Building an Online Course Based on the E-Learning Standards:
Guidelines, Issues, and Challenges

Permanand Mohan

Author

Permanand Mohan (pmohan@tstt.net.tt) is a Lecturer in the Department of Mathematics and Computer Science at the University of the West Indies in St. Augustine, Trinidad and Tobago. Correspondence concerning this article should be addressed to: Permanand Mohan, Department of Mathematics and Computer Science, The University of the West Indies, St. Augustine, Trinidad and Tobago.

Abstract

This paper provides a comprehensive set of guidelines for building an online course based on the e-learning standards. It discusses the steps that should be followed to build a digital repository that facilitates the storage, retrieval, and reuse of the learning resources that comprise an online course in a standard way. The paper also examines various shortcomings associated with adopting the e-learning standards that threaten the viability of widespread reuse of learning resources. Finally, the paper highlights research challenges that must be surmounted in order to gain the benefits of reusable digital learning resources.

Résumé: L'article présente une suite exhaustive de directives qui portent sur la création d'un cours en ligne en fonction des normes de l'apprentissage en ligne. Il présente les étapes qui doivent être suivies pour créer une logithèque numérique qui facilite le stockage, le repérage et la réutilisation des ressources pédagogiques que l'on trouve normalement dans un cours en ligne. L'article examine aussi les diverses lacunes associées à l'adoption de normes d'apprentissage en ligne qui mettent en péril la viabilité de la réutilisation globale de ressources pédagogiques. En dernier lieu, l'article met en évidence les défis en matière de recherche qui doivent être relevés afin de tirer profit des ressources d'apprentissage numériques réutilisables.

Introduction

Over the past five years, there has been tremendous interest worldwide in the concept of a reusable digital learning resource, usually referred to as a *learning object* (Wiley, 2001; Downes, 2001; Littlejohn, 2003). A *learning object* can be defined as an independent and self-standing unit of learning content that is predisposed to reuse in multiple instructional contexts (Polsani, 2003). According to proponents of the learning objects' approach, there are numerous benefits that accrue when learning objects created by others are reused and

re-purposed in new instructional situations. Presumably, this would lead to a considerable reduction in the time and effort taken to produce new content, compared to developing the content from scratch. The reused content may even be of a higher quality than if developed from scratch, similar to well-designed and well-tested software components.

To obtain the desired benefits of reusability, there are numerous issues that must be addressed. First of all, learning objects must be designed with a view to promoting reuse and re-purposing them in different instructional contexts. This involves consideration of a number of design issues such as instructional design and context. Second, it is important for learning objects to be standardized to facilitate sharing by different individuals and organizations. Standards are required for packaging learning objects so that they can be used in different computer platforms and learning systems without modification. Standards are also required for cataloging learning objects using a common vocabulary so that they can be located on the Web by potential users. Finally, standards are required for facilitating the storage, discovery, and retrieval of large numbers of learning objects stored in digital repositories.

Because of the importance of standards for achieving the expected benefits of reusability, there have been a number of efforts worldwide aimed at developing standards to facilitate different aspects of learning object reuse. An important standardization effort is content packaging. The Content Packaging (CP) specification developed by the IMS Global Consortium (IMS, 2003a) is intended to facilitate the interoperability of digital learning resources in different Learning Content Management Systems (LCMSs) (Greenberg, 2002). The IEEE 1484.12.1-2002 standard for Learning Object Metadata (LOM) (Learning Technology Standards Committee [LTSC], 2004) specifies standardized *metadata* on each learning object and is intended to facilitate the discovery, management, and exchange of learning objects by learners, instructors, and automated systems over the Web. Another important standard is the IMS Digital Repositories Interoperability (DRI) specification (IMS, 2003b). DRI is intended to facilitate the interoperability of learning object repositories, enabling users worldwide to access learning object repositories in a uniform manner.

The "Creating the Standardization Content" section of this paper provides practical guidelines on using the CP specification and LOM-based metadata to create interoperable packages of learning objects at different levels of granularity. It also discusses difficulties associated with applying the guidelines and presents solutions to these problems. Theoretically, any system (e.g., LCMS) compliant with the specification can then import a content package and extract the learning objects stored in that package. Practically, this is not always possible, and this section presents the results of an experiment that attempted to import learning objects packaged with the CP specification into a popular commercial LCMS.

Next the "Creating a Standardization Content Repository" section of this paper provides various guidelines for creating a digital repository to store learning objects that could be made available to others. In particular, it explains how the DRI specification can be used to facilitate the storage and retrieval of learning resources in a standardized manner. This section also gives an overview of the current state-of-the-art with learning object repositories, touching upon both existing systems and research prototypes.

Despite the volume of the literature on the concept of a reusable learning object, this author is not familiar with any study that validates on a wide scale the benefits of the learning objects' approach. Even with the presence of a significant amount of e-learning standards for learning objects, it is yet to be shown how reusability can be achieved in an effective and practical manner. The "Issues and Challenges" section delves into possible reasons for this state of affairs and identifies various challenges that have to be surmounted before the benefits of the learning objects' approach can be obtained. This paper concludes by discussing various research directions that can be explored to tackle these challenges.

Creating Standardized Content
Structure and Semantic Markup of Content

Learning objects can exist at different levels of granularity or aggregation. At the lowest level of the granularity hierarchy are *assets,* which are normally single files such as an image, some text, or a video or audio clip. The next level of granularity occurs when assets are aggregated into some meaningful structure called an *information object*. Structured information objects can then be aggregated into coarser-grained components, and the process repeated again and again to produce increasingly coarse-grained learning objects (Duncan, 2003). The objects at each level of aggregation can all be considered learning objects, although they may correspond to the educational terms of topics, lessons, modules, or courses. Figure 1 shows a learning object hierarchy that can be used as a basis for creating increasingly granular learning objects. A similar aggregation hierarchy is used in the Shareable Content Object Reference Model (SCORM) (ADL Technical Team, 2004).
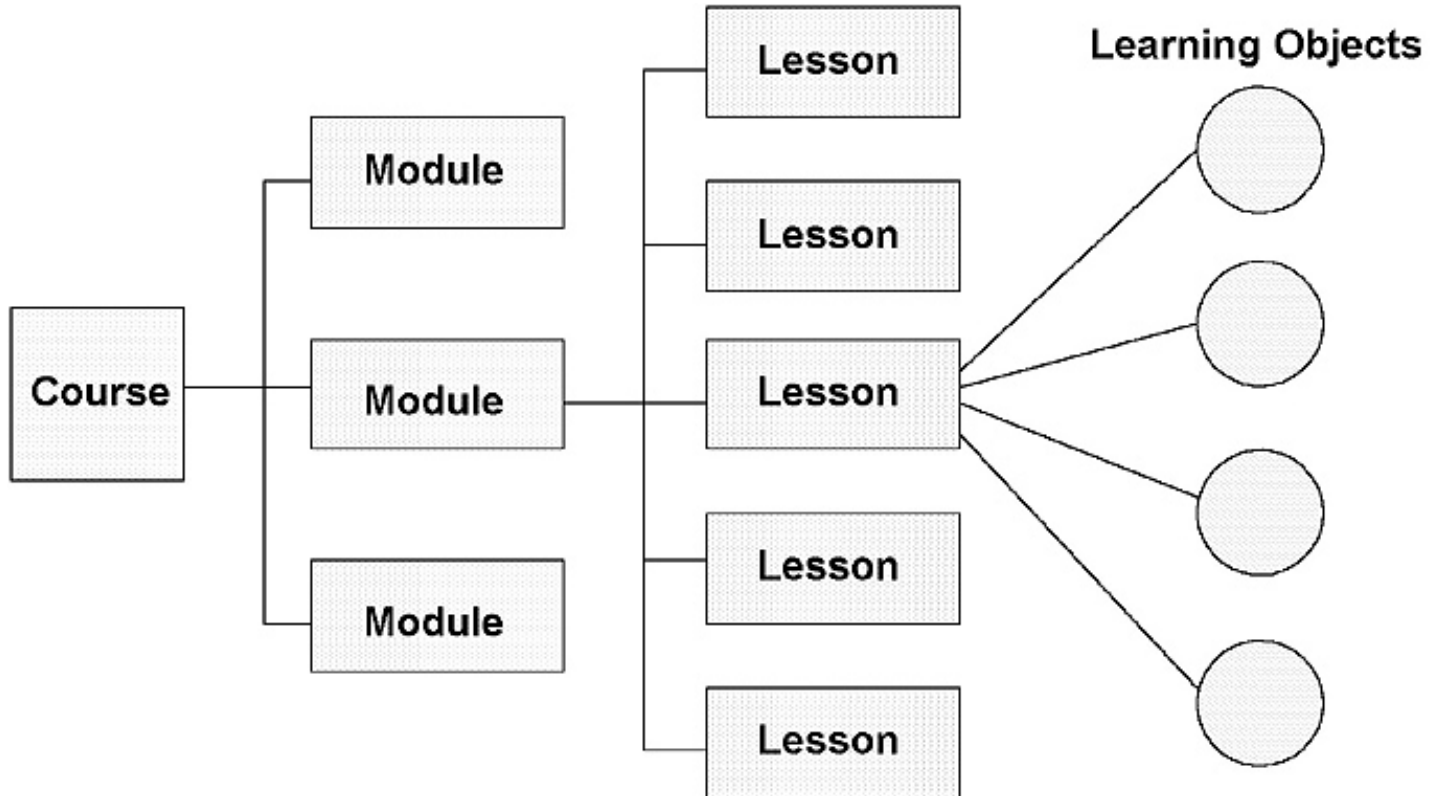

Figure 1: Learning Object Hierarchy

An important aspect of learning object development is designing the physical content

structures to impart knowledge and skills of a domain. These structures need to be organized in different ways in different disciplines. For example, an element named *Algorithm* would be appropriate for a computer science learning object, but not a chemistry-learning object. Clearly, a wide range of markup tags is required to define content in different disciplines. If the content of a learning object is specified using standard markup tags, an intelligent computer-based tutoring system can attach semantic meaning to its structure and display the different components of the learning objects according to the present needs. For example, an LCMS may choose not to display certain elements given the preferences of certain learners, or it may choose to render another element in a special way, given certain learning characteristics of the learner.

Standard markup languages also allow the possibility of dynamic replacement of content in certain parts of a learning object, during an instructional interaction. For example, an assessment element within a learning object can be replaced with an assessment element belonging to a similar learning object, or with a learning object that provides a more detailed assessment of the learning objective(s). Indeed, the assessment element can be skipped altogether. Thus, a standard markup language provides the opportunity to reuse more fine-grained components of a learning object, enabling a wider variety of learning object re-purposing.

Süss (2000) developed a language called the Learning Material Markup Language (LMML) for structuring the content of learning objects (which he calls the *conceptual document structure*). In the LMML, inheritance hierarchies are used to create markup languages for different disciplines such as computer science, music, and finance. While the different disciplines have their own domain-specific structure, these are ultimately derived from a common conceptual and modular structure of knowledge. The LMML provides an XML binding for learning objects, enabling content for the different disciplines to be validated against the binding.

The computer science tags used in the LMML include tags such as *Algorithm*, *Code*, *Listing*, *Block*, and *Line*. Higher up the LMML hierarchy are tags such as *Conclusion*, *Remark*, *Definition*, *Example*, and *Exercise*. These tags are applicable to many domains and can thus be shared among many disciplines. For example, they can be automatically used to markup computer science learning objects as well as learning objects for tutoring music. The object-oriented structuring of tags in LMML is an elegant way to specify and reuse tags in different domains.

One notable standardization effort for marking up the content of learning objects is the IMS Question and Test Interoperability (QTI) Specification. This specification allows multiple-choice and other similar questions to be specified in a standard way using XML, enabling consistent rendering of test items by different computer-based tutoring systems. The QTI had it origins in the Tutorial Markup Language (Browning, Williams, Brickley, & Missou, 1997), a superset of HTML that was designed to separate the semantic content of a question from its screen layout or formatting.

Creating the Content Packages

An IMS content package consists of two major parts, a special XML file called the *manifest*that describes the content and organization of the package, and the actual physical files that make up the package. Each instance of the manifest contains the XML elements described in Table 1.

Table 1:

XML Elements in Content Package

Element Name	Description
Metadata	Describes the manifest as a whole
Organizations	Describes zero or more organizations of the content within the manifest
Resources	Contains references to all the actual learning resources and media elements needed for the manifest
(sub)Manifest	One or more optional, logically nested manifests

The IMS CP specification can be used to package all the learning objects at the different levels of aggregation. Starting at the bottom of the hierarchy shown in Figure 1, a folder must be created to store all the resources (XML content files, style sheets, image files, audio files, video files, etc.) that comprise a learning object. After developing the physical resources, they must be placed in this folder and a manifest created. Finally, the resources that comprise the learning object and its associated manifest must then be zipped together to create the interoperable archive. This process must be repeated for each leaf-level learning object.

While it is fairly straightforward to use a text editor to manually create the manifest file based on the CP specification and then zip the resources together, the process is quite time-consuming and prone to errors. An open source tool known as *Reload* provides a graphical editor which can be used to create the hierarchical structure of the content package and specify the resources that correspond to each item in the content package. The editor also allows the specification of metadata for the different components of the content package.

After creating the leaf-level learning objects, it is then necessary to aggregate these learning objects into *Lessons*, the next higher level learning objects in the hierarchy. To facilitate the aggregation, it is necessary to create a higher-level folder representing a lesson. This folder must physically contain all the folders corresponding to the leaf-level learning objects. In Reload, a lesson learning object (i.e., content package) can be created by specifying its component learning objects as sub-manifests. This requires the manifests representing the

leaf-level learning objects to be physically copied into the manifest being created for the lesson.

It should be noted that the current specification does not allow a nested manifest to be included by means of an external reference. Thus, a sub-manifest must be physically inserted into the higher-level manifest, resulting in increasingly bigger files at successive levels of aggregation. This shortcoming may be alleviated in a future version of the CP specification by being able to refer to external elements using a technology known as XML Inclusions (XInclude) (World Wide Web consortium, 2004).

It should also be mentioned that the learning objects developed for the leaf-level nodes in the hierarchy must be context-free. However, a lesson necessarily comes with a context. This context can be provided by developing separate content that provides a context to "glue" the other learning objects in place in a lesson—or even in a module. Contextual content can be created as normal Web content in XML files. However, it is not considered to be learning objects, since it is highly specific to a particular learning situation, reducing the likelihood of heavy reuse (South & Monson, 2000).

The approach to keep the folders of the contained learning objects as sub-folders of the lesson folder makes it easy to create the zipped archive representing the lesson as a whole since the entire contents of the lesson folder are simply zipped together. A system or person getting access to a lesson learning object then has various options. For example, the lesson learning object can be used in its original form without change. However, the lesson learning object can be easily dis-aggregated into its constituent learning objects, since each of them is a learning object in its own right, and each is accompanied by its own manifest file. Thus, coarse-grained learning objects can be dis-aggregated and re-purposed according to the current needs. After creating the content packages representing lessons, the lesson learning objects are then aggregated into module learning objects using the same process of aggregation. Finally, the module learning objects are aggregated into a content package representing the entire course.

When creating content packages, it is difficult to make changes to a higher-level learning object (e.g., module) and have the changes propagate downwards to the lowest level learning objects and vice versa. Using the preceding guidelines, the manifests of nested packages are physically inserted into the higher level manifests. This requires the manifests of aggregate learning objects to be manually modified when changes are required (such as insertion, deletion, or reorganization of learning objects). The folders containing the physical learning objects also have to be manually modified when changes are made since there is an exact mapping between the manifest structure and the folders containing the learning resources. This is an avenue for future research and will involve consideration of versioning issues.

Metadata Standards

Reusable learning objects in a digital repository are useless if they cannot be discovered and subsequently reused and re-purposed by other content producers and instructional

designers. The discovery and management of learning objects in a repository can be considerably simplified by providing descriptive information on each learning object. This information is called *metadata* and facilitates the search, evaluation, acquisition, and use of learning objects by learners, instructors, or automated systems. However, metadata by itself is useless in promoting reusability. To be effective, metadata needs to be standardized internationally and adapted locally. Recognizing the importance of standardized metadata for describing a wide array of resources across the Internet, the Dublin Core Metadata Initiative (DCMI, 2003) developed a "core" set of elements to facilitate the discovery and retrieval of networked information. There are fifteen elements in the Dublin Core Metadata Element Set (DCMES): *Contributor, Coverage, Creator, Date, Description, Format, Identifier, Language, Publisher, Relation, Rights, Source, Subject, Title*, and *Type*. Each element is defined using a set of ten elements from the ISO 11179 standard for the description of data elements.

The most important metadata standard for learning objects is the IEEE 1484.12.1-2002 Standard for Learning Object Metadata (LOM). LOM uses 86 data elements to describe a learning object and these are grouped into nine categories: *General, LifeCycle, Meta-Metadata, Technical, Educational, Rights, Relation, Annotation*, and *Classification*. DC-Ed is yet another metadata specification for learning objects. DC-Ed is a set of extensions to the DCMES—it even contains three elements from LOM, and is geared towards more general kinds of digital learning resources, such as those that were not educationally purposed from inception (Sutton & Mason, 2001).

In order to describe the learning objects in a repository, there are thus several choices of metadata schema. However, given the standardization of the LOM, this is the best choice for describing learning resources designed from inception with an educational purpose in mind. DC-Ed can be used for other types of learning resources or for those situations when only a basic set of descriptors is needed. The following fragment of XML code gives an example of a partial LOM metadata record where the elements closely resemble those found in a library catalog card.

```
<lom>
<general>
      <title>
            Design Patterns: Elements of Reusable Object-Oriented Software
      </title>
      <catalogentry>
            <catalog>ISBN</catalog>
            <entry>0-201-63361-2</entry>
      </catalogentry>
      <language>en</language>
      <keyword>Object-Oriented</keyword>
      <keyword>Programming</keyword>
      <keyword>Computer Software</keyword>
      <keyword>Reusability</keyword>
</general>
</lom>
```

The CP Specification allows metadata to be inserted into each content package, allowing metadata for a learning object to travel wherever the learning object travels. Thus, if someone locates a particular learning object in a repository, metadata for the different components in the package can be extracted and examined. This enables the different components of the content package to be reused and re-purposed as necessary. However, it is also important for metadata to be stored separately from the physical learning objects to facilitate searching and management of the learning objects in a repository. Indeed, it is common for repositories to only store metadata records and the metadata refers to the location of the physical learning object (this type of repository is really a "referratory"). An example is shown below. In order to access the actual learning object, a potential user of the learning object would have to visit the URL specified in the *location* element.

```
<technical>
    <location>http://www.lor.org/inherit.zip</location>
</technical>
```

Regardless of where the actual learning objects are stored, it is important for a repository to provide a set of services to allow its metadata to be queried, and to allow the metadata to be easily changed by individuals authorized to do so. Moreover, it is desirable for repositories to provide a common set of external services so that potential users of learning objects can direct queries in the same way to different repositories. This is important if learning object repositories were to become widely available in the years to come. Since these repositories would be created and maintained by different organizations and individuals, there is a clear need for interoperability.

Describing Content Packages With Metadata

The LOM standard can be used to describe content packages of learning objects with metadata. It is also possible to use "application profiles" such as the Canadian Core Learning Resource Metadata Application Profile (CanCore, 2004) that are based on the LOM. The Reload tool provides an interface to capture metadata elements belonging to LOM Standard. To use another application profile, the necessary files defining the profile are placed in the Reload metadata helper directory. If the user chooses this profile before entering the metadata, Reload automatically generates an interface to capture data for only the elements defined in the chosen profile. Once the metadata is entered, it is associated with the content package currently open in the editor. However, metadata can be also specified for the items within the package itself.

A bottom-up procedure should be used for entering metadata, since coarse-grained content packages are created out of finer-grained packages. First, metadata should be entered for the lowest level content packages. Next, lesson content packages are built out of these low-level learning objects, and metadata for each lesson is specified. The modules for the course are then assembled out of the lessons, and metadata for each module is entered. Finally, metadata is specified for the course as a whole, after it is assembled from modules.

Experiments with WebCT®

After developing a set of learning resources conforming to the e-learning standards, it is important to determine the extent to which these learning resources could interoperate with other compliant LCMSs. An experiment was conducted with a set of learning resources that were assembled together using Reload based on the CP specification. The experiment involved importing the learning resources into WebCT® Campus Edition 4.1. WebCT® is a commercial LCMS (Greenberg, 2002) that is widely used in tertiary institutions around the world. According to the documentation, WebCT® should be able to import content packages that are compliant with the IMS CP Specification 1.2. Several of the content packages created with Reload were first uploaded to the WebCT® system and then imported using the *Import* option. A control group consisting of the example content packages supplied by IMS with the 1.2 specification was also used to compare results. In particular, the *All_Elements* content package was uploaded.

When the *All_Elements* content package was uploaded to WebCT®, the error "Import failed" was generated, and a number of warnings were listed. One of these warnings was that the package type could not be determined. On closer examination of the other warnings, it appeared that WebCT® was unable to parse the two sets of metadata tags in the manifest file of the content package. These were both deleted and the import process was then repeated. Another set of errors was generated and the *All_Elements* content package still could not be imported into WebCT®. However, the original list of warnings was no longer displayed.

An attempt was then made to import the content packages created with Reload into WebCT®. However a similar set of errors occurred. The cause of these errors is not clear and no further tests with WebCT® were conducted. However, the documentation seems to suggest that only content packages created by the Microsoft LRN tool and Microsoft

PowerPoint can be imported into WebCT® and that packages created manually will not work. As a final experiment, the *All_Elements* content package and the Reload learning objects were tested for conformance with the IMS Content Packaging Schema Version 1.2. Both sets of content packages tested positively for conformance.

Creating a Standardized Digital Repository
Standards for Interoperability

The IMS Digital Repositories Interoperability (DRI) Specification can be used to facilitate the interoperability of learning object repositories and thus enable users (individuals and software agents) worldwide to access repositories containing digital learning resources in a uniform manner. The specification recognizes two different repository types: one reflecting established practice for repository interoperability and the other representing repositories that are able to implement the XQuery and SOAP-based recommendations put forward in the specification. The first type of repository includes those that utilize the Z39.50 standard and the Open Archives Initiative-Protocol for Metadata Harvesting (OAI-PMH), and is not discussed here because of space limitations.

The second type of repository makes heavy use of XQuery and SOAP to achieve interoperability. The W3C XQuery language is still in the working draft stage. It uses the structure of XML to express queries across all kinds of XML data sources. SOAP is the Simple Object Access Protocol (SOAP) and is a W3C Recommendation. It provides a simple and lightweight mechanism for exchanging structured and typed information between peers in a decentralized, distributed environment using XML.

To understand how a DRI-compliant repository works, consider an XML client application that receives a request from a user to find the set of all learning resources with the keyword "Reusability". The XML client converts this request into an XQuery statement based on the LOM standard. The XQuery statement for the given example would be as follows:

```
collection('metadata')//general [keyword='Reusability']/..
```

When this statement is executed, the query engine examines the *general* element of each metadata record in the collection maintained by the repository. It then checks the *keyword*sub-element for equality with the desired string. If the test succeeds, the parent of the*general* element is returned. This corresponds to the *lom* element representing the entire metadata record. The result of this query will therefore be all the metadata records where one of the keywords is equal to "Reusability". Of course, the query can be easily re-stated to return a subset of the information in the metadata record.

After formatting the query in XQuery, the XML client uses the SOAP protocol to send the query to one or more DRI-compliant repositories. The following fragment of code shows what the SOAP message may actually look like.

```xml
<?xml version='1.0' ?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
    <env:Header>
        <imsmsg:header xmlns:imsmsg= />
    </env:Header>
    <env:Body>
        <imsmsg:msgType>Search</imsmsg:msgType>
        <imsmsg:XQuery>
        collection('metadata')//general [keyword='Reusability']/..
        </imsmsg:XQuery>
    </env:Body>
</env:Envelope>
```

When a repository receives the SOAP message, it extracts the XQuery statement from the *Body* element, executes the query against its database of metadata records, and returns the results packaged in a similar SOAP message. Similarly, when the XML client receives the message, it extracts the results from the *Body* element of the SOAP message and presents the information to the user. Using the metadata, the user can determine the general suitability of the learning object and may then choose to view the actual learning objects.

Implementation

The technical aspects of creating a repository to store digital learning resources are fairly straightforward. A hostname has to be obtained for uniquely identifying the computer in which the metadata records are to be stored. The metadata records can be stored in a relational database with an XML mapping provided between the XQuery statements and the relational tables and associated SQL statements. Alternatively, the XML metadata records can be stored natively in an XML database. The repository must also provide a set of services to facilitate querying and management of the metadata records. Essentially, these services will accept SOAP requests and perform the desired operations.

The content packages representing the learning objects must be stored somewhere and referred to by the metadata records. The packages can be stored in a directory structure in the repository itself or elsewhere. In order for a learning object to have a unique reference on the Internet, a URI can be used. However, problems of maintaining the metadata records can arise if a learning object should subsequently change its location. Previous links to the learning object also become invalid. To this end, a Digital Object Identifier (DOI, 2004) can be used. A DOI is a permanent, globally unique identifier. It is usually maintained by a DOI registry service, which charges a small fee to maintain the physical addresses pointed to by a DOI. The DOI approach is currently being adopted by the large publishing houses (which supported its development in the first place) and by digital libraries such as the ACM Digital Library.

Current Status of Digital Repositories for Learning Resources

Over the past few years, several different types of digital repositories have been developed for storing learning objects. Several of these have been research prototypes and some have actually had real-world deployments. None of these repositories is DRI-compliant since the specification was finalized after most of them were developed. Global repositories are based on the client-server approach and usually maintain links to learning resources stored elsewhere on the Web. A number of global repositories for learning objects appeared over the past few years, but some are no longer being actively supported, perhaps indicating the cessation of project funding. These include TeleCampus (2004), the Campus Alberta Repository of Learning Objects (CAREO, 2004), and the Multimedia Educational Resource for Learning and Online Teaching (MERLOT, 2004). The first two repositories use the LOM standard to catalogue learning resources. One common feature of these repositories is that they do not physically store the learning objects. They are essentially global catalogs for learning objects that are available at different levels of granularity.

Telecampus is no longer actively maintained, but at one time had over 66,000 courses and programs listed (mostly harvested from Web sites). At the other extreme, CAREO lists only about 3800 learning objects. MERLOT sits in-between with about 9300 learning objects. The growth rate of each repository has been less than 25% over the past year. Concerning the issue of cost, most of the resources listed by CAREO and MERLOT are freely available. However, more than 80% of the learning objects in TeleCampus were available on a commercial basis. It is likely that the increased use of learning object repositories will also depend on how issues such as payment, copyright restrictions and trust are dealt with in a global environment.

Other kinds of repositories that have developed as research prototypes include peer-to-peer (P2P) repositories such as the Portal for Online Objects for Learning (POOL) (Richards & Hatala, 2002), Edutella (Nejdl et al., 2002), and LOMster (Ternier, Duval, & Neven, 2003). Yet another type of repository is one that is based on a *brokerage* service such as the EducaNext portal, which is powered by the Universal Brokerage Platform (Simon, 2003).

Issues and Challenges

The preceding section mentioned a number of digital repositories being used around the world to store learning objects. However, while both commercial and non-commercial learning object repositories continue to be developed, one property shared by most of them is the lack of a critical mass of learning objects (Ternier, Duval, & Neven, 2003). Indeed, the examples given in the previous section suggest a very slow growth rate for learning objects, given the intense interest generated in the field over the past few years. One suggestion is that the problem is due partly to the time-consuming procedure of filling in many metadata fields in a repository (Neven, Cardinaels, Duval, Ternier, & Vandepitte, 2003). This view is supported by Richards, McGreal, and Friesen (2002) who argue that the full set of 86 elements in LOM is not suited to direct implementation since it entails a huge classification effort.

The main goal of the LOM standard is to facilitate the discovery and reuse of learning objects in new instructional situations. However, several authors have pointed out that the absence of a critical set of information in the LOM is likely to hamper the reuse and re-purposing of learning objects (Wiley, 2001; Mohan & Brooks, 2003; Farance, 2003). For example, Farance argues that it is difficult to assign values to several data elements in a consistent manner due to their imprecise definitions (e.g., *Aggregation Level*, *Interactivity Level*, *Semantic Density*, and *Context*), giving doubts as to their usefulness.

In an attempt to deal with the problems mentioned above with the LOM, several "application profiles" such as CanCore and SingCore have been developed. CanCore contains only 36 elements from LOM that are considered essential for promoting the discovery and reuse of learning objects (Richards, McGreal, & Friesen, 2002). The problem can also be tackled by automatically generating as many metadata fields as possible before data entry by a human user (Neven et al., 2003). However, the problem of populating digital repositories with learning objects may also be related to the lack of a reward system for developing learning objects and entering the metadata, and it is suggested that an extended timeframe is required for widespread sharing and adoption of learning objects (Koppi & Lavitt, 2003).

Even if a critical mass of learning objects were to be stored in a learning object repository, it is unlikely that current metadata schemes would be expressive enough to facilitate the kind of sophisticated searching likely to be performed by an instructional designer. For example, consider how an online catalog of a library is used. To search the catalog, a user enters values in various metadata fields such as title, subject, and author. The library system searches the catalog and returns the records satisfying the query. On many occasions, the user will not have prior knowledge of the items returned by the query. A trip to the library and manual perusal of the items in question often helps the user to determine if an item is suitable for his/her needs. If a digital repository returns a list containing hundreds of learning objects that satisfy a query, it is quite impractical to examine each one individually to determine if it is appropriate.

From a pedagogical perspective, a significant amount of information is required to support the reuse of learning objects in new instructional contexts. This information includes association of a learning object with an appropriate body of knowledge (or domain conceptualization), specifying the learning objectives and learning outcomes of a learning object, and indicating the teaching and learning strategies employed by the learning object (Mohan, 2004). However, the current metadata specifications do not include elements to directly specify such information, reducing the potential for reusing learning objects.

An important aspect of the learning object approach is that content producers can develop high quality learning objects and make them available at a fraction of the development cost to others. This probably explains why a lot of the early interest in learning objects was sparked by individuals from the e-learning industry. However, Wiley (2003) points out that every major content creation industry has seen its core product line exploited and freely traded on the Internet in recent years. Because of this, he believes that an "educational object economy" in which large amounts of commercial content are available for purchase

will never materialize. Even if learning objects were freely available, there is the related problem of allowing users to make changes to intellectual property. If this is not controlled, individuals may obtain a learning object, make a few minor changes and then upload the learning object to another repository, claiming ownership. Given the point previously made about a reward system, it seems that uncontrolled access to even freely available learning objects is not in the best interests of learning object producers.

Conclusion and Future Research

This paper provided a set of guidelines for creating reusable, interoperable learning objects based on the CP specification and the LOM standard. It carefully described the steps to be followed and pinpointed some of the limitations of the approach. Tools are now starting to appear to support the process, and we can expect more sophisticated tools for learning objects to be developed in the near future. However, current commercial support for the kinds of learning objects mentioned in this paper is weak, given the experiences reported with WebCT®. The paper also discussed the process of creating a digital repository of learning objects based on the DRI specification and highlighted some current repository efforts around the world. Despite the amount of attention that reusable learning objects have received, it is surprising that successful reuse of learning objects on a wide scale is yet to be reported. The paper suggested reasons for this and identified some challenges to the learning objects' approach.

There are several research avenues that can be investigated to tackle some of the challenges pointed out in the paper. The first one of these is the automatic generation of metadata. Metadata is both objective and subjective. The *Size* of a learning object in bytes is objective metadata. The *TypicalAgeRange* of a learning object is subjective metadata. It should be a relatively straightforward matter to automatically generate the objective metadata for a learning object, particularly those elements belonging to the *Technical* and *LifeCycle* categories. It should also be fairly easy to generate the *Language* and *AggregationLevel* values for a set of learning objects. For example, if the language for a course learning object is English, it is highly likely that the language for its contained learning objects (modules, lessons, etc.) will also be English.

Recognizing the limitations of the current e-learning standards, research projects are underway to develop sophisticated mechanisms to promote the reuse of learning objects. Much of this research involves applying principles from Artificial Intelligence and the Semantic Web (Berners-Lee, Hendler, & Lassila, 2001). For example, a major research project is being undertaken by a consortium of six universities in Canada, known as the Network of Learning Objects Repositories (LORNET, 2004). One sub-project within LORNET is the investigation of intelligent techniques for representing a learning object on the Semantic Web. Though in its early stage, this sub-project treats a learning object as an agent that is capable of interacting with other learning object agents and adapting itself as necessary to suit the needs of individual learners. Since the reuse and aggregation of learning objects is based on agent negotiation, the limitations of the e-learning standards are downplayed.

References

ADL Technical Team. (2004). *Sharable content object reference model (SCORM) Content aggregation model (CAM) Version 1.3.1*. Retrieved September 21, 2004, from http://www.adlnet.org/screens/shares/dsp_displayfile.cfm?fileid=994

Berners-Lee, T., Hendler, J., & Lassila, O. (2001, May). The semantic Web. *Scientific American, 284*(5), 34-43.

Browning, P., Williams, J., Brickley, D., & Missou, H. (1997). *Question delivery over the Web using TML*. Retrieved September 21, 2004, from the University of Bristol, Professional Development Web site: http://www.lboro.ac.uk/service/fli/flicaa/conf97/browning.html

CanCore. (2003). *Canadian core learning resource metadata application profile*.http://www.cancore.ca/indexen.html

CAREO. (2004). *Campus Alberta repository of learning objects*. Retrieved September 21, 2004, from www.careo.org

Dublin Core Metadata Initiative. (2003). *Dublin core metadata element set, Version 1.1: Reference description*. Retrieved September 21, 2004 fromhttp://dublincore.org/documents/dces/

Digital Object Identifier System. (2004). *Welcome to the digital object identifier system*. Retrieved September 21, 2004, from www.doi.org

Downes, S. (2001). Learning objects: resources for distance education worldwide.*International Review of Research in Open and Distance Learning*, *2*(1).

Duncan, C. (2003). Granularization. In A. Littlejohn (Ed.), *Reusing online resources: A sustainable approach to e-learning*. London: Kogan Page.

Farance, F. (2003, January). IEEE LOM standard not yet ready for "prime time". *IEEE Learning Technology Newsletter, 5*(1), 21-23. Retrieved September 23, 2004, fromhttp://lttf.ieee.org/learn_tech/issues/january2003/learn_tech_january2003.pdf

Greenberg, L. (2002, December 9). LMS and LCMS: What's the difference? *Learning Circuits*, Retrieved September 10, 2004, from www.learningcircuits.org/2002/dec2002/greenberg.htm

IMS. (2003a, June). *IMS content packaging best practices guide: Version 1.1.3 final specification*. IMS Global Learning Consortium, Inc. Retrieved September 21, 2004, fromhttp://www.imsglobal.org/content/packaging/cpv1p1p3/imscp_bestv1p1p3.html

IMS. (2003b, January). *IMS Digital Repositories Interoperability core functions best practices guide: Version 1.0 final specification*. IMS Global Learning Consortium, Inc. Retrieved September 21, 2004, fromhttp://www.imsglobal.org/digitalrepositories/driv1p0/imsdri_bestv1p0.html

Koppi, T., & Lavitt, N. (2003, June). *Institutional use of learning objects three years on: Lessons learned and future directions.* Paper presented at the Learning Objects Symposium

2003, Honolulu, HI, USA. Retrieved September 10, 2004, fromhttp://www.cs.kuleuven.ac.be/~erikd/PRES/2003/LO2003/Koppi.pdf

Littlejohn, A. (2003). (Ed.) *Reusing online resources: A sustainable approach to e-learning*. London: Kogan Page.

LORNET. (2004). Learning Object Repositories Network. www.lornet.org

Learning Technology Standards Committee. (2004). IEEE Learning Technology Standards Committee. Retrieved September 1, 2004, from http://ltsc.ieee.org

MERLOT. (2004). *Multimedia educational resource for learning and online teaching*. Retrieved September 1, 2004, from www.merlot.org

Mohan, P. (2004, July). Design issues for building reusable digital learning resources.*Proceedings of the International Conference on Education and Information Systems: Technologies and applications* (pp. 171-176)*.* Orlando, FL, USA.

Mohan, P., & Brooks, C. (2003, July). Engineering a future for learning objects. *Proceedings of the International Conference on Web Engineering* (pp. 120-123)*.* Oviedo, Spain.

Nejdl, W., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmer, M., & Risch, T. (2002, May). *EDUTELLA: A P2P networking infrastructure based on RDF*. Retrieved September 21, 2004 from http://www2002.org/CDROM/refereed/597/index.html

Neven, F., Duval, E., Ternier, S., Cardinaels, K., & Vandepitte, P. (2003, July). An open and flexible indexation- and query tool for Ariadne. *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications 2003* (pp. 107-114). Honolulu, HI.

Polsani, P. (2003). Use and abuse of reusable learning objects. *Journal of Digital Information, 3*(4). Retrieved September 23, 2004, fromhttp://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani/

Richards, G.R., & Hatala, M. (2002, January). POOL, POND and SPLASH: A peer to peer architecture for learning object repositories. *Internet 2 Conference, Workshop on Peer to Peer Applications in Higher Education*, Phoenix, AZ. Retrieved September 21, 2004 fromhttp://www.edusplash.net/upload/INET2-RichardsandHatalav2.doc

Richards, G.R., McGreal, R., & Freisen, N. (2002, June). Learning object repository technologies for telelearning: The evolution of POOL and CanCore. *Proceedings of theInforming Science + IT Education Conference*, Cork, Ireland, 176-182. Retrieved September 10, 2004 from http://www.edusplash.net/upload/Richa242Learn.pdf

Simon, B. (2003, June). Learning object brokerage: How to make it happen. Proceedings of the *World Conference on Educational Multimedia, Hypermedia and Telecommunications 2003*(1), 681-688. Retrieved September 20, 2004 from http://dl.aace.org/12846

South, J. B., & Monson, D. W. (2000). A university-wide system for creating, capturing, and delivering learning objects. In D. A. Wiley (Ed.), *The instructional use of learning objects*(chap. 4.2).Retrieved September 10, 2004, from<u>http://reusability.org/read/chapters/south.doc</u>

Süss, C. (2000). Adaptive knowledge management: a meta-modeling approach and its binding to XML. In: H. J. Klein (Ed.), *12th GI-Workshop Grundlagen von Datenbanken, Plön*, TR 2005. Retrieved September 21, 2004, from the Christian-Albrechts-Universität Kiel, Germany Web site: <u>http://xml.coverpages.org/suessS00-2000.pdf</u>

Sutton, S.A., & Mason, J. (2001, October). The Dublin core and metadata for educational resources. *International Conference on Dublin Core and Metadata Applications* (pp. 25-31). Tokyo, Japan: National Institute of Informatics. Retrieved September 10, 2004 from<u>http://www.nii.ac.jp/dc2001/proceedings/product/paper-04.pdf</u>

TeleCampus. (2004). TeleCampus Online Course Directory. <u>http://courses.telecampus.edu/</u>

Ternier, S., Duval, E., & Neven, F. (2003, June). Using a P2P architecture to provide interoperability between learning object repositories. *World Conference on Educational Multimedia, Hypermedia and Telecommunications 2003*(1), 148-151. Retrieved September 10, 2004, from <u>http://dl.aace.org/12724</u>

Wiley, D. A. (2001). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (Ed.), *The instructional use of learning objects*(chap. 1.1)*.* Retrieved September 10, 2004, from<u>http://reusability.org/read/chapters/wiley.doc</u>

Wiley, D. A. (2003). *Learning objects: Difficulties and opportunities*. Retrieved September 10, 2004, from wiley.ed.usu.edu/docs/lo_do.pdf

World Wide Web Consortium. (2004). *XML Inclusions (XInclude) Version 1.0*. Retrieved September 21, 2004 from <u>http://www.w3.org/TR/xinclude/</u>