Applying Rules to an Ontology for Project Management

Fran J. Ruiz-Bertol[1], Daniel Rodríguez[2], and Javier Dolado[3]

[1] Tecnalia
Fran.Ruiz@tecnalia.com,
[2] Universidad de Alcalá de Henares
daniel.rodriguezg@uah.es,
[3] Universidad del País Vasco
javier.dolado@ehu.es

Abstract. There is an increasing interest in using ontologies in the area of project and process management and obtain knowledge from ontologies by reasoning. Different approaches are being used for representing knowledge in this field. Based upon previous works on project representations we have developed a basic ontology and we apply rules on it. We used recommended OWL and SWRL languages for defining the ontologies and rules, respectively. Current article shows an example of different types of rules that can be applied on our specific ontology. In this way, we see how further knowledge can be derived and, thus, decision-making for managing projects can be improved.

Keywords: project management, software engineering, ontologies

1 Ontologies for Project Management

Although project management nature need to be continiously registered, tracked and controlled, few knowledge systems have been developed in this field. But during last years it exists a growing interest in specifying how to represent conceptually project derived data [4] [1] [9]. However, data representations proposed during the last decades are not able to gather all knowledge obtained from project and project management activities. Ontologies seems to be the current trend for knowledge representation and knowledge data mining [8] [7].

Ontologies provide a conceptualization of a part of the world that is of interest to the modeller. Although the idea of ontologies is not yet fully mature for its practical and widespread use, W3C states as a recommendation the usage of a language for defining and reasoning about the concepts of an ontology. These languages, OWL for defining domain vocabulary, axioms, taxonomy and relationships, and SWRL, for defining rules over ontologies, allows the information analyst to model ontologies and to reason about them [5].

2 An ontology for project management

Based on other authors' published ideas and on our previous works on project representations [6], we dispose of a basic ontology for the project management. The ontology has define a set of core concepts directly related with project management activities, as well as extensions to the core ontology that provide additional concept definitions and rules for reasoning over data. As there is not a unique way to represent the concepts under the manager's viewpoint, different alternatives to proposed ontological modelling may coexist.

A growing area of research is the interaction and combination of different ontologies for the same field or area of interest, but those issues are beyond the scope of our current research. We simply model our concepts of interest according to our perspective of the representations of project management. Thus, *PM-Core* and *PM-Organization* ontologies contribute to specify knowledge related with project activities (see Fig. 1) and project stakeholders. Fig. 1 shows the taxonomy for a given project, based on the breakdown structure defined in [3]. These ontologies were built with the idea of our specific conceptualization as the main goal. In order to obtain further knowledge from data defined in the ontology, we have extended the core ontology to see how SWRL rules can be applied. We have called this part *Core-extension* ontology, which explicitly deals with problems in project management (see Figure 2). The SWRL rules are applied to this part as it is explained in section 3.

The extension to the core ontology is based on the multiple underlying applications that the concept of *Activity* has. In this case we model some of the generic problems or issues that activities can deal with within the context of a project. The ontology of Figure 2 can be read, in a concise manner, as follows: in every project there are problems, interruptions or issues, which are represented as `Management Issues`. Each `Management Issue` can be managed by an `Activity`, that in our case is modeled as `Activity` ***`Manages Issue`*** `Management Issue`. Each `Management Issue` can be classified in different types `Type Issue`, and there is a subset of problems that can be effectively managed `Type of Issue Managed` in one way or another. `Management Issue` is related to the `Type Issue` with ***`hasIssue`*** and it is also related to `Type of Issue Managed` with ***`hasIssueManaged`***. In summary we see that this part conceptually may assign managerial problems to activities, and that there is a subset that can be effectively managed.

3 Semantic Web Rule Language (SWRL)

The language SWRL has been proposed for adding rules in the process of reasoning about an ontology developed in OWL [2] [5]. SWRL allows reasoning about OWL individuals. As defined by W3C, SWRL extends the set of OWL axioms with Horn-like rules. It combines the sublanguages of OWL-DL, OWL Lite and RuleML. Rules are of the form of an implication between an antecedent (*body*) and consequent (*head*). The intended meaning can be read as: whenever

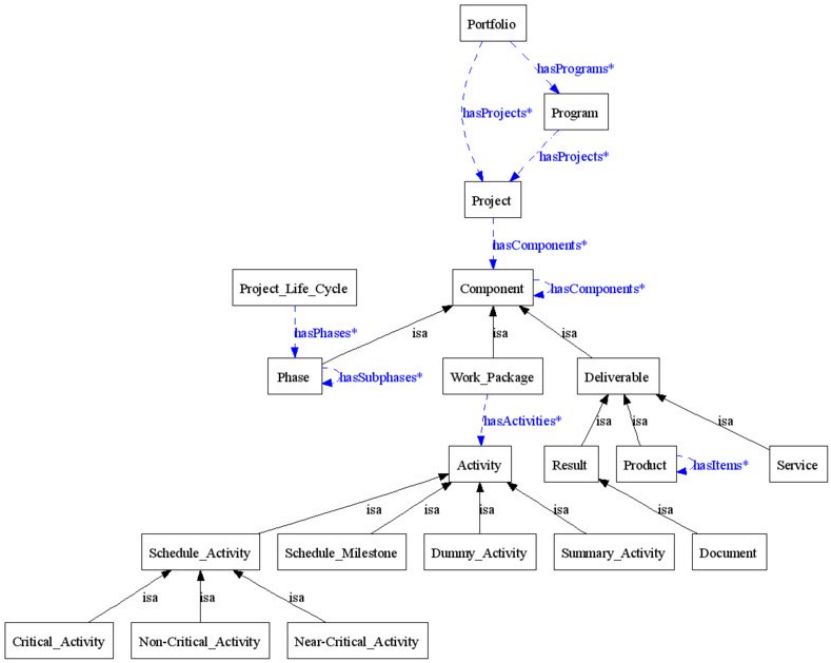

Fig. 1. A simple PM-Core ontology taxonomy. The concept `Activity` will be refined.

the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold.

The classic simple example of the basic rule definition is the expression $\texttt{hasParent}(?x_1,?x_2) \wedge \texttt{hasBrother}(?x_2,?x_3) \Rightarrow \texttt{hasUncle}(?x_1,?x_3)$, which has the obvious meaning that if an individual $?x_1$ has a parent $?x_2$ and $?x_2$ has a brother $?x_3$, then $?x_1$ has an uncle $?x_3$.

From the practical approach, rules are executed using the Jess rule engine in the Protégé environment. Other alternative could use the Pellet reasoning engine but it does not add more value to our development.

4 Applying rules to the ontology

Project Management Ontology provides the knowledge related with project structure and project organization [6]. We propose the usage of this ontology as the basis for defining new knowledge using the rules we propose in this section. These set of rules allows deriving new knowledge based on the current data in the ontology. This knowledge is not explicitly available on the ontology, but it can be derived from ontology data. In order to have comprehensible rules to demonstrate that is possible to create new knowledge from a given project data. For this purpose, we think most amenable data for reasoning is the one related

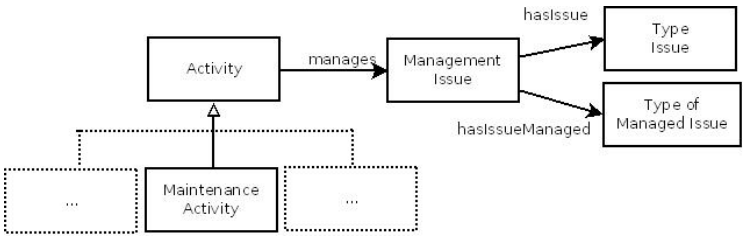

Fig. 2. Extension to the Core ontology with Management Issues.

with issues reporting, changes or event management. We have defined some basic SWRL rules on the extension to the Core Ontology and this allows us to represent management constraints and to add new knowledge to the ontology. Another set of SWRL rules are those that querying to the ontology for getting specific sets of values. These query and reasoning rules show how it is possible to build a reasoning subsystem on the concepts previously defined.

Below we describe part of the rules built in our ontology which can be classified as query and management-reasoning rules. Examples of query rules are:

- Query Rule #1: Get all *Team Members* for a given *Project*.
  `org:ProjectTeamMember(?p)` ⇒ `sqwrl:select(?p)`

- Query Rule #2: Obtain all *Change Requests* that have been initiated by an specific *Team Member* (`Daniel`)
  `pmo:ApprovedChangeRequest(?r)` ∧ `pmo:changeRequestInitiator(?r, p1:Daniel)` ⇒ `sqwrl:select(?r)`

But the most interesting part of rules is not getting derived knowledge, already available in the KBS. It is interesting to check how rules provide support for generating new knowledge based on the current one, when a set of constraints are met. This is the reasoning rules that we put as example for demonstrating new knowledge creation based on triggering given for the antecedent part:

- Reasoning Rule #1: When there is a *Severe Problem* and there is a *Maintenance Activity*, then the *Management Issue* is managed by that *Maintenance Activity*.
  `pmo:ManagementIssue(pmo:SevereProblem)` ∧ `pmo:MaintenanceActivity(?a)` ⇒
  `pmo:hasIssue(pmo:SeveralProblem, pmo:GeneralProjectReview)` ∧
  `pmo:hasIssueManaged(pmo:SevereProblem, pmo:reactive)` ∧
  `pmo:managesIssue(?a, pmo:SevereProblem)`

- Reasoning Rule #2: If there is some *Trivial Problem* labeled as *Reactive* and there is some kind of *Maintenance Activity*, then the *Maintenance Activity* takes responsibility over the *Management Issue*.
  `pmo:ManagementIssue(pmo:TrivialProblem)` ∧
  `pmo:hasIssueManaged(?i, pmo:reactive)` ∧ `pmo:MaintenanceActivity(?a)`
  ⇒ `pmo:managesIssue(?a, ?i)`

– Reasoning Rule #3: If appears a *Personnel Problem* (e.g. somebody gets sick) and that should be covered at *Internal* level, and there is one *Activity* for evaluating team composition and levelage, then assign that *Management Issue* to that *Management Activity*.
`pmo:ManagementIssue(pmo:PersonnelProblem)` ∧
`pmo:hasIssueManaged(?i, pmo:internal)` ∧ `pmo:ManagementActivity(?a)`
⇒ `pmo:managesIssue(?a, ?i)`

For defining all the rules, we have used the Protégé environment, having Jess as reasoning engine for executing the rules. This combination allows us to develop a full set of reasoning rules for creating new knowledge based on the `antecedent` ⇒ `consecuent` SWRL rule definitions. Fig. 3 shows the execution of Reasoning Rule #3 in this environment.

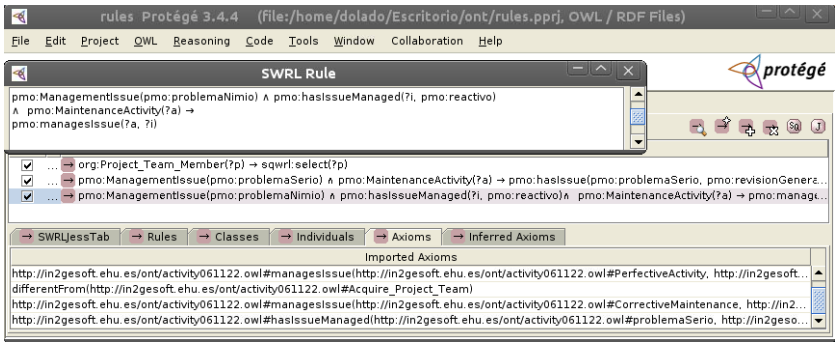

Fig. 3. Execution of Reasoning Rule #3 in a Jess-Protégé environment

5 Conclusion and future work

The work presented here outlines the possibility to extract implicit knowledge from ontology. We have shown, using different rule examples, how to apply rules defined in SWRL to create new knowledge from explicit data defined in the ontology. Altought we are concious that the current rules only show a limited data mining, we are confident that in the future, rules can be used for defining not only implicit new knowledge, but being capable for creating a true knowledge based system for project management. We are concious about the big problem that project management represents in the software development, and the information need at different management level.

Here we demontrated that the applying of rules to ontologies contributes positively in reasoning about facts that underlies from explicit data defined in the ontology.

We have developed two classes of rules: one for querying and the other for reasoning. Queries cannot strictly be considered rules, since they do not derive additional behaviour. Further work is needed to organise and/or to formalise the reasoning rules, specifically for project management. While the conceptual structure of project management has been studied in the last decade, few effort has been employed to understand the different ways of reasoning in management. Its formalization through SWRL may provide a route to help the managers' endeavours, but there is a lot of work to do for organising all the rules used by managers.

Future work is related on generating more complex reasoning rules that allows project managers to have up-to-date information about any issue that is ocurring during the project development.

Acknowledgments

This work has been partially supported by project UPV/EHU Onproiek EHU08/ 40.

References

1. Abels, S., Ahlemann, F., Hahn, A., Hausmann, K., Strickmann, J.: PROMONT - A project management ontology as a reference for virtual project organizations. In: Proceedings of the On the Move to Meaningful Internet Systems OTM Workshops (OTM'06). pp. 813–823 (2006)
2. Dong, H., Hussain, F.K., Chang, E.: Application of Protégé and SPARQL in the field of project knowledge management. International Conference on Systems and Networks Communication pp. 74–80 (2007)
3. Institute, P.M.: A Guide to the Project Management Body of Knowledge (PMBOK® Guide). Project Management Institute, 4th edn. (2008)
4. Rabelo, P., Do Amorim, S.: Ontology, management of project process, and information technologies. Proceedings of the 6th European Conference on Product and Process Modelling - eWork and eBusiness in Architecture, Engineering and Construction, ECPPM 2006 pp. 295–302 (2006)
5. Rodríguez, D., García, E., Rodríguez-Solano, C.: Defining software process model constraints with rules using OWL and SWRL. International Journal of Software Engineering and Knowledge Engineering 20(4), 533–548 (2010)
6. Ruiz-Bertol, F.J., Dolado, J.: Una ontología para la gestión del conocimiento de proyectos software. Revista Española de Innovación, Calidad e Ingeniería del Software 4(1), 6–22 (2008)
7. Sarantis, D., Charalabidis, Y., Askounis, D.: An ontology for stakeholder collaboration and knowledge exploitation in e-government project management. ACM International Conference Proceeding Series pp. 61–67 (2009)
8. Sarantis, D., Askounis, D.: A project management ontology as a reference for egovernment projects. In: International Conference for Internet Technology and Secured Transactions (ICITST). pp. 1–8 (2009)
9. Tserng, H.P., Yin, S.Y., Dzeng, R., Wou, B., Tsai, M., Chen, W.: A study of ontology-based risk management framework of construction projects through project life cycle. Automation in Construction 18(7), 994–1008 (2009)