Ontology for Organizational Learning Objects based on LOM Standard

André Luís Andrade Menolli
Computer Science Department
State University of North Paraná – UENP
Polytechnic School
Postgraduate Program in Computer Science - PPGIa
Pontifical Catholic University of Paraná – PUCPR
Brazil
menolli@uenp.edu.br

Sheila Reinehr, Andreia Malucelli
Polytechnic School
Postgraduate Program in Computer Science - PPGIa
Pontifical Catholic University of Paraná – PUCPR
Curitiba, Brazil
malu@ppgia.pucpr.br, sheila.reinehr@pucpr.br

Abstract – Organizational learning is an area that helps companies to improve their processes significantly through the reuse of experiences. An area that may help in this way is social learning. Collaborative tools, such as social network and wiki, enable collaborative work and are important facilitators of social learning process. However, collaboration is one of the several necessary components for learning. Therefore, it is important that all acquired knowledge be organized to be reused faster, easily and efficiently. Therefore, we propose using learning objects to organize the content inserted in collaborative tools. There are some learning object metadata to describe relevant learning objects characteristics and to catalog them. As these metadata are proposed to describe educational learning objects, they do not contemplate organizational characteristics, important for knowledge-intensive organizations. Moreover, the metadata are formally modeled through the XML-Schema language, which has a lack of expressiveness. Thus, trying to solve these limitations, the paper presents an ontology for organizational learning object based on IEEE LOM standard. The paper describes the ontology building process, following all the activities proposed in Methontology. Some experiments to evaluate the ontology are also presented

Keywords - ontology; organization learning; learning object; LOM.

I. INTRODUCTION

Knowledge is an essential property for companies in contemporary economics. More than ever before, knowledge has been spread out among individuals, teams and organizations. Thus, the capacity to create, acquire, integrate, implement and disseminate knowledge has emerged as a fundamental competence for organizations in general [1]-[2]. To be successful, companies must not only explore current knowledge but must also continuously invest in the search for new knowledge as strategic options for future decisions and as a way to develop competitive edge [3].

Many researchers have tried to identify factors that could help, or even automate learning in the corporate environment, some of them in the software engineering area, because it is a knowledge based processes area. Among these researches, some of them focus on collaboration and communication between developers. This occurred mainly with the advent of new technologies, like social network, wiki and blogs, facilitating the communication among people.

Through these new technologies and social environments, virtually anyone can create knowledge and make it available to be accessible and possibly useful to others. Therefore, the learning happens socially, with people creating and sharing knowledge dynamically. However, for social learning occur some characteristics are needed, such as trust in the social relationships and a way for discussions and ideas exchanging leading to collective knowledge construction. Hence, instead of designing technologies that "teach" the learner, the new social learning technologies will perform three main roles: 1) support the learner in finding the right content; 2) support the learner to connect the right people; and 3) motivate/incentivize people to learn [4].

Nevertheless, although most of these tools support collaborative work, this kind of tool does not provide ways to achieve the required characteristics for learning occur satisfactorily. The collaborative tools often provide an efficient way to collaborate and create knowledge, such as wiki, that according to Kimmerle et al. [5] may help both the process of internalization and externalization of knowledge, using the constructivist approach [6]. Externalization occurs through the writing of texts, which leads to the realignment or improvement of cognitive schemes. Internalization occurs through bits of information from wiki, which are decoded and incorporated in internal structures of existing knowledge. This creates new knowledge entities in the person's cognitive system, new associations among knowledge entities and new schemes.

However, besides collaborative tools, it is necessary other mechanisms for learning takes place within organizations. Thus, we consider that contents included in collaborative tools can be organized as learning objects (LO), facilitating the search, evaluation and reuse of information.

A LO is defined as any digital or non-digital entity that may be used, reused, or referenced for learning, education or training [7]. Polsani [8] states that a learning object is a content independent and autonomous unit, which may be reused in several teaching contexts. In this perspective, the need to manage reusable resources has driven the development of several metadata specifications in order to represent learning content. Thereby, among several metadata standards to represent content as a learning object stand out the Learning Object Metadata (LOM), Sharable Content Object Reference Model (SCORM) and Dublin Core.

The metadata specifications are useful to describe educational resources, and thus to facilitate interoperability and reuse between learning software platforms, as they represent the vocabulary describing the different aspects of the learning process. However, the main drawback is that the meaning of the specification is usually expressed in natural language. Although this description is easy to understand for humans, it would be difficult to be automatically processed by software programs [9]. To solve this issue, ontologies [10] come handy to describe formally and explicitly the structure and meaning of the metadata elements; that is, ontology would semantically describe the metadata concepts. Ontologies can describe a hierarchy of concepts connected by subsumption relationships, a concept more aligned with taxonomies; or a structure where axioms are added to express relationships among concepts and to limit their intentional interpretations [11]. Axioms make ontology more expressive by allowing the use of inference mechanisms.

The use of ontologies in education is not a new idea since [9], [12] and [13] analyze the application of ontologies in the education area. However, all these researches are towards to educational environment, and they do not contemplate organizational characteristics and needs.

In this context, this paper aims to propose an Ontology for Organizational Learning Objects (OOLO), based on LOM standard.

The paper is structured as follows: in the next section, the motivations to construct the proposed ontology and the limitations of the LO standards in an organizational environment are discussed. Section III describes the LO standards, which are the starting point for the proposed ontology; Section IV introduces the proposed ontology; Section V describes the evaluation and discussion; and, finally, Section VI presents the final considerations.

II. THE NEED FOR ORGANIZATIONAL LEARNING OBJECTS ONTOLOGY

In knowledge-intensive organizations, business processes are typically complex and weakly structured [14]. Therefore, they are not capable of being a direct base for the development of business process that supports knowledge infrastructures [15]. A commonly used approach to overcome this problem is to identify and model organizational knowledge processes based on business processes that visualize relevant, executed knowledge work in different ways [16].

The knowledge creation in the organizational level is a process in which the organization extends the knowledge created by the individuals and consolidates it as part of the organization knowledge net [17]. This process occurs inside an interaction community that crosses organizational borders and intra and inters levels [18]. We consider that the information created inside the organization can be organized in learning objects, aiming to better organize their content, facilitating the knowledge reuse.

In [19] is presented a semantic collaborative organizational environment, combining semantic resources, organizational learning and learning concepts, to support organizational learning in software development organizations thus maximizing team members learning.

Fig. 1 provides a general overview of the proposed architecture for an organizational learning semantic environment [19]. The architecture is subdivided into two key tiers: the application tier and the organizational memory tier, which is organized into three sub-tiers: interoperability, manipulation and knowledge.

The *Application Tier* is responsible for the user interaction and provides subsidy for the content inclusion, creation of instructional modeling domain, besides of present an interface to search in the organizational memory. This tier is composed of two components with distinct functions [19]:

(i) user interaction components: composed by the collaborative tools like wikis and whiteboards. These tools are configured according to MDI, to promote easy inserting of information respecting the domain of knowledge;

(ii) semantic search: allows semantic research to be carried out in the organizational memory based on searches for the consultation language for the Resource Description Framework (RDF), a SPARQL[1].

---

[1] http://www.w3.org/TR/rdf-sparql-query/

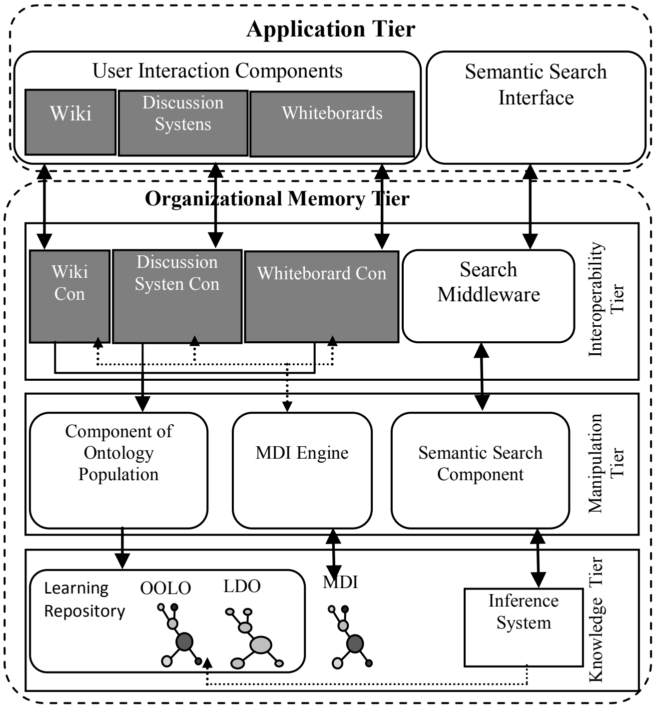


Figure 1. Semantic collaborative organizational environment. Adapted from [19]

The *Organizational Memory Tier* is responsible for storing all knowledge generated in the application tier, as well as the manipulation of this content to organize it according to the representation defined by the organizational memory and their respective tiers: interoperability, manipulation and knowledge [19].

The *Interoperability Tier* provides an extensibility mechanism to allow the incorporation of new collaborative tools in the described architecture. In addition, it makes the interaction with the application tier homogeneous, providing a common knowledge representation language. This tier is composed of a collection of connectors that interact with the application tier to provide representation of the extracted knowledge from collaborative tools and submit them to lower tiers, organizing collaborative tools at the same time respecting the definitions proposed by ontologies located in knowledge tier, especially MDI. Another responsibility of interoperability tier is to interpret inferred knowledge from the lower tiers to make them available to the application tier [19].

The Manipulation Tier is responsible for the manipulation of data in the upper tiers and forwards them to the knowledge tier. This tier has three key components with different functions [19]:

(i) MDI engine: that is responsible for providing information on the MDI structure, sending them to the collaborative tools connector to synchronize them and organize them correctly, according to what has been defined in the MDI.

(ii) Component of Ontology Populations: this component populates the LDO from the content inserted in application tier. To this it, is used the ontologies population technique [20], thus creating learning objects and units of learning, with contents generated in the application tier.

(iii) Semantic search component: organize the consultations for the inference engine and controls inferred knowledge, organizing the results, handling errors, exceptions and unexpected behaviors during execution.

The last tier of the proposed architecture is the *Knowledge Tier*. The knowledge tier is fundamental to the proposed architecture. It receives the information provided by different tools and organizes them into learning objects, and these are the base to create units of learning [19].

The content and material introduced in application tier is organized in units of learning. A unit of learning defines a general module of an educational process, like a course [21], and due to this fact, its content need be organized in a correct sequence, according to the domain.

Finally, in the knowledge tier there is the inference system, which carries out searches on LDO and competence ontology. As a result of this process, inferred knowledge is forwarded to the manipulation tier.

Thus, the objective of this architecture is the generation of specific knowledge objects, through the exchange of knowledge among team members, according to educational and domain models, developed by the organization, organizing knowledge to be reused and easily assimilated [19].

Consequently, one base component in the architecture proposed in [19] is OOLO, that is an ontology to represent the content included in application tier as LO, so in this paper we show this ontology.

To organize the content into LO it is necessary to use some LOs metadata. However, the existing standards are focused on educational domain and do not contemplate some important organizational aspects as: What is the person role that posted the object? Who did review the material? Who can use the object? Can the object be used outside the organization? Which project inside the organization can access the object? What is the object's context and domain? Was the content produced within the organization?

Moreover, all the main learning object standards, like LOM, SCORM and Dublin Core are formally modeled through the XML-Schema language, and there is a lack of specific ontologies. Nevertheless, there are some proposed ontologies to learning domain, like [22] that presents an ontology to design and management of material to assess the teaching learning process according to the IEEE Learning Object Metadata Standard, and [9], that presents an ontology to represent the semantics of the IMS Learning Design.

However, according to [23] there is no consolidated public Web Ontology Language (OWL) ontology to define the metadata standard properties [23]. The main XML-Schema limitations are [24]:

- Hierarchical (*is-a*) relations between two or more concepts can not be explicitly defined. Therefore, there are no inheritance mechanisms to facilitate the representation of concept taxonomies.
- Relations properties can not be defined. XML-Schema language does not provide primitives to represent neither mathematical nor taxonomic properties (disjoint and exhaustive partitions) of a relation.
- General and formal constraints (or axioms) between concepts, attributes, and relations can not be specified. These axioms describe more precisely the semantics of the concepts as they constrain how the instances of the concepts can be created.

However, even with these limitations these standards are consolidated and widely used. The next section presents some standards that are the basis for the proposed ontology.

III. LEARNING OBJECTS STANDARDS

The LOs metadata can help to provide LO reuse in different locations [25]. The reuse consists of an efficient way to readapt the LOs content for different contexts and users. The LOs metadata describe the relevant LO characteristics, used for cataloging their reusable LO repositories, which can later be retrieved by search engines or used by Learning Management Systems (LMS) to compose learning units [26].

Standards Associations such as IEEE (1484.12.1 Standard for Learning Object Metadata) and ISO (SC 36 WG 2 – Information Technology for Learning, Education, and Training) have created working groups to develop proposals for objects structuring and categorization (metadata), aiming to support the LO cataloging to be properly recovered and reused [26]. Thus, these metadata represent a way to organize data from LO to provide communication between different computing environments, as well as, its accessibility and usability, and ensure its interoperability. Some of these metadata are:

- LOM: describes important LO features with the purpose of facilitating the search and use of LOs for instantiation by learners and instructors or automated software processes [7].
- SCORM: is a collection of standards and specifications for web-based e-learning. SCORM is a specification of the Advanced Distributed Learning (ADL) initiative, and focuses on interoperability and reusability of LO. SCORM introduced a complex idea called sequencing, which is a set of rules that specifies the order in which a learner may experience content objects [27].
- Dublin Core Metadata Initiative (DCMI): metadata developed by NISO (National Information Standards Organization) consisting of fifteen elements to describe learning resources [28].

The ontology proposed in this work, as explained in the next section, is based on the LOM standard. LOs described by LOM have a set of nine categories, which characterize the object. Table I presents these categories, as well as the corresponding descriptions.

TABLE I. LOM BASIC METADATA STRUCTURE [4]

Categories	**Description**
General	Groups the general information that describes this learning object as a whole.
Lifecycle	Groups the features related to the history and current state of this learning object and those who have affected this learning object during its evolution
Meta-Metadata	Groups information about the metadata instance itself.
Technical	Groups the technical requirements and technical characteristics of the learning object.
Educational	Groups the educational and pedagogic characteristics of the learning object.
Rights	Groups the intellectual property rights and conditions of use for the learning object.
Relation	Groups' features that define the relationship between the learning object and other related learning objects.
Annotation	Provides comments on the educational use of the learning object and provides information on when and who created the comments.

IV. ONTOLOGY FOR ORGANIZATIONAL LEARNING OBJECTS

The Ontology for Organizational Learning Objects (OOLO) aims at helping to organize the content created in the organization, especially software development organizations. For the proposed ontology creation, we followed the activities outlined in Methontology [29], which are:

- Specification: establishes the ontology purpose and scope. Answer questions as: Why the ontology is being built? and What are the intended uses and end-user ?
- Conceptualization: organizes and structures the knowledge acquired during knowledge acquisition using external representations that are independent of the knowledge representation and implementation paradigms in which the ontology will be formalized and implemented next.
- Formalization: transforms the conceptual model into a formal or semi-computable model.
- Implementation: builds computable models in an ontology language.

Each developed activity is described in the next sections.

A. Specification

Due to the differences of using LO in an educational and organizational environment, it is important to adapt the LO standards to support significant organizational characteristics.

Thus, the first step to define the ontology was to analyze the LOM, Dublin Core and SCORM standards. A mapping between the properties of the three standards was established. This mapping showed that the Dublin Core properties match with the LOM properties. However, the ontology proposed in this work is based on LOM due to the fact of being a standard planned to facilitate the search; acquisition, evaluation and use of LOs. Besides facilitating the exchange of LOs, it allows the development of repositories and also takes into account cultural diversity and linguistic contexts in which the LOs and their metadata are reused [30]. In addition, LOM is one of the most widely used metadata for describing LOs and serves as a basis for other standards such as SCORM [30].

Some properties of the LOM standard were selected and other three properties, not provided in any other standard, were included. These properties aim to organize the information according to organizational characteristics, showing the projects or groups that can access it, and whether the information can be accessed only by the person within the organization or externally too. In addiction, it is provided information about the role of the person who created or inserted the content. Table II shows the properties defined to create the proposed ontology, and a comparison with the existing standards, showing which properties are mandatory (x), which are optional (OP) and which are not supported in each standard.´

B. Conceptualization

The knowledge domain was structured in a conceptual model that describes the problems and solutions in terms of the identified domain vocabulary during the activity specification. A concept map is a schematic structure to represent a set of concepts embedded in a propositions network [20]. Therefore, through this model it is possible to check all concepts and relations, and to analyze in a clear and concise way the knowledge modeling. Fig. 2 presents the conceptual map for the proposed ontology.

The OOLO concept map organizes the properties defined in Table II, showing the relationships between concepts. It is organized into categories and properties. Furthermore, each object has a relationship with a concept person. A person can create, approve, edit or access the object. The concept person is defined using FOAF (Friend of a Friend). FOAF has as main objective to link people and information about these through the Web [31].

TABLE II. A COMPARATIVE OF THE OOLO PROPERTIES AND MAIN LO STANDARDS

		Standards			
Categories	**Properties**	**OOLO**	**SCORM**	**DC**	**LOM**
General	Identifier	x	x	x	x
	Title	x	x	x	x
	Language	OP	OP	x	OP
	Description	x	x	x	x
	Keyword	OP	x	x	OP
Life Cycle	Version	x	x	-	x
	Status	OP	x	-	OP
	Role	OP	OP	x	OP
	Date	x	OP	x	x
Technical	Format	x	x	x	x
	Location	x	x	-	x
	Source*	OP	-	-	-
	Type	OP	-	-	OP
	Artifact Type*	OP	-	-	-
Educational	Interactivity Type	OP	OP	-	OP
	Learning Resource Type	x	OP	x	x
	Context*	OP	-	-	-
Rights	Copyright	OP	x	-	OP
	Use*	x	-	-	-
	Scope*	OP	-	-	OP

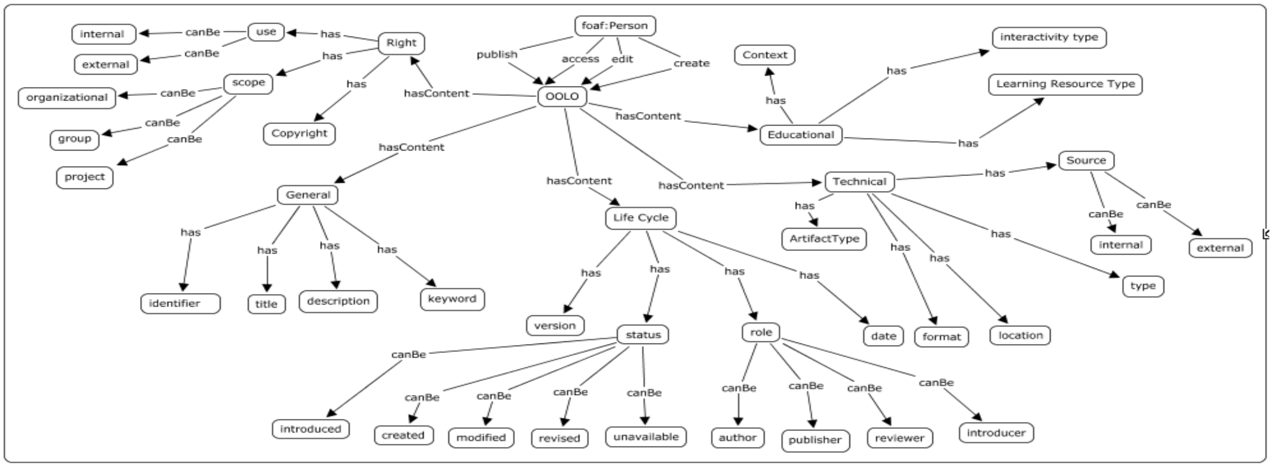


Figure 2. OOLO Concept Map

C. Formalization

The first step was to create a formal model of the conceptual map presented in Fig. 2. For this purpose, we used a class diagram because it facilitates to visualize the concepts taxonomy covering axioms and properties. Covering axioms are designed to require that individuals of a class must be individuals of one of its subclasses [32].

After modeling the class diagram showed in Fig 3, two other steps were performed. First was created a base schema, containing the properties, their attributes, data types and value space. This schema is showed in Table III, and was adapted from [7]. After this step, the formal axioms were defined. These axioms are introduced to constrain their interpretation and well-formed use [33]. The axioms describe more precisely the semantics of the concepts as they constrain how the instances of the concepts could be created [32]. Table IV shows the main axioms defined in first order logic using the ontology predicates.

Besides the classes proposed in Table II, the class diagram also presents classes to represent the people that interact with the LO. We used the classes defined in FOAF to represent the individuals. Each individual can create, modify, evaluate or access a LO.

We also define a role class, which represents the role that can play an individual within the company. So this way, in the future we would be able to classified the quality and trustworthy of a LO, depending on the role of the individual who created and evaluated it.

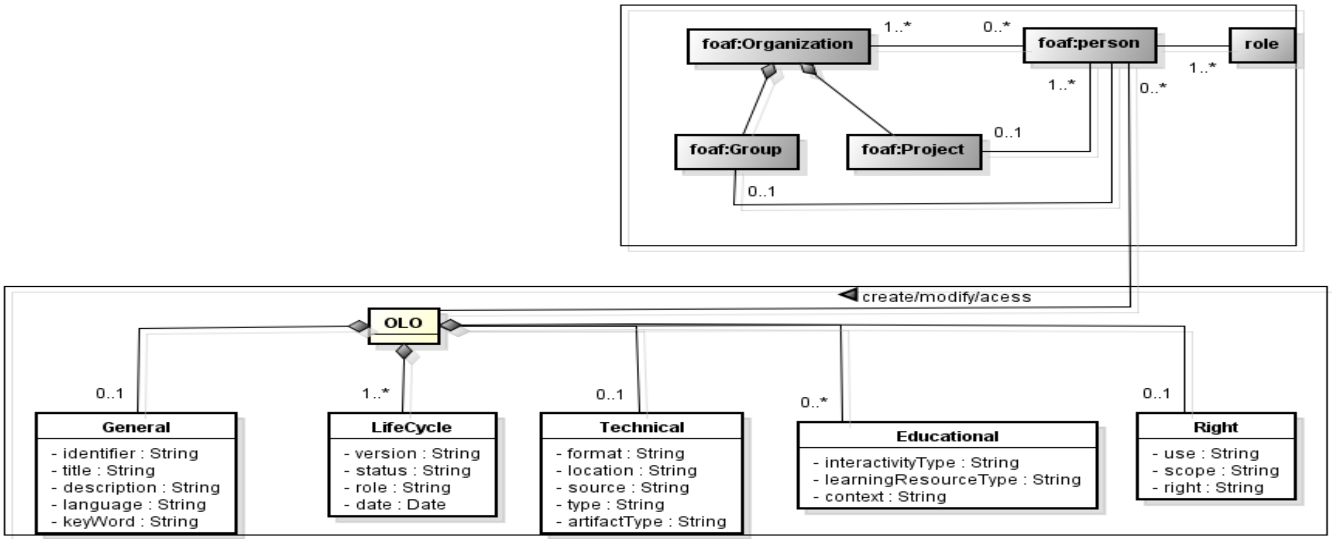
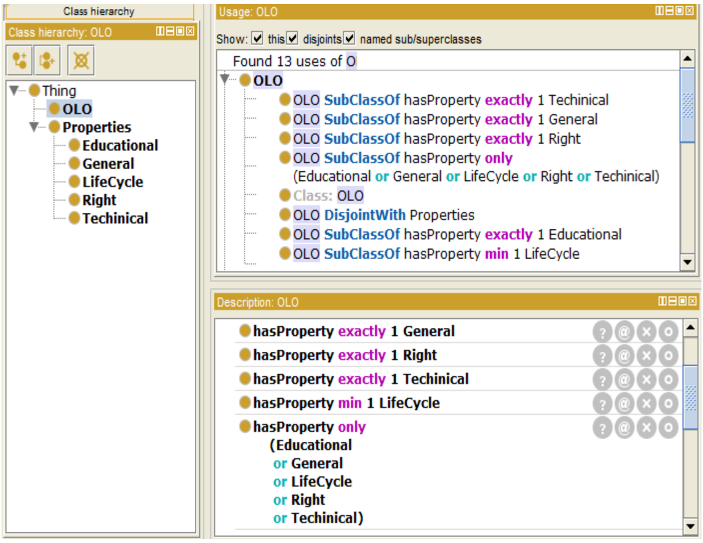


Figure 3. OOLO Class Diagram

D. Implementation

The ontology was implemented using the Protégé ontology editor [34] and it was represented in OWL. The Fig. 4 shows part of this implementation. On the left side the class hierarchy may be observed and on the right side the axioms that define the classes.



Thing
- OLO
- Properties
  - Educational
  - General
  - LifeCycle
  - Right
  - Techinical

Found 13 uses of O

OLO
- OLO SubClassOf hasProperty exactly 1 Techinical
- OLO SubClassOf hasProperty exactly 1 General
- OLO SubClassOf hasProperty exactly 1 Right
- OLO SubClassOf hasProperty only (Educational or General or LifeCycle or Right or Techinical)
- Class: OLO
- OLO DisjointWith Properties
- OLO SubClassOf hasProperty exactly 1 Educational
- OLO SubClassOf hasProperty min 1 LifeCycle

- hasProperty exactly 1 General
- hasProperty exactly 1 Right
- hasProperty exactly 1 Techinical
- hasProperty min 1 LifeCycle
- hasProperty only (Educational or General or LifeCycle or Right or Techinical)

Figure 4. Implementation of the OOLO in Protégé

TABLE III. OOLO BASE SCHEMA. ADAPTED FROM [7]

Categories	Properties	Value space	Data type	Example
General	Identifier	-	-	-
	Title	-	LangString (smallest Permitted maximum: 1000 char)	("en", "The life and works of Leonardo da Vinci")
	Language	languageid = langcode ("-"subcode)*	CharacterString (smallest permitted maximum: 100 char)	"pt", "pt-BR", "en", "it"
	Description	-	LangString (smallest permitted maximum: 2000 char)	("en", "In this video clip, the life and works of Leonardo da Vinci are briefly presented. The focus is on his artistic production, the Mona Lisa.")
	Keyword	-	LangString (smallest permitted maximum: 1000 char)	("en", "Mona Lisa")
Life Cycle	Version	-	LangString (smallest permitted maximum: 50 char)	("en", "1.2.alpha")
	Status	created, introduced, modified, revised, unavailable	Vocabulary (State)	-
	Role	author, introducer, publisher, reviewer	Vocabulary (State)	-
	Date	-	DateTime	"2001-08-23"
Technical	Format	mime types based on iana egistration (see rfc2048:1996) or "non-digital"	CharacterString (smallest permitted maximum: 500 char)	"video/mpeg", "application/x-toolbook", "text/html"
	Location	repertoire of iso/iec 10646-1:2000	CharacterString (smallest permitted maximum: 1000 char)	"http://host/id"
	Source*	internal,external	Vocabulary (State)	-
	Type	-	CharacterString (smallest permitted maximum: 1000 char)	"text/explanation"
	Artifact Type*	Stakeholder Request, Business Case, Glossary, Risk List, Use Case Model, Deployment Plan, Software Architecture Document, Analysis Model, Project Model, Implementation Model, Test Plan	Vocabulary (State)	-
Educational	Interactivity Type	active,expositive, mixed	Vocabulary (State)	-
	Learning Resource Type	exercise,simulation, questionnaire, diagram,figure, source code, graph,índex, slide,table,narrative text,exam,experiment, problem,lecture,	Vocabulary (State)	- -
	Context*	-	LangString (smallest permitted maximum: 1000 char)	("en", "General") ("en", "e")
Rights	Copyright	yes,no	Vocabulary (State)	-
	Use*	external, internal	Vocabulary (State)	-
	Scope*	group, project organization	Vocabulary (State)	-

TABLE IV. FORMAL AXIOMS OF OOLO

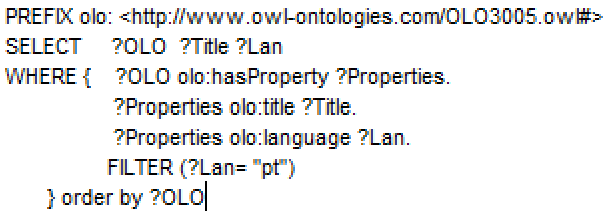
Description	***Formal Description***
A status can be created, edited, revised or introduced and there is only one status for each object's life cycle.	$\Im \models \forall s\exists x\exists y\exists w\ [Status(s) \wedge Created(x) \wedge Edited(y) \wedge Revised(z) \wedge Introduced(w) \rightarrow (isStatus(s,x) \vee (isStatus\ (s,y) \vee (isStatus\ (s,z)\ v\ (isStatus\ (s,w)]$ $\forall x[Created(x) \rightarrow \neg Edited(x) \wedge \neg Revised(x) \wedge \neg Introduced(x)\ ]$ $\forall x[Edited(x) \rightarrow \neg Created\ (x) \wedge \neg Revised(x) \wedge \neg Introduced(x)\ ]$ $\forall x[Revised(x) \rightarrow \neg Created(x) \wedge \neg Edited(x) \wedge \neg Introduced(x)\ ]$ $\forall x[Introduced(x) \rightarrow \neg Created(x) \wedge \neg Edited(x) \wedge \neg Revised(x)\ ]$
For each contribution, its type can be made as author, publisher, introducer or reviewer. And only one contribution type can be made in each life cycle.	$\Im \models \forall s\exists x\exists y\exists z\exists w\ ContributionType(s) \wedge Author\ (x) \wedge Editor(y) \wedge Reviewer(z) \wedge Introducer(z)) \rightarrow (IsContributionType(s,x) \vee IsContributionType(s,y)\ v\ IsContributionType(s,z)\ v\ IsContributionType(s,w))]$ $\forall x[Author(x) \rightarrow \neg Editor(x) \wedge \neg Reviewer(x) \wedge \neg Introducer(x)\ ]$ $\forall x[Editor(x) \rightarrow \neg Author(x) \wedge \neg Reviewer(x) \wedge \neg Introducer(x)\ ]$ $\forall x[Reviewer(x) \rightarrow \neg Editor(x) \wedge \neg Author(x) \wedge \neg Introducer(x)\ ]$ $\forall x[Introducer(x) \rightarrow \neg Editor(x) \wedge \neg Author(x) \wedge \neg Reviewer(x)\ ]$
The source of the material can be internal or external. The sources are disjoint.	$\Im \models \forall x\exists y\exists z\ [Source(x) \wedge Internal(y) \wedge External(z) \rightarrow temSource(x,y) \vee temSource(x,z)]$ $\forall x[Internal(lx) \rightarrow \neg\ External(x)]$ $\forall x[External(x) \rightarrow \neg\ Internal(x)]$
The use of the object can be restricted within the organization, or may be released to be used by anyone. The types of use are disjoint.	$\Im \models \forall x\exists y\exists z\ [Use(x) \wedge Internal(y) \wedge External(z) \rightarrow hasUse(x,y) \vee hasUse(x,z)]$ $\forall x[Internal(x) \rightarrow \neg\ External(x)]$ $\forall x[External\ (x) \rightarrow \neg\ Internal(x)]$
The scope of use of the object can be used only internally within the organization is, within a project or within the group. The scope must be unique for each object.	$\Im \models \forall x\exists y\exists z\exists w\exists r\exists s\ [Uso(x) \wedge Internal(y) \wedge hasUse\ (x,y) \wedge Scope(z) \wedge ScopeOrganizaation(w) \wedge ScopeGroup(r) \wedge ScopeProject(s) \rightarrow hasScope(z,w) \vee hasScope(z,r) \vee hasScope(z,s)]$ $\forall x[ScopeOrganization(x) \rightarrow \neg\ ScopeProject(x) \wedge \neg\ ScopeGroup(x)]$ $\forall x[ScopeProject(x) \rightarrow \neg\ ScopeOrganization(x) \wedge \neg\ ScopeGroup(x)]$ $\forall x[ScopeGroup(x) \rightarrow \neg\ ScopeProject(x) \wedge \neg\ ScopeOrganization(x)]$
The General property has only Identifier, Title, Language, Description and keyword, and the attribute Title and Language are mandatory	$\Im \models \forall x\exists y\exists z\exists w\exists r\exists s\ [General(x) \wedge Identifier(y) \wedge Title(z) \wedge Language(w) \wedge Description(r) \wedge Keyword(s) \rightarrow (hasAttribute(x,y) \vee hasAttribute(x,z) \vee hasAttribute(x,w) \vee hasAttribute(x,r) \vee hasAttribute(x,s)\ ]$ $\forall x\exists y\exists z\exists r[\ General(x) \rightarrow hasAttribute(x,y) \wedge hasAttribute(x,z) \wedge hasAttribute(x,r)\ ]$
The Life Cycle property has only attributes Version, Status, Type and Date Contribution, and the attributes Version, Status and Date are mandatory.	$\Im \models \forall x\exists y\exists z\exists w\exists r\ [LifeCycle(x) \wedge Version(y) \wedge Status(z) \wedge ContribuitonType(w) \wedge Date(r) \rightarrow (hasAttribute(x,y) \vee hasAttribute(x,z) \vee hasAttribute(x,w) \vee hasAttribute(x,r)\ ]$ $\forall x\exists y\exists z\exists r[\ LifeCycle(x) \rightarrow hasAttribute(x,y) \wedge hasAttribute(x,z) \wedge hasAttribute(x,r)\ ]$
The Technical property has only attributes Format, Location, type and origin, and the attributes Format and Location are mandatory.	$\Im \models \forall x\exists y\exists z\exists w\exists r\ [Technical(x) \wedge Format(y) \wedge Location(z) \wedge Type\ (w) \wedge Source(r) \rightarrow (hasAttribute(x,y) \vee hasAttribute(x,z) \vee hasAttribute(x,w) \vee hasAttribute(x,r)\ ]$ $\forall x\exists y\exists z\ [\ Technical(x) \rightarrow hasAttribute(x,y) \wedge hasAttribute(x,z)]$
The Educational property has only attributes InteractivityType, LearningResourceType and Context, and the attribute LearningResourceType is mandatory	$\Im \models \forall x\exists y\exists z\exists w\ [Educational(x) \wedge InteractivityType(y) \wedge LearningResourceType\ (z) \wedge Context(w) \rightarrow (hasAttribute(x,y) \vee hasAttribute(x,z) \vee hasAttribute(x,w)\ ]$ $\forall x\exists y\exists z\ [\ Educational(x) \rightarrow hasAttribute(x,z)]$
The Right category has only atributes Use, Scope and Copyright, and the attribute Use is mandatory	$\Im \models \forall x\exists y\exists z\exists w\ [Right(x) \wedge Use(y) \wedge CopyRight\ (z) \wedge Scope(w) \rightarrow (hasAttribute(x,y) \vee hasAttribute(x,z) \vee hasAttribute(x,w)\ ]$ $\forall x\exists y\exists z\ [\ Right(x) \rightarrow hasAttribute(x,y)]$
An object only has General, Life Cycle, Technical, Educational and Right properties. Each object has exactly one Properties General, Technical, Educational and Right, and at least one property Life Cycle.	$\Im \models \forall x\exists y\exists z\exists w\exists r\exists s\ [Object(x) \wedge General(y) \wedge LifeCycle(z) \wedge Technical(w) \wedge Educational(r) \wedge Right(s) \rightarrow (hasProperty(x,y) \vee hasProperty\ (x,z) \vee hasProperty\ (x,w) \vee hasProperty\ (x,r) \vee hasAttribute(x,s)\ ]$ $\forall x[(x) \rightarrow hasProperty(x,y)\ \ exactly\ 1]$ $\forall x[(x) \rightarrow hasProperty(x,w)\ \ exactly\ 1]$ $\forall x[(x) \rightarrow hasProperty(x,r)\ \ exactly\ 1]$ $\forall x[(x) \rightarrow hasProperty(x,s)\ \ exactly\ 1]$ $\forall x[(x) \rightarrow hasProperty(x,z)\ \ min\ 1]$

V. EVALUATION AND DISCUSSIONS

In order to evaluate the OOLO, an application scenario was created, inserting software engineering wiki pages as LO instances. The objective was to organize some contents as LO, defining properties for each content. So, it was defined the title, url, date, the content source (internally or externally) to the organization, the learning type, among others. After insert the instances, the ontology was able to make inferences, as for example, to infer the object consistence or even the object language content. Then, we carried out queries using the SPARQL [35], within the Protégé. The queries aimed to look for specific LOs, as for example, find out LOs created within the organization, LOs with specific permission of use in a project, among others LOs.

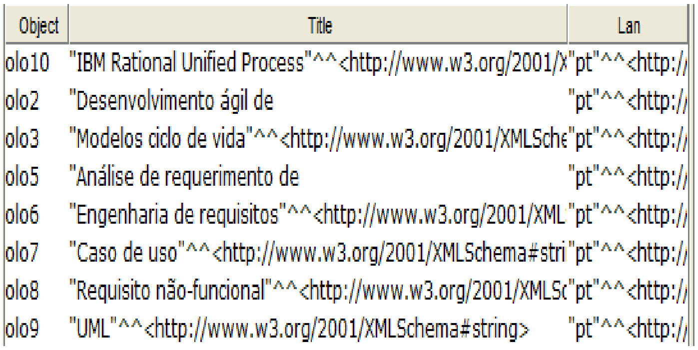
An example of the executed queries is presented in Fig. 5. The objective of this query is to return all LOs that are in Portuguese language, showing the objects and their titles

Fig. 6 presents the query results. This query finds out all Portuguese LO available, and shows their titles and language.



```
PREFIX olo: <http://www.owl-ontologies.com/OLO3005.owl#>
SELECT    ?OLO  ?Title ?Lan
WHERE {    ?OLO olo:hasProperty ?Properties.
           ?Properties olo:title ?Title.
           ?Properties olo:language ?Lan.
          FILTER (?Lan= "pt")
   } order by ?OLO
```

Figure 5. SPARQL query



Object	Title	Lan
olo10	"IBM Rational Unified Process"^^<http://www.w3.org/2001/X	"pt"^^<http:/
olo2	"Desenvolvimento ágil de	"pt"^^<http:/
olo3	"Modelos ciclo de vida"^^<http://www.w3.org/2001/XMLSche	"pt"^^<http:/
olo5	"Análise de requerimento de	"pt"^^<http:/
olo6	"Engenharia de requisitos"^^<http://www.w3.org/2001/XML	"pt"^^<http:/
olo7	"Caso de uso"^^<http://www.w3.org/2001/XMLSchema#stri	"pt"^^<http:/
olo8	"Requisito não-funcional"^^<http://www.w3.org/2001/XMLSc	"pt"^^<http:/
olo9	"UML"^^<http://www.w3.org/2001/XMLSchema#string>	"pt"^^<http:/

Figure 6. SPARQL query results

In addition, different objectives can be achieved according to the query formulation, as example, to find objects considering only a certain language, a source, the keywords, the person who created or has permission to access the LO.

Unlike the LOM standard, where hierarchical taxonomies, relation properties, and semantic constraints between the LO properties can not be represented in XML-Schema; ontology defines precisely the concepts semantics, there should be no misinterpretations or errors when the instances of the concepts are created and managed in runtime.

In addition, with the ontology it is possible to check the consistence of the LOs in both the design and runtime phases, by using a general reasoner following the axioms associated to the language in which the ontology is represented. Moreover, using ontology it is easier to organize and reuse the LOs. An ontology is formally defined, so we can use a reasoner to carry out inferences with the LOs ontology.

Finally, we can improve the LOs, creating an axiom to classify the trustworthiness of each LO according to the person's role that created or inserted it.

VI. FINAL CONSIDERATIONS

The paper presented the OOLO ontology, based on LOM standard and fragments of the FOAF ontology. The ontology represents organizational learning objects taking into account educational and organizational properties.

LOM standard is defined in XML-schema and this language has some limitations as lack of expressiveness to represent hierarchical taxonomies, relations, properties, and semantic constraints between the LO, properties and categories. The use of ontology to define a LO standard can help to reduce these problems.

The proposed ontology is based on LOM standard and add organizational properties for helping companies to organize their information, facilitating the content reuse and improving the organizational learning, or even using the objects to create a generic context of learning environments, using, for instance, the Educational Concept Map [36]. Additionally the OOLO includes axioms that allow greater expressivity to classes and it can be integrated into learning environments. A simulation was carried out and the results showed that it is possible to use the OOLO to organize LOs.

ACKNOWLEDGMENT

This work is being developed with the financial support of Fundação Araucária (Foundation in Support of the Scientific and Technological Development of the State of Paraná, Brazil).

REFERENCES

[1] A. Takeishi, "Knowledge Partitioning in the Interfirm Division of Labor: The Case of Automotive Product Development," Organization Science, vol. 13-3, pp. 321-338, 2002.

[2] D. J. Teece, G. Pisano, and A. Shuen, "A Dynamic Capabilities and Strategic Management," Strategic Management Journal, vol.18 -7, pp. 509-533, 1997.

[3] V. Sambamurthy, A. Bharadwaj, and V. Grover, "Shaping Agility through Digital Options: Reconceptualizing the Role of Information Technology in Contemporary Firms", MIS Quarterly, vol. 27-2, pp. 237-263, 2003.

[4] J. Vassileva, "Toward social learning environments," IEEE Transactions on Learning Technologies, vol. 1-4, pp. 199–214, 2009.

[5] J. Kimmerle, J. Moskaliuk, and U. Ccress, "Learning and Knowledge Building with Social Software," Proc. 9th International Conference on Computer supported collaborative learning, 2009, pp. 459-468.

[6] J. Piaget, "Problems of equilibration, " Proc. Topics in cognitive development," vol. 1, New York, 1977, pp. 3-14.

[7] IEEE, "Draft Standard for Learning Object Metadata, " http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf, July, 2002.

[8] P. R Polsani, "Use and abuse of reusable learning objects," Journal of Digital information, vol. 3-4, 2004.

[9] R. R, Amorim, M. Lama, E. Sánchez, A. Riera, and X. A. Vila, "A Learning Design Ontology based on the IMS Specification, " Educational Technology & Society, vol. 9, pp. 38-57, 2006.

[10] A. Gómez-Pérez, M. Fernández-López, and O. Corcho, "Ontological Engineering, " Berlin: Springer, 2004.

[11] N. Guarino, "Formal Ontology and Information Systems," In the Proceedings of Formal Ontology in Information Systems, pp. 3-15, 1998.

[12] P. Mohammed, and P. Mohan, "Contextualizing Learning Objects Using Ontologies," Computational Intelligence, Blackwell Publishing, vol. 23-3, pp.339-355, 2007.

[13] I. Bittencourt, E. Costa, M. Silva, and E. Soares, "A computational model for developing semantic webbased educational systems,". KnowledgeBased Systems, vol. 22-4, pp.302–315, 2009.

[14] M. J. Eppler, P. M. Seifried, and A. Ropnack, "Improving knowledge intensive processes through an enterprise knowledge medium," Proceedings of the ACM SIGCPR Conference on Computer Personnel Research, 1999.

[15] M. Strohmaier, and K. Tochtermann, "B-KIDE : A Framework and a Tool for Business Process-Oriented Knowledge Infrastructure Development, " Knowledge and Process Management, vol. 12- 3, pp. x 171-189, 2005.

[16] M. Strohmaier, "A business process oriented approach for the identification and support of organizational knowledge processes," Proceedgins of fourfh Oldenburger Fachtagung Wissensmanagement, Potenziale, Werkzeuge, 2003.

[17] I. Nonaka, and H. Takeuchi, "The knowledge-creating company: how Japanese companies create the dynamics of innovation, " Nova York, Oxford University Press, 1995.

[18] E. Wenger, "Communities of Practise: Learning, Meaning and Identity," Cambridge University Press, Cambridge, UK, 1998.

[19] A. L. Menolli, A. Malucelli, and S. Reinehr, "Towards a Semantic Social Collaborative Environment for Organizational Learning," Proc. International Conference on Information Technology and Applications (ICITA 2011), IEEE Press, 2011, pp. 65-70.

[20] H. Tanev, and B. Magnini, "Weakly Supervised Approaches for Ontology Population," Proceeding of the conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge, Netherlands, 2008.

[21] IMS Global Learning Consortium, "IMS Learning Design Information Model", Version 1.0 Final Specification, from http://www.imsglobal.org/learningdesign/ldv1p0/imsld_infov1p0.html, 2003.

[22] L. Romero, and H. Leone. "Una ontología para el dominio de evaluación del aprendizaje". Martín Llamas Nistal, Carlos Vaz de Carvalho, y Carlos Rueda Artunduaga, editores. TICAI2007: TICs para el aprendizaje de la ingeniería. Isbn 978-84-8158-380-9 pags.:17-24 , 2007.

[23] J. C., Gluz, J. C., and R. M. Vicari, "An OWL Ontology for Metadata IEEELOM, DublinCore and OBAA," Brazilian Symposium On Computers In Education, pp. 204-213, 2011 .

[24] Y. Gil, and V. Ratnakar, "A Comparison of (Semantic) Markup Languages," 15th International FLAIRS Conference, Pensacola Beach, Florida, USA, 2002.

[25] C. L. D. Dias, A. Kemczinski, S. Lucena, J. Ferlin, and M. Hounsell, "Open standards: applicability in Learning Objects (LOs)," Brazilian Symposium On Computers In Education, 2011 .

[26] Brasil, "Ministério da Educação. "Objetos de aprendizagem: uma proposta de recursos pedagógico", Brasília: Mec, pp.154, 2007.

[27] ADL, "Advanced Distributed Learning," Scorm 2004 4th Edition, http://www.adlnet.gov/Technologies/scorm/default.aspx, July, 2004.

[28] NISO "Niso Standards. ANSI/NISO Z39.85 - The Dublin Core Metadata

[29] M. Fernández, A. Gómez-Pérez, and N. Jurino, "Methontology: From Ontological art Towards Ontological Engineering," Symposium on Ontological Engineering, Stanford, Califórnia, 1997.

[30] S. R., Gomes, B. F. Gadelha, and A. P. Mendonça, "Functional Learning Objects and Limitations of Current Metadata," Brazilian Symposium On Computers In Education, 201, 210, 2005 .

[31] FOAF Vocabulary Specification 0.98. http://xmlns.com/foaf/spec

[32] M. Horridge, H. Knublauch, A. Rector, R. Stevens, and C. A. Wroe, "Practical Guide To Building OWL Ontologies Using The Protéegé-OWL Plugin and CO-ODE Tools," Edition 1.0. The University Of Manchester, 2004.

[33] A. J. Pretorius, "Ontologies - Introduction and Overview," Mosaic A Journal For The Interdisciplinary Study Of Literature, pp. 1-13, 2004.

[34] Stanford. The Protégé Ontology Editor and Knowledge Acquisition System. http://protege.stanford.edu/index.html.

[35] SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query.

[36] G. Adorni, M. Coccoli, and G. Vivanet, "Educational Concept Maps: a Knowledge Based Aid for Instruciotional Design," Software Engineering and Knowledge Engineering (SEKE), USA, July 7-9, 2011.