Designing Flexible Learning Object Repositories: Balancing Flexibility and Delegation in Ontological Characterizations

Salvador Sánchez-Alonso[1], Jesús Soto[2] and Miguel-Angel Sicilia[1]

[1]Computer Science Department, University of Alcalá, Ctra. Barcelona km. 33.6, 28871 Alcalá de Henares, Spain

{ salvador.sanchez, msicilia }@uah.es

[2]Faculty of Computer Science, Pontifical University of Salamanca, Paseo de Juan XXIII, 3 - 28040 Madrid, Spain

jesus.soto@upsam.net

Learning Objectives

- To have an overview on current ontological models for learning objects and learning object metadata.
- To have an overview of current approaches to the representation of learning objects in repositories.
- To be aware of the trade-offs between flexibility and delegation of tasks to software in the field of learning objects' management.
- To deepen on the role of ontologies in the design of a new generation of flexible learning object repositories.

Executive Summary

The concept of learning object is the basis of a new instructional design paradigm for Web-based learning. Learning technologies based on this model put the emphasis on reuse as a key characteristic of learning contents and activities. However, a plethora of definitions of learning object currently exist, ranging from anything and everything, to digital objects that support learning in a particular or specific context. The existence of so many different definitions and conceptualizations somewhat introduces confusion in the way these educational entities are understood and managed, hampering interesting possibilities such as the interaction between different learning management systems or the interchange of learning objects created and designed by different organizations, among others.

In the last years, several efforts towards the standardization of learning technology have resulted in the emergence of specific terminologies that are used to name and classify learning elements. Most of these efforts underline the importance of providing metadata information for content items in a standardized format as a crucial step to reusability. Writing metadata information about the learning objects, in the form of records describing their content, facilitates a number of processes such as storage, search, retrieval, or composition of new learning materials as an aggregation of others, making it possible the public availability of quality educational materials through learning object repositories. Unfortunately, most current learning object repositories provide room for including metadata records, but lack a conceptual model that clearly establishes what a learning object is and what their associated metadata should be.

Recent studies about the unification of the existing concepts into a single definition of learning object suggest the possibility of coexistence of all the available characterizations. The need for a conceptual model supporting the different visions that each different definition of learning object entails, asks for a new generation of flexible repositories where all the existent conceptualizations have a place. In this scenario, the taxonomization of all the definitions can be the basis of a neutral conceptual model based on a clear, formal semantics. This sound semantic model, defined by an ontology schema engineered using the Web Ontology Language –OWL–, postulates as a tool for research on achieving flexibility in description, while retaining specific degrees of orientation to enabling automated functions or delegating tasks to agents. The ontology proposed in this chapter serves as the basis of a new generation of learning object repositories: the so-called *semantic repositories*. The design and implementation issues of the core functions of a semantic learning object repository enabling a variety of learning object ontological characterizations will be described through the examination of a working prototype, the SLOR.

The SLOR –Semantic Learning Object Repository– prototype is an archetype of a semantic repository. It has been specifically designed for the creation and management of learning object metadata with integration and exchange purposes, allowing the repository clients to add, retrieve, modify and search for learning elements regardless the definition of learning object used by the learning object creator. With the aim of illustrating the design of a flexible learning object repository through the use of ontological characterizations, this chapter presents the architecture of the SLOR prototype as well as some examples of operation.

Introduction

Recent efforts in the standardization of learning technology (Friesen, 2005) have resulted in the emergence of specific terminologies that are used to name and classify learning actors, activities and artifacts (elements in a general sense). McGreal (2004) has recently attempted to unify existing concepts into a single definition, as current conceptualizations of the term "learning object" range from "anything and everything" to "digital objects that are marked in a specific way for educational purposes". Furthermore, Downes (2004) has introduced the notion of resource profiles as "the multi-faceted, wide ranging description of a resource". These works represent an attempt to reach a conclusion after the discussion and deliberation surrounding learning objects that has tried to state what should be considered to be "basic or fundamental classes of resources". These recent essays raise the fact that repositories of learning-oriented entities require higher degrees of flexibility in their characterizations of learning objects and related concepts. The inexistence of a common vocabulary, as well as the coexistence of different definitions of learning object, signal the need for a new generation of flexible repositories where all the existent conceptualizations have a place and users could benefit from higher richness of description. For the purpose of this chapter, flexibility is thus understood as the ability of a learning-based system for dealing with learning objects or any related elements described according to different models or conceptualizations.

Metadata is oriented to enable functions, as explicitly stated in Greenberg's definition of metadata: "structured data about an object that supports functions associated with the designated object" (Greenberg, 2002). According to this definition, there exist a tension between flexibility of description and the specificity and detail required for concrete uses of metadata. For example, the degree of interactivity characterizing a learning object should be stated in its metadata record as accurately as possible, making use of the terms in a list of appropriate values (often called *vocabularies*) or in a domain ontology: 'very low', 'low', 'medium', 'high', etc. However, too much flexibility in description, i.e. the use of several vocabularies according to different classification criteria (cultural, educational, etc.), could be the cause of malfunctioning such as inaccurate search results, because systems that rely on some values might not understand others. This becomes more important in the case where the functions that are desired to be supported are to be automated fully or partially. The concepts of scenario and learning object contract have been recently proposed as conceptual frameworks to delineate the execution semantics of these delegated functions:

- *Learning object design by contract* (Sicilia & Sánchez-Alonso, 2003) is a technique that defines contractual relationships between a learning object and the context in which it is to be used. It basically consists on stating, in the form of declarations called contracts, a collection of logical assertions on the requirements of use of the learning object and its expected learning outcomes.
- A *scenario* (Sicilia & Lytras, 2005) defines a situation in which a learning object is capable of engaging in according to the metadata information provided for it, thus connecting the concept of metadata with a context-specific definition of learning objects.

Formal ontology as a discipline (Welty & Guarino, 2001) is aimed at studying possibilia, so that it can be used to compare learning element representations according to the flexibility of their coverage, and term subsumption properties. In fact, ontological representations can play an important role as support for sound semantic models that fulfill a number of new requirements related to automation, such as search, retrieval or composition of new learning materials from others that already exist. The existence of ontology-based schemas becomes essential when some of the functions are to be delegated to automated or semi-automated systems, following the Semantic Web vision (Berners-Lee, Hendler & Lassila, 2001).

For the purpose of this chapter, delegation will thus be understood as the ability of a Learning Management System (LMS) to automate or semi-automate some tasks such as the mentioned, based on the existence of semantic models describing learning object-related concepts.

This chapter introduces an ontological schema that attempts to serve as a tool for research on achieving flexibility in description, while retaining specific degrees of orientation to enabling automated functions or delegating tasks to agents. In addition, design and implementation issues of the core functions of a repository which enables a variety of learning object ontological characterizations are described.

Learning object definitions, metadata and repositories

Current standardized e-learning systems are centered on the concept of learning object (Wiley, 2001), which can be characterized (using one of the most often-cited definitions) as "an independent and self-standing unit of learning content that is predisposed to reuse in multiple instructional contexts" (Polsani, 2002). This concept of learning object is the basis for a new instructional design paradigm for Web-based learning that emphasizes reuse as a quality characteristic of learning contents and activities. To date, however, "there is no commonly accepted definitive definition of learning objects" (McGreal, 2004). Readers interested in a deeper analysis of the numerous definitions of *learning object* available, are encouraged to have a look at the discussion at the beginning of this book.

Writing metadata information about the learning objects, in the form of records describing their content, facilitates a number of processes such as storage, search and retrieval from distributed repositories, as well as the composition of new learning materials as an aggregation of others. Accepted metadata specifications and standards make learning objects interoperable and reusable, but a number of shortcomings regarding current learning object metadata still exist:

- On one hand, current standards are purposefully descriptive instead of normative: they are intended to give information about the contents or the format of the learning object, but do not generally entail explicit run-time semantics for LMSs that use the learning object (Sánchez-Alonso & Sicilia, 2005). An exception to this is the IMS Simple Sequencing Specification, which allows representing "the intended behavior of an authored learning experience such that any learning technology system can sequence discrete learning activities in a consistent way that includes explicit runtime support" (IMS, 2003).
- On the other hand, the information in a learning object metadata record is not, as currently defined in international standards such as LOM (LTSC, 2002), machine consumption-oriented. In fact, most metadata in current learning object repositories are no more than an overall content identification and description, thus providing limited value from the viewpoint of delegation. In addition, information in metadata records is mostly in the form of unstructured texts written in natural language, such as faculty

member's peer reviews and comments, a kind of information that software agents would find it difficult to process. All this hinders the possibility of programming applications capable of "behaving" according to the information in a learning object metadata record (for example, to initiate a sequence of actions in a LMS according to given metadata elements).

These problems seriously hamper the automation or semi-automation of some processes such as search, retrieval or composition, thus limiting learning objects reusability.

Nowadays, the emergence of a global space for learning object-based education fosters the creation of the denominated learning object repositories. This kind of repositories, such as MERLOT (http://www.merlot.org/), CAREO (http://careo.ucalgary.ca/), Ilumina (http://www.ilumina-dlib.org/) or NLN Learning Materials (http://www.nln.ac.uk/materials/) to name a few, store learning object metadata, providing access to a large variety of educational materials and tools in order to promote sharing and exploration of knowledge. Current repositories are in fact Web-based organized catalogs of online learning materials –animations, case studies, lecture presentations, exercises, simulations, tutorials–, but also other related resources such as peer reviews, learning assignments, or user comments about the learning objects stored in these repositories. Learning object repositories are, in general, aimed to help faculty members to find online teaching materials for their courses, thus avoiding "reinventing the wheel" every time they need to create a new course. Very often, the educational materials in these repositories are available free of charge. Other common features provided to users are browsing and searching capabilities. Figure 1 shows the subject tree that allows to browse materials through a number of subcategories, as well as the search engine (top right) that allows direct access to resources, from the MERLOT home page.

Fig. 1: MERLOT search and browsing capabilities

It is important to remark that current learning object repositories, such as MERLOT, only store metadata information related to the learning objects, but not the objects themselves. Instead, learning objects are generally stored in different places, and access to them is provided in the repository through a hyperlink (see the metadata field “location” in Figure 2). Metadata records are provided to help educators and staff to find, evaluate, and share learning materials. However, even though most learning object repositories provide room for including metadata records, the lack of a conceptual model that clearly establishes what a learning object is and what the metadata descriptors associated to each different conceptualization are, constitutes a major problem. These repositories are called to play a central role in automated approaches to e-learning, since they provide the required support for learning object access and search facilities, oriented not only to humans but to software agents or systems (Sicilia et al., 2005). Consequently, completeness of metadata records becomes a key requirement. The problem is that, in most cases, metadata fields are incomplete and poorly structured. In particular, an examination of the materials in MERLOT shows that different objects include different degrees of completeness and even different metadata fields, what heavily depends on a number of factors:

- The learning object creator’s willingness to associate metadata information at the time of adding their materials to the repository.
- The editing capabilities or tools provided by the repository for learning object creators to associate metadata information to their materials. These capabilities heavily restrict the kind of information that can be stored and its level of formalization.
- The level of cognition or instruction of learning object creators on existing metadata practices and standards.
- The conceptual model of the repository, that is to say, what do the repository creators understand that a learning object is, and what structure the metadata information associated to it should have.

Figure 2 shows the metadata information associated to a particular learning object, “The Song Dinasty in China”, as stored in MERLOT. In this case, information on just a few metadata fields has been provided, mostly in the form of textual descriptions in natural language, which is not particularly useful form the point of view of automation. Software applications performing (by delegation of other agents) automated tasks such as the composition of new materials or the recommendation of specific elements based on pedagogical objectives, would find it difficult to process the information about the contents and educational objectives in this object, as it is included in the metadata field *description*.

Fig. 2: Metadata information for "The Song Dynasty in China", a learning object in MERLOT

The LOM *description* and *coverage* categories allow for the standard language-string specifications prescribed in the standard, but also for ontology-based annotations that will be discussed later. For example, to represent the geographic location of the learning object shown in Figure 2, the concept "China" in the TGN ontology has been used. Similarly, the individual "songDynasty" of the *OpenCyc* class "DynasticFamily" has been used to represent its temporal period. This way, the *coverage* category can be used to state both the geographic location and the temporal period of a particular learning object, by using expressions such as the following ones:

(TGN) situated-In [TGN-nation]: China
(OpenCyc) temporal-Period [OC-DynasticFamily]: songDynasty

Figure 3 shows the metadata information associated to a learning object in NLN Learning Materials, "Fuses and earth for electricity supply". It should be noted how metadata information can be considerably different across different repositories, which has a correlation to the conceptual model of the repository (item number four of the factors enumerated above). Again, software applications performing automated tasks would have difficulties in processing the metadata information about this object, because it is in a non-formal way (natural language). Besides the information shown in Figure 3, each learning object in the NLN Learning Materials repository has an associated metadata record called the "Tutor documentation", which includes additional information about the purpose, learning objectives, approximate study time, type of learning unit, prerequisite knowledge and preparatory activities of the object, as well as an outline of content, among others. This information, although useful, is not a proper metadata record according to existing standards in the field such as LOM, and consequently provides no additional help in the effort to standardize and unify the attributes required to adequately describe a learning object stored in a repository.

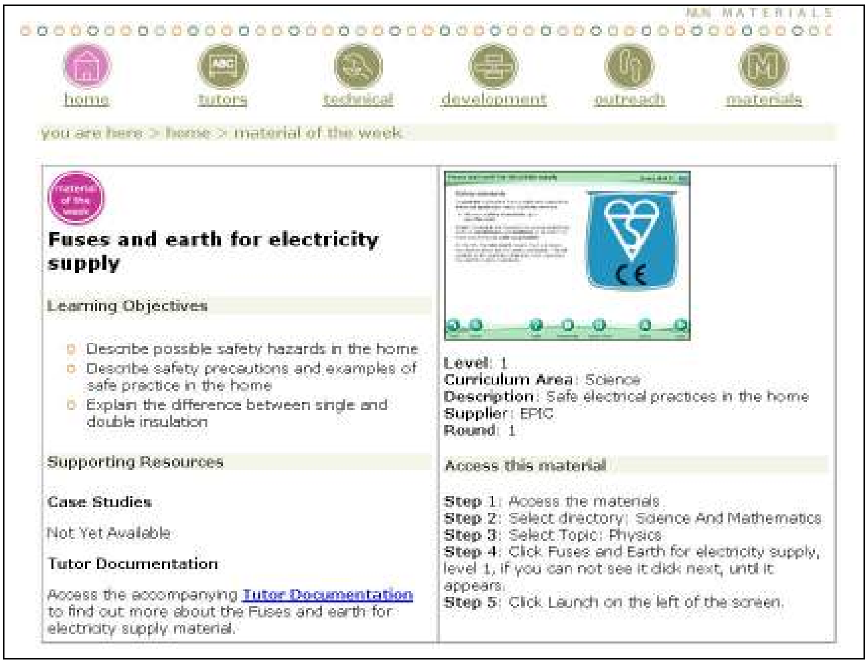

Fig. 3. Metadata information for a learning object in NLN Learning Materials

In considering the examples previously discussed, the need for a shared definition of learning object arises. A recent study by McGreal (2004) about the unification of the existing concepts into a single definition of learning object suggests the possibility of coexistence of all the available definitions. In this study, McGreal states that current terminology considers that a learning object is something that ranges from "anything and everything, through anything digital, to only objects that have an ostensible learning purpose, to those that support learning only in a particular or specific context". The study finally proposes a new definition based on the commonalities between all the definitions, which can be the basis of a new generation of more flexible repositories that support the different conceptualizations.

Regarding the automation or semi-automation of learning object management tasks, there is significant consensus on the need for including metadata information, either together with the learning object, or as a separate unit linked to it. This metadata information should be in a standard form, preferably IEEE LOM (LTSC, 2002) as this is a learning object-specific specification. As it will be analyzed later in this chapter, ontologies can be used to improve the quality of learning object metadata records. In particular, the use of semantic Web ontologies for different purposes can provide the foundations towards higher levels of automation (Sicilia et al., 2004a; Qin & Finneran 2002; Mohan & Brooks 2003).

Ontologies

Ontologies in a nutshell

In the field of philosophy, ontology is defined as the theory of objects and their ties. Therefore, the definition of a shared ontology for a given domain provides criteria for distinguishing different types of objects in the domain (concrete and abstract, existent and non-existent, real and ideal, independent and dependent) as well as their ties (relations, dependences and predication) (Corazzon, 2005). Faithful to this definition, modern science interested in knowledge conceptualization is intensively using ontologies for this purpose, what consequently makes it the main area of application of ontologies. For those looking for an in deep discussion on the topic, an interesting analysis of the philosophical term ontology is provided by Guarino and Giareta (1995). However, the philosophical aspects of the term are beyond the scope of this chapter.

Outside philosophy, ontologies can be understood as conceptualizations that provide an appropriate context for the interpretation of concepts in a given domain. An often-cited definition of the term by Gruber (1993) states that an ontology is “a specification of a conceptualization”. In this sense, ontology engineering becomes of particular interest when applied to conceptual modeling. In ontology edition, which can be easily carried out by using an ontology editor such as Protégé (http://protege.stanford.edu/), classes describe concepts in the domain of discourse. For example, a class of cars represents all cars. Specific cars are instances of this class. The red car parked in the airport parking just outside the terminal building is an instance of the class. A class can have subclasses that represent more specific concepts. For example, both *FamilyCar* and *Roaster* are subclasses (subsets) of the class *Car* in our example. Properties describe features or characteristics of classes and instances: the *color* property has the value *red* for the car in our example. Other properties that instances of the class Car will have are plate number, type of breaks, number of seats, or serial number of engine.

In practical terms, developing an ontology is a creative process that consists of a sequence of actions (Noy & McGuinness, 2001):

- Defining classes.
- Hierarchically arranging the classes in the ontology.
- Defining properties and describing their allowed values.
- Stating the values for properties for instances.
- Creating a knowledge base by defining individual instances of the defined classes.

From the existence of shared definitions in the form of a domain ontology, a property of special interest derives: ontologies are knowledge representations, currently based on description logics (Baader et al., 2003), which allows automated systems to perform tasks according to the inner logics in the ontology. This is the basis for applying the principles of Semantic Web in the domain of the ontology. However, creating a new ontology from scratch is a huge effort, as it implies defining all the classes in higher levels of the hierarchy. It would imply, in the previous example, to define classes such as *vehicle*, *conveyor*, *moving_object*, *object*, and so on, because all these definitions are needed before the class *Car* can be explicitly defined and situated in the right place in the full hierarchy of concepts. For most organizations, this is too much effort. Hopefully, there is a way to avoid defining every time all the concepts in a specific domain from which others derive: the use of the so-called upper ontologies or top-level ontologies.

Upper ontologies are large general knowledge bases that include definitions of concepts, relations, properties, constraints, and instances, as well as reasoning capabilities on these elements. Upper ontologies are limited to generic, high-level, and abstract concepts, general enough to address a broad range of domains. Moreover, upper ontologies do not include concepts specific to given domains, or do not focus on them. Their main objective is that of being used as the basis of a wide variety of intelligent processes, such as e-commerce and educational applications, or natural language understanding tasks. Though several upper ontologies currently exist, the most relevant ones are being considered by the IEEE Standard Upper Ontology Working Group (http://suo.ieee.org/) as part of the effort of creating a standard that will eventually provide "a structure and a set of general concepts upon which domain ontologies (e.g. medical, financial, engineering, etc.) could be constructed".

Opencyc (http://www.opencyc.org), an upper ontology "for all of human consensus reality", includes more than 6,000 concepts, 60,000 assertions about them, an inference engine, a browser for the knowledge base and other useful tools, what makes it one of the major efforts in the field. *Opencyc* is the open source version of the large *Cyc* knowledge base (Lenat, 1995), a huge representation of the fundamentals of human knowledge: facts, rules of thumb, and heuristics for reasoning about objects and events (Witbrock et al., 2005). Other remarkable works are IFF (Information Flow Framework, http://suo.ieee.org/IFF/) and SUMO (http://www.ontologyportal.org/).

Semantic Web ontologies and learning object metadata

Ontologies are a useful instrument for conceptual modeling. In particular, the existence of ontology-based schemas in the domain of discourse is essential when some degree of automation is desired, providing ground for delegation of tasks to automated or semi-automated systems. Assuming this, if a Semantic Web ontology is engineered with the specific aim of modeling the concept of learning object and the surrounding actions associated to the development and deployment of learning objects, definitions for all the elements (classes, objects, actions and properties) surrounding this term will necessarily be part of the ontology. Other elements to be considered would be, for example, all the issues related to each action concept in the ontology.

Learning object repositories play a key role in the vision of reusable learning contents and learning designs, serving as providers for learning-oriented artifacts. Nevertheless, current metadata creation practices result in artifact collections that lack machine-understandable metadata, which seriously hampers opportunities for reuse (Sánchez-Alonso & Sicilia, 2005). Semantic Web ontologies can be used to improve the quality of learning object metadata records, but they are not enough by themselves. In order to respond to requests by returning the adequate resources, repositories are required to be aware of the amount, type and quality of the metadata records they store. Semantic Web Ontologies applied to learning object-related metadata, can serve as:

- A means for the representation of knowledge levels on the learner side.
- A mechanism for the integration of learning object types, essential for the development of systems that are able to select and deliver learning objects.
- A form of knowledge representation aimed to enable richer behaviors than current linear lists of terms (vocabularies) as provided in learning object specifications such as LOM.

- A way to provide reasoning facilities to Learning Management Systems (LMS), enabled by the underlying description logics (Baader et al., 2003).

Previously in this chapter, learning object metadata has been said to be currently oriented to humans, which hampers the possibility of being processed by automated systems. Learning objects are artifacts capable of being used in different scenarios mainly including learning processes, but also system-to-system processes like purchasing, selection, composition or exchange (Sicilia et al., 2004b). Each scenario requires a given set of metadata elements specified in some specific ways. Therefore, providing more and better metadata to learning objects broadens the collection of scenarios in which they may be used, providing also an objective notion of reusability. If the objective is the construction of software that exploits metadata, a representation framework far beyond plain metadata records is required. The use of ontologies in the context of Semantic Web technology can open new scenarios where a higher degree of consistent automation in metadata information management can be achieved. Next section examines the role of semantic Web ontologies in the creation of more flexible learning object repositories. The purpose is to analyze in detail the kind of knowledge representations that can be used as the basis for extended learning object metadata specifications.

An ontology-based approach to learning object repositories

Main definitions

Current learning object repositories provide room for including metadata records, but lack a conceptual model that clearly establishes what a learning object is and what the metadata descriptors associated to each different conceptualization are. Recent studies about the unification of the existing concepts into a single definition of learning object suggest the possibility of coexistence of all the available definitions (McGreal, 2004). According to McGreal's study on existing learning object characterizations, five definitions –ranging from general to specific– coexist:

1. Anything and everything.
2. Anything digital, whether it has an educational purpose or not.
3. Anything that has an educational purpose.
4. Digital objects that have a formal educational purpose.
5. Digital objects that are marked in a specific way for educational purposes.

According to the first definition, the *use* of an object is what determines whether or not the object becomes a learning object, and thus everything that exists (the *universal* concept) can be considered a learning object. Nevertheless, in software-based representations, the only objects considered in practice as existing (in the sense of being able to 'talk' about them) are those that are *represented*. In our case, the scope of representation is that of the different elements of the ontology. Taking *OpenCyc* as a case of ontological representation, the term `Thing`, defined in *Opencyc* as "the collection which, by definition, contains everything there is", subsumes anything that *may eventually be considered* a learning object. This definition has the obvious drawback of not adding any defining characterization to the concept.

Learning can be considered an `Event`, defined in *Opencyc* as “a dynamic situation in which the state of the world changes.” Accordingly, everything that is linked to the representation of learning activities, or declared to have educational purpose in some way, should be considered a learning object. In addition, some axioms could automatically classify some things as learning objects. For example, “every `Book` classifies as a learning object”. These are examples of concrete characterization of classes of learning objects, which can be used for practical applications.

In consequence, the first definition may be interpreted in the following way: “**[1] LearningObject-AsAnything:** learning objects are things that either have been used in learning events or have been provided with descriptions that specify possible usages in learning”. The latter part of the sentence still requires much clarification, but it can be used provisionally until more detailed clarifications are proposed. An example of learning object that fits this definition would be a text book, a pen, or a printed copy of this chapter. Figure 4 depicts both this and the rest of definitions being discussed.

The second definition introduces the concept of “digital object” in an attempt to further specifying that learning objects are artifacts. The term `ComputerFileCopy` is defined in *Opencyc* as “an information bearing thing that is identified as a unit by a unique name, and which is object-like in an important respect”. Examples include individual image files, text files, sound files and executables stored on some `ComputerStorageDevice` (defined as “the collection of devices used by computers to store information”). Consequently, the term `ComputerFileCopy` can be used as a possible characterization of the concept, since it requires unique identification, and is not restricted to “data” but instead subsumes programs in a general sense. Thus the following definition can be used “**[2] LearningObject-AsAnythingDigital:** learning objects are `LearningObject-AsAnything` instances that are subsumed by `ComputerFileCopy`”. An example of learning object that fits this definition would be the PDF version of this chapter.

The third definition introduces a consideration of purpose. In this case, the purpose should be interpreted as something that was present in the act of `Designing` (“the act of designing something, be it clothing, cars, computer chips or buildings”, as defined in *OpenCyc*) the learning object, which entails the associated restriction of learning objects to be `Artifacts`, i.e. at least partially tangible things intentionally created by an `Agent` (or a group of `Agents` working together) to serve some purpose or perform some function, which separates them from “natural” things. This leads to definition [3]: “**LearningObject-AsAnythingWithEducationalPurpose:** learning objects are `LearningObject-AsAnything` instances that have *somewhat* a *record* of the educational purpose put in the object in the act of its `Designing`”. Here “record” is used in a generic sense and it may simply include the trace of the one that created it. Further, it can be considered that some tacit traces of an object again exist, as having being created with an educational purpose. An example of learning object that fits this definition would be a digital unit including both a PDF version of this chapter purposefully designed to serve a particular educational purpose, and a metadata record including specific information on the pedagogical use and goals of the chapter in an educational context.

Since the purpose in the design is an intellectual process, a notion of "record" of it should be introduced. Such purpose may be *internal* to the learning object, e.g. the "objectives" section in a Web page, but it could also be *tacit*, i.e. when it takes a form that is easily recognizable as an educational artifact. This may be the case of slide presentations. That notion of "record" of the purpose is deliberately kept open to divergent interpretations.

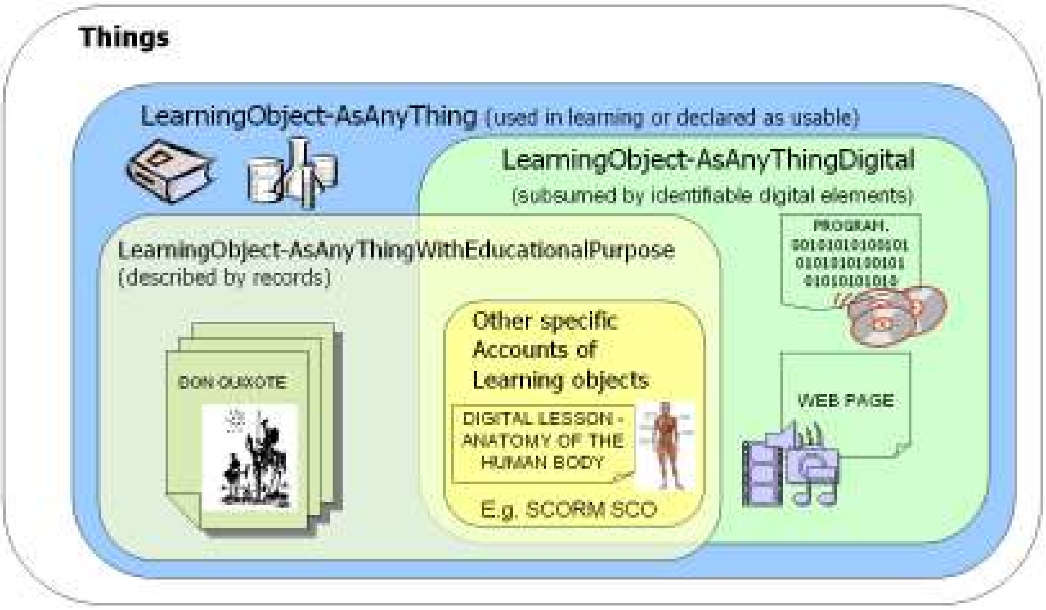

Fig. 4. Representation of different overlapping characterizations for learning objects

Definitions [2] and [3] provide two characteristics that are essential to every learning technology specification, as ADL SCORM (http://www.adlnet.org/scorm/) or IMS Learning Design (http://www.imsglobal.org/learningdesign/). They are concerned with Web contents (i.e. they are digital entities, even though some of them may be considered as digital surrogates to non-digital things), and they have some recorded metadata that is oriented to describe their educational purpose. Note that even in the case that no "educational-specific" metadata is provided, the mere existence of the metadata record in a standardized form is a sign of the fact that the digital entity was conceived for education or learning. In terms of the IEEE LOM standard, this means that even if the annotators do not fill the "Educational" metadata category and leave it empty (which is conformant with the standard), the Web contents can still be considered as learning objects. This finally leads to the definition "**[4] LearningObject-OtherSpecificAccounts:** learning objects are entities subsumed both by *LearningObject-AsAnythingDigital* and *LearningObject-AsAnythingWithEducationalPurpose* that have in addition some specific *form* in their records and contents". This definition can be roughly considered to cover definitions four and five in McGreal's study, even though other interpretations of McGreal's work are also possible. An example of learning object that fits this definition would the previously mentioned digital unit that includes a PDF version of this chapter and a metadata record. Both of them must conform to specific standards, like the SCORM content packaging specification and the IEEE LOM.

Semantics of the main definitions of learning object

In contrast to the definition of metadata by Greenberg (2002), the fact that metadata is created to support some specific function is sometimes overlooked or vaguely acknowledged. Even though some functions are tacit in metadata, e.g. a "subject" metadata element is obviously intended for the function of discovery, or "cost" is intended for a purchase activity, metadata creators are often not concerned with the concrete details of the requirements of the functions that will eventually make use of the metadata records they generate. Each of the above described learning object characterizations entails a different kind of requirement on both the form of the metadata used to describe the object and the kind of functions that can be enabled through them. Table 1 summarizes the main considerations about this issue.

Table 1. Type of functions enabled by the different characterizations of learning objects discussed.

Characterization	Required metadata	Type of functions enabled
[1]LearningObject-AsAnything	None	Human consumption
[2]LearningObject-AsAnythingDigital	None	Human consumption; tacit semantics
[3]LearningObject-AsAnythingWithEducationalPurpose	Something	Human consumption; tacit semantics with purposeful data fields.
[4]LearningObject-OtherSpecificAccounts	Mandated by an `Specification`	Human consumption; tacit semantics with purposeful and previously agreed data fields, including common vocabularies
Ontology-based descriptions conformant to specifications	Mandated by an ontology that represents the `Specification`	All the above plus formal semantics, with room for inference.

If we consider definition (1), then the concept of required (i.e. mandatory) metadata is not applicable. Consequently, the functions necessarily enabled are strictly those related to human consumption. This affirmation does not necessarily state that all the resources in that categorization are not machine-processable, but instead that the freedom in description and denomination of what a learning object is does not guarantee it. An example of metadata according to this definition would be an overall description about this chapter such as the following: *"An introduction to an ontological schema aimed at achieving flexibility in learning object description, with a focus on the automation of functions and the delegation of tasks to software systems"*.

Definition (2) adds the requirement "digital", thus introducing the possibility of exploiting tacit semantics (Seth, Ramakrishnan, & Thomas, 2005), i.e. those that can be extracted by mining and processing the contents of the objects. This includes the use of existing text summarization, keyword extraction and indexing that are used in information retrieval. Unfortunately, this kind of semantics, although useful, does not provide any significant novelty to the tools that are today commonplace in search engines. An example of metadata according to this definition would be an overall description of a digital copy of this chapter, assuming that a clear understanding of all the terms has been reached to avoid ambiguities. Although this metadata information is still human consumption-oriented, terms like *flexibility*, *learning object* or *automation* in the sentence above should have been 'semantically' defined as universally acknowledged keywords.

Definition (3) introduces the requirement for some kind of description. This represents an advance in that actual metadata has to be provided. Nonetheless, this can be simply an annotation in free form, which does not provide much in the general case in terms of opportunities for automated processing. Following the example, metadata according to this definition should include specific information on the purpose (or purposes) of this chapter, as for example: *"After learning this chapter, students will have an understanding of the role of ontologies in the automation of some tasks in e-learning environments"*.

Definition (4) progresses in the formalization of metadata by considering that metadata records must be conformant to some previously agreed `Specification` (defined in *OpenCyc* as "an abstract work that constitutes a description of the properties of a `Situation` or a `SomethingExisting`, and sometimes even entire collections of such things"). This is in fact the current state of learning technology specifications as LOM and SCORM, and provides much improved room for the exploitation of metadata as it incorporates higher levels of structure. An example of metadata according to this definition would be a conforming LOM metadata record including information like the following:

> 1.1.General.Identifier: *"URI", "http://www.slor.org"*
> 1.2.General.Title: *"Flexible Learning Object Repositories"*
> 1.3.General.Language: *"en"*
> 1.4.General.Description: *"An introduction to an ontological schema aimed at achieving flexibility in learning object description, with a focus on the automation of functions and the delegation of tasks to software systems."*
> 4.1.Technical.Format: *"text/html"*
> 5.6.Educational.Context: *"LOMv1.0", Other*
> 6.1.Rights.Cost: *no*
> 6.2.Rights.Copyright and Other Restrictions: *yes*

Last row in Table 1 represents a step further in the degree of structure. Concretely, it mandates that metadata records are provided in ontological terms, but not only as mere translations of the specifications. It requires the descriptions to be connected to large existing ontological structures, which provides increased opportunities needed for inference and exploitation of knowledge. For example, the *Coverage* metadata descriptor in LOM should be expressed through the definitions included in an ontology for geospatial grounding such as the *Getty Thesaurus of Geographic Names* (http://www.getty.edu/research/tools/vocabulary/tgn/), thus providing a more comprehensive and coherent representation of geographical entities. Metadata records annotated this way would enable formal inference in addition to the reuse of knowledge inherent to descriptions connected to large domain ontologies.

Designing flexible repositories through ontological characterizations

It can be stated, at this point, that all the definitions discussed in the previous section can fit together since they are subsumed by others. The recognition of such a diversity of conceptualizations, as well as the taxonomization of them, can be the basis of a neutral conceptual model. This new model would eventually provide users with a number of different functionalities, adapted to each particular concept of learning object, and not necessarily restricted by only one of them.

This approach can serve as the introduction to a new model of learning object repository, where flexibility is dealt with as a key issue for guiding the implementation. Based on a sound semantic model, defined by an ontology schema engineered using the Web Ontology Language (OWL, http://www.w3.org/TR/owl-features/), such a model would include all the definitions in McGreal's study, thus supporting different types of learning objects. In short, the repository clients –final users, agents, and learning management systems– could, among other functionalities, add, retrieve, modify and search for learning objects regardless the definition of learning object used by the learning object creator. For example, client software using a LOM-based model would be able to retrieve learning objects from such repositories, even though the objects that fulfill the requirements of the client system were stored by a different system that was using the SCORM definition of learning object.

From the different characterizations of learning object by McGreal, an ontology has been created. The most important concepts of this ontology appear in Table 2 mapped to the definitions given in the previous section.

Table 2. Mapping McGreal's definitions to ontological terms.

McGreal study terms	Classes in the ontology
Anything and everything	`LearningObject-AsAnything`
Anything digital, whether it has an educational purpose or not	`LearningObject-AsAnythingDigital`
Anything that has an educational purpose	`LearningObject-AsAnythingWithEducationalPurpose`
Digital objects with a formal educational purpose	`LearningObject-OtherSpecificAccounts`
Digital objects that are marked in a specific way for educational purposes	

If learning can be considered an `Event`, any term linked to the representation of learning activities, or declared to have an educational purpose in some way, may be considered a learning object for practical purposes. In Figure 5, all the classes prefixed by "oc" are classes in the original *Opencyc* knowledge base. Therefore, the term learning is represented by the class `oc_Learning`, which represents the definition of learning in *OpenCyc*. The abstract representation from which all the terms in McGreal's terminology derive is defined as the `LearningObject-Generic` class. An instance of this class may be anything `used_in oc_Learning`. The `LearningObject-AsAnything` class encompasses the broad definition of every possible meanings of learning object (a traditional book like *Shakespeare's Hamlet*, a note, applets, all kinds of Web pages, a sheet of paper with schemas or a questionnaire would all fit this definition). The `LearningObject-AsAnythingWithEducationalPurpose` class has been defined to represent objects for which some declaration of their pedagogical purpose exists (such as books with educational purpose, practices, exercises or questionnaires), while the class `LearningObject-`

AsAnythingDigital represents digital objects (such as Web pages, applets, digital pages, software programs and e-books, for example). These two latter characterizations are combined in current specifications of learning technology. In addition, as current learning object standards and specifications suggest, an individual of the class LearningObject-AsAnythingDigital should be linked to at least one LearningMetadataRecord, by stating that the domain of the property hasAssociatedMetadataRecord is the class LearningMetadataRecord. LearningMetadataRecord is a generic term that can be used to derive specific terms supporting each particular specification such as LOM_Record or SCORM_SCO_Manifest. Any specification-specific learning object is, by its own nature, something with a declared educational purpose, which is at least tacit in the standardized schema. Specific accounts of learning object, such as LearningObject-LOM or LearningObjectSCORM_SCO provide room for conceptual lessons with an educational purpose and a specific digital format, such as *Moodle* courses (http://moodle.org/), *ATutor* courses (http://www.atutor.ca/) and the like. Figure 5 shows the relationships between all the terms in the ontology, as described.

In this figure, classes in the ontology are shown as solid-outline rectangles, with several compartments separated by horizontal lines. The name of the class is held in the top compartment. The rest of the compartments hold both the properties of the class, and the restrictions defined for it. Object properties are shown preceded by a circled "o", while restrictions are preceded by a circled "R". For example, the class LearningObject-Generic is linked to the class oc_Learning by an arrow that represents the object property used_in in the corresponding compartment of the class. Note how the arrow is labeled with the name of the property, used_in in this case. The figure also shows the hierarchical relationship between the classes in the ontology. Classes representing more generic concepts appear in the top part of the figure. The hierarchical relationship is shown as a solid path from the more specific classes (e.g. LOM_Record) to the more generic ones (e.g. LearningMetadataRecord), with a hollow triangle at the end of the path where it meets the generic class. Other links, like those labeled with the intersection and the equivalence symbols, represent the kind of formal definition for a class as stated in the ontology. The class LearningObject-AsAnythingDigital, for example, is linked to an intersection symbol by an arrow labeled with an equivalence symbol. This is to represent the fact that the class LearningObject-AsAnythingDigital is defined as the class of objects that are instances of two classes simultaneously. In this particular example, instances of LearningObject-AsAnythingDigital accomplish the properties in both the class oc_Learning and the class LearningObjectAsAnything.

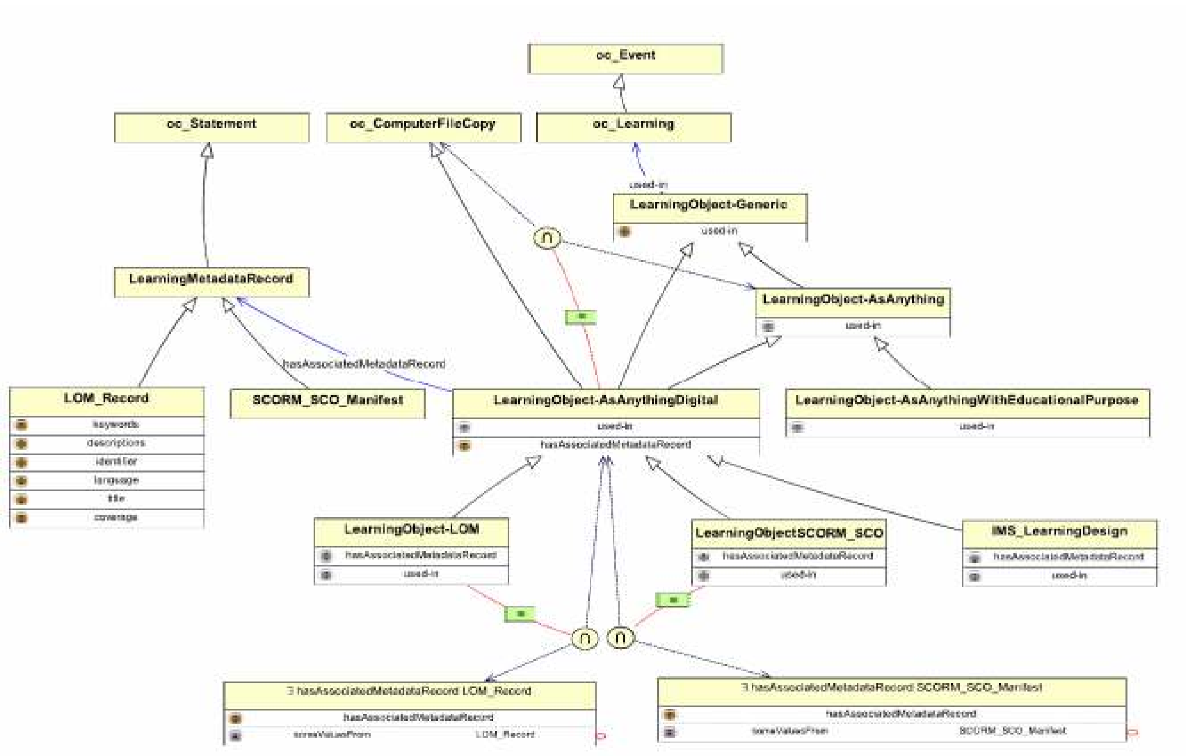

Fig. 5. The SLOR ontology

Semantic Learning Object Repositories

The SLOR prototype

The SLOR –Semantic Learning Object Repository– prototype has been specifically designed for the creation and management of learning object metadata with integration and exchange purposes. Figure 6 depicts the main layers and technologies of a semantic-enabled learning object repository prototype, as an illustration of its architecture. This architecture is structured around three layers: the model interface layer, the model service layer, and the model persistence layer. In order to maintain the consistency between the different layers, the SLOR prototype makes use of the ontology described before in this chapter. Functionalities are grouped in modules, following the principle of scalability.

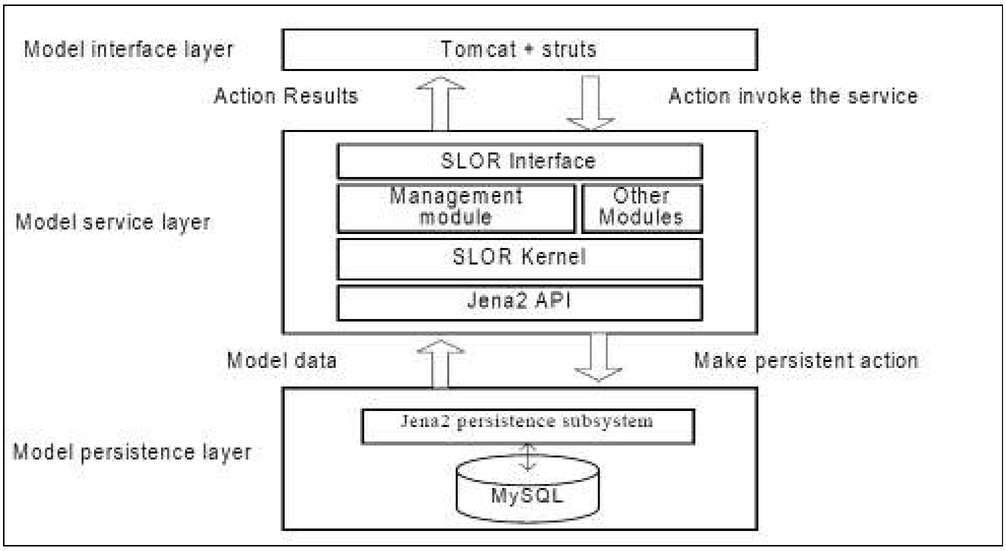

Fig. 6. The SLOR architecture

In order to allow the Model-View-Controller separation of Web logic, the model interface layer is built with *Apache Struts* technology (http://struts.apache.org). The struts actions invoke the SLOR services and retrieve the results, which will be adapted to the current user interface because different learning object models on the client side would probably use a different interface. This case is a good example of the flexibility that the underlying ontological model gives to the prototype: the conceptual model of the client end does not determine the SLOR functionalities, as all the features in SLOR are designed in a model-independent way. Therefore, the results of all the functionalities in SLOR provide results that fit all the different conceptualizations in the ontology. This is an extensible model, prepared to support future conceptualizations that could be included in the ontology.

In order to transfer information between an input form and a struts action, the struts framework has defined the *Action Form Bean* class. Each *Action Form Bean* maps to a class in the ontology, e.g., the *LearningObject_LOMActionForm*, maps the class `LearningObject-LOM`.

The model service layer provides transparent access to different functionalities of the semantic learning object repository:

- The SLOR interface defines a protocol of behavior between this layer and the actions of the user interface. This interface separates the GUI from the SLOR services. The design principles of SLOR interface are based in metadata actions, implemented in modules that enable learning objects accessibility, interoperability, durability and reusability. This interface receives requests from either the Web interface or other agents using the SLOR ontology that are routed to the appropriate module in the following layer.
- The SLOR modules provide a scalable architecture that allows to easily adding new functionalities. Herein, a modular scalable architecture for expanding system functionalities through continued addition of modules is proposed, with the aim of advancing towards a standard architecture for SLOR. In order to include all the

functionalities related to the creation, deletion and update of reusable learning objects, a management module has been implemented.

- The SLOR kernel provides a middleware with basic functions for operating the ontology model. Examples of functions provided by this kernel are: *getIndividuals* (which retrieves all the individuals of a given class in the repository) or *setMultipleProperty* (which inserts all the objects in a list as the values of a multiple property of an instance).
- The Jena (http://jena.sf.net) API is used to handle at low-level the OWL ontology model.

Persistence is the capability to store data structures in files or relational databases. Jena provides transparent persistence for ontology models through the use of a database engine. This feature allows the easy management of storage of great models (instead of using xml files). In SLOR, the persistence of the underlying ontological model is stored in the MySQL database, using the built-in persistence capabilities of the Jena framework.

An example of operation: The SLOR Web interface

As an example of the functionalities provided by the SLOR prototype, some will be described in more detail. For the sake of illustration, the functionalities later described will be invoked through the SLOR Web interface, even though this is neither the only possibility nor preferred one. The SLOR interface is mostly oriented to external agents, which would invoke SLOR methods via an agent-oriented negotiation protocol without the intervention of any Web interface. However, the Web interface is shown here to clarify the explanations and to provide a better set of visually helpful examples.

Creation of metadata records

The creation of a new metadata record is implemented through the *createIndividualLearningObject_LOM* function. This function allows including new learning object metadata in the repository by using a specific interface. The entry fields in the creation form correspond to a given conceptual model of LO, (`LearningObject-LOM` in the example in Figure 3). However, other models –listed in the left hand panel in Figure 7– could be used for storing metadata in the repository.

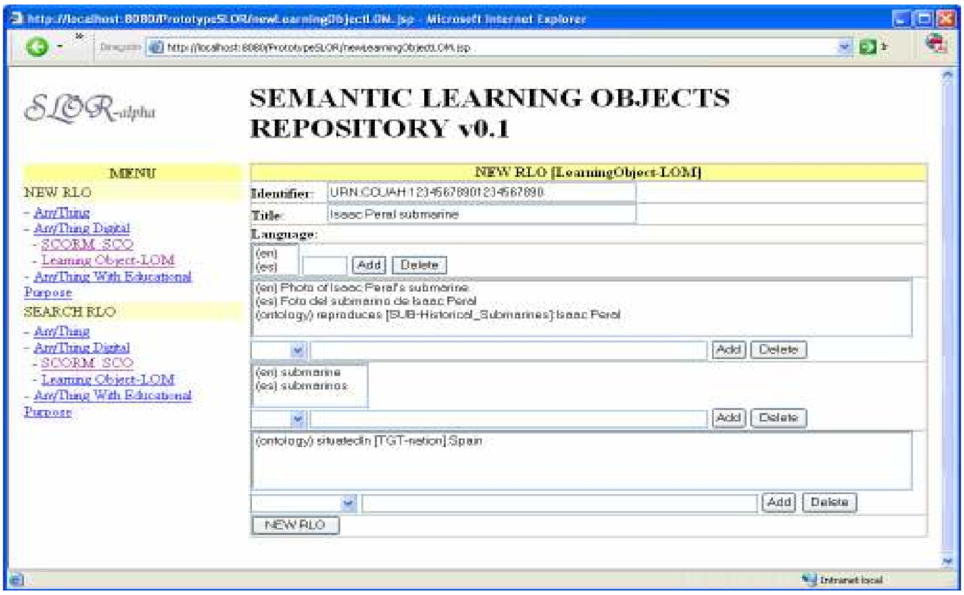

Fig. 7. The creation of a metadata record in SLOR

When the LO metadata creator clicks on the “New RLO” button, the Web client sends a *NewRlo* (new reusable learning object) request to the Tomcat Server, which in turn calls a Struts action that uses the SLOR interface to invoke the method that creates an individual learning object metadata record in LOM format.

This method is part of the SLOR management module. This method obtains a LOM_Record reference by invoking the *getOntClass()* method. Then, an instance (*instMetadataLOM*) is created with the –`LOM_Record`– ontology class reference. Invoking the *addProperty*() method on an individual, allows setting values for all the properties in the *instMetadataLOM* metadata record. As previously discussed in this chapter, metadata information can be linked to specific information items in external ontologies, which is the case of the example shown in Figure 8. The SLOR Kernel class *KernelProperty* is used to handle complex functions (i.e. to set multiple property values through *setMultipleProperty()*). Finally, an individual is created, included in the model and associated to a `LOM_Record`.

Searching capabilities

Metadata information, as stored in SLOR, is linked to ontology elements that evoke concepts. This way, most items in a metadata record are linked to their corresponding elements in a domain-specific ontology for each category, which determines the number and type of values that the item can hold. However, not all the items are linked to ontology elements, as some can be of “simple” data types and thus holding values such as a date, an integer, or the like. Figure 8 shows an example of metadata edition where part of the information in the metadata category *coverage* is set to the value *Spain*, an instance of the *TGN-Nation* class in the previously-mentioned TGN ontology, which evokes the concept of Spain as a country. This scenario gives support to a deep level of search that allows making complex queries, e.g. retrieving elements of the baroque period that are situated in Spain.

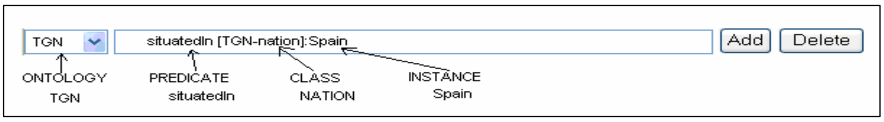

Fig. 8. Linking metadata information to ontology elements in SLOR

Search based on semantic criteria, i.e. "semantic search", is implemented through the *semanticSearchLearningObject*() function. This function allows searching instances of concepts in the ontology model such as retrieving all learning objects marked as "digital" or those learning objects that have an educational purpose. Several restrictions can be defined as part of the searching process. Restrictions allow filtering learning objects on different criteria (pedagogical, economic, or other), thus providing a set of results that better accomplishes the end-user needs.

Browsing capabilities

Learning object browsing is implemented as an ontology-based seeking interface. The browser's role in this model is to allow that any metadata category in LOM (excluding lifecycle and meta-metadata that are not related to the educational purposes of the objects) can be used as top guiding criteria. The ontology terms attached as descriptions are displayed in the browser, and it is finally the user who has the decision on their selection. The result of user selection is a query expression formed by a collection of ontology terms. Queries here are by default interpreted in a contextual basis, i.e. all the requirements selected by the user should be matched in the same context of the leaning object. Figure 9 shows an example of navigation (note that metadata elements in LOM can again be used as filters).

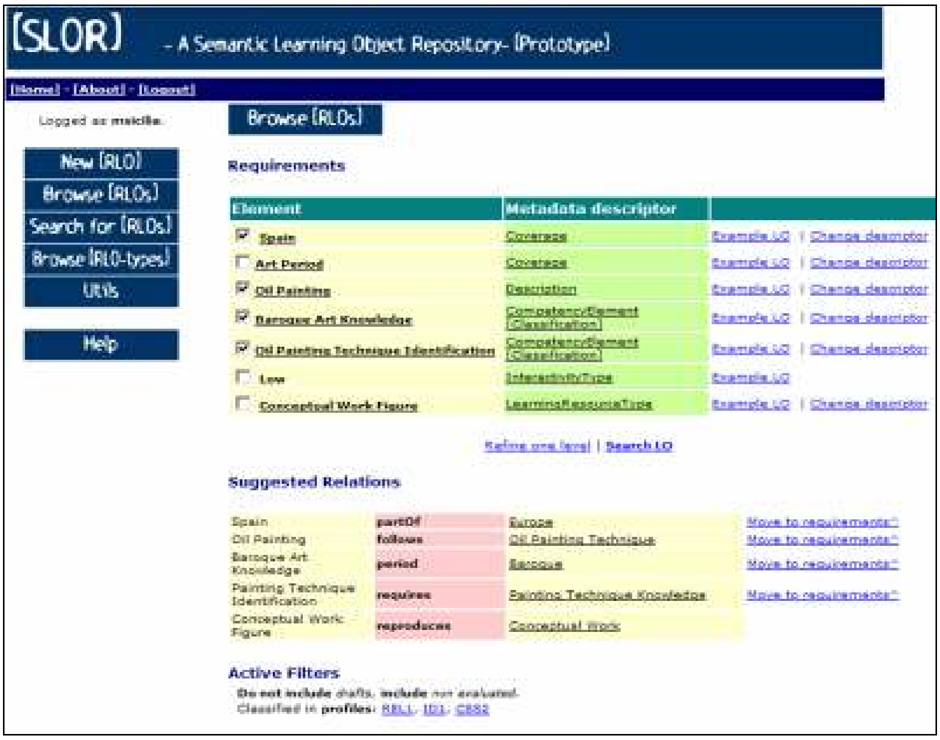

Fig. 9. An example of navigation

Figure 9 shows a query construction step in which several elements are included as requirements. These elements are terms –like “Art_Period”– or instances of terms –like “Spain”– that characterize a particular aspect of the metadata, what is shown in the second column. In particular, “Art_Period” is a term included as part of the *Getty AAT* (Art & Architecture Thesaurus, http://www.getty.edu/research/conducting_research/vocabularies/aat/); a vocabulary “intended to provide terminology and other information about the objects, artists, concepts, and places important to various disciplines that specialize in art, architecture and material culture”. All the underlined elements are actually links to additional information for the ontology element, and the “Example LO” functionality works like a partial search in the sense that it retrieves from the knowledge base a learning object that fulfils that requirement. This mode of fulfillment is an adaptation of that described by García and Sicilia (2003), in which resources linked by relations to the selected terms or instances are retrieved, with a relevance that depends in the form they are linked. For example, the selection of “Spain” in *coverage* would pre-select all the instances connected to that instance with any arbitrary relationship as part of the *coverage* section like “situated in”. In addition, the “Suggested relations” section shows arbitrary terms or instances connected to the ones selected in the requirements area, and the associated elements can be added as requirements. The “refine one level” link traverses one level through the primitive hierarchy as a way to narrow down the search criteria. This navigational structure allows the use of the ontology as a driver for guided search.

Conclusions

In this chapter, an ontological model that supports the learning object definitions in McGreal's study has been described. The ontology here presented serves as the basis for the creation of a flexible learning object repository. A prototype that supports the previous model, SLOR, has been briefly sketched and its implementation architecture described. Further work should progress in the implementation of more functionalities as well as in the validation of the ones implemented so far.

The different conceptions of learning objects as summarized by McGreal [1] lead to different ontological characterizations of learning objects. If a repository of learning object is to cover such different notions, it requires associated definitions for these different characterizations. This chapter has sketched a possible schema for that purpose. In addition, the kinds of functions that are entailed by each of the characterizations have been discussed and examples of operation have been provided.

References

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

Baader, F., Calvanese, D., McGuinness, D., Nardi, D. & Patel-Schneider, P. (2003), *The Description Logic Hanbook*. Cambridge, UK: Cambridge University Press.

Corazzon, R. (2005). *Ontology. A resource guide for philosophers*. Retrieved July 1, 2005, from http://www.formalontology.it

Downes, S. (2004). Resource Profiles. Journal of Interactive Media in Education, 2004(5). Retrieved July 1, 2005, from http://www-jime.open.ac.uk/2004/5/

Friesen, N. (2005). Interoperability and Learning Objects: An Overview of E-Learning Standardization. *Interdisciplinary journal of knowledge and leraning object*, vol. 1(2005), 23-31. Retrieved July 1, 2005, from http://ijklo.org/Volume1/v1p023-031Friesen.pdf

García, E. & Sicilia, M. A. (2003). Designing ontology-based interactive information retrieval interfaces. *Proceedings of the Workshop on Human Computer Interface for Semantic Web and Web Applications* (pp. 152–165).

Greenberg, J. (2002). Metadata and the World Wide Web. In M. Dekker (Ed.), *Encyclopaedia of Library and Information Science* (pp. 1876-1888). New York: Basel.

Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2): 199-220.

Guarino N., & Giareta, P. (1995). Ontologies and knowledge bases: Towards a terminological clarification. In N.J.I. Mars (ed.): *Towards Very Large Know ledge Bases* (pp. 25-32). Amsterdam, NL. IOS Press.

IEEE LTSC (Learning Technology Standards Committee) (2002). Learning Object Metadata (LOM), Final Draft Standard, IEEE 1484.12.1-2002. Retrieved July 1, 2005, from http://ltsc.ieee.org/wg12/

Lenat, D. B. (1995). Cyc: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM*, 38(11), 33-38.

McGreal, R. (2004). Learning Objects: A Practical definition. *International Journal of Instructional Technology and Distance Learning*, 1(9).

Mohan, P., & Brooks, C. A. (2003). Engineering a future for Web-based learning objects. In J. Cueva-Lovelle, B. González-Rodríguez, L. Joyanes-Aguilar, J. Labra-Gayo & M. del Puerto (Eds.), *Web Engineering.*

International Conference, ICWE 2003, Oviedo, Spain, July 14-18, 2003. Proceedings, Vol. 2722 of Lecture Notes in Computer Science (pp. 120-123). Springer-Verlag.

Noy, N. & McGuinness, D. L. (2001). Ontology Development 101: A Guide to Creating Your First Ontology. *Technical Report KSL-01-05*, Knowledge Systems Laboratory, March 2001.

Polsani, P. (2002). Use and abuse of reusable learning objects. *Journal of digital information*, 3(4). Retrieved July 1, 2005, from http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani

Qin, J., & Finneran, C. (2002), Ontological representation of learning objects. In *Proceedings of the Workshop on Document Search Interface Design and Intelligent Access in Large-Scale Collections*.

Sánchez-Alonso, S., & Sicilia, M. A. (2005). Normative Specifications of Learning Objects and Learning Processes: Towards Higher Levels of Automation in Standardized e-Learning. *International Journal of Instructional Technology and Distance Learning*, 2(3). Retrieved July 1, 2005, from http://itdl.org/Journal/Mar_05/article01.htm

Seth, A., Ramakrishnan, C., & Thomas, C. (2005). Semantics for the Semantic Web: The Implicit, the Formal and the Powerful. *International Journal on Semantic Web and Information Systems*, 1(1), 1-18.

Sicilia, M.A., & Lytras, M. (2005). Scenario-Oriented Reusable Learning Object Characterizations. *International Journal of Knowledge and Learning*, 1(3).

Sicilia, M.A., & Sánchez-Alonso, S. (2003). On the concept of learning object "Design by Contract". *WSEAS Transactions on Systems*, 2(3), 612-617.

Sicilia, M. A., García, E., Sánchez-Alonso, S. & Rodríguez, E. (2004a). Describing learning object types in ontological structures: towards specialized pedagogical selection, in L. Cantoni & C. McLoughin (Eds.), *Proceedings of EdMedia 2004 - World Conference on Educational Multimedia, Hypermedia and Telecommunications* (pp. 2093-2097). AACE.

Sicilia, M. A., Pagés, C., García, E., Sánchez-Alonso, S., & Rius, A. (2004b). Specifying semantic conformance profiles in reusable learning object metadata. In *Proceedings of the 5th International Conference on Information Technology Based Higher Education and Training* (pp. 93- 97). Istambul, Turkey: IEEE.

Sicilia, M.A., García-Barriocanal, E., Pagés, C., Martínez, J.J., & Gutiérrez, J.M. (2005) Complete metadata records in learning object repositories: some evidence and requirements. *International Journal of Learning Technology*, 1(4), 411-424.

Welty, C., & Guarino, N. (2001). Supporting ontological analysis of taxonomic relationships. *Data and Knowledge Engineering*, 39(1), 51-74.

Wiley, D. A. (2001). Connecting learning objects to instructional design theory: A definition, a metaphor and a taxonomy. In D. A. Wiley (Ed.), *The instructional use of learning objects* (pp. 3-24). Bloomington, Indiana: Agency for Instructional Technology and Association for Educational Communications & Technology.

Witbrock, M., Matuszek, C., Brusseau, A., Kahlert, R.C., Fraser, C.B., & Lenat, D. (2005). Knowledge Begets Knowledge: Steps towards Assisted Knowledge Acquisition in Cyc. In *Papers from the 2005 AAAI Spring Symposium on Knowledge Collection from Volunteer Contributors (KCVC)* (pp. 99-105). Stanford, USA.