Resourceome: a Multilevel Model and a Semantic Web Tool for Managing Domain and Operational Knowledge*

Diletta Cacciagrano, Flavio Corradini, Emanuela Merelli, Leonardo Vito
Mathematics and Computer Science Department
University of Camerino
Via Madonna delle Carceri 9, Camerino (MC), Italy
{name.surname}@unicam.it

Gino Romiti
LOCCIONI Group
Via Fiume 16, Angeli di Rosora (AN), Italy
g.romiti@loccioni.com

Abstract

In any scientific domain, the full set of data and programs has reached an -ome *status, i.e. it has grown massively. In this scenario, we propose* Resourceome*, a Web-based and semantic Knowledge Management System which extends a previous multilevel ontology-based knowledge manager for the bioinformatic domain as follows: (i) enriching the knowledge model in such a way to describe resource knowledge contextualized in* any *domain (not only bioinformatics), as well as operational knowledge; (ii) implementing a Web-based application for visualizing, updating, querying and managing domain and operational knowledge models; (iii) adding a Web-based and semantic-driven workflow compiler.*

Keywords: Knowledge Management, Ontology, Semantic Web, Workflow.

1. Introduction

The original article on the Semantic Web [2] describes the evolution of a Web of *actionable information*, i.e. information derived from data through a semantic theory for interpreting the symbols. In a Semantic Web, methodologies are studied for describing, managing and analyzing both resources (domain knowledge) and applications (operational knowledge) - without any restriction on what and where they are respectively suitable and available in the Web - as well as for realizing automatic and semantic-driven workflows of Web applications elaborating Web resources.

Motivations and contributions of the paper. To satisfy the above requirements, Knowledge Management Systems (KMS) have to be enriched by ontology languages. To better understand how semantics can improve a KMS in the above sense, consider the following situation, contextualized in the travel service domain[1]. The traveler often has specific preferences regarding transport, accommodation and price. The process of finding the best travel and accommodation is often tedious and imprecise. A travel typically includes several possible means of transport and accommodation, and the traveler must manually contact multiple agencies in order to arrange the best possible travel and accommodation arrangement. Using a semantic-driven workflow compiler on a semantic KMS, able to automatically discover activities/resources as well as to invoke and to evaluate the results of these activities, allows a user to find the best available travel simply by declaring goals and preferences.

However, enriching a KMS by ontology languages is not enough. Semantics should be added to a KMS in such a way to enable the specification of different knowledge forms, both resource and operational knowledge, and their conceptualization in any arbitrary domain. For this reason, we have extended *Resourceome* [4], a multilevel ontology-based knowledge manager for the bioinformatic domain, to be (i) a domain-independent resource and operational knowledge model, (ii) a Web-based management tool for domain and operational knowledge organization, and (iii) a Web-based framework for assembly of semantically well-formed workflows from semantically heterogeneous sources (resources and activities). The revisited knowledge model inherits from the one in [4] the characteristic of splitting into two ontologies respectively domain and resource concepts, in such a way to formalize a resource conceptualization, contextualized in its domain. However, it extends the old model (i) providing a Task Ontology which catches different workflow views and which is independent from concrete formalisms describing them, (ii) implementing social roles in such a way to solve the so-called “Counting-Problem”, and (iii) providing a Web-based software layer to manage the knowledge base as well as to define and execute semantic-driven workflows.

The prototype of Resourceome, available at `http://resourceome.cs.unicam.it:8080`, has been already customized for two different domains - bioinformatics and industry. The first customization has been implemented to handle the unimaginable quantity of data and algorithms to organize and elaborate molecular biology knowledge. The

* *This work has been supported by LOCCIONI Group.*

1. This example can be easily implemented in Resourceome loading two predefined travel service Domain and Task ontologies.

latter has been implemented to satisfy a specific request of the industrial group LOCCIONI[2], in the travel service domain.

Plan of the paper. Section 2 briefly outlines the main features of the Resourceome model for domain and operational knowledge organization. Section 3 is the core of the paper and it depicts in detail the Resourceome knowledge model architecture, while the Web-based Resourceome knowledge manager and the workflow compiler are shown respectively in Section 4 and in Section 5. Finally, Section 6 proposes a short overview of related work, while Section 7 closes the paper.

2. The main features of the Resourceome knowledge model

The Resourceome knowledge organization is fundamentally a multilevel ontology-based model for semantic annotation of *resources* and *activities*, in according to Guarino's approach [7]. It is composed of three levels - Top knowledge, Base knowledge and Application knowledge level (Fig. 1).

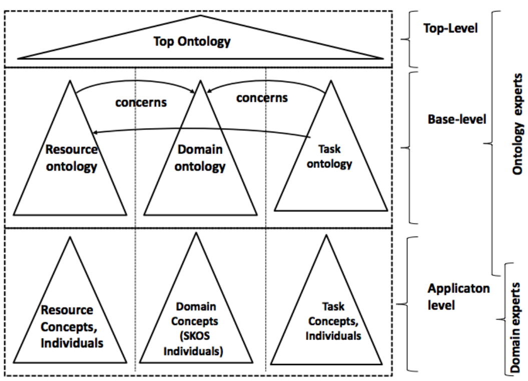

Figure 1. The Resourceome model.

The Resourceome knowledge model implements the following features:

1. *Separation between domain concepts (abstract and concrete concepts) and resource concepts (only physical concepts).*
In Resourceome it is possible to derive specific resources from resource concepts w.r.t. the domain where they are conceptualized. This is obtained by splitting the Domain Ontology (in the sense of Guarino's approach) into two "orthogonal" ontologies - Resource Ontology and Domain Ontology (in our sense) - and by connecting them by a "*concerns*" relation. This mechanism permits to parameterize the knowledge representation w.r.t. the domain. As a consequence, Resourceome is able to formalize a resource conceptualization, contextualized in and concerned to *any* fixed conceptualized domain. Through the "*concerns*" relation, a more specific resource can be simply connected to the domain topics which it refers to, rather than introduced in the Resource Ontology as a more specialized concept. Fig. 2 shows how a resource can be specialized w.r.t. a context either in the classical approach (figure on the top) or in Resourceome (figure on the bottom).

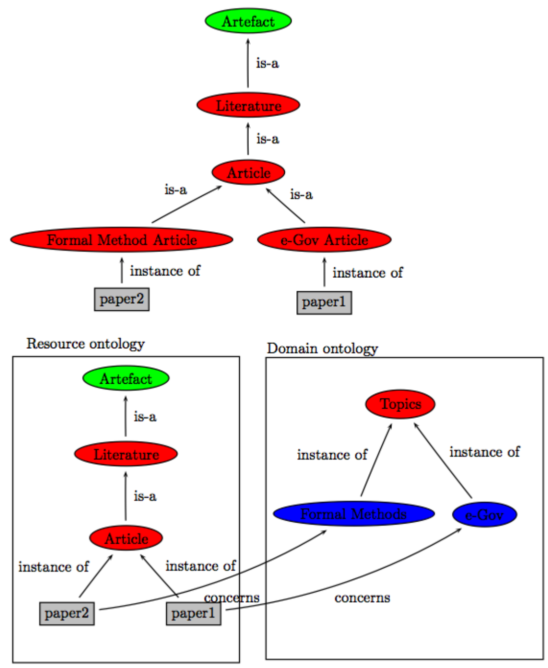

Figure 2. Resource contextualization.

2. *Independence from different workflow views and concrete formalisms.*
The *behavioral aspect* is the most explicit view of a workflow: it basically describes the order in which the different activities are executed. However, a workflow is more than the mere connection of activities: other basic aspects to consider are the *organizational* one - describing the organization structure, the involved *objects* and *roles* and in which way they are involved in the workflow - and *the informational* one - describing the concrete *informations* (or *documents*) associated to objects and roles that are involved in the workflow, how and where they are represented, and how they are propagated among different activities.
Usually, each view is modeled by a specific and suitable formalism: for instance, Petri Nets, organigrams and ER diagrams respectively for behavioral, organizational and informational aspects. For this reason, the Task Ontology in Resourceome has been defined in order to

2. `http://www.loccioni.com`

be independent from concrete formalisms and to catch the essence of each view.

The Task Ontology is connected to Domain and Resource Ontologies by abstract relations, in such a way to describe the execution context of an activity in a semantically rich fashion. These relations link (the most general) activity concept in the Task Ontology with (the most general) domain and resource concepts. Specializations of these relations permit to attach the involved roles, documents and objects to a specific activity, as well as to a domain, i.e. the context in which the activity works. These relations provide the needed support for realizing the semantic-driven workflow compilation process in Resourceome.

3. *Representation of social roles.*
   The notion of *role* is crucial in a workflow scenario. A role is a logical abstraction of one or more physical actors which can perform an activity. The conceptual modeling and object modeling literature considers the roles as anti-rigid and relationally dependent unary predicates. For example, take the role *Student* that is subsumed by the kind *Person*: *Student* is anti-rigid because persons are only contingently students, for example a person can be a student only during a short period of his life. Additionally, *Student* is relationally dependent because, for a person to be *Student* it requires the existence of another entity, namely a certain *University* in which this person *is Enrolled*. Resourceome implements the approach proposed in [9], which it has been proved to solve the so-called "Counting-Problem". Fig. 3 shows how a social role can be described either in the classical approach[3] (figure on the top) or in Resourceome (figure on the bottom).

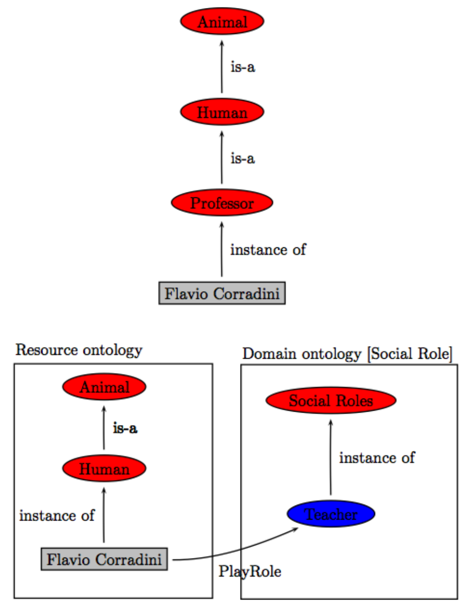

Figure 3. Social roles representation.

3. The Resourceome knowledge model

In this section we will describe in detail the multilevel architecture of the Resourceome knowledge model.

3.1. Top knowledge level

The Top knowledge level is formed by an Upper Ontology describing very general and domain-independent concepts shared across a large number of ontologies. Several standard upper ontologies are available, for instance DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) [8] and SUMO (Suggested Upper Merged Ontology) [15]. The choice of the Upper Ontology concepts depends on what and how the knowledge is going to be described.

3.2. Base knowledge level

The Base knowledge level describes a specific vocabulary by specializing the terms introduced in the Upper Ontology w.r.t. a particular domain of interest.

Domain and Resource Ontologies. The Domain Ontology (in the sense of Guarino's approach) is split into two "orthogonal"[4] ontologies: Resource Ontology and Domain Ontology (in our sense).

The Resource Ontology represents the kind of resources existing in the universe of a domain. It is a representation of a physical world, since it models the types of resources existing in the described domain. It contains the declaration of terminology and relations in the form of OWL-DL entities.

The Domain Ontology represents the semantic relationships between the concepts of the domain. It is implemented with a hybrid OWL-DL/SKOS semi-formal language, in order to provide more flexible and less formal description of concepts and metadata[5]. A domain concept may be both a sub-class of a domain OWL-DL class and a sub-class of SKOS-concept class. On the one hand, SKOS permits to represent concepts as individuals of SKOS-concept class and to join them together with more flexible relations instead of the strict "*is-a*" one. On the other hand, SKOS does not provide strict computational semantics and thus cannot

3. This representation suffers of the so-called "Counting Problem", as stated in [9].

4. As already said in Section 2, the "orthogonality" is realized by a "*concerns*" relation, which allows us to connect the two ontologies and to parameterize the knowledge representation w.r.t. a domain.

5. SKOS is represented in OWL-DL. In detail, in OWL 2 it is possible to have an ontology where the names of individuals can be the same as the names of classes.

be used for performing automated tasks associated to the knowledge represented in the scheme. In the whole, the hybrid OWL-DL/SKOS semi-formal specification of the domain offers a great flexibility and the possibility of creating relationships between individuals and domain/operational concepts: it combines the OWL-Full flexibility - since it can describe relations between resource individuals and domain concepts - and the OWL-DL computational power - since the adopted interbreeding technique does not add non-computable elements.

Task Ontology. The Task Ontology contains the declaration of terminology and relations in the form of OWL-DL entities. The Base level is substantially a meta-model for catching different workflow views (behavioral, organizational and informational aspects), in such a way to be independent from concrete formalisms usually used for modeling them. The pivot of this level is represented by the generic concept of *activity*. Every activity executes in a defined context (domain concept) and includes the following items:

- *Role* (domain concept): A logical abstraction of one or more physical actors, usually in terms of common responsibility or position. An actor may be a member of one or more roles. Roles specify the actor types (e.g. researcher, teacher, etc.) that participate in the workflow and perform activities. For maximum flexibility and reuse, activities are assigned to roles rather than to named users. Using roles instead of assigning a real user's name makes changes easy to manage and it enables a user to easily delegate and reassign activities. Every role has a set of *competencies*, which describe what the people associated with that role are allowed to do.
- A set of *objects* (resource concepts), which the activity eventually inputs, outputs and uses and/or consumes.
- *Document*: The set of informations (resource concepts) that are interchanged as part of an activity.
- A *code* implementing the activity.
- It possibly has *preconditions* and *effects* (rules)[6].
- It is decomposed into more detailed *sub-activities* (task concepts).

As already said in Section 2, the Task Ontology is connected to Domain and Resource Ontologies by abstract relations - in detail, "*hasRole*", "*hasDocument*", "*hasComplexInput*", "*hasComplexOutput*" and "*concerns*". Their specializations permit to attach the involved roles ("*hasRole*"), documents ("*hasDocument*") and objects ("*hasComplexInput*", "*hasComplexOutput*") to a specific activity, as well as to a domain ("*concerns*") or context in which the activity works.

6. Actually, Resourceome does not support rules. This item is introduced for further extensions.

3.3. Application knowledge level

The Application knowledge level introduces very specific concepts depending on the particular domain, resources and activities.

Domain and Resource Ontologies. The Resource Ontology depth hardly grows: as already explained in Section 2, in the most cases inserting more specific resources implies connecting their more generic forms, already in the Resource Ontology, to related domain concepts which they refer to. Eventually, new domain concepts and new general resource concepts are inserted respectively in Domain and Resource Ontologies.

Task Ontology. This level models the semantic (specialization/generalization) relationships among concrete activities, the conditions to be met for an activity specialization to take place, and the context each activity works on. Basically activities can be retrieved by their name, and a minimal set of property fields can store unlimited strings, so that any implementation is possible. This layer is actually a forest, i.e. a tree for each activity: the child (hypoactivity) of a node activity (hyperactivity) is a more specific version of it. The activity also specifies additional constraints about the value of some of the parameters defined by its hyperactivity. These constraints, expressed in any form supported by Domain and Resource Ontologies, are used to determine the applicability of the activity with respect to the execution of its hyperactivity.

4. The Web-based Resourceome knowledge manager

The Resourceome knowledge manager is a Web-based application that provides an intuitive user interface, based on the Resourceome knowledge model, for the representation, visualization, integration, management and querying of domain and operational knowledge using Semantic Web technologies. The Web-based approach differs Resourceome from others visualization tools - such as the plugin OWLViz [11] for Protégé [12] - since it allows collaboration between different users, through the network, simply operating by mean of a Web browser.

Visualization of the Resourceome knowledge model. Intuitive navigation is allowed by an effective Resource, Domain and Task Ontology visualization.

On the Resource Ontology, the resources concerning a particular topic are connected by an arrow to that topic and it is shown only the subtree of the Domain Ontology, having concepts which are concerned by resources of the chosen types. By pressing the button "SHOW INSTANCE TREE", it is possible to visualize the subtree of the Resource Ontology having resource instances (individuals) of the selected types. Once selected a particular resource instance (either in the domain view or on the instance tree) it is possible to see all its relationships with other resources, to look at its attributes and to access it through its URI (if available).

Similarly, by right-clicking on a specific activity in the application level of the Task Ontology, it is possible to visualize the domain concept representing the context of the activity and any resource concepts representing the involved roles, objects and documents.

Management of the Resourceome knowledge model. In the system, two kinds of users are allowed and are equipped with different privileges: administrators and simple users. Users, which may not be familiar with ontologies and metadata schema, can only modify the Application level of Resource and Domain Ontologies. Administrators, who are supposed to have more experience in working with and developing ontologies, have access also to Top and Base levels of Domain and Resource Ontologies, as well as to any level of the Task Ontology.

In detail, in the Resource Ontology users can easily add new (OWL-DL) resource individuals and relationships; attributes and relationships required for the definition of a new resource individual can be visualized in a separate window. On the Domain Ontology, it is possible to add new sub-concepts of a selected one: users can only add (blue-colored) SKOS concepts, whereas administrators can also add (red-colored) OWL-DL concepts. A SKOS parser [14] helps the administrator to translate SKOS user add-ins in term of OWL-DL concepts and relations to uniform and improve the Domain Ontology.

On the Task Ontology, administrators can manage task concepts in the Top and Base levels, as well as either to add/delete/configure concrete activities in the Application level or to specialize any abstract relationships (involving general domain and resource concepts) in term of specialized relationships (involving specific domain and resource concepts).

5. The semantic-driven workflow compiler

As in [1], the workflow compilation process permits to translate a user domain-specific workflow specification into mobile code supported by Hermes middleware[7]. In detail, the compilation process requires three main components: (i) WebWFlow (see Fig. 4) - which allows the user to define graphically a workflow of domain-specific activities, defined in the Task Ontology Application level; (ii) XPDLCompiler - which translates the (role-based) workflow specification into an interactive component-based specification and generates the code to be executed on Hermes middleware. In this case, a repository provides the implementation of each activity as a code template; (iii) Hermes middleware - which supports the generated code execution and mobility. However, differently from [1], the domain and operational knowledge model behind Resourceome allows each activity in the workflow to be automatically linked to its context, roles, objects and documents.

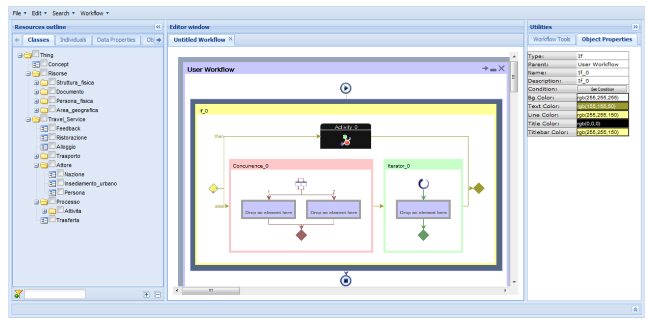

Figure 4. The graphic workflow editor.

6. Related work

Regarding previous work on actionable ontologies for data mining, there have been efforts to provide a systematic description of data and processes for the classical data mining tasks e.g. in systems MiningMart [10], CITRUS [18], CAMLET [17] and NExT [3].

The MiningMart system focuses on propositional data mining from data stored in a relational database; it contains an XML-based meta-model for representing and structuring of information about data and algorithms and it does not provide means for automatic workflow creation. The project CITRUS uses an object oriented schema to model relationships between the algorithms; the system focuses on guiding the user through mostly manual process of building of workflows, by including information about properties and usability of the algorithm in the algorithm description. In the CAMLET system an ontology of algorithms (processes) and an ontology of data structures are defined; however no ontology language is specified and the structure of algorithm ontology does not attempt to formalize the domain systematically.

The most systematic effort to construct a general knowledge discovery ontology is described in [3]. The ontology used by the NExT system is built on OWL-S and provides a relatively detailed structure of the propositional data mining algorithms.

With regard to combining a comprehensive conceptual model of an enterprise with the actual production system and executable workflows, the ARIS methodology [13] and respective tooling support was a major step. ARIS includes not only the control flow perspective, but also the organizational, data, and functional dimensions of enterprises. While ARIS

7. Due to the lack of space, middleware architecture is not discussed here and we refer to [5] for further details.

is based on such a comprehensive conceptual framework, the expressiveness and degree of formality in the various models is rather limited and the links between the various models are quite weak - for instance, querying the process space using logical expressions and machine reasoning is very hard.

In the semantic business process area, the SUPER Project [16] is undoubtedly worth noting. The major objective of SUPER is to raise Business Process Management (BPM) to the business level, where it belongs, from the IT level where it mostly resides now. The project aims at providing a semantic-based and context-aware framework, based on Semantic Web Services technology that acquires, organizes, shares and uses the knowledge embedded in business processes within existing IT systems and software, in order to make companies more adaptive.

7. Conclusions and future work

In this paper, we have proposed *Resourceome*, a Web-based and semantic KMS which extends a previous multilevel ontology-based knowledge manager for the bioinformatic domain [4] (i) enriching the knowledge model in such a way to describe resource knowledge contextualized in *any* domain (not only bioinformatics), as well as operational knowledge, (ii) implementing a Web-based application for visualizing, updating, querying and managing domain and operational knowledge models, (iii) adding a Web-based and semantic-driven workflow compiler.

We intend to improve the workflow compiler in such a way to allow the user to save graphical workflows of domain-specific activities as a semantic workflow logs, i.e. as OWL representations of the corresponding process instances. Since ontologized workflow logs do not only contain simple types, but identify semantic objects instead, they could potentially allow information systems to exchange runtime information. Further, providing means to integrate workflow logs in a unified model, with a formal semantics and unambiguously identified objects, would enable process mining across the execution traces of multiple information systems.

References

[1] Bartocci, E., Corradini, F., Merelli, E. and Vito, L., "Model driven design and implementation of activity-based applications in Hermes", Proc. of the 7th WOA 2006 Workshop, From Objects to Agents (2006).

[2] Berners-Lee, T., Hendler, J. and Lassila, O., "The Semantic Web", Sci Am., 284, pp. 34-43 (2001).

[3] Bernstein, A. and Deanzer, M., "The next system: Towards true dynamic adaptions of semantic web service compositions (system description)", Proc. of the 4th European Semantic Web Conference (ESWC'07), Springer (2007).

[4] Cannata, N., Merelli, E. and Altman, R.B., "Time to organize the bioinformatics resourceome", PLoS Comput Biol, 1(7) (2005).

[5] Corradini, F. and Merelli. E., "Hermes: agent-based middleware for mobile computing", Mobile Computing, LNCS, 3465, pp. 234–270 (2005).

[6] Gruber, T. R., "A translation approach to portable ontology specifications", Knowledge Acquisition, 5(2), pp. 199-220 (1993).

[7] Guarino, N., "Formal Ontology in Information Systems: Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy" (1998).

[8] Masolo, C., Borgo, S., Gangemi,A., Guarino, N., Oltramari, A. and Schneider, L., "Wonderweb deliverable d17. The wonderweb library of foundational ontologies and the dolce ontology" (2002).

[9] Masolo, C., Guizzardi, G., Vieu, L., Botazzi, E. and Ferrario, R., "Relational roles and qua-individuals", Proc. AAAI Fall Symposium on Roles, an Interdisciplinary Perspective, pp. 103–112 (2005).

[10] Morik, K., Scholz, M., "The miningmart approach to knowledge discovery in databases", Proc. of the International Conference on Machine Learning, Springer, pp. 47–65 (2004).

[11] OWLViz. `www.co-ode.org/downloads/owlviz` (15.07.2009)

[12] The Protégé Ontology Editor and Knowledge Acquisition System. `http://protege.stanford.edu` (15.07.2009)

[13] Scheer, A.W., "ARIS - Vom Geschftsprozess zum Anwendungssystem", Springer (1998).

[14] SKOS Parser. `http://oaei.ontologymatching.org/2008/skos2owl.html` (15.07.2009)

[15] Suggested upper merged ontology (sumo). `http://www.ontologyportal.org` (15.07.2009)

[16] SUPER Project. `http://www.ip-super.org/` (15.07.2009)

[17] Suyama, A., Negishi, N. and Yamaguchi, T., "CAMLET: A platform for automatic composition of inductive learning systems using ontologies", Proc. of PRICAI'98, LNCS, 1531, pp. 205–215, (1998).

[18] Wirth, R., Shearer, C., Grimmer, U., Reinartz, T. P., Schloesser, J., Breitner, C., Engels, R. and Lindner, G., "Towards process-oriented tool support for knowledge discovery in databases", Proc. of the First European Symposium on Principles of Data Mining and Knowledge Discovery, LNCS, 1263, pp. 243–253 (1997).