Ontology-based Tailoring of Software Process Models

Ricardo Eito-Brun

Ontology-based Tailoring of Software Process Models

Ricardo Eíto-Brun

Universidad Carlos III de Madrid.

Abstract. This paper describes the development of an ontology to support the tailoring of complex process models in the software industry. The ontology is made up of three layers that represent: a) the process characteristics extracted through the analysis of an international standard, b) the tailoring of this standard made at the corporate level, and c) a final tailoring done in response to client-specific, project-based requirements. The tailoring extends and refines the elements defined on the upper levels to support the customization of tasks and work products for different projects. The resulting ontology supports the tailoring of the process definitions and can also be used as an indexing tool for the easy identification of the requirements and constraints that have an impact on the elaboration of work products and on the planning, execution and monitoring of project activities.

Keywords. Ontologies; Software Process; Software Engineering; Requirements management; SemanticWiki.

1 Introduction

Software development standards provides organizations with prescriptive process models comprising the list of sub-processes, tasks and activities, their inputs and outputs, and the description of the roles involved in their execution. Compliance with these standards is usually considered a best-practice to ensure the successful completion of IT projects. The components defined in these prescriptive process models may be combined according to different life cycles widely known in the industry: waterfall, incremental, evolutionary, agile, etc. The combination of process models – defining tasks, work products and roles -, with life cycles (waterfall, incremental, etc.) result in methodologies that will be enacted at individual projects and reflected in a project plan. This plan includes the result of tailoring the selected methodology to meet project specific needs.

This process adaptation activity is possible because organizations developing software have at their disposal methodologies providing descriptions of the planned sub-processes, activities and tasks, their sequence and the work products to be generated. De-facto standard methodologies like RUP (Rational Unified Process), OpenUP (Open Unified Process) or international standards ISO/IEC 29110 or ISO/IEC 12207 are examples of these methodologies. Usually, companies adopt a particular methodology to develop their projects. Sharing a common methodology to plan, execute and control projects is a source of savings, as the execution of tasks is predictable, can be measured for further improvement, staff is trained on a common set of tasks and can be easily reallocated between different projects.

But complex projects usually require a wider, more flexible approach to accommodate additional tasks, activities and work products to those defined in the standard or corporate methodologies. Customers impose additional, project specific requirements that may affect both managerial and engineering activities and work products. These customer requirements introduce an additional complexity or understanding of corporate practices, and need to be incorporated or combined with the methodology applied by the company. For example, methodologies usually ask for the regular reporting of the project status to the project's stakeholders; this generic requirement should be reviewed and adapted, in different projects, to the specific customer requirements and preferences specifying different frequencies, layouts, formats or data to be reported.

This paper presents a model that uses ontologies to organize the collaboration framework of software development teams. The proposed solution is not only valid to help engineers organize documents and records, but to serve as a "compass" that facilitates the understanding and comprehension of different requirements stated in separate standards: international standards used as normative frameworks, project corporate guidelines and customer requirements. The paper proposes a general approach that may be easily tailored to support different normative frameworks.

2 An layered-framework for process tailoring

The proposed solution can be defined as a framework made up of three layers:

1. The core layer, based on internationally recognized process standards.
2. The corporate layer, based on the tailoring that individual companies make of the international standards to define their corporate policies and work procedures.
3. The project layer, based on the result of tailoring the corporate layer to accommodate specific requirements imposed by the customer or the project's stakeholders.

The framework – and its representation as an ontology – is based on the process approach. Process approach has become one of the most value paradigms for the organization of work. Companies and organizations are asked to be process-oriented, and attributes related to processes' performance, management and measurement are recognized as the right indicators to assess the maturity of the organizations. In addition, processes have become the way visualize how to do the work and how each person activities relate to those of other partners and colleagues. Following this approach, the starting point of the proposed framework and its representation as an Ontology is a process model comprising a set of tasks and activities, work products and roles shared at the organizational level. The process model and the layered architecture provide the context to understand the impact of the different requirements imposed by standards, corporate guidelines and policies, and customers.

2.1 The Core Layer

To illustrate the proposed approach and define a widely reusable ontology, the study has taken as a reference the process model described in the international standard

ISO/IEC 29110 [1]. This standard constitutes the core of the proposed framework, although it is remarked that any other recognized process standard or methodology could have been used as a reference.

ISO/IEC 29110 "Software engineering -- Lifecycle profiles for Very Small Entities (VSEs)" is a recently published standard elaborated by ISO/IEC JTC 1 "Information Technology" SC7 "Software and Systems engineering " with the purpose of defining standard profiles specifying the software development life cycle processes for Very Small Entities (VSE). VSE are defined as organizations, departments or Project teams with no more than twenty five workers, although the standard could be used in bigger contexts. The contribution of these companies is quite relevant in to the global economy, and VSE are involved and participate in major initiatives developing and maintaining software components that are later integrated into bigger systems by upper-level contractors.

ISO/IEC 29110 tries to give answer to one of the traditional problems in process improvement at small companies: existing models like CMMI or SPiCE may result into complex, hard to complete projects, and impose an unaffordable overhead for this kind of companies, who cannot afford adopting and deploying these process improvement models from scratch as they require a significant investment in time, financial and human resources. ISO/IEC 29110 provides a simplified set of processes, focused on key activities and deliverables that may be adopted before moving to more complex process improvement models.

From this standard, it is possible to identify the main elements, classes and properties of the target ontology.

2.2 The Corporate Layer

Organizations involved in software development usually define and establish a set of work process based on existing standards. This is usually done in response to the complexity of methodologies like OpenUP, RUP or standards like ISO/IEC 12207, whose scope may be too wide and complex for the adopting organizations. A common practice is to tailor or adapt the process model defined in any of these standards. Tailoring removes those aspects that are not applicable to the organization, and emphasizes those that add value to their business. The organization's adopted methodology gives common guidelines to develop projects and to ensure a common understanding of processes among the different teams and staff. It constitutes the basis for a common management across projects.

2.3 The Project Layer

The adaptation of the core processes done at the corporate level must be further developed at the project level. In some cases, projects may be executed directly enacting the process model defined at the corporate level, with no additional customization. In other cases, where the customer provides additional management or engineering requirements, the corporate process model needs to be refined: it is necessary to accommodate and reflect the impact of those requirements in processes and work

products. The project layer is the result of tailoring the process and work product model defined at the corporate layer. The result is a process definition that can be directly enacted or instantiated.

The implementation of this layer can serve as an evidence of organizational maturity: in the software development industry, assessment models like CMMI or SPiCE define maturity level 3 as the capability of tailoring existing corporate procedures to respond to projects specific needs.

3 Designing the ontology

The selection of ontologies, and the OWL language in particular, to represent the knowledge about process models embedded in standards and customer specifications was based on these assumptions:

- Ontologies provide a robust framework to represent concepts, their relations, and incorporate equivalent concepts and terms. Ontologies give the opportunity of managing multiple literals and translations for the same concept, and concepts with an equivalent meaning can be represented using the equivalence relationship built-in the OWL language.
  This aspect is relevant for software process modeling, as there are significant differences in the terminology used by different standards and methodologies to refer to the same concepts.
- The problem under analysis not only requires the creation of a data model or data structure, but the encoding of specific, particular data in a reusable way.
- Ontologies are a standard method to represent knowledge. Process models represented in this language can be easily reused by other applications and serve multiple purposes.
- The modular characteristics of ontologies make possible the management of process models and requirement set in separate modules. These modules can be maintained in separate files and combined as needed to create links and dependencies.
- The use of XML makes possible the exchange of the data with different process modeling tools supporting XML-based vocabularies like SPEM (Software Process Engineering Modeling) or BPMN (Business Process Modeling Notation).
- Integration of the ontology with the Semantic Wiki tool is easier.

The design of the ontology starts with a set of competence questions [2, 3]. These are related to the information needs of the ontology's end users. In this particular case, typical questions are related to the constraints that the set of applicable requirements impose when planning or executing tasks or elaborating a work product or deliverable, for example:

- Which is the content that is expected for a particular document?
- Is there any requirement asking for reporting activities?
- Which is the format that needs to be used to deliver the software and related documentation?

- Which is the version of the notation that needs to be applied in the design documents?

One of the concerns leading to the design of the framework is that complexity is related to the fact that applicable requirements are spread across multiple specifications, and it is necessary to give engineers a common point of access to them: a place where they can easily locate all the requirements impacting a particular activity or work product, to ensure that activities are compliant with both corporate and client specific requirements. In other words, to answer competence questions the ontology needs to incorporate all the elements needed to contextualize the requirements in the work processes. These are in turn the result of the three-layer refinement described in the previous section.

The first components of the ontology correspond to those in the "core layer", extracted from the ISO/IEC 29110 international standard. This is a document set, and the ontology elaboration has focused on one of its guides: ISO/IEC 29110-5-1-2, "Software Engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 5-1-2: Management and Engineering Guide - Basic VSE Profile". It corresponds to the Basic Profile. This profile defines processes for small companies and projects not developing critical software. The Basic Profile includes two processes: Software Implementation and Project Management. For each process, the related activities, tasks and work products are defined. This information constitutes the basis of the core layer.

When modelling the ontology, processes, activities, tasks and work products in ISO/IEC 29110 have been modelled as subclasses of the generic Processes, Activities, Tasks and WorkProducts classes. This decision is based on the fact that the framework considers the possibility of having multiple instances of these items tailored to the different projects. It is also needed to keep the consistency with the metamodel defined by SPEM (Software Process Engineering Metamodel), a metamodel used to define process models used by the software industry. For example, the SI.6.6 task "Perform delivery according to Delivery Instructions" within the SI process is modelled as a subclass of the generic Tasks class: the different projects that enact the process following this process model shall instantiate the SI6.6 task within their projects. In addition, when it is necessary to customize or adapt the task definition and characteristics, process architects can create subclasses tailored according to the corporate or customer requirements.

ISO/IEC 29110 provides with the core elements on the ontology, including a process meta-model with the different types of elements the standard deals with: processes, activities, tasks, work products, process objectives, process outcomes and roles. Similar meta-models for software processes have been proposed in the professional and academic literature in software engineering [4, 5, 6]. As previously stated, from this meta-model additional classes can be directly derived from the content of the ISO/IEC 29110 standard. Each process, activity, task, process objective and work product constitute a separate class. ISO/IEC 29110-5-1-2 makes possible to derive additional elements for the target ontology, mainly, the object properties that represent relationships between classes, including the relationships that delineate the se-

quencing and dependencies between tasks and activities. Dependencies between activities are in fact based on the usage that activities make of the work products generated by other activities. In this layer of the ontology, an initial set of relationships are identified, among others:

- includesActivity – isIncludedInProcess, between Processes and Activities.
- includesTask – isIncludedInActivity, between Activities and Tasks.
- usesInput – isUsedAsInput, between Tasks and Work Products.
- generatesOutput – isGeneratedAsInput, between Tasks and Work Products.
- participatesIn – isDoneBy, between Tasks and Roles.
- produces – isProducedBy, between Roles and Work Products.
- dependsOnTask – mustPreceed, between activities, and between tasks, to establish sequencing and order.

To ensure that elements in the ontology are properly documented, for each element the following properties are systematically filled with the definitions in the ISO/IEC 29110 standard.

The second phase in the design of the ontology corresponds to the addition of specializations to reflect the particular tailoring of the standard made by the company. In the case of the ISO/IEC 29110, the description provided for activities, work products and tasks may result quite general, and tailoring may consist of adding additional details, usually in the form of data properties that will be added to subclasses of the classes derived from ISO/IEC 29110. This tailoring can be completed using interviews or questionnaires that identify particular aspects of the process implementation made by the company.

The third phase in the design of the ontology corresponds to the incorporation of the characteristics derived from the project specific requirements imposed by the customer. To complete this exercise, the following heuristics are proposed:

- If the requirement requires the incorporation or the adaptation of a new tasks, role or work product, a new subclass shall be added to the ontology as a subclass of the class being extended or tailored.
- The tailoring of the existing class may consist of incorporating new data – individuals - or object properties to represent a characteristic of the element that was not present in the superclass. The limits of Semantic Web ontology languages to represent default values of data properties are one of the major constraints for the tailoring process.

When doing the tailoring of an existing class it is necessary to maintain the traceability between the change incorporated and the specific requirement that justifies and provides the reasons for the change. This traceability ensures that all the requirements in the applicable specifications are taken into account and covered by the process model that the project will enact. But keeping this traceability using the Semantic Web languages is complex, due to the constraints to represent properties of properties. RDF and OWL focuses on binary relations between individuals or between an individual and a value. The proposed solution for dealing with the traceability is based on

the use of ontology patterns for representing n-ary relations in OWL described in the W3C note "Defining N-ary Relations on the Semantic Web". The pattern represents the relation between more than two items as a class, being individual instances of the related classes instances of the relation. Additional properties provide binary links to each argument of the relation. This corresponds to the use case 1 described in the W3C document: additional attributes describing a relation. This case is used to represent additional attributes describing a relation (the example in the W3C note is a patient having a health problem with a specific probability). In the proposed pattern, an individual is created to represent the relation itself, containing links from the subject of the relation to this instance, and links from this instance to all participants that represent the additional information. Going back to the case under discussion, the individual corresponding to the Tasks "Reporting to the Customer" has a has_periodicity property whose value is another individual (_:Periodicity_Relation_1) that encapsulates the requirement and the standard from which it is taken.

A different approach to model this case is based on incorporating to the ontology the set of applicable specifications and their requirements. This implies having separate classes for specifications and requirements. Particular requirements and documents will be represented as individuals of these classes. This would make possible the creation of relationships between classes representing tasks and work products and requirements, but it would not make possible adding constraints at the property level to indicate that its value is affected by a particular requirement. Regardless these aspects, the ontology supporting the process model must consider requirements from standards, and must provide the necessary classes and instances to handle these objects.

4 Views of the Process-framework: the Ontology's End User Perspective

Different projects demonstrate the value of ontologies as valid tools for knowledge organization in software projects [7, 8]. In the case of the ontology described in this paper, it is aimed for process analysts in charge of modelling and engineers involved in the planning, execution and control of the project. All the staff participating in the project should have a clear knowledge of the processes that shall be enacted during project execution, and should know the impact that customer requirements and applicable standards have on their activities and on the product deliverables they need to produce, verify or validate.

The process-framework organized in the three layers described above support two different views: the first one oriented to the process engineers and quality departments in charge of customizing and tailoring corporate processes, and the second oriented to the engineers working on a particular project. Both views are based on the OWL ontology that serves as the conceptual infrastructure of the framework.

For process engineers, the ontology provides the model and the data that support a complex, corporate information system, whose complexity become greater as additional standards, projects and customer requirements are added to the framework. The

advantage of using an ontology is that the tailoring of the processes may be reused across different projects applying the same standards: process models can be further adapted by adding additional subclasses to the ontology.

For engineers and staff involved in project execution, the ontology-based approach makes complex, related requirements affecting tasks and work products explicit, and put them in the context of the daily activities. Usually, company staff has the necessary knowledge to follow the procedures defined in the corporate quality system or guidelines. In the opposite, when dealing with customer-imposed, project specific requirements and standards, there is a risk of missing some aspects requested by these requirements, as staff is not used to apply them systematically in previous projects. The ontology serves this purpose and reduces or even eliminates this risk.

Staff involved in projects need to be given an application where they can get answers to these two questions: which are the Customer requirements that have an impact on my activities? Is there any Customer requirement I need to take into account during the creation or validation of a specific work product? This application is delivered as a web publication built on top of Semantic Wiki. In this web, engineers can easily find detailed information about each tasks and work products. As the volume of information increases, the framework helps organize information and reduces the complexity implicit to the fact of dealing with additional requirements to those imposed by the organizational work procedures.

For the publication of the resulting process adaptations (each project shall have its own adaptation) Semantic MediaWiki tool has been selected. Experiences with this tool are documented in previous research [9, 10]. The advantage of this tool with respect to other web publishing tools is that it supports searching and browsing to easily identify those requirements having an impact on the activities and work products in the process model. In addition, Wikis have become a popular tool to keep technical documentation and share knowledge between technical staff. Using a tool already known and deployed in a wide range of companies, help ensure the success of the process framework, as users do not need to learn how to use a new tool or keep separate bookmarks. Content edition capabilities provided by Wikis is also a relevant feature, as engineers can suggest additional relationships to those initially identified when modeling the particular adaptations of the corporate process model.

5 Conclusions

Different ontologies have been proposed in the academic and professional literature to model software development processes and to create mappings between standards and methodologies. The case described in this paper demonstrates the feasibility of using software process model ontologies to improve the access to information in complex projects that need to be compliant with requirements stated in different sources.

Starting from a core process model based on ISO/IEC 29110, subsequent adaptations of the standard at the corporate and at the project level are modelled by adding new elements to the core ontology. The core ontology identifies a set of tasks, activi-

ties, work products and roles that may be extended or tailored in response to the requirements imposed by the client.

Dealing with requirements coming from different sources implies a risk, as staff involved in project planning and execution may disregard relevant requirements and information. In addition, dealing with separate documents and standards has a negative impact on productivity, due to the difficulties to remember at a given time all the applicable constraints. The proposed solution ensures a single, shared point of access to search and browse applicable requirements – whatever their source - and assess their impact on activities and work products. When working on a specific deliverable or tasks, engineers can get the list of requirements that need to be considered.

The basis of the proposed solution is an OWL ontology organized in three layers:

- The first layer, Core Layer, corresponding to the process model taken as a reference and based on a well-defined standard. ISO/IEC 29110 has been used in this prototype. This layer provides an initial list of processes, activities, tasks, roles and work products.
- The second layer, Corporate Layer, corresponds to the process definitions obtained after the tailoring of the processes defined in the Core Layer. This tailoring is based on the corporate standards.
- The third layer, Project Layer, corresponds to the definition of the processes done for specific projects, considering the constraints imposed by requirements imposed by the customer. Process definitions at this layer shall be enacted at project initialization.

The ontology incorporates specifications imposing requirements. They may be international standards or customer-provided. These requirements are linked to other ontology classes representing activities, tasks and work products to provide traceability. These links or object relations enable the creation of a web publication where users can easily browse requirements impacting on tasks and work products. To ensure a wider access to the knowledge encoded in the ontology, this is incorporated into a Semantic Wiki that gives engineers the choice of incorporating additional knowledge about the impact of requirements. The semantic wiki support the work of engineers involved in projects as well as the work of process engineers.

The prototype demonstrates the feasibility of using ontologies to represent the knowledge embedded in corporate and project-specific software process models and link them to particular requirements from different sources, building an integrated view of the processes and the constraints that impact project activities and work products, avoiding one of the most relevant challenges that engineers face today: information overload and intellectual control of multiple inputs affecting their activities.

6 Bibliography

1. Boucher, Q. et al.: Towards Configurable ISO/IEC 29110-Compliant Software Development Processes for Very Small Entities. In: Winkler, D., O'Connor, R., Messnarz, R. (eds) Systems, Software and Services Process Improvement, pp. 169-180 (2012)

2. Corcho, O: Building Legal Ontologies with METHONTOLOGY and WebODE. In: Law and the semantic web : legal ontologies, methodologies, legal information retrieval, and applications. Berlin: Springer, pp. 142-157 (2005)
3. Iqbal, R.: An Analysis of Ontology Engineering Methodologies: A Literature Review. Research journal of applied sciences, engineering and technology. 6 (16) 2993-3000 (2013)
4. Alexis, O.: Rationale modeling for software process evolution. Software Process: Improvement and Practice 14(2) 85-105 (2009)
5. Calero, C., Ruiz, F., Piattini, M.: Using Ontologies in Software Engineering and Technology. In: Ontologies for Software Engineering and Software Technology. Berlin : Springer. 49-102 (2006)
6. Uwe A. et al.: Ontology-Guided Software Engineering in the MOST Workbench. In: Ontology-Driven Software Development. Berlin : Springer. pp 293-318 (2013)
7. Villela, K. et al.: The use of an enterprise ontology to support knowledge management in software development environments. Journal of the Brazilian Computer Society (11)2 45-59 (2005)
8. Smite, D. et al.: An empirically based terminology and taxonomy for global software engineering. Empirical Software Engineering. 19 (1) 105-153 (2014)
9. Baumeister, J., Reutelshoefer, J., Puppe, F.: KnowWE: a Semantic Wiki for knowledge engineering. Applied Intelligence. 35(3) 323-344 (2011)
10. Ribaud, V., Saliou, P.: Using a Semantic Wiki for Documentation Management in Very Small Projects. In: Sánchez-Alonso, S., Athanasiadis, I. (eds.) Metadata and Semantic Research. Berlin: Springer , 119-130 (2010)