Using Conceptual Lattices to Represent Fine Granular Learning Objects through SCORM Meta-Objects

Luciano Silva, Pollyana N. Mustaro, Denise Stringhini and Ismar F. Silveira
Mackenzie Presbyterian University, Brazil
lucianosilva@mackenzie.br
polly@mackenzie.br
dstring@mackenzie.br
ismar@mackenzie.br

Abstract: Ideally, learning resources should be built over a shared pool of fine reusable granular learning objects. However, in order to avoid contextual lacks, dynamic creation of such resources would mostly rely on the conceptual relationships among learning objects inside a repository. These conceptual relationships, as well as the learning objects creation, are best established if students' learning styles are considered. Common standards like Sharable Content Object Reference Model (SCORM) do not have tools to provide conceptual relationships among fine granular learning objects. This paper presents a conceptual lattice-based architecture for using SCORM to provide an effective mapping of conceptual relationships among learning objects.

Keywords: learning objects, conceptual lattices, SCORM, granularity.

1. Introduction

Learning objects (LOs) constitute any digital entity that can be used, reused or even referenced during a technology-mediated educational process (IEEE-LOM, n.d.; Millar, 2003; Wiley, 2000). The proposition of learning objects configures the possibilities reusability, manageability, interoperability, and accessibility (South and Monson, 2000). Granularity becomes a key concept for LOs when they are designed to be reused in different contexts. LO granularity is related to the level of detail contained in a LO, as well as its size, decomposability and potential of reuse. Finer levels of granularity are desirable since the smaller an LO is, the more likely it is to be reused in different contexts. In addition, learning object repositories constitute resource providers that contain and manage LOs in order to make them referable, accessible and distributed in large scale. This mechanism can be extended and become richer through the establishment of conceptual modelling of relationships which could include learning styles (Curry, 1987). Learning styles describe categorisation ranges that take into account the ways people naturally – and unconsciously – perceive information and build knowledge through them. However, the implementation of the presented characteristics constitutes a challenge that involves different proposals of metadata standards, such as SCORM (Sharable Content Object Reference Model) (ADL, n.d.), IMS (Instructional Management Systems) [10], Dublin Core (DCMI, 2004) or LOM (Learning Object Meta-Data) (IEEE-LOM, n.d.).

Currently, the major scheme used and supported by LORs is SCORM (ADL, n.d.). This standard tends to be globally accepted, considering the current support it receives by learning objects repositories. This standard is based on three basic documents: the Content Aggregation Model, the Run-Time Environment and the Sequence and Navigation Model. The Content Aggregation Model (CAM) allows quantitative and qualitative annotations about learning objects. The Run-Time Environment (RTE) defines the operational environment that is necessary for the object execution. The Sequencing and Navigation (SN) model defines a linear order for the exhibition of LOs. Particularly, the SN document is largely discussed in literature, especially considering the Activity Trees limitations. Basically, the Activity Trees are responsible for the navigation among objects. The navigation scheme defined by these trees is interpreted by the Learning Management System (LMS) in a traversal order. The current scheme could actually restrict the apprentice navigation through the objects. The navigation pathway of this model is limited because it does not consider the necessities of learning personalisation, especially the differences related to how apprentices interact or answer to a specific learning environment. According to this, LORs need to consider the representation of learning styles diversity to enhance the effectiveness of educational processes with ICTs (Information and Communication Technologies). This paper proposes a navigational scheme among SCORM objects based on the Conceptual Lattices Theory and Learning Styles concepts using a dynamical graph navigation transformation. This proposal includes the introduction of annotations and links with semantic structure via XLink (Wilde and

Reference this paper as:
Silva L et al (2006) "Using Conceptual Lattices to Represent Fine Granular Learning Objects through SCORM Meta-Objects" *The Electronic Journal of e-Learning* Volume 4 Issue 2, pp 141 - 148, available online at www.ejel.org

Lowe, 2002) technology within the SCORM objects. This allows a personalised and non-linear treatment for point-to-point navigations between objects.

2. Granularity of learning objects

Firstly, we consider the following fact: the usual process to develop digital learning processes results in large monolithic content. This content is hard to be reused because it lacks granularity. On the other hand, such content could be described as a well-structured, highly reusable, low-coupled learning objects set that could be arranged in order to provide a more adaptive, learner-centred content. This could be explained by some essential elements: first of all, digital learning content is often planned in an ad hoc way, since its content is too much problem-specific, being driven to a given knowledge domain. Besides, such development often uses tools and techniques that usually do not separate content from presentation. Learning objects' development usually includes a variety of tasks and procedures, such as instructional and hypermedia design, text analysis and production. Furthermore, there are other tasks to be performed, like course authoring, software tools development, content integration and evaluation, training and establishment of a lifelong computer-mediated learning organisational culture. Defining a reusable architecture for more effective learning objects retrieval would noticeably diminish costs related to the development of new courses, thus contributing to make the task of creating new learning objects faster through reuse.

Besides, relationships among LOs must be equally ubiquitous through different levels of granularity. These relationships must be transparent, being kept away from the courses developers' point of view. These requirements could be fulfilled by applying techniques like conceptual lattices (Davey and Priestley, 2002), whose precise building and navigation relies on psychological and educational theories, like Kolb's Learning Styles Theory (1984).

3. Conceptual lattices

The theory of partially ordered sets and lattices (Davey and Priestley, 2002) has been successfully applied to the modelling of hierarchical systems and has produced many contributions in several computational areas as Artificial Intelligence, Category Theory, Semantics of Programming Languages and Concurrency Theory. The concept definition involves a complex philosophical question. However, a concept is formally determined by an extension and an intension. All objects belonging to a concept form an extension and an intension is an attribute set shared by such objects. Normally, the enumeration of all objects and attributes related to a concept represents a hard, sometimes impossible task. Thus, in many practical cases, the set of objects and attributes is restricted to discrete and finite ones. In order to clarify the preceding definitions, let us consider a context to learning concepts about the Solar System:

Table 1. Context adapted from Davey and Pristley (2002) for Learning of Solar System. Objects are formed by Planets and attributes are related to Astronomical Observations of size, distance from sun and moon's Presence/Absence.

Planet	Size	Distance from Sun	Does it have a moon?
Mercury	small	near	no
Venus	small	near	no
Earth	small	near	yes
Mars	small	near	yes
Jupiter	big	far	yes
Saturn	big	far	yes
Uranus	medium	far	yes
Neptune	medium	far	yes
Pluto	medium	far	yes

Conceptual lattices are better visualised using a Hasse Diagram, a directed graph that exhibits order-covering properties through a hierarchical diagram. Theoretically, two Hasse Diagrams would be necessary in order to represent the possible orderings: using objects and attributes. Figure 1 depicts objects and attributes related to lattices from a Solar System context, without edge orientation:

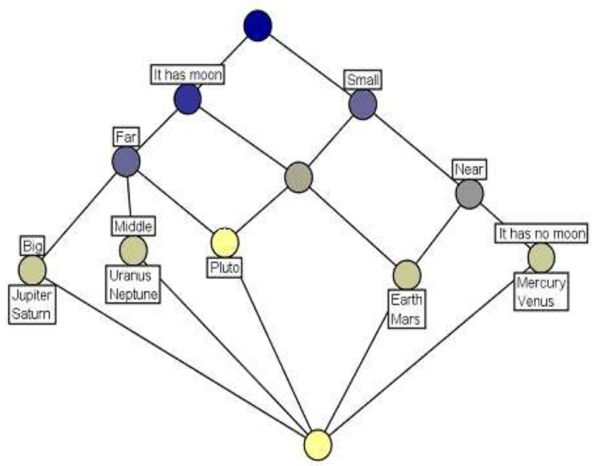

Figure 1: Combined Hasse Diagram related to Solar System context. Annotations below nodes

represent object subsets, while those above indicate attribute subsets.

The combined diagram allows navigation in both vertical directions: ascendant and descendant. The ascendant navigation starts on lattice lowermost and it allows an easy way of obtain objects. For instance, in the preceding diagram the lowermost has no related object (empty set) while all the planets arise in the next order level. The central node, even without annotation, inherits objects belonging to the level immediately below (Earth, Mars and Pluto), obeying the inclusion ordering. The lattice utmost in the ascendant navigation contains, by inheritance, all objects in the context.

The descendant navigation allows the easy attainment of attributes. In the lowest part of the navigation we have an empty set again since all objects in the context share no common attribute subset. The central node has no annotation as in the ascendant navigation, but using inheritance we conclude that it is small and owns a moon. It is not difficult to see that the utmost of descendant navigation presents all attributes in the context.

4. Learning styles

According to Cognitive Psychology, learning styles represent the individual preferred ways of perceiving and processing information, which are the responses to educational stimuli (Alonso, 1993). In this case, as Curry (1987) has pointed, the analytical diagnosis of learning styles considers theoretical models that emphasise the preferences related to instructional context, information process, social interaction or even individual personality. David Kolb (1984) developed a proposal that considers this framework. His proposal of learning styles representation is based on a bi-dimensional scale that results in four categories: converger, assimilator, diverger and accommodator.

The converger learner is an inquirer and has a better performance in situations that involve a correct response, problem solving, decision-making, and deductive hypothetical reasoning. The assimilator apprentice presents inductive reasoning and is supposed to develop theoretical models from multiple observations and analyse situation from different points of view. The diverger student is an observer that uses reflection and feeling to construct models and presents high performance in applications that involve lateral thought. Finally, the accommodator learner appreciates the activities that include creativity, autonomy, and presents high degrees of immediate adaptability. Table 1 summarises the key characteristics of Kolb´s classification:

Table2: Kolb’s learning style characterisation.

Learning Style	Learning Characteristics	
Converger	Abstract Concept	Active Experimenta-tion
Assimilator	Abstract Concept	Reflexive Observation
Diverger	Concrete Experience	Reflexive Observation
Accommo-dator	Concrete Experience	Active Experimenta-tion

These elements collaborate to organise, structure or select the learning objects in a LOR that are more appropriate for a given learning context and offer subsidies for the establishment of educational approaches. Besides, it is possible to use instructional design elements to constitute a systemic educational architecture. Considering different theories, heuristics and methods for analysis and definition of guidelines from learning styles that personalise learning experience (Mustaro et al., 2006). From combining learning styles characteristics and instructional design framework it is possible to establish didactical proposals for learners. In the converger case, the activities can be associated to problem solving situations that present a single answer or choice the best solution from structured tasks that consider errors as elements of learning process. For the assimilator apprentice, one perspective includes development of exercises that involve logic steps and theoretical model for problem solving. The diverger learner is characterised by his questioning behaviour, being able to establish relationships between content, previous experiences, etc. when manipulating a case study, for example. At last but not least, the accommodator student is creative and presents interest for working in solving of real problems or role-playing. In this case accommodator learners can transpose scene characteristics and generalise them in other contexts. To analyse this scenario and determine a person’s learning style it is necessary to use the Kolb instrument of evaluation (Learning Style Inventory – LSI), which is based on identification of apprentices’ perceptual and processing preferences.

With these pieces of information, it is possible to model LORs. However, the computer applications related to identifications of learning styles is not the focus of this paper, however it is necessary to present kinds of learning activities that could be related to the dimensional model. Concrete Experience Learners' dimension could be guided by experiment observation, simulation systems, movies or role-playing activities. In the Reflexive Observation dimension, it is possible to develop activities such as guided reading, portfolio writing, conceptual maps elaboration and thematic discussion in groups. Abstract Conceptualisation dimension explores research papers, Computer-Assisted Instructional software (CAI) and other activities that deal with learning individual timing. The last dimension, Active Experimentation, presents trends to explore case studies, assignment problems or laboratory experiments. The study presented in the following items introduces the proposal of using conceptual lattices to implement SCORM adaptive navigation rules that consider students' learning styles. The described schema provides information that allows the understanding of students' distinct ways of learning. From the standpoint of information process preferences, it is possible to design the educational process to use adaptive learning objects with Conceptual Lattice-based SCORM for personalisation of apprentice experience.

5. Conceptual lattices and learning styles

Considering Kolb's model presented in Section 4, it is possible to infer that convergers and assimilators use structured logical sequences of information to learn. They could use axiomatic logical resources that could be defined by properties (or attributes). Thus, they could develop a complex theory based only in the subject's properties or axioms. It is also possible to affirm that convergers and assimilators could learn the logical characteristics of the solar system by studying only its properties. Convergers use a simple path through attributes to achieve their objectives. The learning path defined by convergers is depicted in the figure 2. Assimilators can pass twice over a graph node in order revise or rethink some knowledge. Their navigational transformation is shown in figure 3. Again, taking Hasse's diagram as a viewpoint, divergers and accommodators will follow the ascendant navigation direction, which means, using objects. They will start from the planets (objects) and will traverse their characteristics (attributes). They will repeat this process until they have learned all the planets - the entire Solar System. Their learning paths are depicted in figures 4 and 5.

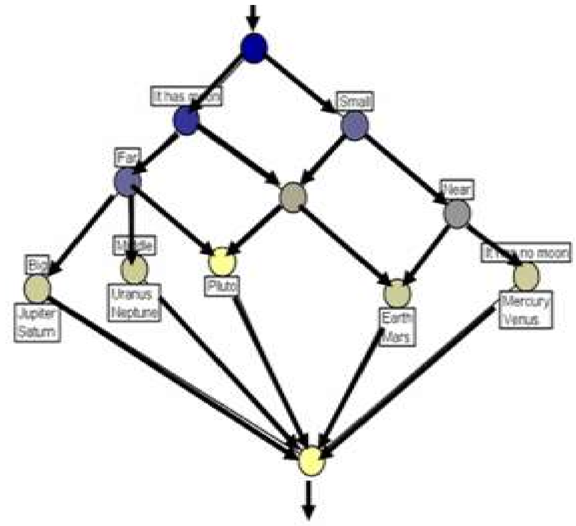

Figure 2: Convergers' learning path through conceptual lattice

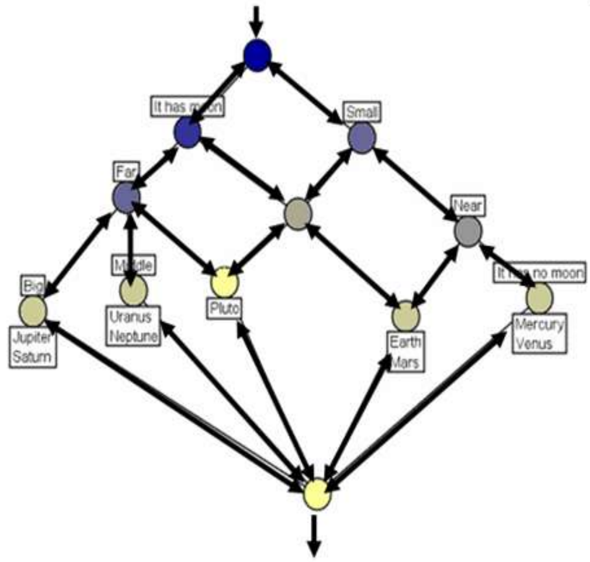

Figure 3: Assimilators' learning path through conceptual lattice.

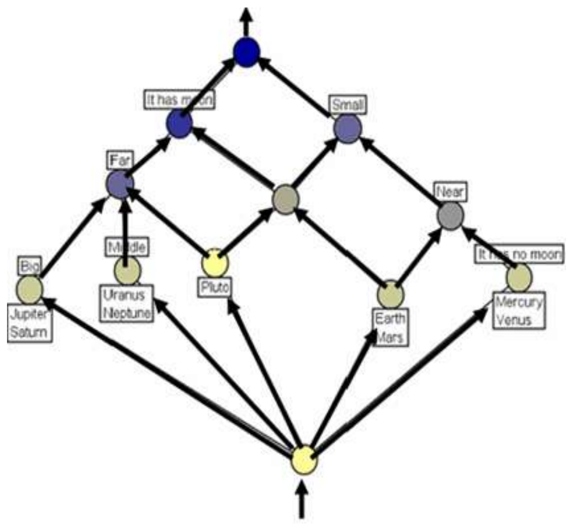

Figure 4: Divergers´s learning path through conceptual lattice

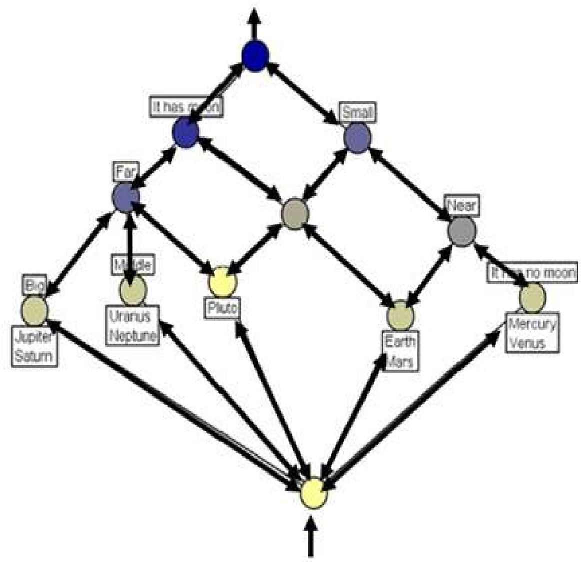

Figure 5: Accommodators' learning paths through conceptual lattice.

It must be noted that both approaches will present the same result: even by learning through attributes (planets properties) or through fragments (planets as objects), all students will learn the entire content.

6. Metadata standards and fine granular learning objects

IEEE Learning Object Metadata – LOM (n.d.) uses the Aggregation Level to describe "the functional granularity" of a learning object. SCORM also considers different levels of granularity, classifying them in the following scales:

- The smallest level of aggregation, e.g. raw media data or fragments.
- A collection of level 1 learning objects, e.g. a lesson.
- A collection of level 2 learning objects, e.g. a course.
- The largest level of granularity, e.g. a set of courses that lead to a certification.

Wagner (2002) defines five different granularity categories. In this classification, highly atomic LOs are called "content assets"; the next level of granularity refers to "information objects", or "molecular" LOs. SCORM considers assets as a first granularity level. SCOs (Sharable Content Objects SCOs, as well as metadata about aggregation itself and its individual components.

SCORM is a reference model for the packing and aggregation of learning objects allowing their usage from any compatible LMS. The SCORM model was presented by ADL (Advanced Distributed Learning) and it is defined by three documents: Content Aggregation Model (CAM), Run-Time Environment (RTE) e Sequencing and Navigation (SN) (ADL,n.d.). The basic unities in SCORM are the SCOs, which represent the learning objects that compose a course structure. The navigation among these objects must be defined in such a way that it can be followed by the LMS. This phase of a SCORM package definition is called content aggregation and it is accomplished by the creation of a XML file with the navigation rules among the objects. Thus, a SCORM package must contain a manifest file according to the IMS Global Learning Consortium rules. The package contains its content declaration, the content navigation order and the placement of the SCOs' physical files.) represent the second granularity level, being self-contained learning objects that meet additional technical requirements needed for interoperability with LMSs. Third SCORM granularity level includes Content Aggregations, which could contain assets or

Besides the manifest and the SCOs' files, the SCORM package includes description files for every SCO in order to facilitate their manipulation. These files are known as meta-data files which basically include information like author, title, version, creation date, technical requirements, educational context and objective. A manifest file could refer to others, called sub-manifest files. The LMS uses these files to establish the navigation order among SCOs. The Activity Tree defines this order. A learning activity could be a resource ("leaf" activity) or could be composed by different sub-activities. Besides, the activities have start and end points defined, as well as associated final tests. The passing from one activity to the other depends on successful attempts through the final tests. Thus, the sequence followed by the LMS is based on the Activity Tree traversal that is derived from de manifest(s) file(s). This structure, however, is not always the best alternative for a course. Based on this limitation, some work has been presented by researchers to propose alternative sequencing structures.

Different authors have been discussed limitations to the SCORM meta-data model. Abdullah (2004) points out that the SCORM model version 2004 follows the IMS proposal (2004), but it is too simplistic once it does not provide mechanisms for the effective implementation of adaptive learning objects. Gomes (2005) stands out the current standards' limitations, regarding comprehensive representation of functional learning objects. Simões (2004) proposes a SCORM extension, which can support transversal information to the learning objects, such as evaluation rules, curriculum or bibliography. Chang (2005) proposes using Petri Nets to represent the SCORM activity tree. The goal is to provide a linear visualisation of the traversed flow, as well as to allow the skipping through some chosen lessons. The authors support that such a scheme may be useful in collaborative environments (that are not covered by SCORM).

7. SCO annotations and links

The navigation among SCOs is highly dependant on the Activity Tree structure. This section proposes an alternative implementation of Activity Trees by introducing XLink navigation annotations directly on SCOs, dispensing further navigation structures on these trees. The previously presented navigation structure for objects and attributes do not depend on linear navigation. In order to map the principles described in this section with minimum impact on SCO annotation, will be adopted the following construction order: annotations in objects and attributes, link introduction among SCO objects and link annotation.

7.1 Annotations in objects and attributes

The annotation process in objects and attributes will be performed in META-DATA section of SCO manifest file, by using the tag keyword available in lom namespace (Davey and Priestley, 2002) and described by the SCORM Content Aggregation Model (SCORM CAM). The given annotation syntax follows:

```
<manifest ... >
 <metadata>
 <lom:lom
  xmlsns:lom="http://ltsc.ieee.org/xsd/LOM">
  <lom:general>
   <lom:keyword > <!-- An object/attribute -->
    <lom:string>    object    |    attribute
</lom:string>
    <lom:string>  Object  or  attribute  name
</lom:string>
   </lom:keyword>
  </lom:general>
 </lom:lom>
 </metadata>
</manifest>
```

Each SCO encapsulates a set of objects and attributes. Each object or attribute uses a keyword tag, described by two strings: a type (object or attribute) and a name. The type string is important for inheritance evaluation, which will be performed in the navigation process.

7.2 Links between SCOS and their annotations

The navigation structure between SCOs is still target of intense discussion as showed in the document SCORM Sequencing and Navigation (n.d.). To attend the navigation purposes in the hierarchical model treated in the Section 4, the XLink technology (Wilde and Lowe, 2002) is highly adequate. Links using XLink allow bi-directional navigation, processing rules and multiple directions. Links will be added to SCORM manifest using a new tag inside lom namespace called navigation:

```
<manifest ... >
  <metadata>
 <lom:lom>
  <lom:general>
   <lom:navigation> <!—A navigation link -->
    xlink:type   =  "extended"  xlink:to     =
address of next
                            SCORM object
    xlink:from   = address of previous SCORM
object
    xlink:arcrole  =   link   processing   rule
xlink:show       = "replace"
    xlink:actuate = "onRequest">
       Navigation text...
    </lom:navigation>
   </lom:navigation>
  </lom:general>
 </lom:lom>
 </metadata>
</manifest>
```

Each navigation tag represents a basic navigation unit: we know the previous address (from), the next address (to) and what we must do on the present object (arcrole). As an implementation rule, we use the actuation rule under request (onRequest) and its behaviour should occur by taking the place of the current content (`replace`).

One of the most important issues related to the proposed link consists of the arc processing role: if we are using an ascendant navigation, the set of valid objects, including the one you are on, is formed by the node itself plus their ancestral nodes; by using a descendant navigation, we consider the attribute related to the node plus their ancestral attributes. Furthermore, it is also possible to verify previous requirements in the navigation, process that resemble the tags in namespace imsss (IMSSS, n.d.).

Indicating an XML parser, normally based on SAX or DOM processing models, may specify the link-processing role. In this context, treatment roles can be interpreted as concerns that crosscut order relations and could be implemented by using Aspect-Oriented Programming (Kiczales, 1997). This approach allows us to work with an external entity, the aspect, in which we can introduce a complex treatment to arc roles with a low impact on SCO structure. Furthermore, we can replace the treatment role without changing the SCO and setting free the LMS to control the navigation task among SCORM objects.

8. The hierarchical structure and SCORM meta-object

The SCORM units built in the previous section will now be organised in a hierarchical structure in order to reflect the conceptual lattice requirements. Lattice nodes are mapped to SCOs and order relationships use links specified with XLink. These elements form a whole unit represented by a manifest file. Each manifest file is an integrating part of a bigger object, called SCORM meta-object or SCO-meta. Two annotations are essential for the SCO-meta: the initial SCO to start the navigation process and the direction to follow. Only the initial SCO will be mapped as a sub-manifest of the SCO-meta. This fact will permit the reduction of necessary space required to store the remaining hierarchy. The initial SCO for navigation purposes, as well as the direction, can be easily implemented in a manifest file by using the tag <organisation>, jointed with the tag <resource>:

```
<manifest>
 <organisations>
  <organisation>
         <item identifier=”Initial”
     Identifierref=”RInitial" />
  </organisation>
 </organisations>
 <resources>
  <resource
identifier=”RInitial"href=”manifest0.xml” />
 </resources>
</manifest>
```

An initial object in the navigation hierarchy will always be identified with the name Initial, which is placed in the parameter identifier inside the tag item. Finally, the LMS only needs to know the initial SCO and then transfer it to the navigation control. This responsibility delegation allows achieving great flexibility to navigation by associating complex and dynamic behaviors to SCO transitions.

8.1 Learning styles in the SCORM meta-object

Unfortunately, SCORM does not foresee learning styles in the manifest file. However, without a right learning style, the LMS does not know how to pass the navigation control in the lattice. In order to provoke a minimum impact in the manifest file, we propose the usage of the field Parameters in the item tag inside the organisation. We define a field called Style, which could assume the following values: converger, diverger, accommodator or assimilator. The following code exhibits a more complete SCORM metafile including style navigation for a converger learner:

```
<manifest>
  ...
 <organisations>
  <organisation>

        <item          identifier=”Initial”
Identifierref=”RInitial"
      Parameters=”? Style=converger”>
   </item>
        </organisation>
       ...
 </organisations>
 <resources>
  <resource             identifier=”RInitial"
href=”manifest0.xml”>
  </resource>
     ...
 </resources>
</manifest>
```

The Style parameter regulates how the navigation control is performed: with the converger value, we have a strict descendant navigation; by using the assimilator value, we can navigate in both descendant and ascendant direction with attributes; with the diverger value, we have again a strict ascendant direction and, finally, by using the accommodator value we gain two directions with objects.

9. Conclusions and further work

The specification of interoperability, accessibility and reusability in reference models is of great interest for Learning Management Systems development. In particular, the SCORM standard represents a great alternative to encapsulate learning objects data. Besides its general usage, SCORM has some deficiencies as the fact that it lacks sophisticated navigation mechanisms. This paper presented a proposed architecture to navigate through a SCORM objects net via conceptual lattices with dynamical graph navigational transformations. These lattices allow navigation among objects and attributes in a bi-directional way. The architecture is based on the introduction of annotations and links via XLink technology that is highly applied to integrate XML documents. The annotations and links produce a low impact on the current SCORM structure and make possible the building of complex SCORM objects nets through simple constructions. Links among objects could be endowed with qualified semantic processing. Besides, they allow the abstraction of connections as aspects among the manifest files associated with the learning objects and styles.

The approach used in this paper for learning styles respects some learners' individual characteristics even if it could be considered a simplistic form to face learning styles. Furthermore, specific learning styles effectively guide a dynamic graph navigational transformation. Future work will include more sophisticated learning styles classifications, as well as the study of more elaborated conceptual models as, for example, hybrid models involving Petri Nets and Conceptual Lattices. Besides, the automatic SCORM objects generation via conceptual lattices and their dynamical rewriting could be a powerful tool to help the learning objects development with high cohesion, fine granularity and high adaptability for different navigational styles. The development of a tool for learning styles discovery is also being planned.

References

Abdullah, N. A., Bailey, C., Davis, H. (2004) “Synthetic hypertext and hyperfiction: Augmenting SCORM manifests with adaptive links,” in *Proceedings of the 15th ACM Conference on Hypertext and Hypermedia*, Santa Cruz, EUA, pp. 183–184.

ADL (n.d.), Advanced Distributed Learning. [online], http://www.adlnet.org

Alonso, C. M. (1993) “Educational Technology and Learning Styles. Rethinking the Roles of Technology in Education,” in *The Tenth International Conference on Technology and Education,* USA, Massachusetts Institute of Technology and The University of Texas at Austin, pp. 1277–1279.

Chang, W., Lin, H. W., Shih, T., Yang, H. (2005, January). SCORM Learning Sequence Modeling with Petri Nets in Cooperative Learning. *Learning Technology*, IEEE Computer Society. 7(1) [online] http://lttf.ieee.org/learn_tech/issues/january2005/ #Toc98675005

Curry, L. (1987). *Integrating concepts of cognitive or learning style: A review with attention to psychometric standards.* Ottawa, ON: Canadian College of Health Service Executives.

Davey, B. A. and Priestley, H. A. (2002) *Introduction to Lattices and Order*. 2nd. Ed. Cambridge: Cambridge University Press.

Dublin Core Metadata Element Set, Version 1.1: Reference Description (2004), DCMI, Dublin Core Metadata Initiative. [online] http://www.dublincore.org/documents/dces/

Gomes, S. R. Gadelha, B. F., Mendonça, A. P., Amortti, M. S. M. (2005) “Objetos de Aprendisagem Funcionais e as Limitações dos Metadados Atuais,” in *Anais do XVI Simpósio Brasileiro de Informática na Educação – SBIE,* Juis de Fora, Brasil, pp. 211-221.

Learning Object MetaData. (n.d.) IEEE LTSC. [online] http://ieeeltsc.org/wg12LOM/

Learning Resource Meta-data Specification - Version 1.2.4 – XSD Schema, (2004). IMS Global Learning Consortium. [online] http://www.imsglobal.org/metadata

IMSSS (n.d.), IMS Global Learning Consortium. [online] http://www.imsglobal.org/xsd/imsss

Kiczales et. al. (1997) “Aspect-Oriented Programming,” in: *Proceedings of the European Conference on Object-Oriented Programming (ECOOP’97)*, Springer-Verlag, pp. 220–242.

Kolb, D. A. (1984) *Experiential Learning*. Experience as the Source of Learning and Development. New Jersey: Prentice Hall.

Millar, G. (2003). Learning Objects 101: A Primer for Neophytes. Sidebars: Inside out. *British Columbia Institute of Technology.* [online] http://online.bcit.ca/sidebars/02november/inside-out-1.htm

Mustaro, P., Silveira, I. F., Omar, N., Stump, S. M. D. (2006 – in press) Structure of Storyboard for Interactive Learning Objects Development. In Koogan, A. and Harman, K. (ed.). Learning Objects and Instructional Design. EUA: Informing Science Press.

Simões, D.,Luís, R., Horta, N. (2004) “Enhancing the SCORM Metadata Model,” in *Proceedings of the 13th international World Wide Web Conference*. New York, EUA, pp. 238–239.

Specht, M., Kravcik, M., Klemke, R., Pesin, L., Hüttenhain, R. (2002) “Adaptive Learning Environment for Teaching and Learning in WINDS,” in P. D. Bra, P. Brusilovsky, and R. Conejo (Eds.) AH’2002, LNCS. Springer, pp. 572–575.

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* [Online] http://reusability.org/read/ chapters/south.doc

Wagner, E. (2002). Steps to Creating a Content Strategy for Your Organisation. eLearning Developers' Journal. [Online] http://www. elearningguild.com/pdf/2/102902MGT-H.pdf

Wilde, E and Lowe, D. (2002) *XPath, XLink, XPointer and XML: A Practical Guide to Web Hyperlinking and Transclusion*. Pearson Education.

Wiley, D. A. (2000) “Learning Object Design and Sequencing Theory,” Ph.D. thesis, Brigham Young University. Provo, EUA