation with Java requires longer time and more lines of code than the implementation with Phyton.

  Another factor is the cycles of execution. We think that the systems would improve in time of execution because we save cycles Currently, there is an exchange of messages with the implementation of SCORM; however this exchange of messages does not exist with the described above.

- Our second idea of implementation consists of the creation of a Java library. That library would provide us with the SN functionality by means of the implementation of its rules. RTE would use the library to access to the standard sequencing. That makes it possible to identify the next SCO to be launched at run-time by using its methods.

  This static implementation of the Sequencing and Navigation, is determined by the convenience of a real separation of the different parts of the standard, being at run-time when the SN rules are interpreted, without this SN being executed in parallel with the RTE.

  In the definition of the standard, the division of this one in three parts exists, but in spite of that, it is very difficult to distinguish clearly within the LMS the own functions of the SN and those of the RTE. That causes that it is complicated that a work group develops a system in which each member of the group is in charge of a module, since there are many functions that overlap between these two parts and the own LMS.

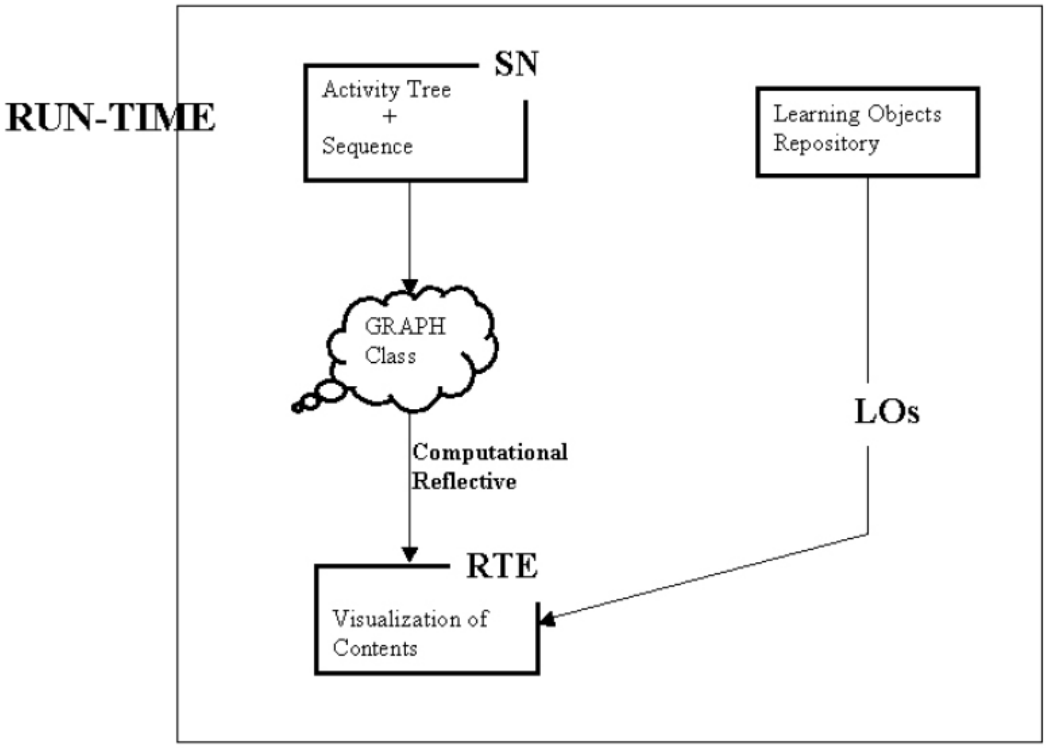


Fig 1. System with the dynamic implementation

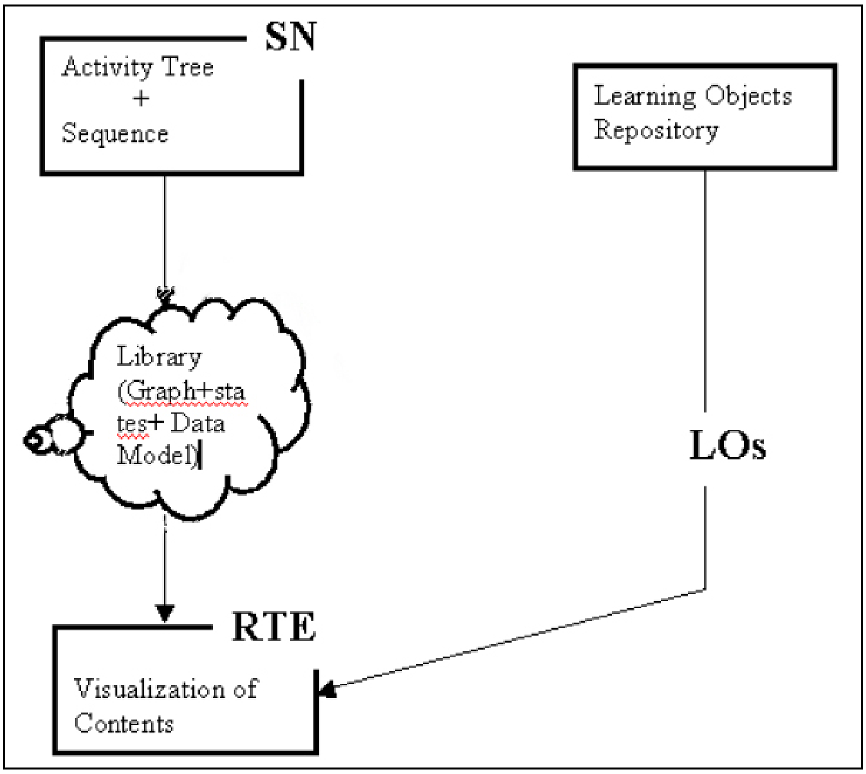


Fig 2: System with the static implementation

The using of a library allows us to encapsulate the SN functionality. Moreover, it makes it possible to work with this part of the standard in an asynchronous way, both in design and implementation time, as in run-time. In this way, the development of a totally modular SCORM e-learning system is facilitated. The input of the library to make this process would be the Activity Tree, the present state and the RTE data model, and the output would be the next learning object identified to be launched.

The Java library would provide an only method to us, which would be the one in charge of the interpretation of the own rules of the SN. In order to evaluate these rules the SN uses the RTE Data Model, the results of the user's interaction, and the own sequecing strategy that forms the Activity Tree.The selection of Java language in order to implement the library, is based mainly on its suitability to incorporate the library in systems implemented in this language.

That, together with the ease of use of the language and the own features of the same one, like the portability, turn Java into a language adapted to our objective.

Conclusions

The final conclusions of this article are:

- We propose an implementation of Sequencing and Navigation independent of the rest of the standard to distinguish clearly within the LMS the different functionality of both, the SN and the RTE. This separation can favour the reduction of the cycles of execution.

- The library provide us the encapsulation of the SN, in addition it allows us to work with it in an asynchronous way.

- The use of Computational Reflective lets us separate the semantic of RTE and SN, but at the same time it respects the orientation of the implementation of the SCORM.

- The use of the Java language facilitates the incorporation of the library in programs developed with this language, and it allows us to make use of the features of the language.

References

1. Francisco Ortin, Benjamin Lopez, J. Baltasar García Perez-Schofield.Separating Adaptable Persistence Attributes through Computational Reflection. IEEE Software, Volume 21, Issue 6. November 2004.

2. Web Site of Phyton. http://www.python.org/

3. Web Site of Sun. http://java.sun.com/

Mª del Puerto Paule Ruiz
Dpto. Computer Science- University of Oviedo (Spain)
paule@uniovi.es

Sandra Guerrero Blanco
Dpto. Computer Science- University of Oviedo (Spain)
sgd@telecable.es

Sagrario García Santos
Dpto. Computer Science- University of Oviedo (Spain)
tioda6@yahoo.es

Sequencing and Navigation in Simulation-based Training

Distributed learning systems are now beginning to explore the edges of their currently constrained world. Recent efforts have begun to investigate how inter-operating distributed e-learning applications can include formats previously excluded by their proprietary development and delivery environments and by the limitations of their architecture. ADL, focused first on an effort to overcome the limitation of proprietary formats, resulting in the introduction of SCORM and (soon) CORDRA. However, SCORM conformance seemed like a stifling limitation to instructional developers accustomed to using more powerful instructional designs than SCORM 1.2 supports. However, given the new structures available within the SCORM 2004 specification as well as new architectures combining SCORM with other platforms, new instructional possibilities are wide open. IAI has been exploring these capabilities, especially with respect to combining instruction conforming to SCORM with simulation conforming to the HLA (High Level Architecture) standard for simulation. Some details of this work can be found in Haynes, Marshall, Manikonda and Maloor (2004).

SCORM 2004 provides huge improvements in the instructional architectures that can be implemented, due to implementation of new sequencing and navigation rules. SCORM sequencing and navigation (SSN) can now be used to individualize the instructional experience, thereby making it more efficient; motivating; and effective. Also, learners can experience greater control over their personal learning experience by selecting options best suited to their preferences for detail, learning modality, or level of interaction with other learners, for example. Other stakeholders (employers, schools, or the military, for example) can also benefit from more complex learning environments that can facilitate better quality training and better assessment information about learners performance.

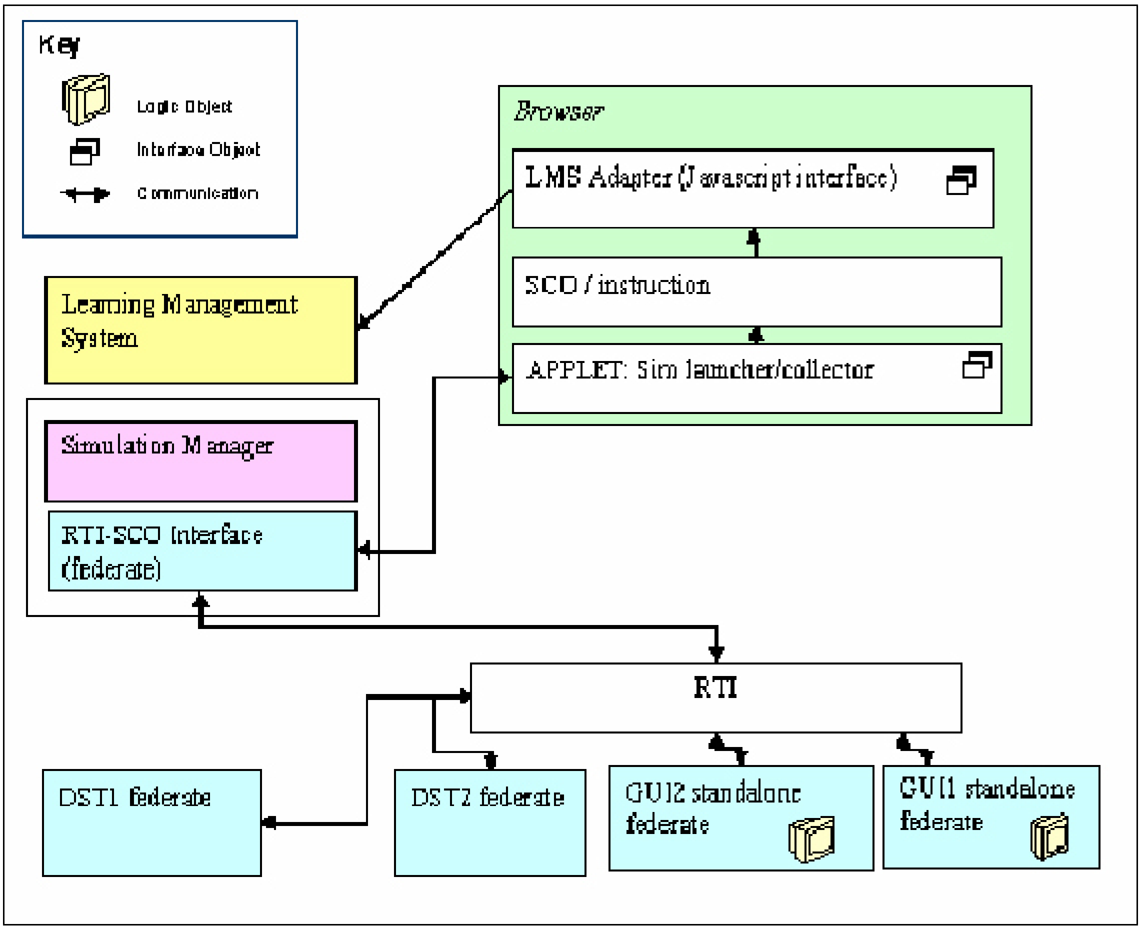


Figure 1: SITA Architecture

SITA (Simulation-based Intelligent Training and Assessment) is a prototype system (SITA was sponsored by the Joint ADL Co-lab in Orlando, under BAA) designed to provide a model of combining simulation-based training (conforming to HLA) with instruction conforming to SCORM. SITA includes a variety of sequences of instructional events that can be used to explore the 'edges of the envelope' in terms of SSN in SCORM. SITA includes instruction in declarative knowledge (learning 'what' and 'why'), procedural knowledge (learning 'how,' 'when' and 'where') and applied knowledge (using the prior knowledge to perform one or more tasks). While the specific instantiation of this architecture uses an air traffic control task, of course, any distributed simulation conforming to HLA could be substituted.

The SITA instructional architecture includes SCO's for pre-assessment, didactic instruction, interactive instruction, individual skill practice, team skill practice, knowledge assessment, individual performance assessment and team performance assessment. A SCO launches the simulation, , which then communicates with the simulation federates and the LMS is achieved via the RTI-SCO Interface, the Simulation launcher/collector applet and the LMS Adapter.. Figure 1 illustrates this technical architecture.

While the one-hour course is artificially complex to demonstrate the full scope of possible instructional architectures, it is useful as scenario for considering how and where simulation can be used effectively to achieve various types of outcomes. Currently in SITA, simulation is used for skill practice and performance assessment, both for individuals and teams. Table 1 describes the instructional activities in SITA and the role of SSN in the instructional architecture.

Instructional Activity	SCORM Sequencing and Navigation
A *Pre-test SCO* gathers information about he trainee's current state of knowledge and/or performance.	SSN can be used to make the assessment adaptive.
Based on the pre-test, a *Didactic Instruction SCO* introduces the trainee to the new principles of air traffic control called 'collaborative regional flow control, (CRFC)' by description and examples of types of constraints that can be employed to create different effects on airspace efficiency.	SSN is used to direct individual learners to begin their learning experience with different SCOs, depending upon the evidence of their pre-existing knowledge, derived from the pre-test.
With the *Interactive Instruction SCO*, the trainee then learns more specifically about each of the constraints and how they are represented in the CRFC. The SCO uses the simulation's GUI to manipulate each constraint separately, with instruction in what, when, and how to perform the manipulation	SSN is used to direct the trainee's learning experience to include those SCO's dealing with constraints not mastered (as evidenced by the pre-test). Performance on each SCO is evaluated separately (within the SCO), which is only exited upon achieving a level of proficiency.
Following completion of the interactive SCO's AND attainment of threshold scores, The CRFC-DST simulation (see Satapathy, 2002 for detail) is initialized with a scenario selected for the trainee's proficiency level. The trainee practices over a series of until s/he attains a criterion efficiency score. Individual simulation may be followed by team simulation, following the same logic.	SSN is used to permit navigation among different scenarios depending on outcomes in previous simulations. These can be individual simulations, followed by team simulations covering larger regions of airspace.
Assessment SCO's follow instruction and include of two types: performance assessment using simulation, and knowledge assessment using conventional assessment items. Assessment data	SSN is used to make knowledge assessment adaptive, and to identify appropriate scenario(s) for the performance assessment such that what the trainee demonstrates s/he *knows*, is then assessed in

is reported to the LMS.	simulation to determine whether s/he can *use* that knowledge in task performance. SSN can also provide remediation.

Table 1: SCO's and SSN in SITA

The uses of SSN in SITA are but the beginning of exploration into only one of the many possible instructional architectures that are now supported in SCORM-conforming learning environments. Future R&D efforts will undoubtedly introduce a variety of new instructional architectures into the SCORM distributed learning world/

References

1. Satapathy, G., Manikonda, V., Robinson, J. and Farley, T. (2002). En-Route Sector Metering using a Game-Theoretic Approach. AAAI Game Theory and Decision Theory Workshop.

2. Haynes, J., Marshall, S., Manikonda, V., and Maloor, P. (2004) Enriching ADL: Integrating HLA Simulation and SCORM Instruction using SITA (**S**imulation-based **I**ntelligent **T**raining and **A**ssessment). Interservices/Interindustry Training Education and Simulation Conference. Orlando. December.

Jacqueline A. Haynes
Intelligent Automation, Inc.
jhaynes@iai.com

Preetam Maloor
Intelligent Automation, Inc.

SCORM Compliant Web-Based Content Management Tool for Extending Viable Information to Rural Community in an Affordable Way

Abstract

In the present agricultural scenario, there are two levels at which information is handled. At one level are the researchers and scientists who generate information and at the other, are the rural people who apply it. The most critical problem pertaining to this information flow (intra/inter level) is the availability of comprehensible information and that of information interoperability and reusability. The information developed by the scientific community is abstruse and incomprehensible for the rural people and this introduces a sense of divide between the two. The actors involved in this digital information flow are also not regular computer users (in some cases, they are not even literates and are non-conversant in English, in which most of the literature is available), which makes it even harder to overcome. This paper proposes a web-based multilinguistic tool for extending viable and localized information to rural community in an affordable way.

This paper is based on the research work done at ICRISAT [2] (International Crop Research Institute for Semi-Arid Tropics, Hyderabad) under its VASAT [3], Virtual Academy for Semi- Arid Tropics, initiative.

I. INTRODUCTION

There presently is an information divide between the scientific community that includes the people who do all research and actually are information creators or generators, and the agrarian communities who apply this information in the fields. It is for bridging this particular divide that the farm extension workers come in. These are the people (can be literate or semi-literate farmer/person) who have required amount of agriculture knowledge and the minimal required computer skill (like booting and browsing etc). The content usually created is full of scientific terms, definitions and analysis. But what a farmer requires is something that, in a very simple and localized manner, gives him the required knowledge.

II. WEB-BASED CONTENT MANAGEMENT TOOL

This section describes the features and functionalities of the tool, in context of the requirements mentioned above.

A. Information Workflow

The information flow requires an interface between the high-level knowledge workers and extension workers. As is always the case, the information generated by the researchers is abstruse and incomprehensible from farmer's point of view. This information in the form of SCORM version1.2 [1] -compliant objects is stored in the central repository, which acts as the database. The farm extension workers can then access collection of objects, customize them (according to local preferences) using our tool and pass it over to farm families, who then acquire the required knowledge. Moreover, the content generated at both levels is sharable, reusable and interoperable (only for SCORM compliant objects). The functionalities in our system are divided into three categories: localization of information, generation of SCORM compliant objects, and defining pedagogy (creation of instruction).

B. Localization of Information

Farm families, through extension workers, can access the necessary nformation. The information thus obtained can be customized using our tool. The customizing tool gives user the power of extracting content from a variety of documents that are not only available in our system but elsewhere, with the help of drag-drop multilingual editor. Any content in any local language can be added. New content generated, can be locally used and can also be uploaded onto the repository (so that it can be shared and reused). The unique property of this tool is that it gives the users power of localizing information, in a very simple (just drag-drop) way, with minimum required skills. The user only deals with simple text and images, without having the knowledge of any of the scripting or markup languages. Most of the processing required is done on the server. All that is required at the client side is the internet connection and the browser. This functionality fulfills the need of contextualization “in an affordable way”, which is an imperative issue.

C. Generation of SCORM compliant objects

The Content Management Tool also gives the users option of creating SCORM version 1.2 [1] compliant objects. These objects can be html pages, text files, animations, images or a group of these objects. Here the user defines the level of granularity. The lowest level reusable objects are called assets and group of assets make up an SCO (Sharable Content object). These SCOs are interoperable and reusable and mainly constitutes the flow of information amongst the peers. For this, user has to fill the required tags [1],[3] ranging from minimally required to all, as defined by the ADL community. This ensures the sharability and reusability of information in the horizontal direction. Here also the interface provided is built keeping in mind that the users are not regular computer users. The implementation complexity is hidden from the users of the system, thus they need not have any technical background (like having knowledge of XML or other software skills).

D. Defining Pedagogy

This functionality gives users the option of defining an instruction (course/module/tutorial) online. The instruction can be created with the help of objects available in the repository or by new user defined objects. The Tool also has an inbuilt player, which can be used for playing the instruction. User can thus, online generate the instruction and can play it as a course. The SCORM compliant course/module created through the system can be downloaded and played on any other SCORM compliant LMS or player, without any changes. This fulfills our requirement of interoperability and reusability at the instructional level.

III. CONCLUSIONS

At present, the most critical problem concerning information flow in agriculture scenario is that of availability of comprehensible (localized) information and of content compatibility (sharability, reusability and interoperability). This paper presents a Web-based Multilingual Content Management Tool specifically designed for keeping these two problems and the target users (farm ext. workers and agricultural scientists) at focus. Since, the content generated by the system is in accordance with SCORM specifications [1] as well as sequencing and navigation capabilities [7], the content can run on different SCORM Compliant Learning Management Systems. Presently the system has minimal functions, catering mainly to needs of rural and the agricultural domain. But the system is incorporated and hence can be made usable for other domains too.

REFERENCES

[1] ADL Technical Team, “Sharable Content Object Reference Model (SCORM) Version 1.2”, http://www.adlnet.org/index.cfm

[2] ICRISAT, International Crop Research Institute for Semi-Arid Tropic,http://www.icrisat.org.

[3] VASAT, Virtual Academy for Semi-Arid Tropic, http://vasat.org

[4] ICRISAT, “Legume Pod Borer”, www.vusat.org/learning/agri/Pigeonpea_htm/est/Attack%20Flower%20&%20%20Buds/legumepodborer.htm

[5] IMS Global Learning Consortium, “Developing Tools and Content” http://www.imsglobal.org/develop.cfm

[6] Jin-Tan David, Yang Chun-Yen Tsai, “An Implementation of SCORMcompliant Learning Content Management System - Content Repository Management System” http://csdl.computer.org/comp/proceedings/icalt/2003/1967/00/19670453.pdf

[7] ADL Technical Team, SCORM Sequencing and Navigation, Version 1.3.1, July 2004

Dr. V. Balaji
International Crops Research Institute for the Semi-Arid Tropics, Hyderabad, India
v.balaji@cgiar.org

Anant Arora
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat, India

Anuj Jain
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat, India

Gaurav Goyal
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat, India

Anant Arora
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat, India

Gauravi Dubey
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat, India

Saurabh Singh
Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, Gujarat, India

SCORM Learning Sequence Modeling with Petri Nets in Cooperative Learning

Abstract

Learning sequence showed the learner progress and learning status. It is useful to collect and model the learning sequence information to discover the learning disability or analysis learning efficiency. In cooperative learning, learners learned in groups, the learning sequence provided instructor to realize the students' learning progress. In order to model SCORM sequence, we use petri net in this paper.

1. Introduction

SCORM [1] establishes a technical foundation of e-learning via standardization. It is focused on defining reusable learning objects, developing new content models, developing learner assessment models, and creating new models for sequencing content. The three main portions are the Content Aggregation Model (CAM), the Run-Time Environment (RTE) and the Sequencing and Navigation (SN). CAM defines the responsibilities and requirements for building content aggregations (e.g., course, lessons, modules, etc.). CAM contains information on creating content packages, applying metadata to the components in the content package and applying sequencing and navigation details in the context of a content package. The SCORM Run-Time Environment communicates with its underlying Learning Management System (LMS) that is both machine independent and operating system independent. Sequencing and navigation define the learning sequencing behavior and the learner navigation.

2. Mapping Petri Nets onto SCORM Sequences

We adopt the traditional Petri Net [2], with refinement and additions. Our DCPN model is defined as a directed graph PN = (P, T, F, W, M0); where

- P = {p1, p2,…, pm} U {cp1, cp2,…,cpn} is a finite set of places that consist of two subset, ordinary place subset (circle) and control place subset (double circle), respectively. This is different from an ordinary Petri Net where there is only one type of place.

- T = {t1, t2,…, tK} is a finite set of transitions that are drawn using bars.

- F : {P * T} U {T * P} is a finite set of arcs representing the flow relations .

- W: F® I is a weight function, I = {1, 2 , ... } representing set of nonnegative integers. The function W can be extended to incorporate a weight k. Therefore, $W_k : F \xrightarrow{k} I$ represents a set of k parallel arcs with the same source and destination.

- M0: P $\rightarrow$ { IC1, IC2, IC3,…} is the initial marking (represented using dots), that assigns color tokens to each place in the net, IC={n_1, n_2, n_3, …}, is a nonnegative set of integers representing numbers of color tokens.

The DCPN model is able to accomplish the SCORM routing constructs easily, including flow and choice. Flow displays a straight linear learning path. It ensures that the learner progresses through the content aggregation in a pre-determined order. Choice allows the learner to jump and select other lessons in an arbitrary order. In Figure 1, and Figure 2, we illustrate these routing constructs of flow and choice, respectively. In Figure 2, the learner can choose "Lesson1" or "Lesson 2" by firing transition 1 or transition 2, after he has completed "Introduction".

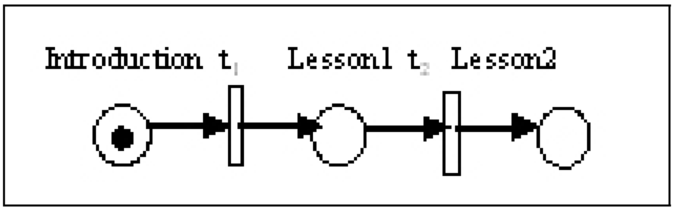


Figure 1. Flow Construct.

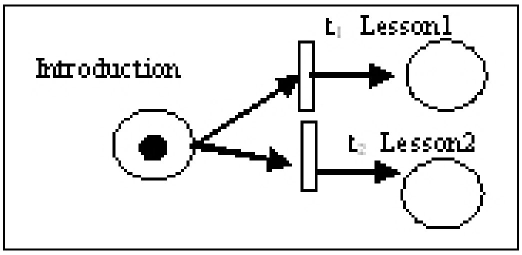


Figure 2. Choice Construct.

We identify other SCORM constructs as follows. Skip is the action when the learner wants to ignore some learning material temporarily in a linear structure. We add the control place with an initial token and use the choice structure to allow a learner to either go through the content in a linear manner, or skip one of the learning materials. In Figure 3, the double circle represents the control place. If transition t4 is fired then the learner will skip “Lesson2” and learn “Lesson3” immediately. The reason of the input arc of t4 to be two-way is to preserve the learning opportunity of skipped learning material (i.e., Lesson2). After a learner learned Lesson 1, Lesson 3 and Lesson 4, the learner has an opportunity to study the skipped Lesson 2.

Limit Condition describes a condition under which an activity is not allowed to deliver. Since SCORM does not requires the evaluation of any time-based limit conditions, our model focuses on the maximum number of attempts for the activity. By the number of token in the control place, we limit the times that the learning material can be read. In Figure 4, because each number of tokens in input places (cp1 and Lesson1) is equal to (M(Lesson1)=1) and greater than (M(cp1)=2) each weight of the directed arcs connecting the places to transition t1 simultaneously, transition t1 is said to be enabled. If we fire t1, the number of tokens in input places decrease (M(Lesson1)=0, M(cp1)=1) and the number of token in output place increases (M(Lesson2)=1). In other words, we utilize the number of tokens in a control place to limit the number of times the transition can be fired.

Suspend describes the situation where the learner needs to terminate learning activities temporarily, in which case the LMS should record the break point in order to restart learning. In our model, one type of arc can represent Suspend by labeling the arc with “s”. This can distinguish whether the source place of an outgoing arc terminates or is temporarily paused when the token leaves that place.

The above concepts can be mapped onto the features of Petri Nets to construct a framework. An entire activity tree may be replaced by a single place or transition for modeling at a more abstract level, or places and transitions may be replaced by subnets to provide more detailed modeling. In our model, we make use of the token number and the weight of an arc to determine whether the Roll Up condition is satisfied.

We illustrate using the Exam model in Figure 5. When transition t1 is fired, the token is moved from Question1 to cp1 to judge whether the answer is correct. If so, the net fires t5 and moves a token from cp1 to cpcorrect; otherwise, the net fires t4 and move token from cp1 to cperror. After all questions are finished, the number of tokens in $cp_{correct}$ and cp_{error} represents the total number of correct and error answers respectively. And, we use the weight of the arc connecting to t10 to set the filter of whether the exam is passed. In order to calculate the sum of the correct questions, we didn’t divide the module into individual question. In the example, we simulate the real operation of an exam.

If the learner passes an exam (pretest or posttest), the module's learning objective has been satisfied and the module becomes disabled – the learner is unable to select any of its lessons. In Figure 6, we draw the Abstract Module 1 in the SCORM example. We can observe that the arc connects from Ppretest to two transitions t1 and t2. Ppretest is the abstract place that can contain Question1, Question2, and Question3 from Figure 5, and thus has a hierarchical relationship with Figure 5.

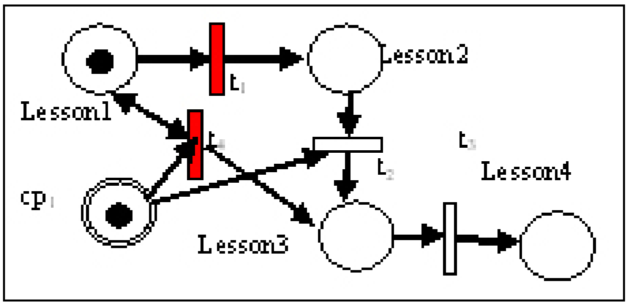


Figure 3. Skip Construct

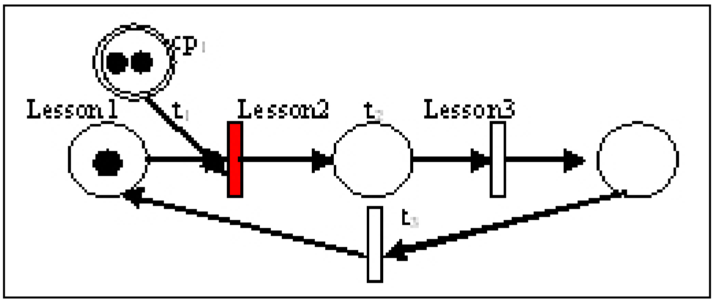


Figure 4. Limit condition (attempt=2) Construct

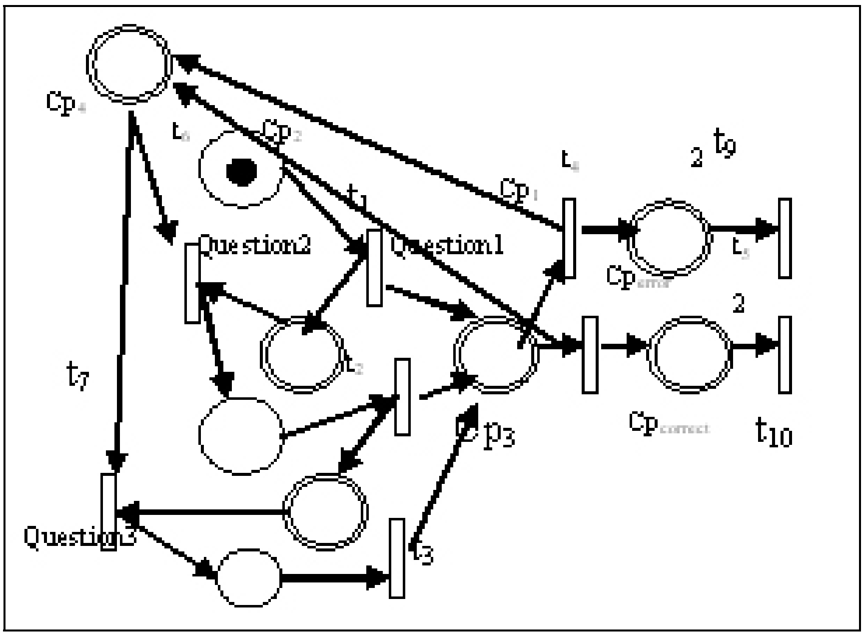


Figure 5. Exam subnet

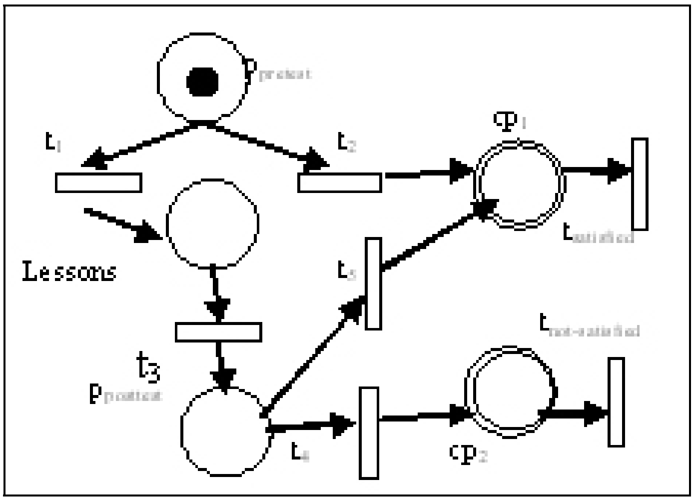


Figure 6. Abstract Module 1 subnet

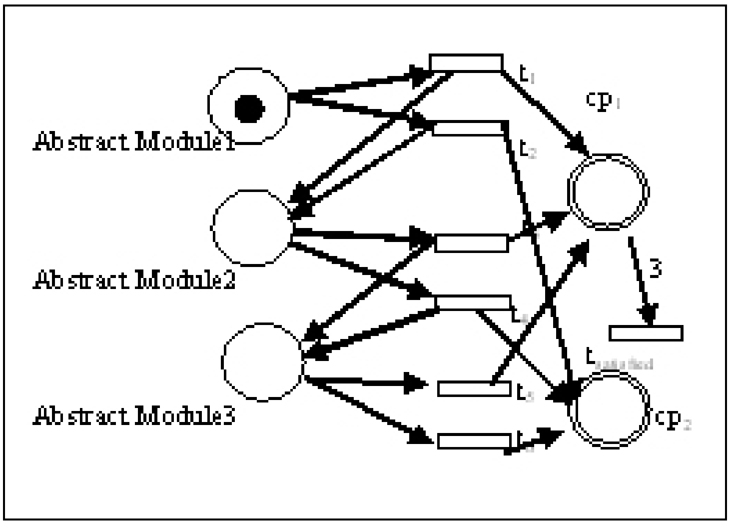


Figure 7. Modules subnet

The output of Ppretest is assessed by t1 and t2. If the learner entered into the Abstract Model 1 and passed the Pretest, he/she can ignore the immediate Lessons and the Posttests to experience the next Module. If the learner does not pass the Pretest, he/she is directed to the Lessons, and once completed, must take the Posttest. The Posttest is not selectable by the learner. It is only encountered after ‘flowing’ through the Lessons.

The same operation to construct the Modules is demonstrated in Figure 7. The subnet replacements for Abstract Module 1, Abstract Module 2 and Abstract Module 3 each have output transitions that are reached depending on which module is satisfied.

3. Supporting Collaborative Learning

The main contribution of this paper is in the newly proposed technology of collaborative learning in SCORM. In several studies [3], collaborative learning was shown to be effective in distance education.

Our DCPN can model all the flows of collaborative learning that are key to distance learning, as we explain in the following sections.

The most important feature of collaborative learning (CL) is to form a group with heterogeneity such as having different backgrounds, various learning paths, and diverse instruction styles. According to this feature, collaborative considerations should be addressed before we design tools for collaborative learning. One issue is what capability is necessary and the other is whether the number of groups is satisfied. The solution is DCPN utilizing token color through an exam to verify the learner capability and the token number to validate the specified number of groups. The learner should take an exam to tell the system what role he can play before entering into collaborative learning. As shown in Figure 8, taking the Exam and dividing into two categories (black token and white token), p3 collects the results and waits until the condition is satisfied. That is, if there are two black tokens and one white token, the transition can be fired and then three learners make up a team to enter into the CL mode.

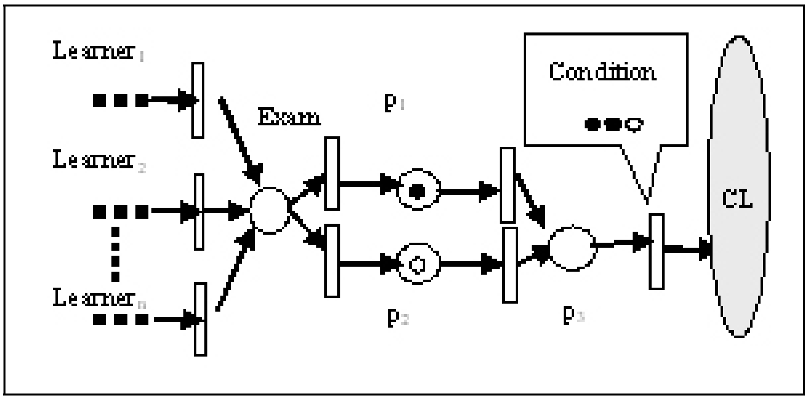


Figure 8. Conditions before collaborative learning mode

After entering CL mode, a team member has to learn some knowledge or skill before working together to solve a problem as their mission. In our model (shown in Figure 9), the "s" above the arc represents the LMS break point of suspending the lesson to enter into chartroom mode to facilitate the learner getting assistance from other partners. When the learner gets the necessary background information, he/she can go back to the break-point and keep going. In order to confirm that the learner not only achieved the global mission but also the personal mission of a collaborative learning, we identify individual accountability using places.

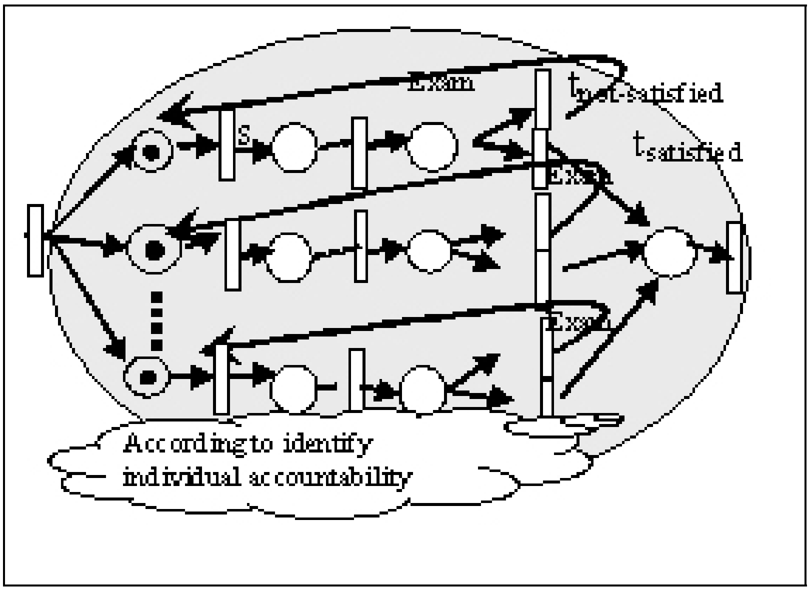


Figure 9. CL Construct

4. Conclusion

Distance-learning Color Petri Net for the SCORM Sequence Specification using subnet oriented aggregation and firing rules. The advantage of mapping DCPN on the SCORM sequence is to make all instructional elements traceable, via a visualized activity tree. Authors using can preview the learning content after authoring, understanding the sequence and navigation behavior through the help of a visualized DCPN. We believe that cooperative learning and learning content management are important to e-Learning. As a contribution, the newly proposed DCPN provides a starting point for the formal definition of navigation and collaboration behavior of users using SCORM compliant courseware. With this model incorporated into a visual tool, we have found that instructors are able to understand and control instruction flow easily.

Reference

[1] ADL Technical Team, Sharable Content Object Reference Model (SCORM) Version 1.3 Working Draft 1, Advanced Distributed Learning (ADL), October 22, 2003

[2] J. L. Peterson, Petri Net Theory and the Modeling of Systems (Englewood Clifs, NJ:Prentice-Hall, 1981).

[3] T. K. Shih, J. C.-S. Hung, W.-C. Ko, W.-C. Chang, and N. H. Lin,“Collaborative Courseware Authoring Based on SCORM Metadata,” in Proceedings of the International Conference on Computers in Education, Hong Kong, China, Dec, 2003

Wen-Chih Chang
Department of Computer Science and Information Engineering
Tamkang University, Tamsui, Taiwan, R.O.C.

H. W. Lin
Department of Computer Science and Information Engineering
Tamkang University, Tamsui, Taiwan, R.O.C.

Timothy K. Shih and Hsuan-Che Yang
Department of Computer Science and Information Engineering
Tamkang University, Tamsui, Taiwan, R.O.C.

nbird@mail.mine.tku.edu.tw

Simple instruction and assessment workflow templates

Overview

This brief paper presents a couple of learning and assessment activity templates that can be implemented using SCORM 2004 sequencing. Even though the implementation may be a SCORM activity tree and rules, the focus is on the learning workflow and not the underlying tree. The templates described in this paper are designed to be used alone or nested at any level of depth. Typically, the second template may be nested into a placeholder in the first template. The templates provide typical adaptive learning sequence workflows, such as pretest followed by selective instruction for objectives identified as needing instructions, remediation, assessment, etc. The instruction template supports field-dependent as well as field-independent learning styles. The templates assume that the competencies to be achieved or assessed are clear, and that they "roll up" to some "overall" competency. For example, they might be sub-skills of a more complex skills or different facets of a competency.

Why templates

Templates for learning workflow can help automate the gathering of properly tagged learning content relevant to identified objectives into an instructionally sound package.

Templates can be used by normal human. The SCORM 2004 sequencing specification is very arcane. It should be possible to take advantage of it without having to learn its intricacies, just as it is possible to drive a car without knowing how to tune an engine.

Activity tree generation

The templates do not include any content. They are not as themselves SCORM conformant Simple Sequencing activity trees, but they can be used to generate Simple Sequencing activity trees. They are intended to be used in an authoring environment that mediates the generation of the actual activity tree in XML format on behalf of the author, and in which content assets and SCOs are created or obtained separately. Typically, after ensuring that the objectives are clearly identified, such a tool might present the author with a simple workflow diagram similar to those shown in this paper, and ask the user to populate the placeholders. Where there is a decision to be made by that author, it is always very simple (yes or no, allow choice or not).

Note that the workflow models in these templates are not applicable only to single user SCORM content, or limited to online instruction. The same workflows could apply to a personal training program or to instructional workflows where a team must achieve a certain level of competency. This is one of the reasons why the specific learning resources or assessment instruments are not specified here. For SCORM applications, online content and online assessments implemented as a SCOs would of course work quite well.

Instructional design

The workflow model is based on good old learning workflow principles: Figure out what you need, skip what you don't need, and retry until you succeed or you fail. The activities to be sequenced may be very simple or very complex. They might in fact use different strategies than the workflow of the template within which they are embedded. For simplicity, the described here templates do not include the option recommended in another Ostyn paper which suggests to use different remediation methods if remediation is necessary:. If the instruction did not work the first time, it may indicate that a different method should be used for remediation rather than just repetition. This instruction template, however, includes an option to let the learner choose between two basic strategies often associated with basic learning styles: Field dependent learners require guidance and hand holding. Field independent learners tend to learn better by independent exploration, and to get annoyed by too much guidance. An author may choose to use one, or the other, or to leave the learner the option to choose. At this point the SCORM does not have support for a standard data model for learning styles; if it did, this decision could of course be automated.

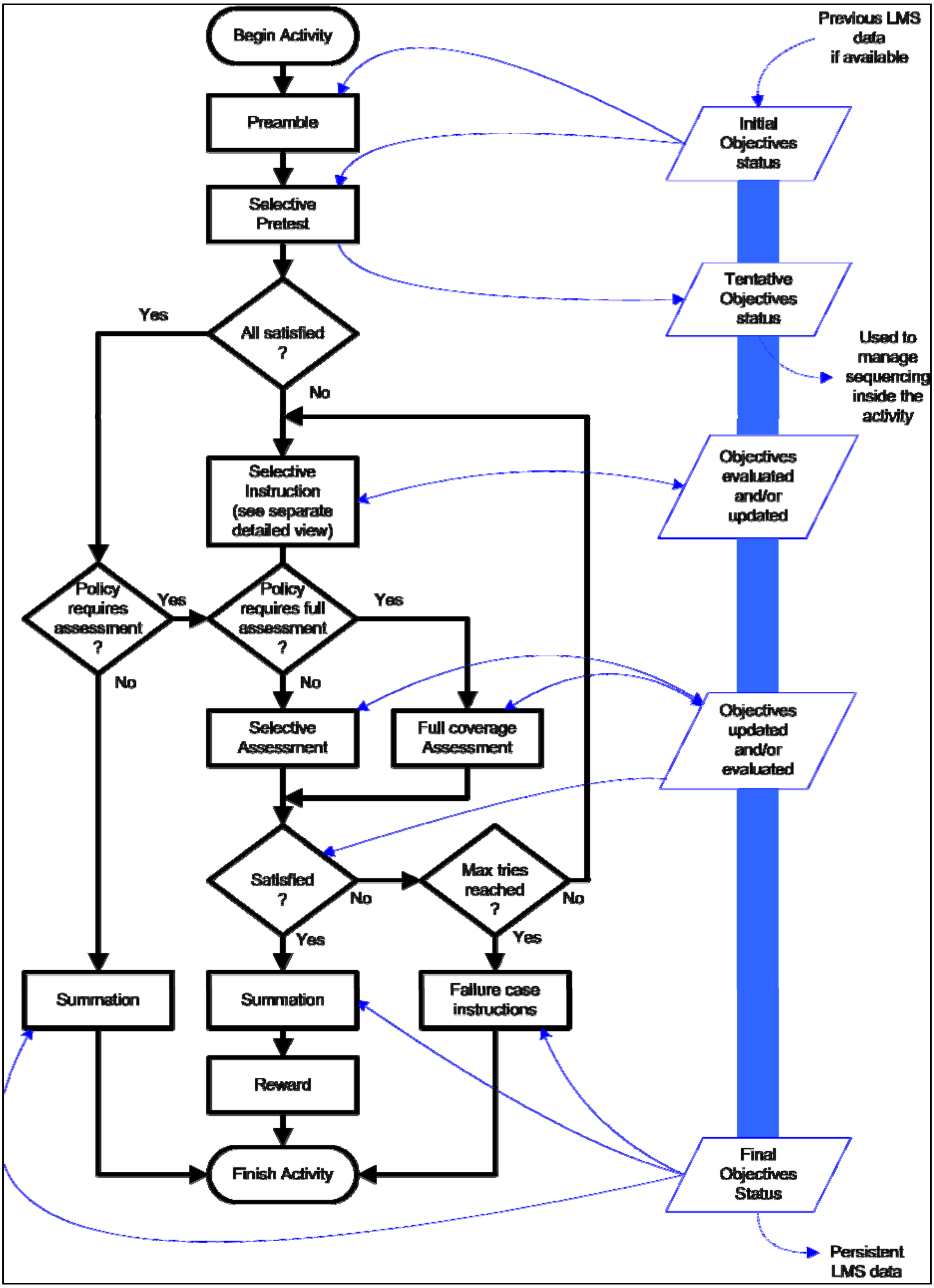


Figure 1 - Instruction and assessment template workflow

Activity tree nodes and activity titles	Has content (SCO, asset or sub-manifest)	Flow	Choice	Auto skip	Max tries	Objective mapping
O Topic Title		Y	N			Main objective
O Preamble	X					Objective1 .. ObjectiveN
O *(invisible) Pretest*		Y	N			
O Selective pretest	X					Objective1 .. ObjectiveN
O Summation *(use only if full assessment not required)*				Y		Objective1 .. ObjectiveN
O *(invisible - try or retry)*		Y	N		n or unlimited	
O Learn (Topic Title)	(see Selective Instruction)					Objective1 .. ObjectiveN
O Assessment *(designer chooses full or selective)*						Objective1 .. ObjectiveN
O *(invisible - used if success)*				Y		
O Summation	X	Y	N			Objective1 .. ObjectiveN
O Reward	X	Y	N			Objective1 .. ObjectiveN

Figure 2 - Activity tree for instruction and assessment template

Each rectangle in the workflow diagrams in Figure 1 and Figure 3 represents a placeholder for a SCO or asset, or for a sub-activity that may be generated from another template. Some of the placeholders may be left empty. In that case a generator tool using the template would adjust the generated activity tree accordingly.

Figure 2 and Figure 4 show a summary of the data to be generated for each activity tree.

Conclusion

Reusable, adaptive strategy templates that support multiple objectives can be designed and reused in flexible ways. The focus should be on the activity workflow rather than on content structure. The approach suggested in this paper starts from the learning objectives and the instructional strategy, and looks for learning resources to that can be used in appropriate learning activities.

Even though the implementation may be a SCORM activity tree and rules, what you really want to specify first is the learning workflow and not the underlying tree that defines a content structure. This underlying tree and the syntactical complexity of sequencing rules should remain hidden in an implementation layer. This may suggest a different way to design and build some authoring tools.

Metadata for learning resources should support discovery for use in templates that provide activities for different stages in the learning process. Metadata should include not only the identification of a competency definition or learning objective, but also some value from a standard vocabulary that identifies what the learning resource's purpose is in relation to that competency or objective, such as introduction, tutorial, assessment, practice, summation, and so on. As of this writing, no such common vocabulary seems to exist that would be representative of various practices as well sound theory.

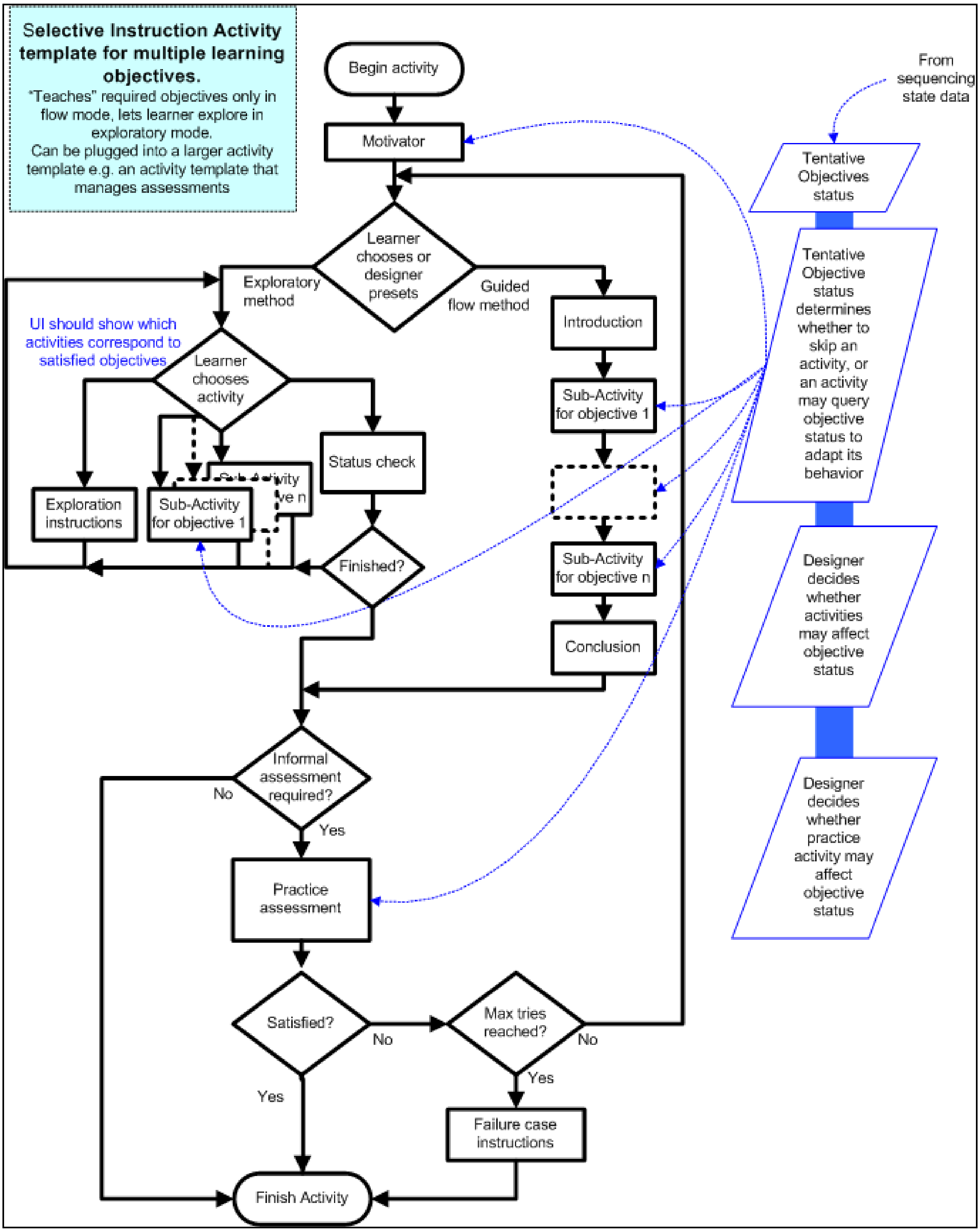


Figure 3 – Workflow in selective instruction activity template

Activity tree nodes and activity titles	Has content (SCO, asset or sub manifest)	Flow	Choice	Auto skip	Max tries	Objective mapping
o Learn (topic title)		Y	N			
o Motivator	X					Objective1 .. ObjectiveN
o *(invisible - try or retry)*		Y	N		n or unlimited	
o Choose how to learn		N	Y			
o I prefer to explore		N	Y		unlimited	
o Exploration instructions	X					Objective1 .. ObjectiveN
o Topic 1	X					Objective1
o Topic 2	X					Objective2
o Topic n	X					ObjectiveN
o Finish exploration	X					Objective1 .. ObjectiveN
o Guide me		Y	Y *(if allowed by designer)*			
o Introduction	X					Objective1 .. ObjectiveN
o Topic 1	X			Y		Objective1
o Topic 2	X			Y		Objective2
o Topic n	X			Y		ObjectiveN
o Conclusion	X					Objective1 .. ObjectiveN
o Practice assessment	X					Objective1 .. ObjectiveN
o You are out of tries *(normally hidden)*	X			Y		Objective1 .. ObjectiveN

Figure 4 - Activity tree for selective instruction activity template

Claude Ostyn
Ostyn Consulting
Claude@Ostyn.com

Using Topic Maps to Support Non-linear Navigation for UK Defense E-Learning Programs

1. Introduction

Cranfield University supports military colleagues at the UK Defence Academy in the delivery of a wide range of educational courses relevant to the needs of the defense sector. Recently we have been engaged in developing quality web-delivered distance learning courses for the British Army as part of the Review of Officer Career Courses (ROCC) initiative. The ROCC initiative aims to provide 'life-long learning' for Army officers up to the point of retirement. Our particular concern has been with developing courses aimed at officers in the early and middle periods of their careers. These are known as Military Knowledge 1 and Military Knowledge 2 (MK1 and MK2) and provide in total some 120 hours of self-directed learning, covering, as the titles suggest, basic knowledge of military doctrine, service functions and organization, together with coverage of relevant science, technology and project management topics. For logistical reasons, MK2 is the first of the courses, as of December 2004, to go live online.

The MK learning materials are a multimedia and interactive activity rich resource. Input from instructional designers has ensured that content is supported by clear statements of learning outcomes, activities that support learning and summaries and formative assessment questions that support self-directed learning and revision. The materials are also designed to be an effective on the job 'just-in-time' training and reference tool. Topic maps have been defined to aid non-linear navigation through the course content down to the level of individual 20-40 minute lessons. The maps provide learners with a conceptual view of content at course level (Knowledge Map) and lesson level (Lesson Map). In the paper, we describe the form and function of these maps and how they relate to various sequencing and navigation specifications. Our implementation also supports non-linear access to particular topics within individual lessons, together with the ability to track progress down to intra-lesson topic level. We also discuss how our MK delivery system is relevant to the broader class of hypertext adaptive tutorial systems and associated constructivist and conversational pedagogies.

2. Implementation Overview

The MK course is divided into parts, modules, sections and lessons (Appendix A). Each lesson is further divided into as many as five topics. The ROCC team are interested in tracking progress, but not necessarily formative assessment results, of MK students at the topic level. Although our current implementation is based on the IMS Simple Sequencing Specification (IMS SS) [1], we have designed the product to achieve the required degree of tracking by packaging each lesson at topic level to SCORM Content Aggregation Model 1.3 (CAM) [2] specifications and capturing activity status tracking data as defined by SCORM Run-Time Environment 1.3 (RTE) [3] and SCORM Sequencing and Navigation 1.3 (SN) [4]. Our full implementation of CAM, RTE and SN will be dependant on the availability of a UK Ministry of Defence approved SCORM 2004-conformant LMS, expected in late 2005.

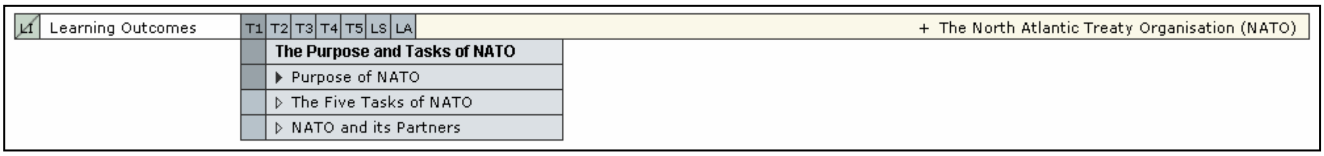


Figure 1 - Lesson Navigation

The recommended path within lessons is forward and backward which guides students in a linear manner from introduction through to summary and concludes with a formative assessment. However, students are permitted to browse lesson content or to go directly to any given topic within the lesson via the lesson's internal navigation (Figure 1). Students are also provided with a Lesson Map which allows them to view a dynamically generated 'cluster' hierarchy and navigate to a specific 'leaf' activity (Figure 2). MK lessons are developed using Macromedia Flash MX, JavaScript and XML and launch in a new browser window which occupies the entire screen of a 1024 x 768 resolution monitor.

The SN book states that "SCORM does not place any requirements on the LMS or content as to what navigation controls are visible, how they are rendered, how they are triggered or what navigation events they trigger". This welcome lack of navigation specification in the requirements has enabled us to empower students to freely define their own path through the lesson by providing them with multiple navigation options to suit their individual style of learning.

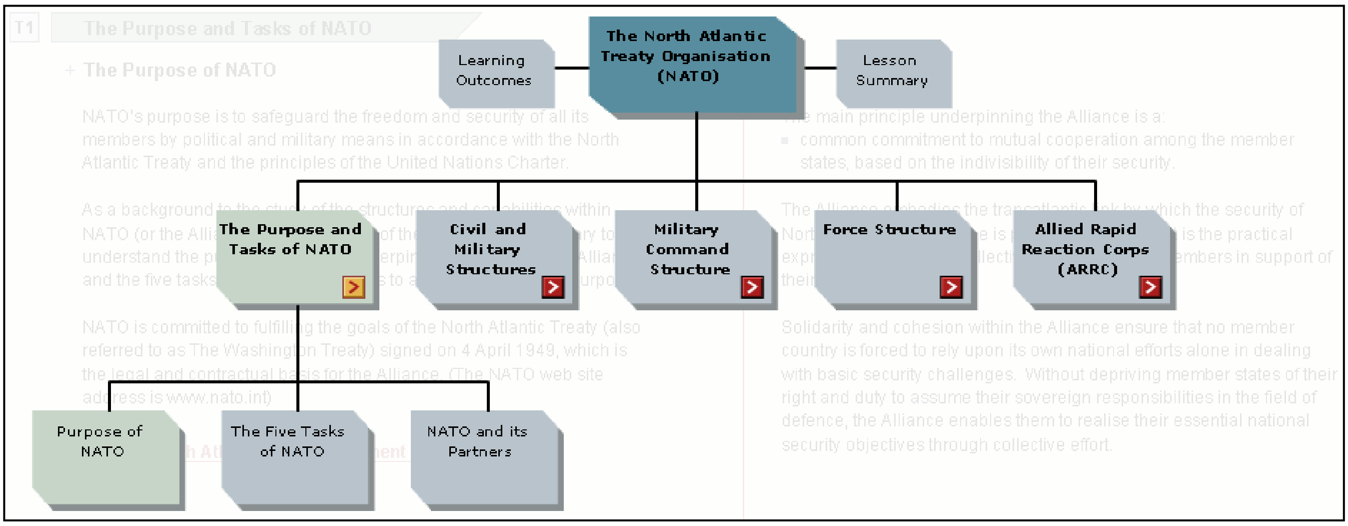


Figure 2 - Lesson Map

3. Navigation Strategy

The MK courses are structured in such a way as to give total freedom to the student in terms of the order in which each lesson, and indeed each topic is completed. Sound instructional design principles were applied to the intra-lesson navigation but we also wanted to provide students with a visual representation of a relatively complex course and, ideally, a mechanism for browsing and launching lessons. Our solution (Figure 3) was to create an interactive Knowledge Map which allows macro navigation at course level as well as micro-level status details for each lesson including the cmi.completion_status. Individual lessons as independent 'organizations' are still responsible for getting and setting run-time student tracking data but those values are also available to the Knowledge Map for visual display to the student as required.

When employed as a self-directed training programme, the Knowledge Map allows MK students to effectively manage and track their own progress and receive dynamic updates of their status for each lesson. Over time, we intend to evolve our pedagogical model by using the data already being captured programmatically to determine a student's path through the course and implement the full capabilities of SN to dynamically apply adaptive sequencing rules where appropriate. Although this is not a current requirement of our military client, we (Cranfield University), as noted below, intend to implement this evolved model for R&D purposes.

4. Concluding comments

As early as the 1970s, Pask, Scott and colleagues designed an adaptive tutorial system incorporating a hypertext topic map that supported constructivist, conversational learning [5, 6, 7]. Arguably, current technologies still fall short of being able to support the full implementation of Pask and Scott's (ibid) CASTE (Course Assembly System and Tutorial Environment). Topic maps are increasingly being used in support of learning activities (see, e.g., [8, 9]) and adaptive web-based teaching systems are receiving more considered attention (see, e.g., [10, 11]). Our MK system uses topic maps to support learner navigation and tracking. As part of its requirement to serve as a browsable resource, adaptive tutoring with progress through lesson content contingent on summative assessment attainments was not a feature of the design. The next step for the Cranfield e-learning R&D team is to prototype such a system both as a demonstrator of pedagogic principles and also to serve as a CASTE-like sophisticated, flexible learning environment for the study of student learning under different pedagogic regimes (learning designs) and constraints.

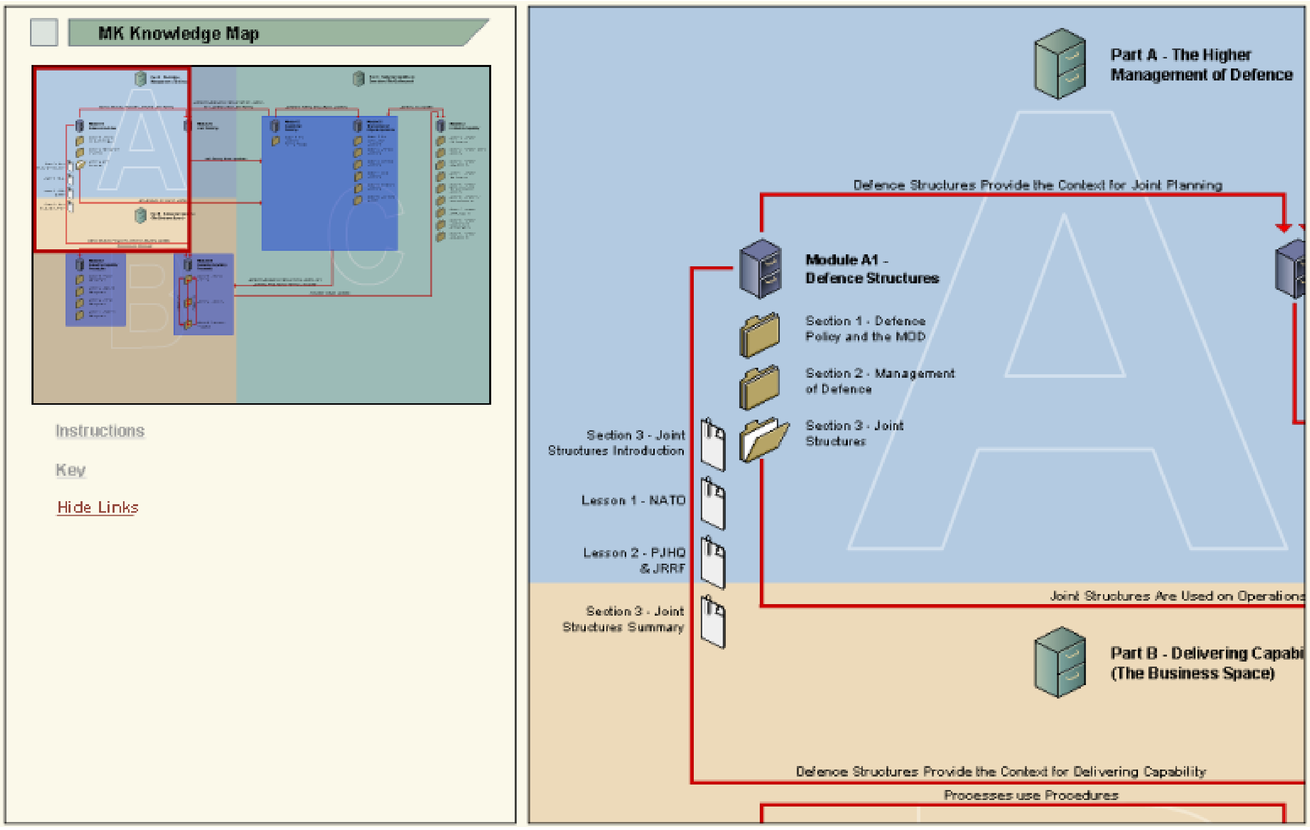


Figure 3 – MK Knowledge Map

5. References

[1] IMS Global Learning Consortium, Inc, *IMS Simple Sequencing Best Practice and Implementation Guide*, http://www.imsglobal.org/simplesequencing/index.html.

[2] Advanced Distributed Learning, *Content Aggregation Model Version 1.3*, http://www.adlnet.org/.

[3] Advanced Distributed Learning, *Sequencing and Navigation Version 1.3*, http://www.adlnet.org/.

[4] Advanced Distributed Learning, *Runtime Environment Version 1.3*, http://www.adlnet.org/.

[5] Pask, G. and Scott, B. (1973). "CASTE: a system for exhibiting learning strategies and regulating uncertainty", *Int. J. Man-Machine Studies*, 5, pp. 17-52.

[6] Pask, G. (1975). *Conversation, Cognition and Learning*, Elsevier, Amsterdam.

[7] Scott, B. (2001). “Conversation theory: a dialogic, constructivist approach to educational technology”, *Cybernetics and Human Knowing*, 8, 4, pp. 25-46.

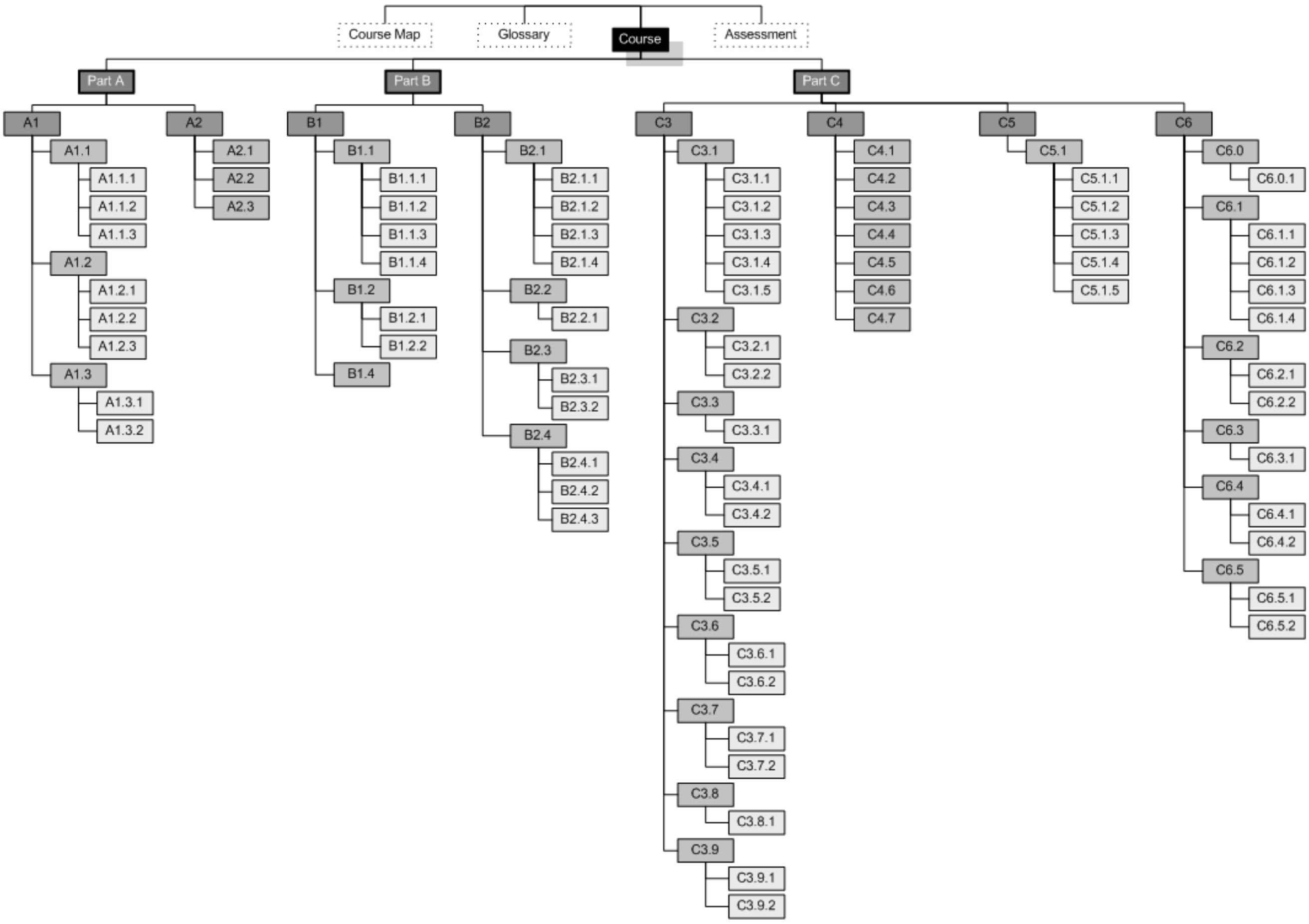
[8] Park, J. and Hunting, S. (eds.) (2003). *XML Topic Maps: Creating and Using Topic Maps for the Web*, Addison-Wesley, Boston, MA.

[9] Dichev, C., Dicheva, D., and Aroyo, L., (2004) *Using Topic Maps for Web-Based Education*, Advanced Technology for Learning, Vol. 1, No. 1.

[10] Brusilovsky, P., Stock, O. and Strapparava, (eds.) (2000). *Adaptive Hypermedia and Adaptive Web-Based Systems*, Springer, Berlin.

[11] De Bra, P., Aroyo, L., and Chepegin, V., (2004) *The Next Big Thing: Adaptive Web-Based Systems*, Journal of Digital Information, Volume 5 Issue , Article No. 247, 2004-05-27, http://jodi.ecs.soton.ac.uk/Articles/v05/i01/DeBra/.

Appendix A. Military Knowledge 2 Course Activity Tree



Zach Johnson
Defence Academy of the United Kingdom
Cranfield University
z.johnson@cranfield.ac.uk

Bernard Scott
Defence Academy of the United Kingdom
Cranfield University
b.c.e.scott@cranfield.ac.uk

ITS and SCORM

The Sharable Content Object Reference Model (SCORM) defines the communication between online Learning Management Systems (LMS) and courseware content. As SCORM is likely to extend to new instructional affordances, Intelligent Tutoring Systems (ITS), their ideology, and their implementation become important features on the SCORM landscape.

A defining characteristic of ITS is the option space that is available for presentation to the learner by the system, at any instant. For any ITS of reasonable complexity, exhaustive search is intractable: with n options available, the number of options is exponential. Classical AI uses both search and planning to optimally traverse such option spaces. Search and planning remove from consideration paths that have low or zero probability of being optimal.

Basic search and planning have recently made their way into SCORM 2004, which operates over a content organization, such as a course, of learning objects called Sharable Content Objects (SCO). SCOs themselves are meant to be independently reusable units of courseware

The Sequencing and Navigation (SN) model allows the course designer to define learning activities, which may be conditionally sequenced based on information derived by tracking the learner's progress. The key concept of the SN model is the Activity Tree. The Activity Tree is a branching data structure derived from the Content Organization defined in a Content Package. Items in a Content Package are mapped by the LMS to Learning Activities in the Activity Tree, with SCOs and launchable Assets as the leaves. In SCORM2004, the Content Package has been annotated by the course designer with sequencing rules, so the LMS can use the Activity Tree to sequence a course at runtime.

Each sequencing rule has an "if-then" structure, where the "if" conditions use the learner's tracking information. The "then" sequencing actions include eliminating a node or a node's children from being sequencing candidates, exiting a parent node or all nodes above that node, retrying the current or all nodes, and sequencing the next or previous nodes in a pre-order traversal. These sequencing rules may be used to create a continuum of controlled sequencing. At one extreme of the continuum, the learner is free to choose any Learning Activity in any order. At the other end, the learner can only move forward, and tracking information is used so that the leaner skips over material they already know.

SCORM has sequencing abilities that approach those of ITS. However, an unavoidable consequence is that the closer SCORM sequencing approaches the sequencing of an ITS, the smaller the unit of instruction in a SCO becomes, and the less independently reusable the SCO becomes. This dichotomy illustrates that in SCORM SS a delicate balance must be maintained between adaptive sequencing and independent reuse.

In our comparison of ITS and the SCORM, we focus primarily on sequencing. Clearly AutoTutor, our example ITS, has a number of components outside the SCORM's focus, e.g. natural language processing components to interpret student utterances and assess student performance. However, these kinds of components could be supplied by the LMS or SCOs via some extension, e.g. plug-ins or remote services. Similarly the format of the content is not a concern, because the SCORM Content Aggregation Model is sufficiently general to represent all media that might be presented in a tutoring session.

To examine the similarity between ITS strategies and SN in SCORM, we see that AutoTutor is an LMS operating on two levels of pre-organized knowledge structures, called Curriculum Scripts. The CS specify all possible tutor utterances and on-screen display information. This information is organized in a tree-like structure. AutoTutor, functioning as an LMS, simply 1) provides a seed question, 2) gets a response from the student, 3) evaluates the response, 4) selects the next item and 5) delivers the next message to the student. The five steps are simple enough that one observes similar behavior in almost all CBT or WBT. We observe that ITS differ from ordinary CBT or WBT in the details of the last three steps. Namely, evaluation of the response, selection of the next item, and deciding how to deliver the item.

In other publications, we offer a formal mathematical model for Intelligent Tutoring Systems. This model serves as a formal description of AutoTutor. We will largue that this is general enough to model ITS based on other learning theories. The basic idea of the model is similar to a typical Skinnerean Model, namely, it concentrates on the behavior of Tutor-student interaction.

Xiangen Hu
The University of Memphis
Memphis, Tennessee
x.hu@mail.psyc.memphis.edu

Andrew Olney
The University of Memphis
Memphis, Tennessee

Regular article: **Aligning Campus Portals with Learners' Needs**

A Preliminary Study on the Implementation of Campus Portals in Iranian Higher Education Communities

Abstract

Effectively developing and deploying campus portals can dramatically increase productivity and profitability of research and education. The cutting edge of this initiative lies in aligning portals with students' current needs. Our study aims at identifying these needs and provides a preliminary theoretical framework for portal developers to benchmark their objectives according to educational requirements. The study is mostly done based on local observations and experience of its conductors within higher education communities in Iran. The result of this primary study paves the way of implementing campus portals in the Iranian higher education communities which will be paced by the authors of the article in the near future.

Keywords

Portal- Campus Portals - Higher Education- E-learning, Information Technology

1. Introduction

E-Learning usually refers to "learning that is delivered or enabled via electronic technology" (Sun Microsystems, 2002). It encompasses learning delivered via a range of technologies such as the internet, television, videotape, intelligent tutoring systems, and computer-based training.

E-Learning is a subset of the larger worlds of both “information technology” and “education and training”. It can be valuable when used as a part of a well-planned and properly supported education and training environment, but e-learning is not a magic bullet that replaces or renders obsolete existing pedagogical theories and approaches.

Many learning and technology professionals believe that e-learning will have “arrived” when we stop referring to it by a separate name and begin considering it as an integral part of a complete learning environment.

Recent advances in the availability and speed of Internet access and in the power and availability of personal computing platforms have dramatically increased the opportunities for the use of collaborative environments and other distributed learning technologies. As a result, a wide range of new products are being developed and many new companies have entered the learning technology market.

New categories of products continue to emerge, some providing new capabilities and others combining existing functionality into new product configurations. It can be a challenge to determine how these systems relate to each other and how they fit into a complete e-learning environment. The emergence of e-learning does not mean that existing software applications are obsolete. Systems such as Student Administration, Human Resources, and Library Management provide critical components of e-learning environments. The challenge is to integrate these systems effectively with e-learning application services.

This has been done today in what is being called as "Campus Portal". Campus portals merge a wide range of educational applications into an integrated web-based system. These portals are designed and developed at many modern colleges and universities within recent years and are becoming more popular as useful tools in offering academic services. They are being used by both the current and distant students. Some colleges even offer their portals to their staff as official media for internal communication. These capabilities have proved portals as effective systems of e-learning.

The remaining question is: "How can we align portals with students' current needs?" Sometimes, adding new contents or services to a portal may answer this question; but, changing needs of students makes the job harder and requires enough flexibility of portals both in nature and usage.

We have made an effort in this study to identify those needs of Iranian students which may be satisfied through implementing portals and to provide a theoretical framework for portal developers to allign their objectives according to educational requirements.

2. E-learning Needs in Iran

In 1992, Peter Drucker predicted that in the next 50 years, “schools and universities will change more drastically than they have since they assumed their present form 300 years ago when they organized themselves around the printed book” (Drucker, 1992, p. 97), but what about developing countries? Can be it true again for such states?

The history of e-learning in Iran at present time did not exceed than 5 years, yet from a realistic point of view we might say that e-based learning in Iran has had a 3 year experience and even younger.

A successful campus portal should be designed and implemented so that real needs of learners might be satisfied. These needs can be summarized as following according to recent studies (Dilmaghani, Noori and et al 2003):

- Realistic comprehension concerning the process of learning [1]
- Proper implementation of computer hardware and software [1]
- Strong IT education [1]
- New IT infrastructure [2, 3]
- Enough experienced IT professionals [1]
- Realistic point of view or strategic program for higher education [3]
- Sufficient budget and equipment [3]
- Real learning stimulus [1]
- Preparedness for an active information society and new technology [4]
- Stable political, social and economic situations [3]
- Compatible educational resources for e-leaning [3]
- Information literacy [3]

We may categorize basic needs for e-learning in Iran in four main classes including: Social & Cultural, Economic, Technological and finally, Academic. All of these categories have their own characteristics which should be addressed in a realistic manner.

3. What are Portals?

At the most basic level, portals gather a variety of useful information resources into a single, “one-stop” Web page, helping the user to avoid being overwhelmed by “info glut” or feeling lost on the Web. But since no two people have the same interests, portals allow users to customize their information sources by selecting and viewing only the information they find personally useful. Some portals also let you personalize your portal by including private information (such as your stock portfolio or checking-account balance).

Put simply, an institution's portal is designed to make an individual's Web experience more efficient and thereby make the institution as a whole more productive and responsive. But portals have an economic and social impact that extends far beyond any basic functional definition. Eighty nine percent of the estimated fifty eight million people using the Web in the United States use some type of portal.

It is estimated that over 20 percent of the Internet's retail e-commerce is portal-based. And though portals have historically been developed from search-engine-based sites (e.g., Yahoo, Excite, Lycos, Alta Vista) or ISP-based sites (e.g., AOL, Earthlink, Prodigy), their value goes far beyond a Web page containing a directory of URLs. One author described a portal as a place to start your day and get a little news. It is an epicenter of the Web experience, a "home base," a place to return to when you get lost, a place to keep your information, a place from which to communicate with others, and a trusty guide to all things 'Web.' [7]

4. The Potential Value of a Portal for Higher Education

Portals are also used to support learning communities, which are groups of people with interest in a particular topic or subject area. The portal provides a way to identify people with similar interests and provides collaboration tools and content sharing to members of these communities.

Portals bring together the e-learning tools, content and delivery environment and organize them into logical groupings based on the role of the individual accessing the portal. Each organization using a portal will define and organize detailed roles based on their needs, but some common overall roles are content developer, instructor, advisor, administrator, and learner.

In the higher education space, schools implement these portals as an integral part of the school community and learning environment. Portal technology and services are available from a range of vendors including specialized vendors like Campus Pipeline, course management system vendors like Blackboard, and Student Administration products such as PeopleSoft.

Many students are adults in the real world—they are employees and parents—and certainly elements of a community portal will be critical to building lifelong loyalty and retention. Portals should serve as an important publishing medium for the campus, sending some information to everyone but customizing other information to meet the needs of different segments of the community and allowing individuals to personalize their own portals. Using polling technologies, portals can serve as a valuable tool for real-time institutional research and strategic planning.

Portals look both inward and outward and can provide a powerful medium for campuses to communicate with off-campus constituencies: prospective students; parents; students who work or commute; alumni; business and government partners.

Existing budgets may already support these information management and communication functions, although organizational responsibilities may be diffuse. If a campus is prepared to reorganize around a knowledge management strategy, it is possible to achieve large returns on investment by redefining information resources as a service.

The return on investment should be measured both in terms of cost savings and in qualitative terms, measured by extending and revitalizing the sense of participating in a campus community.

So should a campus develop a portal? Increased efficiency alone suggests yes, but there are other benefits that make a personalized campus portal not only desirable but imperative. We believe that the value of a portal to a campus is that it can be used to engage constituent groups, empower them with access to information resources and communication tools, and ultimately retain them by providing a more encompassing sense of membership in an academic community.

In the academic space, particularly in higher education, publishers are making content available in most subject areas. They are repurposing their existing content for web delivery as “course packs” or “cartridges” that run on widely used course management systems such as WebCT and Blackboard. [7]

5. Conclusion

It is obvious that deploying advanced higher education institutes and colleges equipped with modern e-learning facilities is one of today's urgent needs in developing countries like Iran. But the sustainability of such learning systems depends on making sound and realistic pedagogical strategies. New learning technologies need to be targeted so that they may develop applied learning skills in the students. Today the success of an educational program is highly tied to those web-based applications it may provide for its clients.

There has been a great deal of studies on the methods of developing e-learning in Iran and many challenges or problems have been determined upon the results of such studies. As the writers of this article have proposed, developing e-learning portals could be considered as a solution for the hazed situation of online higher education in Iran. E-learning portals are developed based on students' real needs in an online environment. Using advanced countries' experience in the field of implementing e-learning portals may bring fruitful results for the Iranian higher education community such as: content management, developing IT-based skills, university-industry cooperation, educational competition, self-confidence, creativity and many other useful outcomes. The remaining point is the policy of Iranian higher education authorities toward the implementation of new learning technologies including e-learning portals.

References

[1]. Noori, M. (2003). "Traditional Education or Learning with Computer", *Virtual University Conference at Kashan Payam-e Noor College: Conference Proceedings*. Kashan: Payam-e Noor.

[2]. Giveki, F. (2003). "Learning New Methods in Distance Higher Education", *Virtual University Conference at Kashan Payam-e Noor College: Conference Proceeding*s. Kashan: Payam-e Noor.

[3]. Dilmaghani, M. (2003). "National Providence and Virtual Education Development Capabilities in Higher Education", *Virtual University Conference at Kashan Payam-e Noor College: Conference Proceedings*. Kashan: Payam-e Noor.

[4]. Momeni, N. (2003). "First Successful Experience of Distance Learning in Iran", *Virtual University Conference at Kashan Payam-e Noor College: Conference Proceedings*. Kashan: Payam-e Noor.

[5]. Drucker, P. (1992). *Managing for the future: The 1990s and beyond.* New York: Penguin.

[6]. Sun Microsystems white paper—“e-Learning Application Infrastructure”, January 2002

[7]. Looney, M. "Portals in Higher Education", Educause Review July/August 2000

Alireza Hejazi
MS Student of IC Management
Malek-e Ashtar University of Technology
Tehran, Iran
hejazialireza@yahoo.com

Mitra Dilmaghani
Member of Scientific Board
Malek-e Ashtar University of Technology
Tehran, Iran
mitra_dimaghani@yahoo.com

Regular article: **Developing a Collaborative Virtual Learning Model for P2P Grid Environments**

Introduction

The need for computing in support of education continues to escalate. Until recently, everyone assumed that educational computing required desktop computers. Today wireless-enabled laptops make it possible for students to use their time more efficiently, access databases and information from the Internet, and work collaboratively. Through this flexible learning approach, students can succeed in selectively incorporating critical input from their peers and instructor, then revising their documents based on their own interpretation of facts and theory. This technology will soon give students full-time access to computation and wireless connectivity, while expanding where educational computing can take place to the home and field. This is an important equity issue because these computers will provide much of the educational benefit of more expensive computers in an inexpensive format that has many advantages over desktops. Connectivity for these devices will soon be the norm rather than the exception. As they become more functional (e.g computing power and battery technology) and more connected (e.g. peer-to-peer grids), the possibility for completely new and unforeseen application increases.

However, the conventional ubiquitous learning media is based on the textual-based devices that are restricted to obtain selectable information. The new generation technology has been promoted to provide various types of information to meet the personal needs of users at any time. Currently different type of multimedia can be processed by variety of ubiquitous devices making it possible for virtual and collaborative learning environment to be our new eLearning reality. Indeed we can not use the many available packages that are traditionally used for Web-based distant learning (e.g. Blackboard, WebCT, WebFuse, CoSE, TopClass, WebEx, VNC, SCORM, and Tango) because they luck supplying some intrinsic ubiquity capabilities as well as they do not deal reusable open-source learning materials and rely only on the traditional Web-Based infrastructure. Therefore it is of substantial benefits to Internet and ubiquitous devices users if we can have an integrated collaboration environment, which combines non-textual streaming and instant-messaging into a single easy-to-use anywhere at anytime application. This effort requires creating a more general framework to cover the wide range of collaboration solutions that allow different users and devices from different communities to collaborate.

Design Framework for Ubiquitous P2P Grid Collaboration

Generally speaking, the integration of heterogeneous systems into one collaboration system requires the achievement of the following goals [1]:

- Different kinds of application endpoints should join/leave in the same collaboration session.
- Different providers for multipoint multimedia and data collaboration should be connected together to build unified multimedia and data multipoint channels.
- A common user interface should be present for all the collaboration participants using different multimedia and data application endpoints.

Many current technologies and protocols can contribute to the achievements of these goals. However, there is no uniform designing framework which can grantee reusability and adaptively for diverse, ubiquitous and heterogeneous collaborating devices/users. In this article we are proposing a framework based on the Model-View-Controller (MVC) design pattern, which is often been used by applications that need the ability to maintain multiple views of the same data. The MVC pattern hinges on a clean separation of objects into one of three categories — **models** for maintaining data, **views** for displaying all or a portion of the data, and **controllers** for handling events that affect the model or view(s). Because of this separation, multiple views and controllers can interface with the same model.

Part 1: The Model

Model represents an internal representation of a semantic model of the problem of interest. For an educational model need to be used within a collaborative environment, the model needs to comply to a learning object standard. There are many standards that we may use for defining reusable learning objects[2]. In a country like Canada the standard used is CanCore [3]. In this direction the model must be described using a schema and the resulting learning object must have its representative metadata as well as its learning contents (Figure 1).

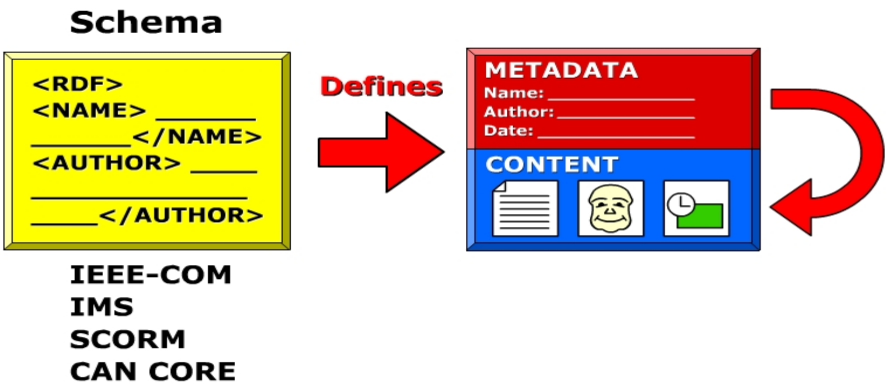


Fig.1: The Learning Object Model.

However, in a peer-to-peer grid environment, the capabilities of peer machines vary and accordingly the model requires to be flexible. Flexibility ensures that the modeled learning objects can be interpreted/transcoded/optimized according to the capabilities of the receiving device. Flexibility also means that any change at the metadata must imply a change in the actual contents. Indeed, the model becomes more important if the data contents include multimedia. In this direction the most common model that can be used for representing virtual scenes and multimedia is either Scene Graph or the DOM Tree. Although both models can be adopted for sharing and collaboration purposes there is still no relation between the metadata and the content of a virtual scene. In this direction, Fiaidhi [4] developed a Virtual SceneBeans model based on scene graph that can relate between the metadata and its contents for the purpose of representing learning objects with dynamic scenes. On the other hand, the Document Object Model (DOM) tree model has been largely used by the W3C SVG standard [5] in which an XML document type is used for describing two-dimensional graphics and animations. However, SVG is significantly more complex than SceneBeans, mainly because it specifies all properties of the scene within the XML document. SceneBeans, in comparison, interprets the document as commands to load and compose external components that define and animate the scene. In addition, SceneBeans can easily be transferred in a Web Service that makes them more accessible to the other collaborating peers. Figure 2 illustrates the two roles of the model in the MVC design pattern.

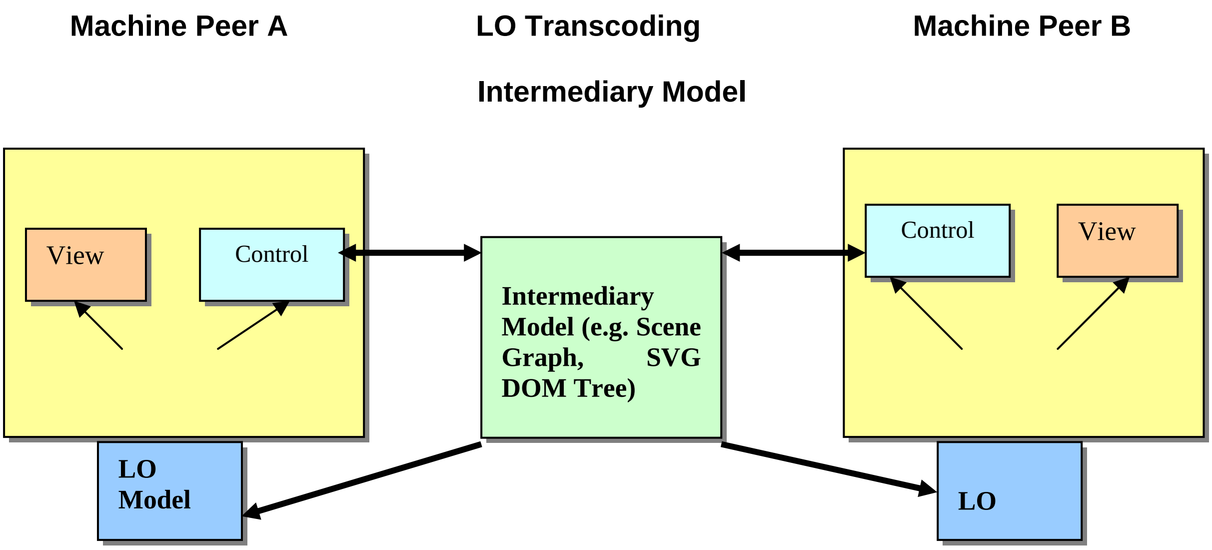


Fig. 2: The role of model within the MVC based model.

Part 2: The Controller

The controller is the code that determines the overall flow of the application model (e.g SceneBean, SVG DOM) within the P2P Grid environment. Basically it comprises one or more struts actions, configuration files, and servlets that manage the accessibility of the various requested virtual leaning objects. This means the control needs to be responsible for coordinating various activities within each peer node as well as between peers and brokers. Within each peer node the control should be responsible for interpreting queries and returning the right sequence of requested and relevant learning objects. In this direction the control will represent a small Learning Content Management System (LCMS) (Figure 3).

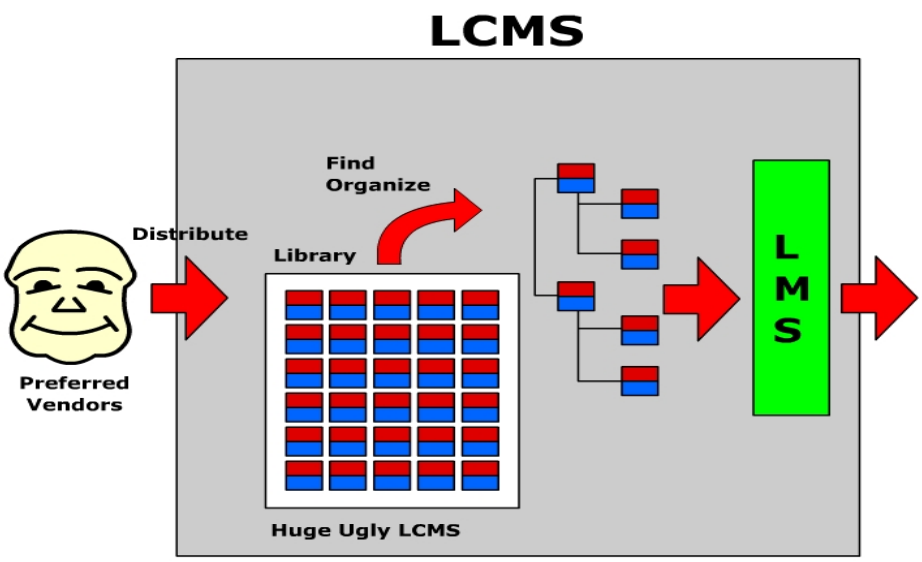


Fig.3.The LCMS Controlling Part.

However, the control should be also responsible for coordinating the various events and messages between peers and/or brokers. This practically means that the control should manage the communication and sharing of learning objects between peers. As far the communication is concerned, this mean the networking infrastructure suitable for ubiquitous and p2p collaboration. In this direction we have very limited open sources that can deal with open sources messages and objects represented in XML-Like source code. In fact on the horizon there are only the Narada and JXTA in which one of them can be used for this purpose (or both through using a proxy[6]connection). Moreover, the control needs to use further primitives to control synchronous and collaborative events. In this direction the XGSP (XML General Session Protocol) protocol can be used[7] to control collaborative sessions when learning objects such as the Virtual SceneBeans (VSB) is used. Finally the control should manage the routing of messages between peers or brokers. However, routing is also effected by the way peer to peer grid is created. In this direction one can use structured grid creation algorithms(e.g. CAN,CHORD) or unstructured grid creation algorithms(e.g. PRV, Hypergrid). The CHORD algorithm is a self-organizing algorithm[8] that uses consistent hashing to map nodes onto a circular space which can have a positive effect on routing messages. Figure 4 illustrates the role of different parts of controllers within our MVC based model.

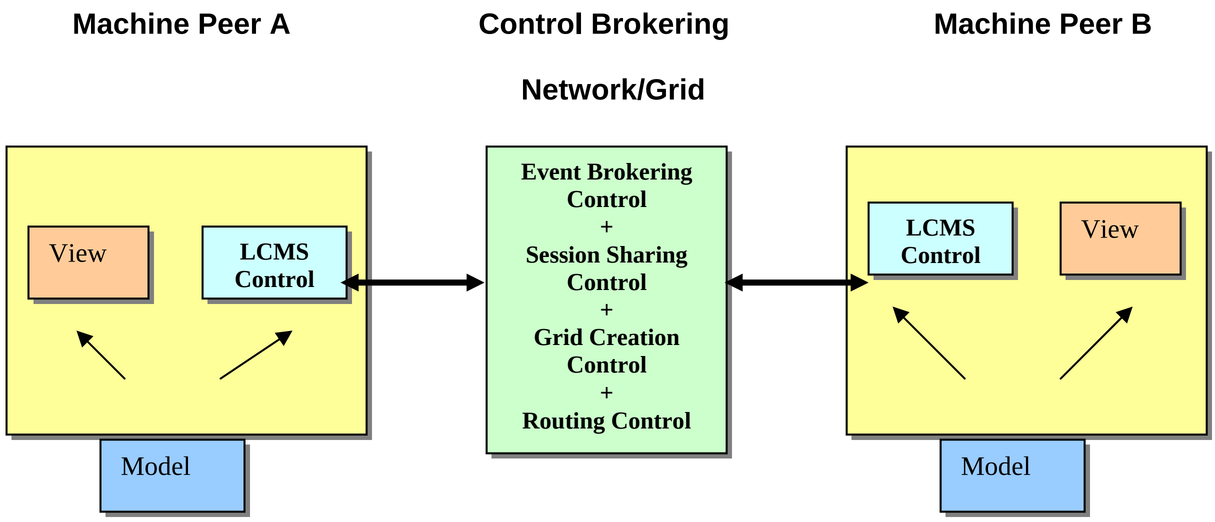


Fig. 4: The role of control within the MVC based model.

Part 3: The View

The view is the code that registers itself as a listener to certain parts of the application's underlying business and functional logic, as represented by the model. The model then notifies all registered views whenever there is a change in the data. Completing the cycle, the controller receives user actions and dispatches them to the model. This idea can be realized using a bean interface. This interface represents an animation loop that runs continuously which alternately calls two other methods (e.g step then render). Step and render are both called on any arriving SceneBean, which is the root of the scene graph, and are then called on the subnodes so that they are called on all nodes in the scene graph. The reason that they are separate methods and not all done in the same method is that if there are many views we may have to render the scene many times before stepping on to the next frame. Such primitive implementation of the rendering process based on Java beans was introduced recently by Martin Baker (www.martinb.com*)* and Fiaidhi[4]. *However, Martins rendering APIs* supports only VRML type events which is somehow restrictive. However, one can use the Virtual SceneBeans, SVG, X3D or WJ3D instead of VRML in describing virtual scenes.

Conclusion:

This article introduces a generic design model for collaborative virtual learning based on the Model-View-Controller (MVC) design pattern, which solves the traditional design problems by decoupling data access, business logic, and data presentation and user interaction. This article introduced a first level refinement of the MVC model within peer-to-peer grid environments. More refinements are expected to be generated during our ongoing research project at Lakehead University to establish the LU Virtual Learning and Training centre.

References:

[1] S., Singhal, and M. Zyda, (1999), Networked Virtual Environments*: Design and Implementation*. Addison Wesley.

[2] N. Friesen (2004). Three Objections to Learning Objects. In McGreal, R. (Ed.). Online Education Using Learning Objects. London: Routledge/Falmer.

[3] G Richards., R. McGreal and N. Friesen (2002), Learning Objects Repositories for TeleLearning: The Evolution of POOL and CanCore, IS2002 Proceedings of the Information Science & IT Education Conference, June 19-21 2002, Irland.

[4] J. Fiaidhi (2004) Virtual SceneBeans: a Learning Object Model for Collaborative Virtual Learning surroundings, Informatics in Education: Int. Journal, Vol.3, No.2, 2004. (191-218)

[5] S. Lee, et al., (2002) Ubiquitous Access for Collaborative Information System using SVG, SVG Open Conference, Zurich, Switzerland, July 15-17, 2002.

[6] Geoffrey Fox, Shrideep Pallickara and Xi Rao (2002), A Scaleable Event Infrastructure for Peer to Peer Grids Proceedings of the ACM Java Grande ISCOPE Conference 2002. pp 66-75. Seattle, WA.

[7] G.C.Fox, and S.Pallickara (2002), "The Narada Event Brokering system: Overview and Extensions", In Proc. of the 2002 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'02) http://grids.ucs.Indiana.edu/users/shrideep/narada

[8] I. Stoica, R. Morris, D. Karger (2003) M. F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications. IEEE/ACM Trans. on Networking, 11(1): 17-32, February 2003.

Jinan A.W. Fiaidhi
Department of Computer Science,
Lakehead University, Canada
jinan.fiaidhi@lakeheadu.ca

Sabah M.A. Mohammed
Department of Computer Science,
Lakehead University, Canada
sabah.mohammed@lakeheadu.ca

Regular article: **Using interpretive ethnography to explore mandatory laptop use in undergraduate teaching and learning practice: research in progress**

1. Research Outline

This research in progress paper presents a case for the use of an interpretive ethnographic methodology to understand the experience of use of laptops in teaching and learning in the Quinn School of Business (QSB), University College Dublin (Fisher et al., 2004). To date, Griffith University (Crebert et al., 2004); De Montfort University (Brown, 1998) Oxford Brookes University (Breen et al., 2001) and Henley Management College (ul-Haq et al., 2003) have all undertaken research which seeks to understand the role that laptops play in teaching and learning practice.

At this point in the research process in QSB, a more in-depth approach is required to understand the situated use and practice (Lave and Wenger, 1991, Lave, 1985) of laptop technologies. An interpretive ethnographic methodology provides us with rich descriptions of the surrounding social factors that influence the situated mandatory student use of laptops in teaching and learning.

2. Interpretive Ethnographic Methods

There is a tradition of interpretive qualitative methods use in teaching and learning experience (Lagemann and Shulman, 1999). Understanding the underlying social, cultural and organisational practices associated with the activity of teaching and learning is highly complex.

Interpretive ethnography data collection will capture depth, narrative, meaningful and rich description (Tedlock, 1994) from individuals and groups involved in "socially shaping" (Baskerville and Myers, 2004, Klein and Myers, 1999, Kline and Pinch, 1999) laptop technologies and provide understanding on the daily practices of academic staff and students (Mitchell, 2000) in more detail.

2.1 Method Limitations

Interpretive ethnography has limitations as a research method: David (2003) documents social exclusion as inherent in the research process; Schultze (2000) discusses constraints on participant articulation; Buchanan (2000) identifies researcher/researchee dualities experienced as worker/ researcher within in the same organisaiton; Northcott (2001) portrays the closeness of the researcher/researchee relationship in as useful in identifying different learning styles in classroom activities. Interpretive ethnography is time consuming, administration heavy and organisationally subjective requiring the ethnographer to spend a significant amount of time in the field (Myers, 1999) and highly dependent on the researchers' own knowledge and experience (Schultze, 2000).

2.2 Data Collection

Interpretive ethnography research has included questionnaire and interviews to capture personal experiences (Goodyear et al., 2003); multiple questionnaires and open ended questioning to explore student approaches to studying (Cantwell and Scevak, 2004); observation to explore student perceptions of their experiences in learning (Edwards et al., 2003) and questionnaires to assess approaches to learning in undergraduate business courses (Byrne et al., 2002).

In the QSB interventionist exploratory research is underway to investigate current experiences of academic staff and students in order to explore the impact of mandatory student use of laptops in teaching and learning practice. Open exploratory interviews are currently being conducted with academic staff and students; recruited using theoretical sampling. A method of locating and engaging with "gatekeepers" (Hammersley, 1995) and "informants" (Van Maanen, 1998) from the field is underway to locate participants for further discussion and to obtain "thick description" (Geertz, 1973) of the Laptop experience.

2.3 Pilot Studies

Pilot data collection has been undertaken in the QSB using triangulation to capture information on the experience of use of laptops in teaching and learning practice. Table 1 and Table 2 list ethnographic data collection methods, piloted over two years of research in the field, and will now be discussed.

Table 1 details research which was conducted within one consecutive academic year (2004/3004).Data points below showed that academic staff and students used laptops in very different ways for both instructional and learning activities, both inside and outside of the classroom.

Table 1: Interpretive Ethnographic Data Collection Points 2003/ 2004: Pilot Study - 1

#	TYPE	THEME / FOCUS	DATA	PARTICIPATION
1	Focus Groups	Experience & Expectation	62 Participants	Staff / Students
2	Questionnaire 1	Learning Experiences	317 / 500 Responses	Students
3	Questionnaire 2	Technology Experiences	174 / 530	Students
4	General Observation	Use of Technology	Reflective/ Critical Journal	Researcher
5	Meetings	Student / Staff Experience	16 Meetings	Staff / Students
6	Interviews	Student / Staff Experience	5 Meetings	Staff / Students
7	Documents	General Documents	Emails, Reports ect	QSB

Table 2 details Pilot Study 2 and is currently in use to extend themes identified in Pilot Study 1. Increased use of in-depth interviews using "open, interventionist and exploratory" methods have been used while recruitment of staff/ student participants has used theoretical sampling (Miles, 1994); to explore the impact of laptops on the process of situated learning (Lave and Wenger, 1991).

Table 2: Interpretive Ethnographic Data Collection Points 2004/ 2005: Pilot Study - 2

#	TYPE	THEME / FOCUS	DATA	PARTICIPATION
1	Questionnaire 1	Learning Experiences	Underway	Students
2	Questionnaire 2	Technology Experiences	Underway	Students
3	General Observation	Use of Technology	Reflective/ Critical Journal	Researcher
4	Observation In-Class	Student / Staff Experience	In Planning	Staff / Students
5	Meetings	Student / Staff Experience	Underway	Staff / Students
6	Interviews	Student / Staff Experience	In-depth and Exploratory	Staff / Students
7	Documents	General Documents	Emails, Reports ect	QSB

3. Next Steps

Observation of laptop use in class will commence in September 2005, for the next consecutive academic year. General observation, meeting attendance and in-depth interviews will continue, along with documentation collection which add socio-historical context. Focus groups and questionnaires will peripherally add to core ethnographic data over the next year. Finally, laptop use in teaching and learning plays a transformative role.

Interpretive ethnography provides a methodology from which to explore social practice and extend our understanding of experience of use of laptops in situated teaching and learning.

4. Acknowledgements

The authors would like to thank Intel (Ireland) for funding this research; staff and students of the Quinn School of Business, UCD for permission to conduct field research and the Centre for Teaching and Learning, UCD for their continued support and supervision.

References

Baskerville, R. & Myers, M. D. (2004) Special issue on action research in information systems: making IS research relevant to practice - foreword. *MIS Quarterly,* 28**,** 1-8.

Breen, R., Lindsay, R., Jenkins, A. & Smith, P. (2001) The role of information and communication technologies in a university learning environment. *Studies in Higher Education,* 26**,** 95-114.

Brown, S. (1998) Reinventing the university. *Alt-J: Association for Learning Technology,* 6.

Buchanan, E. A. (2000) Ethics, qualitative research, and ethnography in virtual space. *Journal Of Information Ethics,* 9**,** 82-87.

Byrne, M., Flood, B. & Wells, P. (2002) Approaches to learning of European business students. *Journal of Further and Higher Education,* 26**,** 19-28.

Cantwell, R. H. & Scevak, J. J. (2004) Engaging university learning: the experiences of students entering university via recognition of prior industrial experience. *Higher Education Research and Development,* 23.

Crebert, G., Bates, M., Bell, B., Patrick, C.-J. & Cragnolini, V. (2004) Developing skills at university, during work placement and in employment: graduates perceptions. *Higher Education Research and Development,* 23.

David, M. (2003) The politics of communication: information technology, local knowledge and social exclusion. *Telematics and Informatics,* 20**,** 235-253.

Edwards, R., Crosling, G., Petrovic-Lazarovic, S. & O'Neill, P. (2003) Internationalisation of business education meaning and implementation. *Higher Education Research and Development,* 22**,** 183-192.

Fisher, L., Butler, M., Keenan, P. & O'Neill, G. (2004) Laptop use in teaching practice: current research in the Quinn School of Business, University College Dublin. *Learning Technology,* 6**,** 51-54.

Goodyear, P., Asensio, M., Jones, C., Hodgson, V. & Steeples, C. (2003) Relationships between conceptions of learning, approaches to study and students' judgements about the value of their experiences of networked learning. *Alt-J: Association for Learning Technology*, 11**,** 17-27.

Hammersley, M., & Atkinson, P (1995) *Ethnography: Principles in Practice*, London: Routledge.

Klein, H. K. & Myers, M. D. (1999) A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS Quarterly,* 23**,** 67 - 93.

Kline, R. & Pinch, T. (1999) The social construction of technology. In Mackenzie, D. & Wajcman, J. (Eds.) *The social shaping of technology.* 2nd ed. Ballmoor, Buckingham, Open University Press.

Lagemann, E. C. & Shulman, L. S. (1999) *Issues in education research: problems and possibilities*, Jossey Bass Wiley.

Lave, J. (1985) Situationally Specific Practice - Introduction. *Anthropology & Education Quarterly,* 16**,** 171-176.

Lave, J. & Wegner, E. (1991) *Situated Learning: Legitimate Peripheral Participation,* Cambridge, Cambridge University Press.

MILES, M., & HUBERMAN, M (1994) *Qualitative data analysis: an expanded sourcebook,* CA, Thousand Oaks, Sage.

MITCHELL, P. D. (2000) The impact of educational technology: a radical reappraisal of research methods. *Alt-J: Association for Learning Technology,* 6.

MYERS, M. D. (1999) Investigating information systems with ethnographic research. *Communications of the Association for Information Systems,* 2**,** 1-20.

NORTHCOTT, J. (2001) Towards an ethnography of the MBA classroom: a consideration of the role of interactive lecturing styles within the context of one MBA programme. *English for Specific Purposes,* 20**,** 15-37.

SCHULTZE, U. (2000) A confessional account of an ethnography about knowledge work. *MIS Quarterly,* 24**,** 39.

TEDLOCK, B. (1994) Ethnography and ethnographic representation. IN DENZIN, N. K. & LINCOLN, Y. S. (Eds.) *Handbook of qualitative research.* Thousand Oaks, CA, Sage Publications, Inc.

UL-HAQ, R., STILES, J. & POND, K. (2003) Learning expectations and learning process design. *Active Learning in Higher Education,* 4**,** 168-180.

VAN MAANEN, J. (1998) *Tales of the field: On writing ethnography,* Chicago, The University of Chicago Press.

Lorraine Fisher
Department of Management Information Systems
University College Dublin
lorraine.fisher@ucd.ie

Martin Butler
Department of Management Information Systems
University College Dublin
martin.butler@ucd.ie

Peter Keenan
Department of Management Information Systems
University College Dublin
peter.keenan@ucd.ie

Geraldine O'Neill
Director, Centre for Teaching and Learning
University College Dublin
Geraldine.oneill@ucd.ie