Towards a Standards-based Approach to e-Learning Personalization using Reusable Learning Objects

Owen Conlan, Declan Dagger, Vincent Wade
Trinity College, Dublin,
Ireland.

E-mail oconlan@cs.tcd.ie, daggerd@tcd.ie, vwade@cs.tcd.ie

Abstract: e-Learning systems that produce personalized course offerings for the learner are often expensive, both from a time and financial perspective, to develop and maintain. Learning content personalized to a learners' cognitive preferences has been shown to produce more effective learning, however many approaches to realizing this form of personalization involve considerable re-authoring of existing learning material. To reduce the implementation overhead on the content author and course designer it is imperative to facilitate the maximum reuse of learning resources.

This paper investigates a standards-based approach to delivering personalized learning content to learners using WWW technologies. Detailed is an implementation of a personalized e-Learning service, called the OPen Adaptive Learning Environment (OPAL), which is based on delivering content personalized to the learner's cognitive and presentation learning preferences using aggregation models based on ADL SCORM (ADL).

INTRODUCTION

Mechanisms that facilitate the reuse of learning content allow course authors to increase the life span of the content they develop. As these are often expensive and time consuming to develop the potential for any reuse should be maximized to achieve maximum return on investment. There are a number of factors that increase the potential reuse of learning content –

1. Granularity of the content
2. Effective and descriptive metadata
3. Appropriate packaging of the content and metadata for distribution

These factors, however, do not directly improve the pedagogical quality of the content produced. Learners have been shown to respond well to content and education systems that adapt to their personal preferences. The key to this process is an appropriate learner model constructed either explicitly through an online instrument or implicitly through the learner's interaction with the learning environment.

This paper details an approach to content authoring and packaging that enables the course author to create learning content that is both reusable and capable of supporting different pedagogical approaches through a content aggregation model. Section two details content issues such as packaging standards and granularity, section three shows how content packaging can support some approaches to adaptivity, section four discusses techniques for modeling the learner, section five covers the OPAL Environment and section six gives a summary of the paper.

PACKAGING LEARNING CONTENT

Packages of learning content and associated descriptive metadata are produced to facilitate the portability of learning content between different Learning Management Systems (LMS). The packages usually contain additional information about how the amalgamated learning content should be sequenced and additional metadata that describes the learning resources.

As the learning content, from which the content packages are built, may be stored in searchable repositories, with appropriate metadata, the course author may simply have to search for pre-authored material and combine it into a new offering. This is the model adopted by the EASEL IST project (EASEL) were a basic semantic registry and learning resource repositories were constructed to facilitate metadata searching across multiple schemas.

By re-sequencing the learning content in the package or by adding additional content or by adding content that supports different approaches to learning the course author may produce additional packages tailored to specific learning approaches and learner prior knowledge, while still supporting a common learning objective in the different course offerings. For example, the author may produce three packaged instances of a course, each tailored for a different level of prior knowledge from prospective learners. A simple knowledge pre-test instrument may be used to determine which package is the most appropriate for each individual learner.

Multiple content packages may be aggregated together to assemble an aggregated content package. This enables a learner's requirements to be met from more than one package.

Content Granularity

The size and scope of the pieces of learning content that are combined to form the content packages is a key consideration. For example, if a content package is comprised of only a few pieces of large grained learning content then re-sequencing them to form a new package that supports a different pedagogical approach may not be possible, as the new narrative flow may not make sense in the context of the learner's learning. This issue is paramount in the reuse of any learning object (Conlan et al. 2002).

To facilitate reuse learning objects should be fine-grained, paragraph or diagram size, for example. Content packages should be used where larger grained learning resources are required as they may be constructed using smaller learning resources and/or other content packages. Both IMS Content Packaging and ADL SCORM facilitate the assembly of content packages by combining other content packages or assets, such as in the case of SCOs.

With smaller grained learning objects the course author has greater flexibility in the creation of additional content packages. For example, if learning objects are available at the paragraph level then the course author can add/remove content at this level to produce tailored courses.

Candidate Standards

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 1.1.3 of the specification is focused on defining interoperability between systems that wish to import, export, aggregate, and disaggregate packages of content (IMS CP).

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 1.1.3 of this specification is currently available as a public draft.

ADL SCORM In 1997 the United States Department of Defense (DoD) and the Whitehouse Office for Science and Technology Policy (OSTP) set up the Advanced Distributed Learning (ADL) initiative to push development plans for standardization of Learning Resources. Their standardized approach to creating cross-compatible systems and content is encapsulated in their Sharable Content Object Reference Model (SCORM) (ADL). The SCORM constitutes three key components –

- The Content Aggregation Model (CAM).
- The Run-Time Environment (RTE).
- Content Packaging.

Through these components the ADL aims to meet the high-level requirements of –

- **Reusability**, content persistence over different LEs. A unified method of content markup.

- **Accessibility**, globally accessible content repositories. Metadata search facilities.
- **Durability**, learning resource and system component persistence over time.
- **Interoperability**, create systems and learning resources that are operation system and platform independent.

The purpose of the Content Aggregation Model (CAM) is to provide a common gateway for the composition of learning content from discoverable, reusable, sharable and interoperable sources (ADL). The CAM describes the Content Structure Format (CSF). The CSF allows standalone learning resources to be aggregated into either a complete course or a section of a course. Also described are mechanisms that facilitate the movement of content between different LMSs and repositories.

The three components described by the SCORM CAM for use in building reusable content are –

1. **Assets** A primitive representation of learning content, e.g. an applet, flash animation, image etc.
2. **SCO (Sharable Content Object)** Represents the finest grained learning resource traceable by the LMS e.g. a collection of one or more assets. A SCO should be system independent, i.e. should contain a compatible launch sequence persistent across multiple SCORM compliant LMSs.
3. **Content Aggregation** Involves the virtual mapping of SCOs together to form cohesive units of instruction that can be delivered to the learner through the SCORM Run-time Environment. (ADL)

Through the SCO (Sharable Content Object) markup the content may be described using an XML binding based on the IMS Learning Resource Metadata (IMS LRM) specification XML Binding. This specification has been extended as part of the EASEL IST Project (EASEL) with descriptive elements to facilitate the description of adaptive material (Conlan et al. 2001).

Describing the Content

The reuse of learning resources is aided considerably by appropriate, descriptive metadata accompanying the resource, enabling the content author to assemble new course offerings from existing content. If the metadata used to describe these resources is compliant with a known standard the reuse of the learning resources is extended beyond the bounds of the initial course delivery system.

The XML binding for extended IMS Learning Resource Metadata (Conlan et al. 2001) can be used to describe particular attributes of the content to which a system may adapt. For example a piece of content may include in its metadata –

```
<adaptivity>
        <adaptivitytype name=”learningstyle.vark.visual”>
                <candidate>4</candidate>
        </adaptivitytype>
        <adaptivitytype name=”learningstyle.vark.auditory”>
                <candidate>2</candidate>
        </adaptivitytype>
        …
</adaptivity>
```

This XML snippet describes a piece of content that rates quite highly on the visual scale of VARK (Fleming 1995),(Bruen & Conlan 2002). If the learner’s profile is built using a similar mechanism then the LMS will have a framework in which to compare the learner with candidate content.

ADAPTIVE TECHNIQUES

There are many adaptation effects that may be supported and facilitate using content packaging. Most of these techniques have their origins in Intelligent Tutoring Systems (ITS) and Adaptive Hypermedia Systems (AHS). The most effective uses of content packaging to provide adaptivity, however, involve situations where learning content re-sequencing or replacement produce course offerings that differ to large degree in pedagogical approach.

Curriculum Sequencing Curriculum sequencing is an adaptive technique that can produce very effective differences in how learners perceive learning content. Pedagogical theories such as Kolb/McCarthy place great emphasis on how content is sequenced and the ways different learners react to that learning content.

Content packaging allows a course author to produce several *versions* of the same course, each sequenced in a different manner. For example, one version of the course may start with a case study and derive the theoretical aspects of the material from that study. While another may start with the theory, present some examples and finish with a self-assessment quiz.

The effective production of these different versions assumes that the constituent parts of the packages are sufficiently small-grained that they may be re-sequenced. There may be a need to author some additional narrative content to facilitate the different routes through the content the individual content packages offer.

Contextualization When courses are being offered to corporate learners it is often desirable to use case studies and examples that pertain to those student. For example, if you are offering a programming course for accountants a course that uses accounting examples will be more familiar to the students. If the examples used are individual *grains* of content then that same programming course may be offered to civil engineers by simple changing the example learning objects for more appropriate examples. In this way content packages may be used to store several versions of the same course that differ only in context.

MODELING THE LEARNER

A learner/user model contains explicitly modeled assumptions that represent the characteristics of the student which are pertinent to the system. The system can consult the user model to adapt the performance of the system to each student's characteristics. User modeling allows the system to personalize the interaction between the student and the contents. To achieve effective learning this personalization should put the content in a context that the student can understand and to which they can relate. There are several techniques for modeling the student and refining this model.

The Stereotype Model Creating fixed stereotypes is one of the simplest ways of user modeling (Rich 1989) . New students are categorized and the system will customize its performance based on the category that has been set for the student. A common example would be the notion of novice, intermediate and expert users within a system. This approach is useful when a quick but not necessarily completely accurate assessment of the user's background knowledge is required (Kobsa 1993).

The Overlay Model The overlay model is widely used in the adaptive hypermedia systems in the educational domain. A model of the student's knowledge is constructed on a concept-by-concept basis and updated as the user progresses through the system. This allows for a flexible model of the student's knowledge for each topic (Brusilovsky 1996).

For this model the knowledge domain must be modularized into specific topics or concepts. The complexity of the model depends on the granularity of the structure of this domain knowledge and the granularity of the estimation of the student's knowledge. This estimation is build up by examining the sections the student has read and the test he has performed.

The Combination Model The Stereotype and Overlay techniques of user modeling are often combined in educational adaptive hypermedia systems. The student may be categorized by stereotype initially and then this model is gradually modified as the overlay model is built from information acquired from the student's interaction with the system.

There are a number of implicit approaches that may be used in acquiring and refining the user model. These include –

- The observation of the users direct-manipulative interaction with the software system.
- The analysis of the information which the user retrieves from a database or repository (Kass & Stadnyk 1992) .
- The system can also explicitly ask the user for information (Kobsa 1993) employing mechanisms such as questionnaires and tests.

Learner Model Standards and Specifications

Candidate Learner Model specifications include PAPI (PAPI) , IMS Learner Information Package (IMS LIP) and the GESTALT (GESTALT) user model. As yet none 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.

PAPI PAPI (PAPI) 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 behavior 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 (PAPI) .

IMS Learner Information Package The IMS Learner Information Package (IMS LIP) specification addresses 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 (IMS LIP) 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. IMS LIP v1.0 is currently available as a public draft.

GESTALT GESTALT (Getting Educational Systems Talking across Leading Edge Technologies) (GESTALT) which is funded by the ACTS project, is an educational environment for online learning which extends the IMS metadata definitions. A user profile is constructed from information acquired from the user by asking the user to complete forms displayed by a wizard. The user model is created as an XML document which is then stored physically on the user's machine. The profile information stored includes the user's educational history and technology available to the student.

Some of the profile details acquired include personal details, contact details, qualification details, skill details, learning preferences and mode of delivery. The learning preference is stored as a boolean value of 'Yes' or 'No'. The user profile is created as an XML document that will be stored physically on the user's machine.

Modeling Cognitive Preferences

VARK (Fleming 1995) This technique of learner modeling uses a concept of multi-modal learning strategies, where the student being modeled does not have an outstanding preferred mode of learning but instead uses a combination of two, three or even four of the styles.

For example, one student may prefer a combination of the visual aspect and the kinesthetic aspect, to produce a comfortable learning experience for them. Another may prefer an aural, read-write combination for their learning environment. In rare cases students being modeled score evenly across all four spectrums.

Kolb/McCarthy Kolb's learning style model categorizes learners on two continuums, *abstract/concrete* and *active/reflective*, based on personal preferences for the consumption of new information. It suggests that the learner should cycle through all four styles in order to gain a full understanding of the topic.

The overview of this model is to present the learner firstly with their preferred style of learning, based on the two continuums and then cycle through the other 3 modes.

Bernice McCarthy has applied the original model set out by Kolb to categorize learners in accordance to the different cycles as follows.

1. Innovative Learners: *concrete/reflective.*

   This learner prefers to be shown the practical applications of the new material being presented.

2. Analytic Learners: *abstract/reflective.*

   This learner prefers to be presented with the results of research and well ordered sequential presentation of new material

3. Common Sense Learners: *abstract/active*

   This learner is a "try it and see" learner. Provide guided activities and keep them updated on where they are in the content.

4. Dynamic Learners: *concrete/active*

   This learner prefers the ability to explore other sources of information on the relative topic, hyperlinks to those resources.

THE OPAL ENVIRONMENT

The design approach of the OPAL (OPen Adaptive Learning) Environment architecture follows, in many ways, that outlined in the SCORM documentation specification. This approach has a central *repository* of content remotely stored on a database. The job of the Learning Management System is to access this data, display it to the learner in a coherent manner, and track their progress through the material.

System Components

Figure 1, below, shows the main components of the OPAL environment. The functionality of these components is implemented using twelve Java Servlets, through which the learner interacts with the system. The OPAL Environment is an LMS supporting learning resources that are descriptively tagged by SCORM metadata. The OPAL environment uses SCOs as the basic unit of content storage. The environment has some additional functionality that enables it to perform adaptivity at the level of content packages and at the content delivery stage by employing stretch-text type methods to selectively hide or remove individual types of content.

This mechanism is implemented by the key servlets –

- **AHSQuestionnaire** This servlet queries the learner, by delivering an appropriate online instrument to them and assembles the returned answers from which the learner profile is built.
- **LMSAddNewLearner** This servlet is responsible for authenticating the learner and selecting appropriate SCOs (content packages) to deliver to the learner as a content aggregation model (CAM).
- **LMSCourseServlet** This servlet accesses the CAM and determines how the SCOs should be assembled and the first SCO to be delivered to the learner.
- **LMSDisplaySCO** The LMSDisplaySCO servlet is responsible for the delivery of the content from the different SCOs in the CAM to the learner.

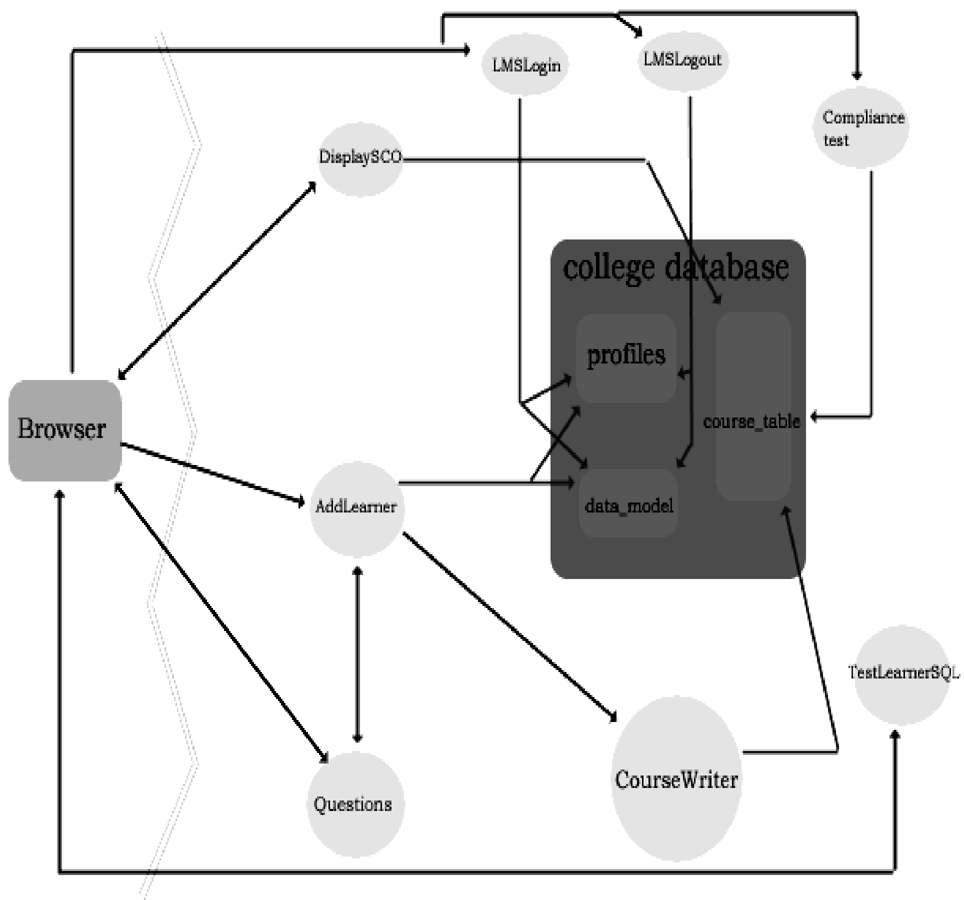

Figure 1: System Components

Building the Learner Profile

The AHSQuestionnaire servlet is responsible for building the learner model and adding it (using the LMSAddNewLearner servlet) to the profiles repository in the system. The model is determined explicitly by querying the learner. For example, in the case of the VARK (Fleming 1995) model the learner is asked 36 simple questions to build a profile of their visual, auditory, read/write and kinesthetic attributes. Once the questions are completed the results are returned to the LMSAddNewLearner servlet where the values are interpreted. This interpreted profile is stored in the profiles repository.

Building the Personalized Course

There exists only a finite number of SCOs in the content repository and it is the role of the LMSCourseServlet to assemble the SCOs selected by the LMSAddNewLearner servlet into a course offering tailored to the learner's profile with a personalized navigation structure. A selection process is initialized based on the learners' profile. This process induces an aggregation model for the learner based on their existing profile. The learning resources' metadata could also be used in this process. This metadata may be described in the manner section 2.3 Describing the Content. Once the servlet has identified the appropriate aggregation model it assembles the content, ready for delivery. This course structure is persistently stored in the course_table in the LMS repository.

Delivering the Personalized Course

It is the role of the LMSDisplaySCO servlet to display a personalized navigation structure to the learner around the selected SCOs. This servlet presents the learner with a simple next/previous navigation buttons for changing pages within the SCO as well as textual links to those pages. Also provided to the learner are links to other sections (other SCOs) and to a course index. This navigation structure is built by examining the structure of the SCO and its constituent learning content.

SUMMARY

This paper describes an approach to developing personalized e-Learning content using content that is reusable both in an adaptive and static manner. The key to this reuse is appropriate metadata and the use of content packages.

Also described was the OPAL Environment as an implementation of an adaptive e-Learning system based on SCOs. As in most Adaptive Hypermedia Systems the cornerstone of the personalization process is the building of an appropriate learner profile. This profile is induced to a selection process to assemble a personalized course for the learner.

REFERENCES

1. ADL. "ADL Sharable Content Object Reference Model", Version 1.2 , http://www.adlnet.org/.
2. Bruen, C.; Conlan, O. (2002): Dynamic Adaptive ICT Support for learning Styles – A Development Framework for re-useable learning resources for different learning styles & requirements. ITTE 2002, Annual Conference of the Association of Information Technology for Teacher Education.
3. Brusilovsky, P. (1996): Methods and techniques of adaptive hypermedia. In P. Brusilovsky and J. Vassileva (eds.), Spec. Iss. on Adaptive Hypertext and Hypermedia, User Modeling and User-Adapted Interaction 6 (2-3), 87-129.
4. Conlan, O., Hockemeyer, C., Lefrere, P., Wade, V., & Albert, D. (2001): Extending educational metadata schemas to describe adaptive learning resources. In Hugh In Proceedings of the twelfth ACM Conference on Hypertext and Hypermedia (Hypertext 2001), pp. 161-162, New York: Association of Computing Machinery (ACM), 2001.
5. Conlan, O.; Wade, V.; Bruen, C.; Gargan, M. (2002): Multi-Model, Metadata Driven Approach to Adaptive Hypermedia Services for Personalized eLearning. In proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2002), P100-P111.
6. EASEL. Educator Access to Services in the Electronic Landscape, IST Project 10051.
7. Fleming, N.D. (1995): I'm different; not dumb. Modes of presentation (VARK) in the tertiary classroom, in Zelmer, A., (Ed.) Research and Development in Higher Education, Proceedings of the 1995 Annual Conference of the Higher Education and Research Development Society of Australasia (HERDSA), HERDSA, Volume 18, pp. 308 – 313.
8. GESTALT. "Getting Educational Systems Talking Across Leading Edge Technologies", Work Package 3, D0301 Design and Specification of the RDS (Resource Discovery Service), 1999.
9. IMS CP. IMS Content Packaging, Public Draft, Version 1.1, December 2000.
10. IMS LIP. IMS Learner Information Package, Public Draft, Version 1.0, December 2000.
11. IMS LRM. IMS Learning Resource Metadata, Version 1.2, http://www.imsproject.com/metadata/.
12. Kass, R. and Stadnyk, I. (1992): Using User Models to Improve Organizational Information. In Proceedings of the 3rd International Workshop on User Modeling. 1992, Dagstuhl, Germany.
13. Kobsa, A. (1993). User modeling: Recent Work, Prospects and Hazards. In Adaptive User Interfaces: Principles and Practice, M. Schneider-Hufschmidt, T. Kühme, and U. Malinowski, (eds.). 1993, North-Holland: Amsterdam.
14. PAPI. Public and Private Information (PAPI), Draft 6 Specification.
15. Rich, E. (1989): Stereotypes and User Modeling. In A. Kobsa and W. Wahlster (eds.): User Models in Dialog Systems, pp. 35-51. Springer, Berlin, Heidelberg.