Hybrid System for Generating Learning Object Metadata

Olivier Motelet – Nelson A. Baloian – José A. Pino
DCC-Universidad de Chile
Blanco Encalada 2120 Santiago, Chile
{omotelet, nbaloian, jpino}@dcc.uchile.cl

Abstract – **Generating LOM for learning material is a complex and tedious task to be manually completed. Therefore, the current trend is to automate this process. However, there are several important issues restricting this approach. First, all the necessary information for generating metadata values is not always available. Second, some LOM attributes are difficult to generate in a rational and objective manner. In order to cope with these issues, this article suggests considering human intervention in the LOM generation process supported by automatic processes: a hybrid system in which both human and computer collaborate for instantiating LOM attributes is described. In this work, a tool implementing this proposal is implemented as a feature of LessonMapper2, which is a graphical editor designed to author lessons as graphs of learning resources.**

Keywords – **learning-object metadata, user-interface for metadata generation, automatic metadata generation, lesson graph**.

I. INTRODUCTION

Learning Object Metadata has been subject of much research in recent years. Yet, it is not a useful tool in educational practice nowadays. We claim that occurs in part, because learning material authors do not have incentives to produce learning object metadata. This paper deals with a proposal to use a hybrid system for generating learning object metadata. The use of such system should speed up the instantiation process and help evaluate the validity of the corresponding values.

There is more than one definition of what a Learning Object is. According to the definition of the IEEE "Learning objects are defined as any entity, digital or non-digital, which can be used, reused, or referenced during technology supported learning" [1]. Wiley defines it as "A learning object is a digital resource that can be reused to facilitate learning." [21]. For Polsani, "A learning object is an independent and self-standing unit of learning content that is predisposed to reuse in multiple instructional context." [20].

Although there are many definitions about what a learning object is, all of them agree in that it is a piece of digital multimedia, which can be used for learning or teaching something. They all also agree that a learning object should be reusable. This usually means that it should be possible to use them in contexts. Certainly, one of the most important goals of having learning objects and repositories containing them is to allow this educational learning resource to be reused, since creation of quality learning material can be very expensive and may only be viable if there is a large community of users. Very often learning objects are complemented with metadata in order to facilitate they storage and retrieval. Metadata are data about data. Some standards for metadata have been already developed like the basic Dublin Core (DC) [15], which defines a set of metadata including mainly objective attributes of the object like author's name, title and granularity. The DC Educational extension [16] was defined in order to include pedagogical attributes. CANCORE [17], IMS Metadata [18] and SCORM [19] are other examples. Nowadays, the IEEE Learning Object Metadata (LOM) [1] specification is generally accepted as the standard for providing metadata to multimedia learning resources. However, sometimes it does not meet all the requirements, since a certain community of practice may require local extensions and modifications. Thus the term Application Profile for LOM (or LOM Profile) has emerged as a means of describing this practical reality. In fact, the metadata definitions used by SCORM and CANCORE may be considered as LOM Profiles since they can easily be translated to the LOM standard. In fact, SCORM has been considered to be a standard for packing courseware content, i.e. learning material, which is used for distance learning and administered by a Learning Management System like WebCT [23], TopClass [24], Blackboard [25] or Docent [26].

As we said, the aim of using metadata for describing learning objects is to promote the sharing of learning material. Nevertheless, studies made about the real use of LOM shows that metadata is often misused or not instantiated [2]. Such results are mostly due to the complexity of the specification: LOM includes almost 70 attributes. Such high number of elements is an obvious obstacle to the generation of LOM because many authors refuse to fill all the metadata. On the other hand, some of the values for the metadata are subjective and make it difficult to assign a value to them. This is especially true for the metadata describing the educational aspects of the object, like semantic density, difficulty, interactivity, etc., which in turn are also the potentially most useful metadata for promoting reuse of learning objects.

Many authors have already proposed the automatic generation of metadata as a way to solve at least partially the problem of the provision of metadata values for learning objects, thus alleviating the work of the author or the learning object "librarian" [3,4]. However, automatic generation of LOM is still limited. In particular, the educational attributes of LOM are very difficult to produce. However, educational metadata are a key topic for building systems allowing retrieval of learning objects in a pedagogically sound manner. Indeed, learning object retrieval should not only be based on text mining techniques analyzing the content of the documents such as search engines like Google or Yahoo! do, but also on the pedagogical context and characteristics for which learning objects were done. Moreover, a text mining technique on the content of the object itself may be not applicable for many learning objects because it may consist of a multimedia file like image, video or just executable code of a simulation program. Educational metadata is certainly relevant for improving learning object retrieval and should be instantiated. Nevertheless, since educational attributes of LOM are difficult to automatically instantiate, it is still necessary to count on human intervention in this process.

This article explores the development of hybrid systems in which both human and computer collaborate for instantiating LOM attributes. Our approach considers the lesson syllabus authoring as a favorable context for such collaboration, in which LOM values may help making the organization of the lesson coherent. In particular, we consider a lesson as a graph in which nodes are learning objects and links define relations between them as defined in the LOM specification. In order to explore this proposal, we have developed LessonMapper2 [12], a graphical front-end for building LOM-based graphs of learning resources.

This article presents and discusses the characteristics of a hybrid system for generating Learning Object Metadata. It presents these characteristics together with a practical implementation. After a brief introduction to the LOM specification in Section 2, Section 3 discusses the progresses and issues of automatic generation of LOM values. Next, section 4 deals with the problems of human-based generation of the LOM values. Then, graph-based lesson authoring is presented as a favorable context for

easily instantiating LOM values. Finally, section 5 discusses the advantages of a hybrid system putting together the expertise of education professionals (authors or librarians) with the potential of automatic systems for generating LOM values.

II. LOM

IEEE Learning Object Metadata specification is a set of about 70 attributes describing technical, educational and general aspects of educational resources. All data elements are grouped into nine categories:

1. *General* – The General category groups the general information that describing the learning object as a whole (e.g. its title, its keywords, its description).
2. *Lifecycle* – The Lifecycle category groups the features related to the history of the learning object (e.g. its version, its current state, the participants to its evolution).
3. *Meta-Metadata* – The Meta-Metadata category groups information about the metadata instance itself rather than the resource.
4. *Technical* – The Technical category groups the technical requirements and characteristics of the learning object (e.g. its format, its size).
5. *Educational* – The Educational category groups the educational and pedagogic characteristics of the learning object (e.g. its interactivity type, its difficulty, its intended end-users).
6. *Rights* – The Rights category includes the intellectual property rights and conditions of use for the learning object.
7. *Relation* – The Relation category groups the relationships between the learning object and other learning objects. Each relation should refer to a link type (e.g *isPartOf*, *isBackgroundFor*, *introducesTo)*.
8. *Annotation* – The Annotation category provides comments on the educational use of the learning object. Each comment is decorated with its author and its creation date.
9. *Classification* – The Classification category describes this learning object in relation to a particular classification system (e.g. ACM Taxonomy).

In this article, we will adopt the following notation to identify the LOM attribute names: a LOM attribute name consists of a string containing a list of names separated by slashes, e.g. *general/title*, where "general" is the category and "title" the attribute name. Only leaf attributes have values. Value types are defined by the specification. There are three groups of value types: (1) Predefined vocabulary values, which means that the values are to be chosen from a predefined set of words with a known unambiguous meaning (e.g. *easy* and *difficult* are vocabulary values for the *educational/difficulty* attribute). (2) Free text, which consist of any string the user may assign to this value. (3)

Primitive types, e.g. identifier, date, time, or integer. A range is defined for most attribute values like e.g. a set of strings for *general/keywords*.

III. Automatic Generation of LOM

Most authors agree on the fact that dealing with metadata cannot be a human task [5]. First, it is argued metadata creation is difficult [2]. It is also a time consuming task for multimedia learning content authors. Nevertheless, expert metadata creators are considered too expensive to be employed in most educational institutions and metadata instantiation generally remains the responsibility of the learning object authors. However, learning object authors have no direct retribution for instantiating LOM: While learners or educational professionals may benefit from metadata when retrieving a certain learning object, the author of this learning object does never take advantage of metadata. So it is difficult to motivate the learning content author to instantiate metadata values if this person does not get any direct rewards from this work, which can be time consuming and tedious. Automatic systems for generating LOM should definitely be useful to relieve this process.

A. Automatic Instantiation of LOM

Most metadata providing technical information (e.g., size, format, creation date, duration, etc.) about an educational resource can be automatically generated without problems by the same applications used to create them [4]. Also some other type of metadata like the author's name and language can be sometimes automatically deduced from the context in which the application is running. Approaches for generating metadata using document contents are based on the process of extracting information identifying the subject of the educational resource (title, keywords, description, classification). However, these techniques are based on text mining and not all educational resources are text-based. In fact, rich educational resources make extensive use of multimedia, which is still difficult to analyze with the same content-based methods. Moreover, most of the useful information for the educational-related metadata remains implicit in the learning material and its usage. For instance, the interactivity of an educational resource or its intended end-user is almost never indicated in the content of this resource.

B. Suggesting and Restricting LOM Values

From another perspective, we can try to find in the context in which the learning material has been used more opportunities for generating the metadata related to the educational description. In particular, techniques exploiting the semantics of the context in which the learning object is used seem to offer interesting opportunities for generating educational metadata. For example, Hatala and Richards [6] relate the semantics of

the already existing metadata in order to generate new ones. In particular, they defined a set of specific rules concerning the inheritance between educational resources (from parent to children), the accumulation (from children to parent), and the content similarity between educational resources. Nevertheless, the correctness of the metadata generated with such techniques remains only approximate. Consequently, the authors claim that such values should be used as suggestions or for validation.

Motelet and Baloian [7] also propose the usage of inference rules based on the relations between a certain educational resource and other material. This work is based on the principle that a semantic relation between two educational resources may impose a certain mutual influence between their metadata values. For instance, let us consider a first learning object, which is explained by a second learning object, i.e. exists a relation of type *explainedBy* between the first and the second object. We may say that this relation implies that the keywords of the first learning resource are quite similar to the keywords of the second. Probabilistic analyses enable to precise how similar are the keywords of both learning objects. For instance, analyzing a repository that we developed in our institution (about 200 learning objects related with each other), it was found that the probability for a certain learning object to have the keywords of a second learning object that explains the first one is 0.7. The same analysis can be done for all the metadata attributes and all relation types between two objects: The result is a set of probabilities for each metadata attribute and each relation type. Applied to a lesson defined as a graph of learning objects, these probabilities can be used for generating possible metadata values for the learning objects of the lesson.

Furthermore, Motelet and Baloian [7] suggest generating rules formulating restrictions on metadata values. For example, some restriction rule may be defined for the attribute *educational/semanticDensity,* which is a subjective measure of the learning object's usefulness as compared to its size or duration. If two learning objects are related with the semantic relation *introducesTo*, i.e. one learning object introduces to the other one, then the *semantic density* of the former is certainly lower than the *semantic density* of the latter. Since such assumption may not be valid for all potential users, rules are created with a simple domain-specific language. Thus, each community may be able to define their own rules according to their needs.

IV. Human-Based Generation of LOM

Manually generating LOM typically means to fill a long form listing the numerous attributes of the metadata specification. Human-based generation tools (e.g Reload Editor) and learning object repository interfaces (e.g. Merlot, or Ariadne) are based on the form metaphor. Nevertheless, most systems provide additional support to the user. For instance, some of them explain the meaning

of the attributes while the user is instantiating them. Other ones propose a list of authorized values (or vocabulary) for each attribute. These values are generally customized to match the needs of a certain community of practice. Nevertheless, the generation process remains tedious [2]. In fact, metadata generation is mostly separated from the learning object authoring process: LOM are usually generated for packaging, classifying and publishing the learning material when there is a wish to share it. Nevertheless, we believe that LOM has potential for other usages. In particular, it may facilitate the characterization of the material during lesson authoring. To allow such practice, human-based generation of LOM may be supported by other user-interfaces, different from form-based ones.

A. Packaging with LOM

Research work about LOM usage points out to the fact that most learning objects available in repositories are coarse grained. In fact, LOM are generally used for packaging and sharing master educational pieces like an entire course or a complex resource (e.g. a simulation system, a flash animation). However, fine-grained learning objects are easier to reuse in various contexts than the coarse ones [8]. Whereas learning object repositories are intended to ease the reuse of pedagogical material, most of them do not yet contain relevant number of varied fine-grained resources (see Merlot, EdNa or Ariadne for a sample). Since education professionals and institutions are still reluctant to spend time and resources to pack and instantiate the metadata of only a few learning material items [2], it seems quite difficult to expect for human-based LOM generation of tens of fine-grained learning material items. Until now, LOM as a packaging tool has shown too few benefits for LOM creators. Other benefits for LOM should definitely be found in order to stimulate their use.

B. Characterizing with LOM

From the beginning of the development of intelligent multimedia for learning, authors have proposed to structure learning material through graphs [9,10]. Indeed, the lesson graph is a key structure for achieving flexibility. The LOM specification includes the notion of relations between two learning resources and so makes it possible to formulate graphs of learning resources. In such graphs, each node corresponds to a piece of material characterized with LOM. We have built LessonMapper2 [12], a graphical application for building graphs of learning resources described with LOM. This tool was developed in Java. In LessonMapper2, learning objects refer to external resources (e.g. an image, a slide, a web site or an activity designed with LAMS) or nested lesson graphs. Navigation in the lesson graph is enabled by a zoomable interface developed with the Piccolo Toolkit [13]. It also uses JDOM so that the metadata of each learning object is directly modified in its XML, LOM-compliant form. LOM RDF [14] is applied in order to define a thesaurus for the attributes.

In a lesson syllabus graph, the graph nodes correspond to learning objects. It often happens that the learning objects of a same lesson have characteristics in common. For instance, the language used in the different learning objects of a lesson is generally the same. In such situation, we believe that human-based LOM generation may benefit from instantiating simultaneously the *general/language* attribute of various nodes of the graph instead of independently repeating this task for each piece of material. In Figure 1, the *general/language* attribute is edited simultaneously for various learning objects. In this example, all learning objects are written in Spanish. With the Simultaneous Edition featured by LessonMapper2, the language identifier can easily be replicated from one learning object to the others by dragging and dropping it.
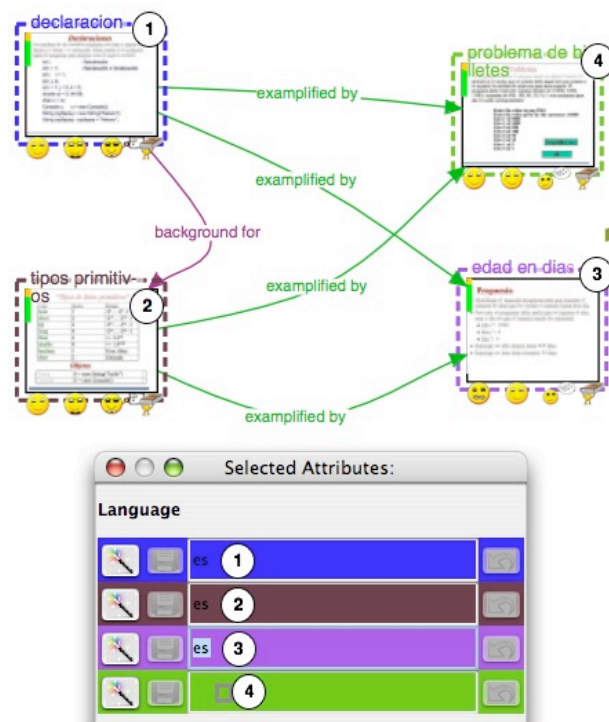

Figure 1. Simultaneous Edition of the "general/language" attribute for four learning objects with LessonMapper2.

Editing simultaneously the same attribute for various resources also eases the comparison between their values. For example, Figure 2 presents the instantiation of the *educational/difficulty* attribute for two learning objects. Since the notion of *difficulty* is directly related to the usage context (e.g. the background of the students or the topic being taught), instantiating this attribute in an isolated manner does not really make sense. The learning objects presented in Figure 2 are dealing with a programming language course. The learning objects "Problema de billetes" and "Edad en dias" are two problems putting in practice the notions of declaration and primitive types. Both problems are quite easy. Nevertheless, one of them is easier than the other. Thus, the instantiation of the *educational/difficulty* for the two learning objects should consider this fact. The Edition in

Comparison featured by LessonMapper2 enables it. The same principle can be applied to most subjective attributes of LOM, i.e. those attributes that are difficult to instantiate in an objective and rational manner. This category concerns almost all the *educational* attributes.
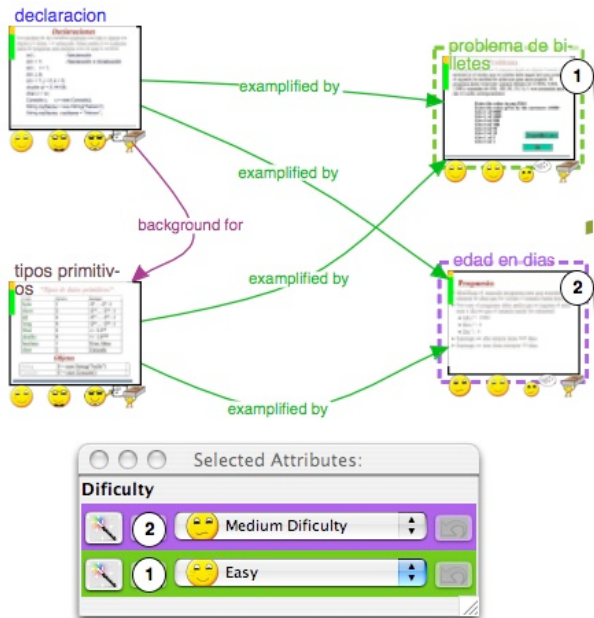

Figure 2.   Edition in Comparison of the "educational/difficulty" attribute for two learning objects.

In LessonMapper2, the learning objects are visually characterized with the values of their metadata. Figure 3 shows the characterization of one of them. This visual information makes easy to check the coherence of the learning design of the lesson: For instance, if a teacher wants her lesson having a linear progress in terms of difficulty, the visual characteristics associated with the *educational/difficulty* attribute of the learning objects of her lesson can help her to ensure this scheme.

Figure 3. Visual Characterization of some metadata attributes of the "Edad en dias" learning object.

This section has shown that manual instantiation of LOM during the lesson authoring process may benefit from specific user-interfaces and could support the teacher in ensuring the coherence of the lesson. However,

the instantiation process of some objective attributes like the size, the technical format or requirements of a learning object are much less interesting to instantiate for the lesson author than the educational characteristics. For these former attributes, automatic generation systems are really necessary. Next section studies the advantages of mixing automatic and manual generation of LOM.

V.   HYBRID SYSTEM FOR LOM GENERATION

In a study about metadata generation methods, Greenberg [11] concludes "*best metadata generation option is to integrate both human and automatic processes*". According to this principle, metadata for educational resources should benefit from collaboration between the human expertise and the advices of automatic processes. A tool supporting such collaboration is a hybrid system between LOM editors and LOM generators. This section presents such a system and details its capacity to support two processes related to LOM generation: the instantiation and the validation.

A. LOM Instantiation

The information produced by the automatic systems presented in the second section may be classified in three groups:

- *The very probable values*: Such values are the results of automatic generation systems like in [3]. They concern objective attributes like the technical format or the size of the document.

- *The probable values*: These values concern the suggestions done by intelligent systems. Typically, these suggestions are extracted from the context like the learning management systems or some related learning material [5,6,7]. These values are generally not reliable enough to serve automatic instantiation purposes, but they may support the task of defining metadata values.

- *The restriction of possible values*: Analyzing the context around a learning object allows to infer some restrictions about the LOM values of this learning object [7]. In the context of a community of practice in which the sets of possible values for LOM are well defined, these restrictions may serve to reduce the scale of possible values.

From the point of view of a human-based instantiation of LOM, all these groups of information are relevant. First, the very probable values may be automatically instantiated without human intervention. Second, suggestions may be displayed to help the user in the process of metadata value instantiation. Third, the restrictions may also be used to speed up the instantiation and limit the cognitive overload related to the large range of possible values for a same attribute [2].
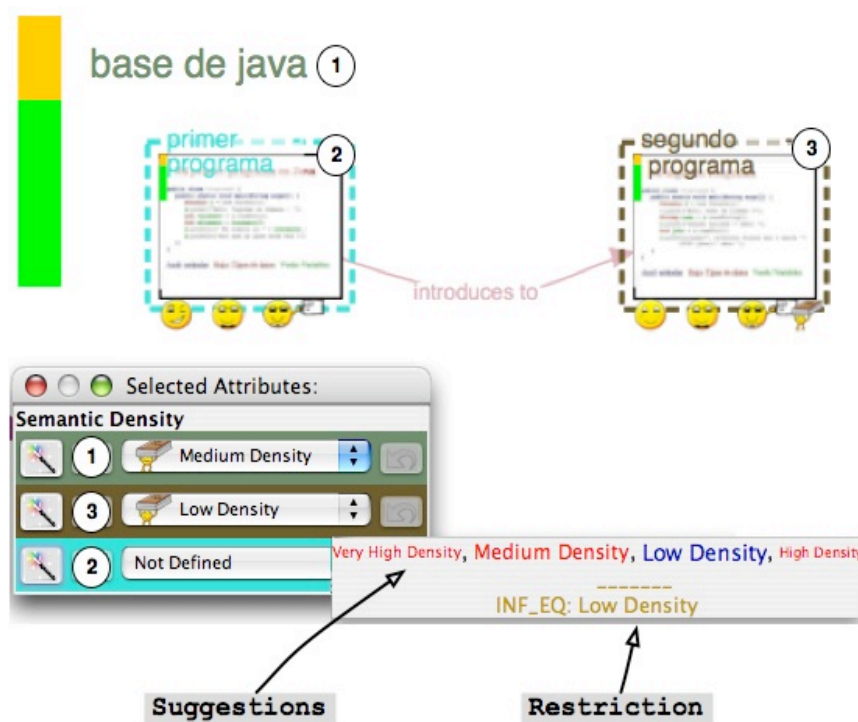
Figure 4. Suggestions and Restriction for the possible value of the *educational/semanticDensity* attribute for the "Primer Programa" learning object. The size of a certain suggestion depends on its probability of being relevant. Note that the font color differentiates the suggested values that do not comply with the restriction from the value (*LowDensity*) that does.

We have implemented part of these concepts in our LOM-based lesson graph builder, LessonMapper2. In particular, LessonMapper2 integrates the generator of suggestions and restrictions presented in [7] and described in Section 3. This system uses the semantics of lesson graphs based on learning objects for generating potential values (suggestions) and value boundaries (restrictions) for LOM attributes. Figure 4 shows the semantic density of three learning objects: "Base de Java", "Primer Programa" and "Segundo Programa". The first learning object contains the last ones. The semantic density of "Primer Programa" is not defined. In this example, suggestions and restrictions were asked for this attribute.

Since "Primer Programa" introduces to "Segundo Programa", the system can deduce a restriction for the *educational/semanticDensity* attribute: The semantic density of "Primer Programa" should be inferior or equal to the semantic density of "Segundo Programa". As described in Section 3, this restriction may not suit the pedagogical considerations of the whole teacher community. Thus, teacher communities using such a system should customize the restriction rules to suit their own needs.

Since "Primer Programa" is part of "Base de Java", the system suggests the value of the semantic density of "Base de Java", *medium density*, for instantiating the semantic density of "Primer Programa". It also suggests the semantic density of "Segundo Programa", *low density*, and the semantic density of other learning objects not displayed in the figure but being also part of the

lesson and indirectly related to "Primer Programa". Suggestions are sized according to their relevance: Suggestions having a high probability of being chosen are bigger than the other ones. For instance, the probability that "Primer Programa" has the same semantic density than its parent, "Base de Java", is about 0.5 whereas the probability of such similarity with the learning object it introduces, "Segundo Programa", is about 0.75. Therefore, the semantic density of "Segundo Programa" is displayed the biggest. As described in Section 3, these probabilities are based on the analysis of a base of learning objects we developed in our institution. This analysis should locally be done in order to suit the preferences of a certain community of teachers. The suggested values that do not comply with the deduced restrictions are painted in a color different from the color of the values that do. In LessonMapper2, once the user has chosen the most appropriate value for her learning object, simple drag and drop allows adopting one of the suggested values.

As described in this section, suggestions and restrictions may be beneficial for supporting the user in instantiating metadata values. Nevertheless, the support system also benefits from the human work since it is based on the metadata values of the lesson elements that the user has already instantiated in the lesson. Therefore, more the user gives values for the lesson metadata, better the system support is.
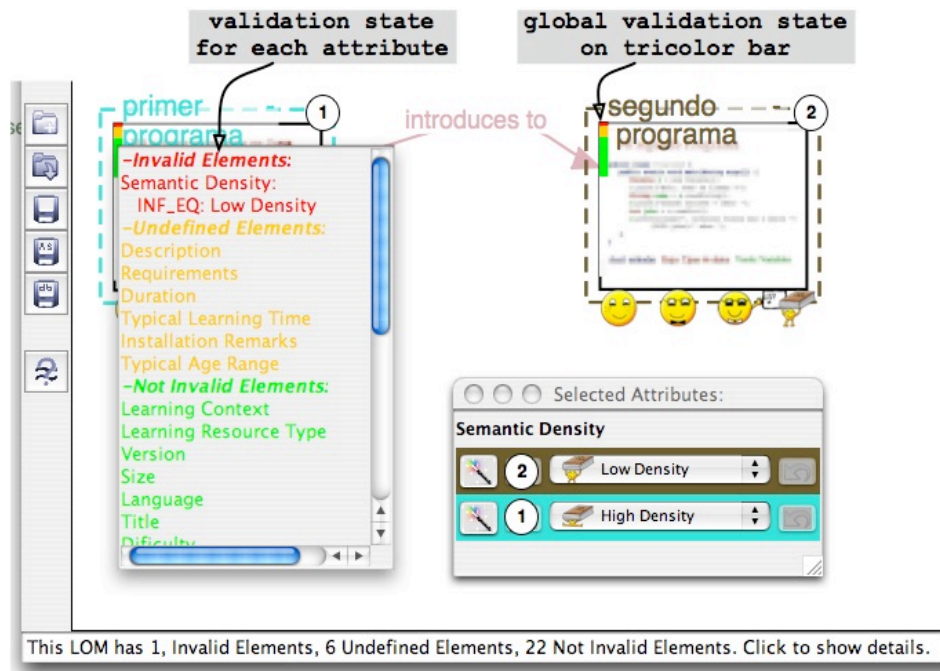
Figure 5. Displaying validation state of the metadata of a learning object with LessonMapper2.

B. LOM Validation

In order to provide full access to an educational resource, it has to be ensured that all the information is annotated with valid metadata. Valid metadata should satisfy a minimum level of completeness and correctness. The analysis of completeness of LOM simply consists in checking the number of instantiated attributes. Complete LOM have all their attribute instantiated. Correctness evaluation is complex because it deals with the semantics of the metadata values. A first step for evaluating correctness consists of checking the validity of the data type. Basically, this test can be done using the XML Schema validation mechanism. Another step consists of checking whether the given value belongs to a set of vocabulary terms. The Reload editor, e.g., enables to check the vocabulary corresponding to a different version of the IMS Metadata definition. Nevertheless, none of the available editors provides a deep checking of the meaning validity, because this topic has to do with both the semantics of the metadata and the semantics of the educational resource. In order to cope with this deficit, Ochoa et al. [3] suggest a framework for using various automatic metadata generation methods for cross-validating metadata.

In LessonMapper2, we propose a different method. It consists of using the restrictions deduced from the analysis of the graph semantics for checking the correctness of the values. Even if this method does not provide sufficiently accurate results to safely validate the correctness of the metadata, it may provide relevant data about metadata coherence for a set of educational resources. As shown in Figure 5, in LessonMapper2, all the learning material items of the graph are decorated with tricolor bars representing the proportion of invalid, undefined, and not invalid elements for the LOM attributes of each learning material item. Invalid elements hold values not satisfying the restrictions deduced by the system. Undefined elements have not yet any value assigned. Finally, not invalid elements hold values successfully passing both tests of completeness and correctness.

The validity bar permits to spotlight the elements forgotten by the instantiation process and prevent from the incoherencies as far as they can be detected by the rule-based system generating the restrictions. The bar is automatically refreshed when a change occurs in the metadata values of graph elements. Figure 5 shows the validity results of the "Primer Programa" learning resource in their detailed form. This view exhibits the list of reasons for eventual issues. In this example, we attached a high semantic density to the "Primer Programa" element. Nevertheless, as described before, there is a restriction deduced for this learning object imposing that its semantic density should be inferior or equal to the low semantic density of "Segundo Programa". Since the current value does not comply with this restriction, the attribute *semanticDensity* is tagged as invalid. If other restrictions are deduced for this attribute, they are combined with the existing one.

In case incompatible restrictions are deduced for a same attribute, this incompatibility can be due to an incoherency in the graph semantics or in the rule definitions. In the former case, the user should check the relations linking the concerned LO with the remaining of the graph. If there is no apparent incoherency, then the problem comes from the rule definition. Therefore, one of the incompatible rules should be identified as incorrect and refined.

VI. CONCLUSION

This article discussed the issues of both automatic and human-based generation of LOM. In order to bypass the problems at stake in these approaches, we suggested an original hybrid system for LOM generation, LessonMapper2. The authoring of the lesson syllabus as a graph of learning resources is shown as a favorable context for such a system. LOM is used not only for packaging the learning material items, but also for characterizing them and potentially helping the design of the lesson syllabus. Our approach also attempts to integrate the results of automatic LOM generation tools in order to assist the instantiation process. In particular, the system provides suggestions and restrictions on LOM values. These features may speed up the instantiation process and limit its cognitive overload. Another advantage of this hybrid system is to evaluate the validity of LOM values. This evaluation is used to highlight the forgotten elements and the incoherent values. In fact, the state of LOM instantiation for each learning material item is permanently shown and automatically updated when a change occurs in the lesson being authored.

On the one hand, the omnipresence of LOM during lesson authoring may be seen as an additional noise disturbing the complex process of designing a lesson. On the other hand, LOM may sound useful if it is considered as a relevant tool for characterizing learning objects and helping the design of the lesson syllabus. We hope that the development of hybrid systems such as LessonMapper2 is a step forward in making this second statement a reality.

Besides, we plan to extend the scope of the concepts presented in this article to the hybrid instantiation of metadata used in other application domains, e.g., competence metadata [22].

ACKNOWLEDGMENT

REFERENCES

[1] IEEE LOM specification. http://ltsc.ieee.org/wg12.

[2] Friesen, N. "International LOM Survey: Report." *ISO/IEC JTC1/SC36 sub-committee,* 2004.

[3] Ochoa, X, Cardinaels K., Meire M. & Duval E. "Frameworks for the Automatic Indexation of Learning Management Systems Content into Learning Object Repositories." *Proc. of ED-MEDIA 2005*, Montreal, pp. 1407-1414, 2005.

[4] Pinkwart, N., Jansen M., Oelinger M., Korchounova L. & Hoppe U. "Partial Generation of Contextualized Metadata in a Collaborative Modeling Environment." *Proc. of 2nd International Workshop on Applications of Semantic Web Technologies for E-Learning - AH 2004*, Eindhoven, Netherland, 2004.

[5] Duval, E., & Hodgins, W. "Metadata matters." *Proc. of DC-2004,* Shanghai, China, 2004.

[6] Hatala, M., & Richards, G. "Value-Added Metatagging: Ontology and Rule Based Methods for Smarter Metadata." *Proc. of RuleML 2003, LNCS 2876*, pp. 65-80, 2003.

[7] Motelet, O., Baloian, N. A. Piwowarsky, B., and Pino J. A. "Taking advantage of the semantics of a lesson graph based on learning objects." *The 13th International Conference on Artificial Intelligence in Education (AIED 2007),* to be published, IOS Press, 2007.

[8] Wiley, D. "The Reusability Paradox", *Connexions Web site.* http://cnx.rice.edu/content/m11898/1.18/

[9] McCalla, G.: "The search for adaptability, flexibility, and individualization: Approaches to curriculum in intelligent tutoring systems." *In: Foundations and Frontiers of Adaptive Learning Environments. Springer*, pp. 91–122, 1992.

[10] Baloian, N.A., Pino, J.A. & Motelet, O.: "Collaborative authoring, use, and reuse of learning material in a computer-integrated classroom." *Proc. of CRIWG. 2003, LNCS 2806*, pp. 199–207, 2003.

[11] Greenberg, J. "Metadata Extraction and Harvesting: A Comparison of Two Automatic Metadata Generation Applications." *Journal of Internet Cataloguing: The International Quarterly of Digital Organization, Classification, and Access,* 6(4), 58-82, 2004.

[12] LessonMapper2, http://www.dcc.uchile.cl/~omotelet/

[13] Bederson, B. B., Grosjean, J., & Meyer, J. "Toolkit Design for Interactive Structured Graphics." *IEEE Transactions on Software Engineering, 30 (8),* pp. 535-546, 2004.

[14] Brase J., Nilsson M., & Palmer M. "The LOM RDF binding - principles and implementation." *Proc. of 3rd Annual Ariadne Conference,* Leuven, Belgium, 2003.

[15] Dublin Core Metadata Initiative. http://dublincore.org. Last accessed Jan. 18 2007.

[16] DCMI Educational Working Group. http://dublincore.org/groups/education. Last accessed on Jan, 2007.

[17] CANCORE http://www.cancore.ca. Last accessed on Jan 17, 2007.

[18] IMS Global Learning Consortium http://www.imsglobal.org. Last accessed on Jan, 2007.

[19] SCORM http://www.adlnet.org. Last accessed Jan, 2007.

[20] Polsani, P. (2003). Use and abuse of reusable learning objects. *Journal of Digital Information, 3(4)*.

[21] Wiley, D. (2001a). Connecting learning objects to instructional design theory: a definition, a metaphor, and a taxonomy, chapter *The Instructional Use of Learning Objects. Association for Instructional Technology.*

[22] J. L. De Coi, E. Herder, A. Koesling, C. Lofi, D. Olmedilla, O. Papapetrou, and W. Sibershi. A model for competence gap analysis. *In WEBIST 2007, Barcelona, Spain, 2007. INSTICC Press.*

[23] WebCT homepage: http://www.webct.com.Last access on Jan. 2007.

[24] TopClass homepage: http://www.wbtsystems.com. Last access on Jan. 2007.

[25] Blackboard homepage: http://wwwblackboard.com.Last access on Jan. 2007.

[26] Docent homepage: http://www.docent.com. Last access on Jan. 2007.

AUTHORS

Olivier Motelet obtained a degree of Engineer from the Ecole des Mines de Nantes (France) in 2000. The same year, he also acquired a degree of European Master of Science in Computer Science from the Vridje Universiteit Brussel (Belgium). His research interests include Learning Management Systems, Document Management and Retrieval, and Human Computer Interaction. He is presently PhD student at the Universidad de Chile.

Nelson Baloian obtained his Degree of Engineer in Computer Science in 1989 from the Universidad de Chile. In 1998 obtained his PhD from the University of Duisburg (Germany). He has been working on the fields of Computer Supported Learning and Distributed Computing since then. He is author of about 50 journal and conference articles on these subjects. He is currently Assistant Professor at the Department of Computer Science of the Universidad de Chile. He is also Visiting Professor of the University of Waseda (Japan) and the University of Duisburg-Essen (Germany).

José A. Pino is Associate Professor of Computer Science and Director of the PhD Program in Computer Science at the Universidad de Chile. His research interests include Computer-Supported Collaborative Work, Human-Computer Interaction and Software Industry Studies. He has served as President of the Chilean Computer Science Society (SCCC) and President of CLEI (the Latin American Association of Universities concerning Information Technology). He has co-authored six books and published research papers in international conferences and journals, including Journal of the ACM, Communications of the ACM, Decision Support Systems, Interacting with Computers and Information Technology and People.