An Architecture for integrating Adaptive Hypermedia Services with Open Learning Environments

Owen Conlan, Vincent Wade, Mark Gargan
Trinity College, Dublin,
Ireland.

+35316081335

{oconlan,vwade,garganm}@cs.tcd.ie

Cord Hockemeyer, Dietrich Albert
University of Graz,
Austria.

+433163808531

CHockemeyer@acm.org, das@wundt.uni-graz.at

ABSTRACT

Adaptive Hypermedia Systems are capable of delivering personalized learning content to learners across the WWW. Learning Environments provide interfaces and support services to aid tutors in course construction and aid learners in navigating those courses. However, most Learning Environments deliver content sourced from local repositories. This content tends not to offer adaptive features that Adaptive Hypermedia Systems are capable of delivering. Adaptive Hypermedia Systems, which are generally Web-based, could be viewed as personalized content services, with the capacity to deliver content to many Learning Environments. However, in order to co-ordinate the correct cooperation of the Adaptive Hypermedia Service and the Learning Environment knowledge and information models need to be exchanged, e.g. learner profile, assessment information, pedagogical constraints. There does not exist, however, a standardized mechanism for integrating Adaptive Hypermedia Services with Learning Environments.

In this paper the requirements for the generic integration of Adaptive Hypermedia Services with Open Learning Environments is explored. The main aim of such integration should be to work within a framework that allows for minimal impact on current Adaptive Hypermedia Services and Learning Environment implementations, while allowing for maximum standards-based interworking. This paper proposes and architecture and interface to support the collaboration of Adaptive Hypermedia Service and Learning Environments.

Categories and Subject Descriptors

D.2.2 [**Tools and Techniques**]: Modules and interfaces

General Terms

Human Factors, Standardization.

Keywords

Adaptive Hypermedia Services, Learning Environments, Integration, e-Learning, Content Interworking, Data Model

1. INTRODUCTION

In the current educational electronic landscape the most prevalent architecture for delivering content to a learner is via a Learning Environment. These environments (sometimes referred to Learner Management Systems or Courseware Support Systems [13]) holds all the content centrally and delivers it to the learner over the Internet, typically via web browser technology.

Developers and publishers of content must design their content for a particular Learning Environment (LE), restricting the reuse and marketability of that content. Typically this restriction is based on the local repository used by the LE or the content management services offered by the LE, e.g. TopClass, WebCT. Advances are being made in the packaging of content to facilitate reuse and integration of content into different Learning Environments (see Section 4.3 IMS Content Packaging), but these advances are concerned currently with static content. If developers of content wish to produce material with complex play rules or high levels of learner interaction they are restricted to the facilities provided by individual Learning Environments.

An alternative is to produce stand alone Adaptive Hypermedia Services (AHS) which are capable of providing these rich play rules and learner interaction. These services would reside as separate servers on the Internet. The benefit in integrating LEs and AHSs lies in combining the strengths of both systems. LEs provide administration and support facilities, while AHSs provide better quality education, personalization and pedagogical control to learners.

This paper explores how current open specifications may be used to achieve seamless integration between such AHSs and LEs. Section 2 discusses the Learning Environments requirements which an AHS should meet to integrate with the LE. In section 3 Adaptive Hypermedia Services and their information requirements are discussed. Section 4 deals with current standards and specifications which may be utilized and enhanced to enable integration between LEs and AHSs. Section 5 continues to propose an architecture for this integration. In section 6 the implementation of an example system is discussed. Section 7 provides a summary of the aspects discussed in the paper.

2. LEARNING ENVIRONMENTS

Most current Learning Environments [3] integrate and store all content to be delivered to learners into a proprietary data storage format. Publishers of content have to author their content for a particular LEs proprietary format or import it into that format. For many of the LEs the content is static, interspersed with assessments. Some LEs use the learner's performance in the assessment information as a control mechanism to determine the flow of the content. Few LEs, however, modify the flow of content through monitoring the learner's progress through non-assessment material, i.e. they do not generally take time factors or multiple visits to a piece of content into account. Lack of standardization on how these content flow mechanisms are represented leaves the publisher of content with a problem – they can either author content with complex play rules for a specific LE and risk making it unfeasible to export that content to other LEs or they can author static content that is easily exported between different LEs with no play rules.

This static approach to the content leads to a 'one size fits all' approach, where the learner's are all presented with the same course content and assessments. The main advantage that LEs offer over the classroom situation is that the learner can set a pace that best suits their learning. In academic and commercial learning situations it is rarely preferable that the learner decides the pace at which they digest the material fully, as constraints such as formal examinations, which tend to occur at fixed points in time, dictate when material needs to be learned by.

Some LEs offer facilities to include remote content, i.e. the content resides on a remote (probably HTTP) server. These LE do not, however, offer standards-based mechanisms to the remote server to request or pass information back to the LE. This facility is usually used to include static content from a remote server because for a remote server to provide adapted content requires integration with a proprietary communications protocol.

From a pedagogical perspective LEs do not offer many services to assist the learner in how they learn. They do tend to offer good management facilities to the course coordinator, such as –

- Learner and class management
- Course assembly and publishing
- Learner tracking across courses
- Summative assessment information

The Learning Environment, therefore, can offer learner management facilities to the Adaptive Hypermedia Services.

2.1 Requirements of the Learning Environment

2.1.1 Control Information

The Learning Environment needs to be informed of when the learner pauses their learning within the Adaptive Hypermedia Service. This control information may be used by the LE to determine whether the learner should be launched back into the AHS upon logging into the LE next time. Some LEs operate on a free navigation system, where the learner can browse to any section within the course and the course is only considered complete when the learner has received an assessment for each section.

LEs using a content management API (see 4.4 Content Interworking below), such as that in ADL SCORM v1.1 [15] have access to the required data model element cmi.core.lesson_status to indicate whether the content has been completed. The AHS can cater for learners that have completed the assessment aspects associated with their required objectives and received an assessment score from the AHS, but have not completed their optional objectives yet by setting the value of this data model element to incomplete (which is part of the restricted vocabulary for this element).

2.1.2 Assessment Information

Learning Environments tend to gather assessment information on the course section level. As different content may return assessment information in a variety of ways the LE often requires the assessment score to be normalized. The AHS may have to return assessment information in a particular format.

3. ADAPTIVE HYPERMEDIA SERVICES

Adaptive Hypermedia Services allow hypermedia content to be delivered to learners in an adaptive manner. In order to be called a service the AHS must facilitate ease of integration in order that learners are able to seamlessly launch, through any LE, and use the AHSs adaptive content. This process, as far as the learner's interaction with the LE is concerned, should appear no different from regular static content. It may be the case that the learner is studying content from both static and adaptive sources to achieve a learning objective. The AHS may, therefore, be called as part of a larger course.

3.1 Requirements of an Adaptive Hypermedia Service

AHS information requirements depend on the type of adaptivity it is providing and on whether it insists on gathering this information for itself. The AHS may require course objectives, learner information and assessment information from the Learning Environment in order to successfully produce content specific to the learner's and tutor's requirements.

3.1.1 Objectives

An Adaptive Hypermedia Service may be capable of delivering more content than the tutor requires the learner to study in order to achieve particular learning objectives within a curriculum. It is necessary that the AHS be able to determine what the required learning objectives are before generating a body of content for the learner.

Learners tend to prefer systems where they maintain control of their learning [10]. If learners have total control over their objectives they may ignore elements of the content that are required within the curriculum. To cater for both approaches the AHS could support two levels of objectives – required and optional. When the learner has completed (and been successfully assessed on) all of the required learning objectives set by the tutor they are considered to have completed the AHS courseware. They may have selected optional objectives which are of interest to them, or that they feel aid their learning. They may continue to interact with the AHS until all of these objectives are also completed.

This two-tiered approach to objectives gives the learner control of their learning, while assuring the tutor that they have, at least, coved the core objectives.

3.1.2 Learner Information

There is a large amount of information that could be passed between the Learning Environment and the Adaptive Hypermedia Service regarding the learner. The main difficulty in passing this information is to represent it in a way that all AHSs and LEs can understand. Candidate specifications include PAPI [12] and IMS Learner Information Packaging (LIP). As yet neither of these specifications deal with the representation of pedagogical aspects of the learner, such as learning styles, prior knowledge or life long learning goals (i.e. career path). Even if there was a recognized standard for passing this information there is no common vocabulary to which AHSs could adhere to process the information. Individual AHSs will however have their own mechanisms and vocabularies for storing pertinent learner information.

The primary requirement of the LE with respect to learner information is, therefore, that they can uniquely identify learners within the AHS. Globally unique identifiers of any nature is a non-trivial problem. The approach suggested here is similar to most ‘solutions’ to this problem – if the LE is running from a given URL and has internal identifiers for learners, then the AHS could use a combination of the LEs URL and internal learner identifier to produce an identifier for internal AHS use. This approach does have one major shortcoming – if the learner is enrolled in a number of LEs the AHS has no mechanism to recognize that the learner may have accessed it from another LE. IMS’s LIP Information Model [9] document discusses this issue concluding that the source of the information record is responsible for the uniqueness of the learner identifier and that the ‘uniqueness of the source (the LE) label is outside of the scope of this specification’.

Another issue with which the AHS may have to contend is whether the learner is already engaged in another course which also utilizes this AHS. If they are the AHS will need to differentiate between the two by asking the LE for a section identifier of some description. This will allow the AHS to use the correct set of objectives required and optional) for the learner.

If the LE cannot reasonably be asked to pass pedagogical information about the learner to the AHS then it is the responsibility of the AHS to pre-test the learner to acquire this information. This pre-test is used to determine the learners competencies (e.g. prior knowledge) and learning preferences (e.g. learning styles and display preferences). This is where the importance of identifying the learner comes to the fore. If we can identify the learner then if they re-enter the AHS we do not need to perform a full pre-test as some of the information determined in previous tests is domain independent, such as some learning preferences. Assessment information may also be used to aid the determining of the learner’s competencies.

3.1.3 Assessment Information

As mentioned above in 2.1 Requirements of the Learning Environment LEs generally require summative assessment information from each section of a course. In [14] one of the sixteen design guidelines for courseware is ‘specifying entry level learner competencies’ for the content. Within the Adaptive Hypermedia Service, however, there are no fixed competencies required as the AHS delivers content that best suits the learner’s competencies. The AHS must, first of all, determine these competencies.

The AHS can use assessment information to determine at what level to pitch the competency pre-testing of the learner. If the AHS can ask the LE for the assessment information of both the learner and their peers for sections of the course completed then it is possible for the AHS to determine how the learner is performing in relation to his peers and thus gauge the difficulty of the pre-test. Assessments should not be too difficult at the beginning of the course or they will discourage the learner [6]. It is also undesirable for prior knowledge assessments to be too easy for the learner as the assessing of the learners prior knowledge may take longer, thus increasing the time before the learner feels they are learning new (relevant) material.

The aim of using previous assessment information is to determine the point within a difficulty scale at which the pre-testing should beginning. At this point certain assumptions may be made about the learners prior knowledge. The task of the pre-test is to confirm these assumptions and determine if the learner has further (non-assumed) prior knowledge that is pertinent to their objectives and the content the AHS will be presenting. Ideally this start point should be positioned just below where the learner’s competencies actually lie, so as to confirm the learner possesses these competencies and determine where the boundary of their competencies lies.

4. STANDARDS AND SPECIFICATIONS

4.1 IMS Learner Information Packaging

The IMS Learner Information Packaging [9] specification address interoperability between internet-based learner information systems. Learner information concerns Learners (individual or group) or Producers (creators, providers or vendors). LIP includes facilities for the Learner to determine which aspects of their information are sharable with other systems. LIP has been designed with four basic requirements in mind –

- Distributed Information
- Scalability
- Privacy and Data Protection
- Flexibility and External References

The last requirement is described in [9] as Learner information includes many constructs, such as learning objectives and learning history, which are in practice represented by different structures in different contexts. Learner information data models must be flexible enough to accommodate this need. An external reference may, in the future, be used by both Learning Environments and Adaptive Hypermedia Services to share learning objectives and learning style information. IMS LIP v1.0 is currently available as a public draft.

4.2 PAPI

PAPI [12] is the IEEE Public and Private Information Specification which is a standard format for the representation and communication of student profiles. The purpose of the specification is to allow the creation of student records which can be communicated between educational systems over the lifetime of a learner.

The profile information for a learner is divided into four areas - Personal information which is for private consumption such as the student's name, address and Social Security Number; Preference information which may be for public consumption, such as the technology available to the student, the learning style of the student, physical limitations or disabilities. This information is collected with the cooperation of the student, i.e. it is negotiated; Performance information which is for consumption by technology. This consists of the observable behaviour of the student and may include grades, reports and logs.

The PAPI specification also incorporates the Dublin Core metadata element set. The information used to construct the user profile is inferred by the system, directly input by the user or is constructed by the user and system in collaboration. PAPI also intends to address the privacy and security issues involved in the storage and communication of user profile [3]

4.3 IMS Content Packaging

IMS Content Packaging is an interoperability specification to allow content creation tools, learning management systems and run-time environments to share content in a standardized set of structures. Version 1.1 of the specification is focused on defining interoperability between systems that wish to import, export, aggregate, and disaggregate packages of content [8].

The primary goal of the IMS Content Packaging specification is to provide a mechanism which, once implemented by producers and vendors, will allow content to be exported between systems with the minimum of effort. Version 1.1. of this specification is currently available as a public draft.

4.4 Content Interworking

The Content Interworking API was initially developed as part of the AICC CMI, more recently developed as part of ADL SCORM and the University for Industry Content Interworking Specification.

It is proposed that the IMS Content Management group will adopt the Content Interworking API as specified in the AICC CMI specification v3.0.1 and currently under implementation by the ADL [15] and by the University for Industry [16].

4.4.1 Aviation Industry CBT Committee Computer Managed Instruction Guidelines

The AICC guidelines [1] provide a method for seamless data flow between different computer based training (CBT) lessons and Computer Managed Instruction (CMI) systems, between different CMI systems, and from CBT lessons created with different authoring systems to a common data store and off-the-shelf analysis tools.

The driving force behind the development of the AICC CMI was to allow content developed by different authors to be used with any CMI system that supports the guidelines. To this end an API was designed that allowed the content to connect to the CMI system. The API principally facilitates the getting and setting of data in the CMI systems data model.

There are two aspects of the AICC approach to enabling interoperability of CMI systems with different CBT systems –

1. Lesson launch: The CMI should have a standard approach to CBT lesson initiation, and
2. Communication: The CMI should have a standard approach to providing information to the CBT lessons, and receiving information from the CBT lessons.

4.4.2 Advanced Distributed Learning SCORM

ADL SCORM [15] (Sharable Courseware Object Reference Model) is based on the AICC CMI guidelines. ADL was established with the purpose of developing a (US) Department of Defense wide strategy for using learning and information technologies to modernize education and training. Version 1 of SCORM defines a reference model to facilitate the interworking of Learning Management Systems (LMS) and content providers material. (LMS is used in the SCORM documentation in place of CMI). The SCORM is based directly on the runtime environment.

The ADL collaborated with AICC members and participants to develop a common Launch and API specification.

4.4.3 University for Industry Content Interworking Specification

A definition for content interworking between UFI [16] endorsed learning materials and the Learning Support Environment (LSE) system is necessary for –

- Consistent launching and running of learning content within LSE,
- Consistent gathering and reporting of assessment data back to LSE database for subsequent inclusion in the Lifelong Learning (LLL),
- Consistent storage of learner inserted bookmarks and annotations added to the content and subsequent display of bookmarks and annotations to learner when requested,
- Monitoring of the consumption and usage of the learning content.

Data is transferred between the content and the LSE via a JavaScript API based on the API for Web implementation of AICC/IEEE CMI standards.

4.4.4 IMS Content Management

The IMS Content Management [7] specification is currently under development. The goal of the IMS Content Management specification is to establish a standard for data interchange and communication between instructional content and run-time environments. This specification should provide cost-effective content interoperability for platform, tool and content developers when implementing the specification in their products.

The following data models have been identified as possible candidates for run-time support over the Content API (i.e. direct interaction between the executing content and the LMS) –

- Bookmarks
- Question and Test Interoperability (item tracking of learner responses)
- Assessment tracking (roll-up of scores and pass/fail status)
- Personal Information (enhance the ability to personalize - possibly found in the IMS Profiles specification)
- Navigation tracking (where, how often, how long etc. for post-analysis)
- Simulations (save settings to re-enter simulation at last point of exit)
- Adaptive Behavior (learning based upon individual attributes, preferences, learning styles etc.)
- Group Information (possibly found in Profiles spec)
- External Interfaces (pass a string to invoke some external app)

It is proposed that the Content Interworking API to be used within IMS Content Management will also be based on AICC SME guidelines and ADL SCORM.

5. ARCHITECTURE

The mechanisms required of an open architecture for integrating Adaptive Hypermedia Services with Learning Environments are standards-based approaches to –

- Importing the location of the AHS into the LE
- Launching the AHS from the LE
- Communicating between the AHS and LE

A commonly understood data model is also required to facilitate the communication of learner information, learning objectives and assessment information.

5.1 Importing the AHS location into the LE

IMS Content Packaging will provide a mechanism for static content to be integrated into a Learning Environment. By importing content as part of an IMS Content Packaging archive, with an XML manifest included, it is envisaged that the content can be automatically installed into the LEs content repository.

With Adaptive Hypermedia Services the content is not shipped in discrete units of material. The AHS is a remote service that the LE has an entry point to. As the LE, such as Microsoft's LRN [11], may already have an IMS Content Packaging import facility a reasonable approach would be to utilize this and enhance it to allow for the *importing* of the location of the AHS as part of an IMS Content Package. This could be imported into a LE in a similar way to static content. The manifest would describe the adaptive service and also contain a URL which the LE would use to launch the service.

Another consideration on the importing of the Content Package is that the learning objectives that the AHS is capable of achieving should also be imported into the LE. Provision should be given to the tutor to specify which learning objectives they would like the learners to achieve within the AHS.

5.2 Launching the AHS from the LE

When the Adaptive Hypermedia Service is launched from the Learning Environment the content it delivers will be coming form a different server than the Learning Environment. The AHS firstly initializes its connection with the LE. The AHS needs to then determine if the learner has been engaged in this course component before. If they have then it can rebuild the adaptive course component for the learner from a snapshot taken prior to the content being paused.

If the learner has not utilized this adaptive course component before then the AHS must determine two things – what the learning objectives are for the learner and whether the learner has been engaged in any course component offered by the AHS before. The learning objectives are used by the AHS to determine the required objectives for the content it is about to deliver. Optional objectives can also be offered based on the required objectives and the learner's prior knowledge. If the learner has used the AHS before from the launching LE this will alleviate the necessity to determine the learning style and possibly some prior knowledge aspects of the learner. It will still be necessary to pre-test the learner as their assumed prior knowledge may have degraded with time and/or they may have learned further relevant knowledge.

5.3 Communication between the AHS and the LE

Communication between the Adaptive Hypermedia Service and the Learning Environment can be achieved using the SCORM Runtime Communication API as used in SCORM v1.1. A subtle modification to the HTML frame layout is required at the AHS to enable calls to API functions residing on the LE from AHS content. The actual API calls used are the same as those used in SCORM v1.1 as the API is designed to get and set values that are separately defined by an external data model [15].

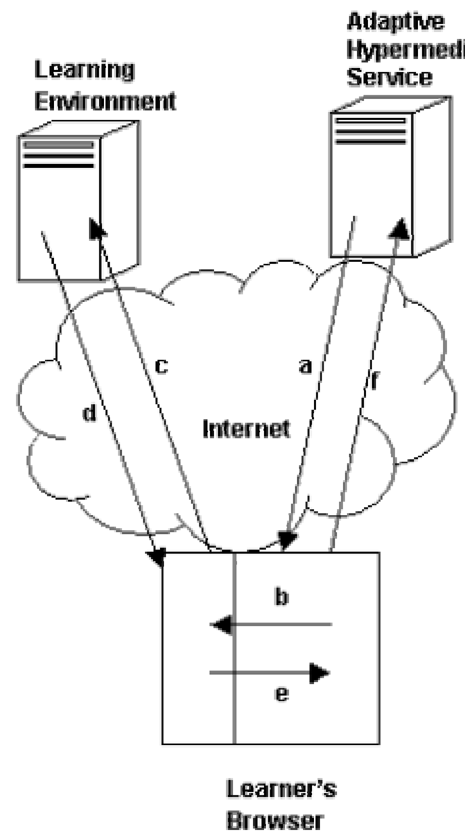

Figure 1. The AHS accessing the LE data model using the Content Interworking API

The remote AHS calls the Content Interworking API to access the data model on the LE using the following process –

a) The learning content (right browser frame) and JavaScript API (left browser frame, hidden) are delivered to the learner's browser.

b) An API function, in the left hand API frame, (e.g. LMSGetValue("cmi.core.leason_status")) is called from the content frame.

c) The API frame communicates the request to the Learning Environment.

d) The Learning Environment returns the value (in this case of cmi.core.leason_status) to the API Frame.

e) The function returns the value to content frame.

f) The value may be passed back to the Adaptive Hypermedia Service.

5.3.1 Common Data Model

The data model required for AHS/LE interaction is similar to that proposed in SCORM v1.1. The key requirements of the data model for this interaction are –

- Learning Environment identifier (to uniquely identify the LE, possibly it's URL).
- Learner identifier (to uniquely identify a learner from a LE).
- Section identifier (to determine, using the elements above, whether the student has attempted this content in the AHS before).
- Section status (completed, incomplete, attempted etc.)
- Section objectives (allows the AHS to query the LE for the learning objectives the tutor wishes learners to achieve).
- Score information (across all sections, for the learner and averages for all enrolled students).
- Score range (the LE may not use percentile scoring and the AHS may need to normalize the scores).

The number of SCORM data model elements has been significantly reduced between version 1.0 and version 1.1. This reduction was to aid harmonization between independent developments of the Content Interworking API (see 4.4 Content Interworking), i.e. many groups where using the same API, but had subtly different data models. Many of the optional elements defined in SCORM v1.0 had not been implemented by other consortia. A complete list of the data model changes between SCORM versions is available [15].

Most of the required data model elements for AHS/LE interaction already exist in some guise in the CMI data model used in SCORM v1.1. Learner identifier, section identifier, section status, section objectives and score range all have equivalents. The LE identifier and summative score information is not available within the data model.

6. IMPLEMENTATION

The mechanism for interfacing LEs and AHSs described in this paper has been implemented using Fretwell-Downing Education's LE [5] and Trinity College, Dublin's AHS [3] within the European Commission funded EASEL [4] project. The data model used is based on SCORM v1.1 [15] with the addition of LE identifier and summative score elements.

Using this model it is possible for the AHS and LE to interact, sharing information about the learner including, but not limited to –

- Learner Identification
- Interface Preferences
- Pedagogical Preferences
- Performance Information

The first piece of information that is requested by the AHS from the LE is the LE identifier. This should be unique, possibly URI based, and allows the AHS to contextualize all further communications. For example, when it asks for the learner's identification it might assume that this identifier is unique within the LE, but not across other Learning Environments.

Learner identification enables the AHS to ascertain if the learner has visited the service before and if they have determine whether it is necessary to pre-test them. For example, if the learner completed another course within the AHS already then it should have pedagogical information that is sufficient to adapt to the learners requirements and abilities. If the AHS has no information, or no appropriate information about the learner then it is necessary to pre-test.

Prior to administering the pre-test the AHS can query the LE for assessment scores for any material the learner has attempted in the course (assuming the AHS is a component of a larger course) and also request the assessment scores for the learner's peers in the course. These values are returned in as normalized values in a range specified by the LE. Using this information the AHS can determine the difficulty at which to set the pre-test, if it has multiple pre-tests aimed at increasing levels of knowledge. Such a pre-test can be used to determine the prior knowledge about a domain before generating the personalized course.

Other extensions that may be made to the LEs data model could cater for –

- Cultural Background
- Preferences & Learning Culture
- Communication Style and Needs
- Cognitive and Learning Style
- Prior Knowledge & Expertise
- Communication Style and Needs
- Learning History
- Objectives and Goals

The primary constraint to adding these complex elements within the data model is that of a commonly understood and accepted vocabulary to describe each extension. This is why the approach taken within EASEL was to allow the Adaptive Hypermedia Service to use a proprietary vocabulary internally, but to return assessment information.

Assessment information is returned to the LE using the normalized range mentioned above. This allows the AHS to inform the LE how the learner has performed in any assessments they have completed, but this is only a summative score for all assessments completed in the AHS. The IMS Question and Test specification was investigated as a candidate for richer communication of learner assessment results, but the uncertainty as to whether an individual assessment would be present in a personalized course meant the information for different learners might be inconsistent.

7. SUMMARY

This paper has described a mechanism for integrating Adaptive Hypermedia Services and Learning Environments and illustrated this approach with an implementation from the EASEL project. The mechanism described is based in current and emerging learning technology specifications and requires minimal modifications to LEs that utilize these specifications for their intended purposes.

By enabling AHSs to be successfully integrated into LEs this mechanism provides the potential for more learners to avail of the pedagogical benefits possible using customized adaptive content.

8. ACKNOWLEDGMENTS

This research is partially funded by the European Commission under the auspices of the EASEL (Educator Access to Services in the Electronic Landscape) [4] project. The EASEL projects goal is to explore technologies which can be brought together to offer course constructors an environment in which they can readily combine existing learning objects to create new online educational offerings. Current proprietary Adaptive Hypermedia Services tend to restrict this kind of integration. As part of EASEL the research conducted will be used to integrate Adaptive Hypermedia Systems into Learning Environments which are based on current WWW educational standards.

9. REFERENCES

[1] Aviation Industry CBT Committee (AICC) Computer Managed Instruction (CMI) Guidelines for Interoperability, Revision 3.0.1, Release 24 November 1999.

[2] Busilovsky, P., and Miller, P., Course Delivery Systems for the Virtual University, Access to Knowledge: New Information Technologies and the Emergence of the Virtual University. Amsterdam: Elsevier Science and International Association of Universities, 2001, 167-206.

[3] Conlan, O., and Wade, P., Novel components for supporting adaptivity in education systems - model-based integration approach, ACM Multimedia 2000, November 2000.

[4] EASEL, Educator Access to Services in the Electronic Landscape, IST Project 10051.

[5] Fretwell-Downing Education, http://www.fdgroup.com/fde/company/home.html.

[6] Gachuhi, D., and B. Matiru, Active Learning, Distance Education-By Design, Symposium 87 Papers. Alberta Correspondence School, Alberta, Canada, pp. 1-21, 1987.

[7] IMS Content Management, Version 0.1, June 2000

[8] IMS Content Packaging, Public Draft, Version 1.1, December 2000.

[9] IMS Learner Information Package, Public Draft, Version 1.0, December 2000.

[10] Laurilard, D. “Rethinking University Teaching: A Framework for the Effective Use of Educational Technology”, Routledge & Kegan Paul. , 1993.

[11] Microsoft Learning Resource Interchange (LRN), http://www.microsoft.com/elearn/support.asp.

[12] Public and Private Information (PAPI), Draft 6 Specification.

[13] Robson, R., Course Support Systems: The First Generation, International Journal of Telecommunications in Education 5(4), 271-282, 1999.

[14] Schaefermeyer, S., Standards for Instructional Computing Software Design and Development, Educational Technology, 1990.

[15] SCORM, Sharable Content Object Reference Model, Version 1.1, Advanced Distributed Learning (http://www.adlnet.org), January 2001.

[16] University for Industry, http://www.ufiltd.co.uk.