Integrating Descriptions of Knowledge Management Learning Activities into Large Ontological Structures: A Case Study

Miguel-Ángel Sicilia [a] Miltiadis Lytras [b] Elena Rodríguez [c]
Elena García–Barriocanal [a]

[a] *Computer Science Department. Polytechnic School.*
University of Alcalá. Ctra. Barcelona km. 33.6
28871 – Alcalá de Henares, Madrid (Spain)
{*msicilia, elena.garciab@uah.es*}

[b] *Department of Management Science and Technology,*
ELTRUN - The Research Center, Athens University of Economics and Business,
47A Evelpidon Str., 113 62 Athens, GREECE
mdl@eltrun.gr

[c] *Computer Science Studies.*
Open University of Catalonia.
Avda. Tibidabo 39-43 08035 Barcelona (Spain)
mrodriguezgo@uoc.edu

Abstract

Ontologies have been recognized as a fundamental infrastructure for advanced approaches to Knowledge Management (KM) automation, and the conceptual foundations for them have been discussed in some previous reports. Nonetheless, such conceptual structures should be properly integrated into existing ontological bases, for the practical purpose of providing the required support for the development of intelligent applications. Such applications should ideally integrate KM concepts into a framework of commonsense knowledge with clear computational semantics. In this paper, such an integration work is illustrated through a concrete case study, using the large *OpenCyc* knowledge base. Concretely, the main elements of the Holsapple & Joshi KM ontology and some existing work on e–learning ontologies are explicitly linked to *OpenCyc* definitions, providing a framework for the development of functionalities that use the built–in reasoning services of *OpenCyc* in KM activities. The integration can be used as the point of departure for the engineering of KM–oriented systems that account for a shared understanding of the discipline and rely on public semantics provided by one of the largest open knowledge bases available.

Key words: Ontologies, data models, knowledge management, learning objects.

Preprint submitted to Elsevier Science *25 April 2005*

1 Introduction

The discipline of Knowledge Management (KM) has evolved and matured in the last decade, resulting in a considerable amount of models, tools and technologies. Diverse perspectives on KM make the field somewhat scattered and even diverging [3], although it can be said that it exists a shared view in several KM concepts related to organizational learning, and also in practical approaches like communities of practice, as pointed out by Scholl *et al.* [17]. This diversity of perspectives has fostered recent efforts oriented towards unifying concepts and providing integrative theoretical foundations for KM. In that direction, the ontology of Holsapple and Joshi (H& J) [5] describes fundamental KM concepts and axioms, and several other authors have also provided integrative views of the diverse perspectives on KM for specific elements. For example, Kakabadse et al [6] provide a taxonomy of knowledge models, and Abou-Zeid [1] provides a multi-layer comprehensive reference model for KM.

In addition, the supporting technologies for socialization, externalization, combination and internalization of knowledge are available and can be applied to build KM solutions of a diverse kind [12]. Formal ontologies [4] have been proposed and applied as the backbone of KM systems [11], and even ontologies specific to certain KM domains exist — e.g. for software development organizations [13]. This has happened probably due to the fact that they provide a formal way to specify semantics of KM artifacts, and they also allow for the development of intelligent tools for knowledge sharing and reuse. Nonetheless, there is a significant effort associated to the engineering of a KM formal ontology, so that *reuse* becomes a key issue in practical situations in which an organization decides to engage in ontology–based KM.

A significant amount of reuse in terminological structures and tools can be achieved by building KM systems on top of existing large terminological bases like *OpenCyc* [1] . *OpenCyc* is the open source version of the *Cyc* Knowledge Base [9], which contains over one hundred thousands atomic terms, and is provided with an associated efficient inference engine. *Cyc* uses as its underlying definition language a variant of predicate calculus called *CycL*, and it attempts to provide a comprehensive upper ontology of "commonsense" knowledge. *OpenCyc* contains many formal definitions that are useful in the development of KM support systems, including basic supporting elements like time and date, descriptions of organizational and customer–related terms, agent–based communication and descriptions of events and activities. These fundamental definitions and the available inferencing, querying and development tools provided by *OpenCyc* conform a basic framework for the implementation of ontology–based approaches to KM.

[1] http://www.opencyc.org/

Process–orientation is at the essence of KM [7] since KM activities are in fact part of business processes. In consequence, ontological approaches to KM should pay a special attention to activities as the dynamic part of KM systems. Here we are especially concerned with the concrete class of knowledge processes that result in *learning* activities, but other kinds of activities could be modelled in a similar way. Our interest in learning activities is directly related to learning processes as those that are supported by current *e-learning* standardized technology [10] , for which some *OpenCyc* previous integration work has been described elsewhere [18,19]. The provision of knowledge representations integrating KM and e–learning standards has been pointed out as an important research direction elsewhere [20].

This paper describes the main integration points of the H& J ontology into the formal structure of *OpenCyc*. The result is by itself an artefact ready as a data model for intelligent applications of KM that explicitly build on common models of KM. In consequence, this represents an advance for the engineering of these kinds of systems, since with previous approaches the ontologies were either not comprehensive (since they were not based on commonly agreed models) or *ad hoc* in their scope and their provision of computational semantics. In addition, the work described here explicitly links learning object–related notions [23] to KM concepts, representing an additional advantage due to the increasing emphasis on metadata-based reuse of learning resources.

The rest of this paper is structured as follows. In Section 2, the main integration issues for KM processes within *OpenCyc* are described. Then Section 3 describes how learning activities are modelled inside the same framework as a concrete kind of KM activity, emphasizing on the integration of existing e-learning standards into the more general framework of KM. Finally, conclusions and future research directions are provided in the last section.

2 Integrating Knowledge Processes in OpenCyc

The recent work of Holsapple and Joshi [5] has resulted in a KM ontology providing a shared view of KM, developed collaboratively by a panel of over 30 KM practitioners and researchers. This ontology provides a foundation for explicit representations of the variety of artifacts and processes that play a role in the discipline of Knowledge Management, and as such, it can be used as a source for Semantic Web approaches to KM. The Holsapple and Joshi general–purpose KM ontology (H&J ontology for short) is described in terms of *definitions* and *axioms*. Definitions start with a letter "D", while axioms begin with a letter "A". Both are followed by a component acronym and a number. The component acronym represents the different components identified in H&J ontology, i.e. *Knowledge Management Conduct*(KMC), *Knowledge*

Manipulation Activities(KMAs), *Knowledge Resources*(KRs), and *Knowledge Management Influences*(KMIs). Definitions and axioms are showed in this paper enclosed in square brackets for reference purposes.

2.1 Basic Definitions

The definition of KM in H&J ontology "An entity's systematic and deliberate efforts to expand, cultivate, and apply available knowledge in ways that add value to the entity [..]"[DKMC1] requires the early definition of "entities" capable of engaging in KM, which are considered to include at least individuals, organizations, collaborating organizations and nations, as stated in [DKMC2-5]. The term `#$Organization`[2] in *OpenCyc* covers all such entities[3] (including nations as `#$GeopoliticalEntity` instances and collaborations as defined by the `#$subOrganizations` predicate). The definition of `#$Organization` further restricts membership by the presence of certain relationships between organization members. Each instance of `#$Organization` can undertake projects, enter into agreements and own property. Such view on organizations is consistent to that of H&J ontology, and is able to model both informal and legally constituted organizations (`#$LegalCorporation`).

The concept of knowledge processor [DKMC10] as a member of an entity can be modelled by the concept of `#$IntelligentAgent`, which are by definition "capable of knowing and acting, and of employing their knowledge in their actions". Humans are by logical definition intelligent agents and certain software pieces may also be, since they are not restricted to not being able to *know* [AKMC10]. The subtype `#$MultiIndividualAgent` fits the definition of collective agents [AKMC11]. The predicates `knowsAbout` and `knows` represent two alternatives for modelling available knowledge. The former is loose and may connect to any concept, while the latter is strict and requires representing justified beliefs as logical assertions belonging to the class `ELSentence-Assertible`. These two differentiated epistemologies allow for defining both formal and informal knowledge, including tacit knowledge, which could also be modelled by diverse predicates like `opinions`, `biases` or `hasEmotionAboutProposition`.

The following definitions summarize the main issues described so far.

Definition 1 ***#$Organization*** *is assimilated to the concept of* ***entity*** *in [DKMC1-5], and represents a group of* ***#$IntelligentAgents*** *that are* ***Knowl-***

[2] The '#$' prefix is the *CycL* convention for constants.
[3] Technically, a `#$Person` is not an `#$Organization` but a specific kind of `#$Agent`, but this can be avoided by considering personal KM as carried out by a single–person organization.

edge Processors *[DKMC9, DKMC10].*

Definition 2 *The different kinds of KM [DKMC2-5] are modelled by the specializations of* ***#$Organization****. Some important ones are yet described in* OpenCyc*, e.g.* *#$Business* *or* *#$LegalGovernmentOrganization.*

Definition 3 *Knowledge in agents can be modelled with various facets by OpenCyc predicates like* **knows***,* **knowsAbout***,* **opinions** *or* **expects***.*

The varying effectiveness of knowledge processors for certain tasks [AKMC7-9] is related to the concept of KMA that is described in the following sub–section.

The definition of Knowledge as "that which is conveyed by usable representations" [DKMC6] can be integrated in *OpenCyc* by considering this usable representations [AKMC2] as information bearing things, i.e. "Each instance of `InformationBearingThing` (or "IBT") is an item that contains information (for an agent who knows how to interpret it) ". The knowledge representation types described in [AKMC1] are similar to some *OpenCyc* subclasses like `SoundInformationBearingThing` or `VisualInformationBearingThing`, and the type of the contents is properly represented by `#$IBTContentType`, which allows a flexible modelling of representation specs. The concept of knowledge artifact [DKR6] has been omitted here, since its definition only refers to a knowledge representation, and does not provide any further function or utility.

Definition 4 *#$InformationBearingThing instances represent usable knowledge representations able of conveying knowledge [DKMC6].* *#$IBTContentType* *can be used to model diverse formats and representations, including MIME types, kinds of internal reports and others.*

Then, the central notion of KM [DKMC1] can be defined in terms of the set of KMAs initiated deliberately by an entity to create value. Nonetheless, the definition in H&J ontology is not precise enough to make directly useful its inclusion in *OpenCyc*.

The broad concept of resource [DKMC7] is properly represented by the predicate `resourceAvailable`, that connects `Agents` (and in consequence, entities as described above) with instances of `#$SomethingExisting`, which encompass not only tangible things but also intangibles like agreements or obligations. Their inherent temporal nature directly capture the change dimension stated in [AKMC5]. Human, material, knowledge and financial resources [AKMC4] can be accommodated as subclasses of `#$SomethingExisting`.

Definition 5 *Resources available to entities [DKMC7] are represented implicitly by that linked to agents through the predicate* **resourceAvailable***.*

Knowledge representations represented by IBTs are a kind of knowledge resource that is easily identifiable (they are content knowledge resources according to [DKR2]), but other knowledge resources that are part of the organization itself [DKR1] require further modelling. Concretely, competencies following the schema described by Sicilia in [22] have been integrated as an extended schema covering the predicates `capableOf` and `skillCapableOf`.

The integration of the basic KM concepts described enables the use of at least the following *OpenCyc* model aspects in KM–supporting applications:

- The variety of Knowledge Representations can be modelled by `IBT` and related classes, resulting in a categorized repository of resources, eventually including fine–grained descriptions of knowledge items and competencies.
- The behavior of knowledge processors inside the organization can be modelled in detail by means of the part of *OpenCyc* dealing with `Agent` interactions, e.g. modelling individual and organizational `goals`, knowledge (`knows`) and actions (`performedBy`).
- Organizational structure can be modelled by the `Organization` term, and `Business` in case of profit–oriented organizations. The predicates `subOrganization` and `parentCompany` can be used to model units and aggregates, respectively.

2.2 Knowledge–Manipulation Activities

H&J ontology describes KM activity in terms of the manipulation of knowledge representation by processors [DKMC11]. The recognizable kinds of knowledge manipulation are referred to as Knowledge Manipulation Activity (KMA) [DKMC12]. Activities in *OpenCyc* are represented as `#$Action`s, which are a collection of `#$Events` carried out (`doneBy`) by a "doer". This generic concept of action can be specialized to represent KMA executions by restricting them to be carried out by intelligent agents. The predicate `ibtUsed` can be used to represent the knowledge representations manipulated by KMAs. In addition, since KM activities are deliberate, it is better to use the subclass `#$PurposefulAction` and the predicate `performedBy`.

Definition 6 *KMA executions can be represented as instances of `#$PurposefulAction`, `performedBy` an `#$IntelligentAgent` and using (`ibtUsed`) `#$IBTs` representing knowledge resources.*

The concept of KM Episode [DKMC15-16] represents executions of KMAs possibly by a collection of processors. These can be modelled by configurations of Actions as modelled by `#$ActionPredicate`, which can be used by the *Cyc* planner to reason about events and dynamics.

Definition 7 *KM episodes of an arbitrary complexity can be modelled by*

#$ActionPredicates and related elements.

Nonetheless, KMA types and KM episode types are generic "templates" characterizing concrete executions. They can be represented as collections of `#$PurposefulActions` and of `#$ComplexActionPredicates`, respectively.

Definition 8 *#$KMA is introduced as a class with KMA instances represented as #$PurposefulActions. Collections of #$ComplexActionPredicates are used to model types of KM episodes, containing interacting KMAs.*

The Knowledge flows concept [DKMA1] defines "the transfer of knowledge from one instance of a KMA to another instance". It can be represented by explicitly asserting the outcomes of KMA. The generic `eventOutcomes` predicate can be used. More specific situation changes, for example, chained activities, can be represented by the `postSituation` and `postEvents` predicates.

Influences in KM [DKMC13] are loosely defined as "factors that can affect resources, processors and processes". Even though this concept is important to model KM events, the forces and outcomes or influences are not detailed in the paper presenting H&J ontology [5], so that they have not been included in the integration described in this paper. A kind of influence that is yet represented in *OpenCyc* is agent–to–agent influences (`influencesAgent`).

Knowledge needs or objectives are another important issue in modelling KMA and KM episodes. Predicates `goals` and `subGoals` in *OpenCyc* can be used to express future–oriented sentences representing agent goals. Since entities and processors are particular kinds of agents, these assertions can be used both for organizational and for individual objectives. Desires and expectations can be expressed by their own predicates, clearly differentiating them from concrete objectives.

Definition 9 *Goals for entities and processors with respect to knowledge can be expressed through the goals predicate in logical form. The objectives of KMAs can be considered as the goals of the processors involved in them.*

The different types of KMA described by H&J ontology [AKMA1], i.e. knowledge acquisition, selection, generation, assimilation, and emission, can be considered as sub–categories of `#$KMA` depending on their purpose and results. Knowledge Acquisition [DKMA3] and Selection [DKMA4] are typically instances of `#$SelectingSomething` events, that can be further decomposed in research or evaluating steps. The difference between them is that the former acts on the organization's external environment, while the later identifies knowledge within the organization. Knowledge assimilation [DKMA5] is essentially connected to learning as described in the following section. Knowledge generation [DKMA6] deals with knowledge derivation, it can be modeled as instances of `#$CreationEvent`, representing the outcomes with the `outputsCreated` predicate. Finally, Knowledge Emission [DKMA7] is related to knowledge projection to the external environment. `#$DistributionEvent` can be used to model such emissions.

The above activity–related definitions enable an implicit definition of the concept of KM as the set of all the KMAs inside an organization. This definition should be complemented with a view on strategic actions (somewhat entailed from [DKMC1]: "systematic and deliberate efforts") and some higher–level view on KM as an aggregated behavior or conduct, as considered in definitions of learning organizations [14]. The predicate `controls` allows for the definition of subordination relations that may be used to model the flow of control from an organization's objectives to their constituents agents. Since organizations are actually agents, they have `goals` and also intentions (`intends`).

The integration of the KMA–related concepts described enables the use of at least the following *OpenCyc* model aspects in KM–supporting applications:

- The sequence of steps for complex KM activities can be generated by planners, simply using the available *OpenCyc* machinery about complex actions.
- Knowledge objectives can be expressed in logical form through goal predicates, thus providing an explicit representation for KM–related behavior.

3 Integrating Learning Activities

Learning in H&J ontology is defined as "a process whereby knowledge resources are modified; an outcome of a KM episode involving change in the state of an entity's knowledge" [DKMC17]. This definition entails that learning is considered as a (positive) change in one or several IBTs, or in some specific cases, in the knowledge attributed to one or several agents inside the organization.

Although the term `#$Learning` is defined in *OpenCyc* as "the collection of all events, brief or extended, in which an agent is acquiring information or know–how", this definition by itself do not supports measurement and subsequent assessment of learning activities. A notion of discrete learning event needs to be introduced to accomplish such goals. The differential account of its definition is consistent with current approaches to contract–based learning object design [21], if they are considered to be knowledge assimilation processes [DKMA5] [4].

Definition 10 *Discrete learning events can be characterized as the difference*

[4] This view of learning objects as facilitators of assimilation of knowledge has been criticized as too narrow elsewhere, e.g. [2]. Nonetheless, it is still the most common view, and other functions can be accommodated in the future.

in the extent of the `knows` *predicate of an agent after the execution of a concrete KMA. This can be expressed by referring to each know–related item through a* `learntIn` *predicate (a specialized inverse of* `eventOutcomes`*).*

Current approaches to Web–based learning are based on the concept of learning object, for which several definitions have been proposed. Reusability is considered to be an essential characteristic of the concept of learning object as the central notion for modern digital learning content design. For example, Polsani [16] includes reuse in his definition of learning object as "an independent and self-standing unit of learning content that is predisposed to reuse in multiple instructional contexts", and Wiley [23] also mentions the term in his learning object definition "any digital resource that can be reused to support learning". Existing work has dealt with the integration of that concept in *OpenCyc* [18][19] taking into account e-learning standards [10]. Figure 1 provides an overview of the main mappings proposed.

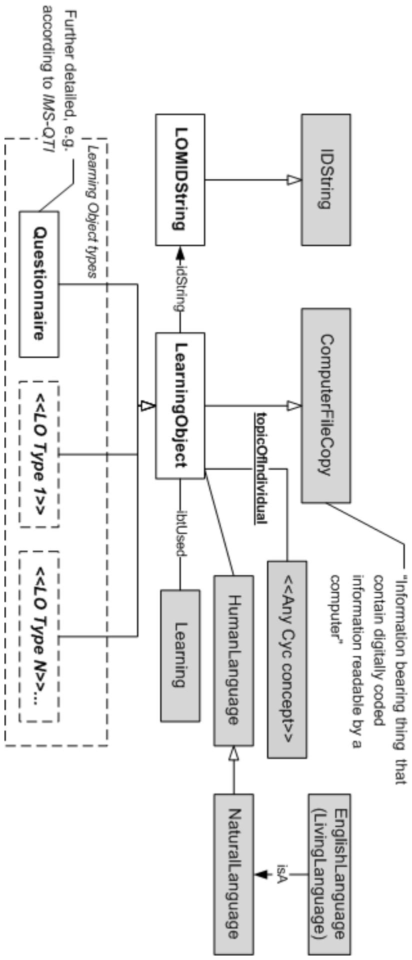

Fig. 1. Summary of the main elements of the OpenCyc mapping of LOM metadata [10] described in [19] and [18].

In the context of KM, learning is considered the outcome of a KM process, so that learning objects become elements in it.

Definition 11 *Learning objects are concrete kinds of knowledge representations oriented specifically to learning.*

Learning objects are considered as resources inside activities of any arbitrary complexity [8]. Since KMAs are provided with detailed definitions, the structure of KMAs can be assimilated to activities, eventually adding some concepts related to specific theories of learning, e.g. construtivists or socio–cultural approaches [15].

The integration of the learning technology concepts described enables the use of at least the following *OpenCyc* model aspects in KM–supporting applications:

- Learning activities' outcomes can be represented inside the ontology, enabling measurement and assessment.
- Learning objects are considered as a specific kind of knowledge representation used in KM processes, and their contribution to learning as such can be measured and assessed.
- Many of the aspects covered in learning object metadata can be assimilated (and in some cases, make more formal) to existing OpenCyc definitions as described in [18].

4 Conclusions and Future Research Directions

The integration of the main concepts of H&J ontology inside the *OpenCyc* knowledge base has been described, and such concepts have been formally linked to ontological definitions related to learning technology described elsewhere [18,19]. The definitions provided are linked to existing commonsense knowledge represented in *OpenCyc*, allowing the use of such knowledge in KM applications of a diverse kind. *OpenCyc* provides a significant amount of concept and predicate definitions that embody diverse aspects of KM, which can be extended and interpreted consistently to come up with a well–equipped knowledge representation for KM applications. Consequently, the practical contributions of the work described include both the pragmatic aspect of providing a knowledge representation for the development of applications, and also the conceptual insights on the ontological commitments that connect shared views of KM and learning resources with commonsense knowledge.

The mapping provided in this paper can be further extended and revised for concrete application profiles, and it is essentially intended to provide a concrete realization of an existing ontology of KM [5], thus sharing with it the objective of providing a foundation for systematic KM research study and practice.

The work described in this paper has still several open issues, notably the modelling of the *context* of knowing [AKMC2], the wide range of knowledge *attributes* [AKMC3], the details of the representation of KM *conduct* concept [DKMC19-20] and an account of *projection* definition in KM [DKMC18].

Future work should integrate detailed ontological views of competency considering explicitly work situations [22], which are required to provide support for automated or semi–automated knowledge gap analysis, and for linking such knowledge needs to the learning contents and activities that may eventually overcome them.

References

[1] E. Abou-Zeid, A knowledge management reference model, Journal of Knowledge Mangement 6(5) (2002) 486–499.

[2] H. Allert, Coherent Social Systems for Learning: An Approach for Contextualized and Community-Centred Metadata, Journal of Interactive Media in Education 2 (2004) .

[3] I. Diakoulakis, N. Georgopoulos, D. Koulouriotis and D. Emiris, Towards a Holistic Knowledge Management Model, Journal of Knowledge Management 8(1) (2004) 32–46.

[4] T.R. Gruber, A Translation Approach to Portable Ontologies, Knowledge Acquisition 5(2) (1993) 199-220.

[5] C.W. Holsapple and K.D. Joshi, A Formal Knowledge Management Ontology: Conduct, Activities, Resources and Influences, Journal of the American Society for Information Science and Technology 55(7) (2004) 593–612.

[6] N. Kakabadse,A. Kakabadse and A. Kouzmin, Reviewing the Knowledge Management Literature: Towards a Taxonomy, Journal of Knowledge Management 7(4) (2003) 75–91.

[7] D. Karagiannis and R. Telesko, The EU-Project PROMOTE: A Process-Oriented Approach for Knowledge Management, in: Proc. of the Third International Conference on Practical Aspects of Knowledge Management (2000).

[8] R. Koper, Use of the Semantic Web to Solve Some Basic Problems in Education: Increase Flexible, Distributed Lifelong Learning, Decrease Teacher's Workload, Journal of Interactive Media in Education 6 (2004).

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

[10] LTCS WG12, Draft Standard for Learning Object Metadata, Technical Report, IEEE Learning Technology Standards Committee (IEEE Inc., 2002).

[11] A. Maedche, B. Motik, L. Stojanovic, R. Studer and R. Volz, Ontologies for Enterprise Knowledge Management, IEEE Intelligent Systems 18(2) (2003) 26–33.

[12] A.D. Marwick, Knowledge Management Technology, IBM Systems Journal, 40(4) (2001) 814–830.

[13] A. Mohame, S. Lee and S. Salim, An Ontology-Based Knowledge Model For Software Experience Management, Journal of Knowledge Management Practice 5 (2004).

[14] A. Örtenblad, On Differences Between Organizational Learning and Learning Organization, The Learning Organization 8(3) (2001) 125–133.

[15] M. Packer and J. Goicoechea, Sociocultural and Constructivist Theories of Learning: Ontology, not Just Epistemology, Educational Psychologist 35(4) (2000) 227–241.

[16] P.R. Polsani, Use and Abuse of Reusable Learning Objects, Journal of Digital information 3(4) (2003).

[17] W: Scholl, C: König, B. Meyer and P. Heisig, The Future of Knowledge Management: An International Delphi Study, Journal of Knowledge Management 8(2) (2004) 19–35. (2004)

[18] M.A. Sicilia, E. García-Barriocanal, S. Sánchez and E. Rodríguez, On Integrating Learning Object Metadata Inside the OpenCyc Knowledge Base, in: Proc. of the 4th IEEE International Conference on Advanced Learning Technologies ICALT'04 (IEEE Press, 2004) 900–901.

[19] M.A. Sicilia, E. García-Barriocanal, S. Sánchez and E. Rodríguez, Describing Learning Object Types in Ontological Structures: Towards Specialized Pedagogical Selection, in Proc. of World Conference on Educational Multimedia, Hypermedia and Telecommunications ED-MEDIA'04 (AACE Press 2004) 2093–2097.

[20] M.A. Sicilia and E. García-Barriocanal, On the Convergence of Formal Ontologies and Standardized e-Learning, Journal of Distance Education Technologies 3(2) (2005) 13–29.

[21] M.A. Sicilia and S. Sánchez, On the Concept of Learning Object "Design by Contract", WSEAS Transactions on Systems 2(3) (2003) 612–617.

[22] M.A. Sicilia, Ontology-Based Competency Management: Infrastructures for the Knowledge-Intensive Learning Organization, in: M. Lytras and B. Naeve, eds., Intelligent Learning Infrastructures in Knowledge Intensive Organizations: A Semantic Web perspective (IDEA Inc. Pub. 2005) *(to appear)*.

[23] D.A. Wiley, The Instructional Use of Learning Objects (Association for Educational Communications and Technology, Bloomington, 2001).