

# Taking advantage of the Semantics of a Lesson Graph based on Learning Objects

Olivier MOTELET<sup>a,1</sup>, Nelson BALOIAN<sup>a</sup>, Benjamin PIWOWARSKI<sup>b</sup>,  
José A. PINO<sup>a</sup>

<sup>a</sup> *Universidad de Chile*

<sup>b</sup> *Yahoo! Research Latin America*

**Abstract.** Lesson graphs are composed of Learning Objects (LOs) and include a valuable amount of information about the content and usage of the LOs, described by the LO metadata. Graphs also make explicit the links between the LOs (i.e. the graph semantics). This article proposes a conceptual model for taking advantage of this information. This model is based on an original diffusion process that copes with the problem of lesson graphs where some metadatas are missing. Two applications of the model are described and were implemented over a previously developed lesson authoring tool whose goal is to facilitate the lesson authoring process.

**Keywords.** Learning Object Metadata, Lesson Graph, Graph Semantics

## 1. Introduction

The last ten years have witnessed the emergence of the concept of Learning Objects (LO) and learning object repositories (LOR). Although LOs have received several definitions in the literature, in this article we will simply consider a LO as a piece of multimedia educational material (a slide, a web page, a simulation, etc.) and its corresponding metadata. One of the main ongoing efforts in this area is the specification of a standard for the metadata characterizing a LO, the so-called Learning Object Metadata (LOM) [7]. Compared with other metadata standards for documents, that mainly address physical attributes of the digital resources, LOM offers a large set of educational attributes, such as the difficulty, the interactivity degree and the description of the pedagogical goals of the LO. This model motivated the development of various systems processing metadata in order to ease LO authoring [5], LO use [2], and LO retrieval [13]. This paper addresses the case of a teacher engaged in a lesson authoring process where the lesson does not consist of a single LO but a graph including many fine-grained LOs. Each node of the graph is a LO and each edge denotes a semantical or rhetorical relation between two LOs. We present a conceptual model for building systems taking advantage of the lesson graph semantics. Two different applications of this model are described and used in order to help the author performing the three tasks: a) defining the context of a query within a

---

<sup>1</sup>Correspondence to: Olivier Motelet, DCC - Universidad de Chile, Avenida Blanco Encalada 2120, Tercer Piso, Santiago, Chile, C.P. 837-0459, Tel (+56 2) 678.4365, E-mail: omotelet@dcc.uchile.cl

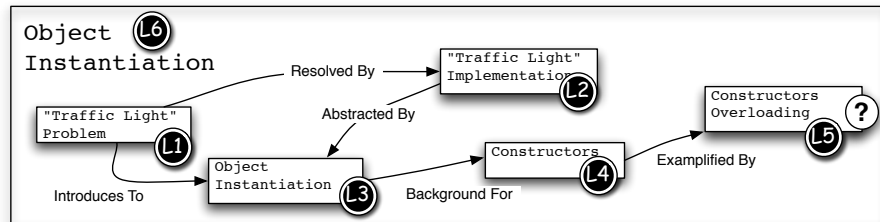


Figure 1. Start of a lesson graph about “object instantiation”

LO repository in order to enhance retrieval precision, b) setting the metadata values of a given LO, and c) checking the appropriateness of the metadata values the user set.

The next section describes the LOM specification and defines the concept of lesson graph. Then, a conceptual model for taking advantage of the graph semantics is presented along with two applications. Next, the potential of these applications to support the lesson authoring process is introduced. Finally, model limitations and outlook are discussed.

## 2. Metadata and Lesson Graph

LOM has about 60 attributes describing technical, educational and general aspects of educational resources. Attributes are identified by a series of names separated by slashes, e.g. *general/title*, where “general” is the category and “title” the attribute name. Attributes can be classified in three groups: (1) Predefined vocabulary values (e.g. *easy* and *difficult* are vocabulary values for the *educational/difficulty* attribute). (2) Free text. (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*.

Many authors have chosen the graph as the most suitable way of structuring the learning material of computer-based learning systems whenever adaptability and flexibility of the learning material is required [9,3]. In a LOM-based lesson graph, the *relation* attribute of LOM is used to describe the links between the LOs of the lesson. Links are typed, e.g., *introducesTo*, *isPartOf*, or *exemplifiedBy*. The set of links defines the edges of a lesson graph in which the nodes are the LOs. Such a graph is called **LO graph**. Figure 1 illustrates a LO graph consisting in LOs and relations among them. Six LOs, labeled from L1 to L6 describe a part of a programming course of an object oriented language. L1 describes the problem (how coordinate traffic lights in a crossroad) and L2 presents its implementation as a Java program. This problem aims to teach object instantiation in a program. L3 and L4 refer to documents defining object instantiation and the concept of constructors respectively. L5 is a node inside the lesson graph whose learning material has still not been defined. L6 is a LO of coarser granularity and acts as a container for L1 to L5.

Most used LOM relation types were often inspired by the DublinCore [1] specification. However, these relations were not originally designed for educational authoring, so they are not well suited to cope with the requirements of lesson authoring. We opted for another extended taxonomy proposed by Trigg [14]. Trigg’s taxonomy defines an extensive set of relations supporting narration, that can be used to define a lesson graph. It defines semantical as well as rhetorical relations allowing the author to use those that

better match the needs of the specific lesson. We asked a group of lecturers working in the Computer Science Department of our university to organize the content of the introductory computer programming course for freshmen as a lesson graph using a certain set of relations. We specifically left out the too generic relations (e.g. *isFollowedBy*) since they give almost no information about a LO context in the graph. Based on their work, we empirically selected a subset of these relations, emphasizing the semantical, rhetorical and organizational aspects of course authoring: *introducesTo*, *assessedBy*, *supportedBy*, *abstractedBy*, *exemplifiedBy*, *comparableWith*, *backgroundFor*, *summarizedBy*, *resolvedBy*, *isPartOf*. Each of these relations has an opposite: a relation from a LO *a* to another LO *b* implies an opposite relation between *b* and *a* (e.g. *isPartOf* defines a reverse relation *hasPart*). In this article, lesson graphs are built using this set.

### 3. Taking advantage of the Graph Semantics

In the literature, the dependency between graph semantics and the metadata values of graph LOs has been explored and we can distinguish two approaches: (1) Using metadata semantic to influence graph semantics. Farrell et al. [2] uses this approach in order to dynamically assemble repository LOs. (2) Using graph semantics to influence metadata semantics. Hatala and Richards [5] uses this method in order to suggest values for missing LOM elements. This article focuses on this second approach.

We call **influence rules** the rules defining the influence of the graph semantics on the metadata semantics. For instance, the rule of [5], *When there is a parent-child (i.e. isPartOf) relation between two LOs, the value of the educational/intendedUserRole attribute of the parent may be strongly suggested to the child*, is an influence rule. The result of this rule is a possible metadata value that may be “strongly” suggested to characterize the child LO. We call **contextualized metadata value (CMV)** the result of computing an influence rule over the **metadata values of a LO graph**.

In [5], the influence rules makes use of the existing metadata values in order to generate CMVs. When some metadata values are missing, this process is directly affected since there may be no input for the rules. Considering that various studies witness that numerous metadata values are missing in the available LOs [4], this could be a serious issue. In order to cope with this problem, we propose a diffusion process in which for a given LO, each influence rule is applied *not only* to the metadata values of the neighboring LOs *but also* to the CMVs previously generated by this rule or other ones. Since this process increases the input scope of the influence rules, the impact of missing metadata value when computing the rules should decrease but at the price of the possible generation of noise, that is unwanted metadata values. We call this recursive process the **context diffusion**.

Figure 2 depicts a conceptual model for generating CMVs using context diffusion. In this model, existing metadata values for a LO graph are processed by a set of influence rules in order to generate CMVs. In contrast with existing approaches for which rule computation is directly done on the metadata values of the graph LOs, in this model, rule processing depends on our context diffusion mechanism.

During context diffusion, influence rules are iteratively applied on both CMVs and original metadata values of the graph until the generated CMVs finally converge. Defining such a non-monotonic process over an existing inference rule system for the seman-

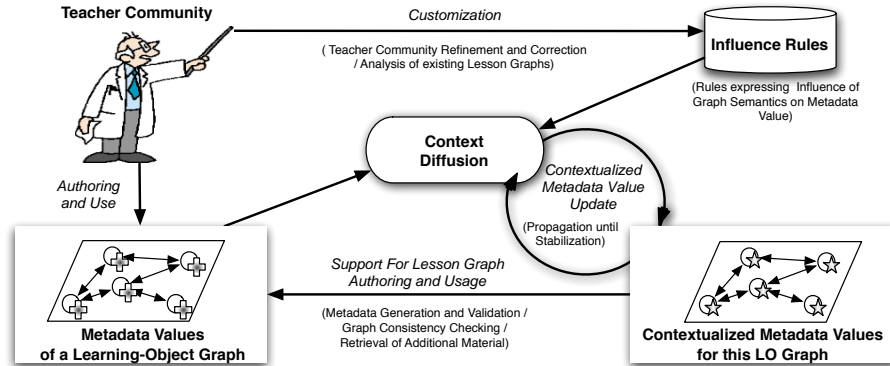


Figure 2. Conceptual model for taking advantage of the lesson graph semantics

tic web such as Jena [6] is a non-trivial task. Therefore, we use a simple push protocol based on propagation: For a certain node, the original metadata values and CMVs of the neighboring nodes are combined with its current CMVs using the influence rules. This update process generates a new set of CMVs for this node. If it appears that the node's CMVs changed during this process, all its neighbors are also updated in turn. Otherwise, the diffusion process stops for that node. Since we consider a lesson graph as a cyclic graph with symmetric edges (see Section 2), update computation should be chosen in such a way that it guarantees the diffusion process convergence.

**Teacher communities** are also part of the model : (1) As they author and use the lesson graphs they benefit from the support provided by using the CMVs, (2) In order to adapt the system to their teaching style and preferences, they can customize the influence rules manually (e.g. defining new influence rules or refining the existing ones) or automatically (e.g. providing existing lesson graphs that the system may analyze in order to deduce the necessary information for customizing the influence rules).

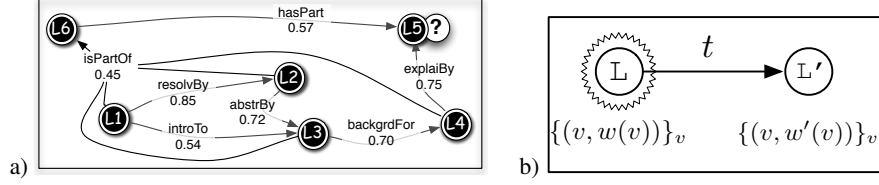
Instantiating the conceptual model presented in this section consists in defining (1) the nature of the CMVs, (2) the influence rules with a restricted language relating graph and metadata semantics, and (3) a CMVs update process that guarantees convergence. Two applications of the model are described in the next section.

#### 4. Applications of the conceptual Model

This section presents two uses of the above described model. In the first one, the graph consistency is analyzed generating restrictions for some of the metadata values. In the second one, similarities among the attribute values of the graph's LOs are used to generate suggestions for the metadata value of the LOs. In this section, we describe for both examples how we defined the influence rules, the CMVs, and the update process.

##### 4.1. Attribute Similarities and Value Suggestion

As stated by Hatala et al. [5], graph semantic analysis may be used to identify similarities between the metadata attribute values of related LOs. In the graph of Figure 1, since L1 introduces L3 we may expect that the attribute `general/keyword` of L1 and L3 share



**Figure 3.** (a) Weights of suggestions for `general/keyword` attribute. (b) Update process of suggestions.

some values. In the example, the values of this attribute are  $\{instantiation, object, method\}$  for L1 and  $\{instantiation, object, new\}$  for L3, sharing two common values out of three.

*Rule Definition.* In order to extend the predefined set of rules of [5], we propose to weight similarities for each couple of attribute and relation type by analyzing an existing repository of lesson graphs (containing about 170 LOs) developed in our institution. We found out that LOs linked with a *introducesTo* relation share the same values for the `general/keyword` attribute with a probability 0.54. Figure 3a is a reduced view of the graph of Figure 1 showing the probability (calculated on our corpus) of sharing the same values between neighboring nodes for the `general/keyword` attribute. Influence rules are defined as a triplet consisting of metadata attribute, relation type, and similarity probability.

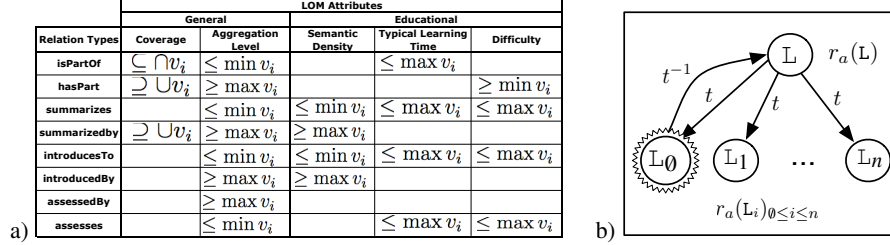
*CMV Definition.* The previous rules are used to generate a list of special CMVs called **suggestions** for each LOs of a lesson graph. A suggestion is a set  $\{(v, w(v))\}_v$  associating a weight  $w(v)$  to all the possible values  $v$  for a certain attribute  $a$  of a LO  $L$ . The weight is 0 when the value is not at all appropriate for  $L$ , while it is 1 when it fits it perfectly. At the beginning of the context diffusion process, we set  $w(v) = 0$  for all possible values  $v$  for the  $a$  attribute except for the original values which have a weight 1.

*Update Process.* Figure 3b depicts a diffusion step: for an attribute  $a$ , changes in the suggestion  $\{(v, w(v))\}_v$  of a LO  $L$  are propagated in order to influence the suggestion  $\{(v, w'(v))\}_v$  of every other LO  $L'$  connected to  $L$ . We note  $p_a(t)$  the probability that  $L$  and  $L'$  share the same value for the  $a$  attribute giving that a relationship of type  $t$  connects  $L$  with  $L'$ . The update process consists of replacing the CMV of  $L'$  with  $\{(v, \max(w'(v), p_a(t) \times w(v)))\}_v$ .

In cases where the same value for a certain metadata attribute is suggested by more than one neighboring LO, the maximum weight is considered. Since on the one hand, the value weights are filtered with the operator maximum and on the other hand, these weights are decreased at each propagation step (since they are multiplied by a probability between 0 and 1), we can guarantee that the process converges.

#### 4.2. Graph Consistency and Value Restriction

Let us consider again the lesson graph of Figure 1. It is plausible to think that for a certain teacher community, the fact that L1 introduces L3 may imply that the content of the L1 LO is simpler than the one of L3 (otherwise the lesson graph would not be consistent). In terms of LOM semantics, it means the value of the LOM attribute `educational/difficulty` of L1 should be lower or equal to the `educational/difficulty` of L3. If L1 introduces to more than one LO, its level of `educational/difficulty` should be compatible (lower) than each element it introduces



**Figure 4.** (a) Examples of restriction rules. (b) Update process of restriction intervals.

to. Since the value of `educational/difficulty` is associated to a predefined vocabulary, we can define an order between the terms of this vocabulary to test the consistency.

*Rule Definition.* The previous assumption about the consistency of LOM attribute values is a special type of influence rule called a **restriction rule**. A restriction rule is defined as the combination of three elements: (1) a metadata attribute name, (2) a relation name, (3) a couple of operators of the set  $\{\leq, \geq\} \times \{\max, \min\}$  (or  $\{\subseteq, \supseteq\} \times \{\cup, \cap\}$  when dealing with a metadata attribute based on a set of elements). For example, the restriction rule for the attribute `educational/difficulty` and the relation `introducesTo` is defined as  $\leq \max v_i$  where  $v_i$  is an attribute value of a LO related with the `introducesTo` connection. Other examples of rules can be found in Figure 4a. Note that the rules can be modified in order to adapt them to other educational contexts and new attributes.

*CMV Definition.* Applying restriction rules to the LO graph results in a special CMV called **restriction interval** for each metadata attribute of each LO. We define the restriction  $r_a(L)$  for the LO  $L$  and the attribute  $a$  as the interval  $[r_a^{lower}(L), r_a^{upper}(L)]$ . The possible values of  $a$  lie within this interval. We can also detect anomalies when the value of  $a$  set by the user does not belong to it. At the beginning of the diffusion process, the restriction interval of the  $a$  attribute of a LO  $L$  is initialized to the whole interval of the possible values ( $[a_{min}, a_{max}]$ ) unless it has been set by the user. In the latter case, the interval reduces to  $[a(L), a(L)]$ .

*Update Process.* Figure 4b depicts a step of the diffusion process: the changes in the restriction interval for a certain attribute  $a$  of a LO  $L_0$  are propagated to the restriction interval of its neighbor  $L$ . Restriction rules are applied to the reverse of the relation type used for propagating the changes: in this case, the relation of type  $t$  connecting the LO  $L$  (receiving the update notification) to  $L_0$  (propagating changes) is used. The update process of a restriction rule considers the  $n+1$  LOs to which  $L$  is connected with relations of type  $t$ . Those LOs are noted  $L_i$  with  $0 \leq i \leq n$  and  $n \in \mathbf{N}$ .

If the relation of type  $t$  imposes a restriction rule  $\leq \max$  for the values of the attribute  $a$ , the update process consists in replacing the restriction interval  $r_a(L)$  of the LO  $L$  by:

$$r_a(L) \cap \bigcup_i ] - \infty, r_a^{upper}(L_i) ]$$

This means that the restriction interval of  $L$  is intersected with an interval consisting of an infinite lower boundary (no effect on  $r_a(L)$ ) and an upper boundary equal to the maximum of the upper boundaries of the restriction intervals of the  $L_i$  LOs (may lower the upper boundary of  $r_a(L)$ ). The update processes for restriction rules of type  $\geq \max$ ,



**Figure 5.** Weighted values for the `Educational/SemanticDensity` attribute. Note that the value `Very High Density` is differently painted showing that it does not enter the scope of the restrictions for this attribute.

$\leq \min$ , or  $\geq \min$  follow similar principles. Note that the diffusion converges since it is only based on monotonically slimming down restriction intervals. If a restriction interval becomes void during the diffusion process, it means that an incoherency occurred in the graph. This incoherency could be due to (a) an incorrect value for the metadata of the LO, (b) some incorrect relations between this LO and the other LOs of the graph, or (c) a contradiction between two restrictions rules that should be resolved. Diffusion follows the same scheme when dealing with metadata attributes based on value sets.

## 5. Model Implementation in a Lesson Authoring Tool

The two applications of the model presented above were implemented in `LessonMapper2`, a Java-software prototype for authoring lesson graphs of LOs characterized with LOM. In this tool, our model is used to facilitate lesson authoring.

Suggestions and restrictions are used to alleviate the metadata generation process. When a lesson author is setting a metadata attribute, a weighted list of possible values is displayed: Suggested values with higher associated weight are displayed with the biggest font as shown in Figure 5. Suggestions not complying with the restrictions are highlighted. Restrictions are also used for detecting incoherences in the graph: all metadata values of the graph LOs are constantly checked. Whenever there is a restriction conflict, the corresponding attribute values are highlighted on the lesson graph. More details about this feature can be found in [11].

Suggestions and restrictions computed for an empty node in a lesson graph (like `L5` in Figure 1) can be used to refine a query on a learning object repository. When querying a repository, searching for a LO to fit into the empty node, the results are checked with respect to the restrictions generated by the system: Results complying with all restrictions are better ranked. Suggestions were used for enhancing retrieval: when querying a learning object repository containing other lesson graphs, the suggestions for the empty node are compared with the suggestions of the proposed results. The results having suggestions similar to the ones of the empty node are better ranked. Ranking information is then combined with a classical keyword-based query processed by the information retrieval system Lucene [8]. Small-scale experiments have shown that this approach effectively enhances the retrieval of LOs compared with Lucene alone [12].

## 6. Conclusion

This article presented a conceptual model using a context diffusion process in order to take advantage of the semantics of a lesson graph based on LOs and their metadata. Two illustrative applications were described: one dealing with the extension of an existing

method for generating metadata value suggestions and the other one dealing with the lesson graph consistency. We also showed how these features were implemented within a tool designed for lesson authoring.

Depending on the lesson graph size, the proposed context diffusion process may decrease the effect of missing metadata values compared with the other existing methods. For instance, since the influence rules for value suggestions are automatically defined for all the couples relation type - attribute, context diffusion extends their scope to all the available values of the graph.

While our system needs at least some metadata value, LOM usage during lesson authoring is generally limited to enable the sharing process. In [11], LOM are used to visually characterize the lesson graph elements, but the effects of this proposal are not evaluated. Moreover, our system requires the lesson author to define explicit relations. Whereas some studies argue for the benefits of the graph structure for building lesson [10], the cognitive weight and the advantage of typed relations on the lesson authoring process remains to be measured. Nevertheless, metadata types and relation types are parameters of our context diffusion process: Adapting them to other specific needs only impacts the influence rules not the diffusion process. For the same reason, while this article has focused on graph-based lesson authoring, our model could be applied to other settings using metadata-based graphs.

## References

- [1] DublinCore. Metadata initiative. <http://www.dublincore.org> last visit on 05/2006, 2007.
- [2] R. Farrell, S. D. Liburd, and J. C. Thomas. Dynamic assembly of learning objects. In *World-Wide Web International Conference WWW 2004, New York*, 2004.
- [3] S. Fischer. Course and exercise sequencing using metadata in adaptive hypermedia learning systems. *J. Educ. Resour. Comput.*, 1(1es):5, 2001.
- [4] N. Friesen. International lom survey report. Technical report, ISO/IEC JTC1/SC36, 2004.
- [5] M. Hatala and G. Richards. Value-added metatagging: Ontology and rule based methods for smarter metadata. In LNCS, editor, *Conference on Rules and Rule Markup Languages for the Semantic Web RuleML*, volume 2876 of *Lecture Notes in Computer Science*, pages 65–80. Springer, 2003.
- [6] JENA. A semantic web framework for java. <http://jena.sourceforge.net/>, 2007.
- [7] LOM. Ieee ltsc p1484.12.1 learning object metadata specification final draft. <http://ieeeltsc.org/wg12LOM/>, 2002.
- [8] Lucene. Full-featured text search engine library in java. <http://lucene.apache.org>, 2007.
- [9] G. McCalla. *Foundations and Frontiers of Adaptive Learning Environments*, chapter The search for adaptability, flexibility, and individualization: Approaches to curriculum in intelligent tutoring systems., pages 91–122. Springer, 1992.
- [10] E. McDaniel, B. Roth, and M. Miller. Concept mapping as a tool for curriculum design. In *Informing Science IT Education Conference (InSITE2005)*, 2005.
- [11] O. Motelet and N. A. Baloian. Hybrid system for generating learning object metadata. In *ICALT*, pages 563–567. IEEE Computer Society, 2006.
- [12] O. Motelet, B. Piwowarski, G. Dupret, and N. Baloian. Enhancing educational material retrieval using lesson authoring context. Technical report, DCC - Universidad de Chile, 2006.
- [13] N. Pinkwart, M. Jansen, M. Oelinger, L. Korchounova, and U. Hoppe. Partial generation of contextualized metadata in a collaborative modeling environment. In *2nd International Workshop on Applications of Semantic Web Technologies for E-Learning AH 2004, Eindhoven, Netherlands*, 2004.
- [14] R. Trigg. *A Network-Based Approach to Text Handling for the Online Scientific Community*. PhD thesis, University of Maryland, November 1983.