Project Context Definition Method

Sergio F. Ochoa, Alcides Quispe, Julio Ariel Hurtado, Cecilia Bastarrica

Computer Science Department, University of Chile
{sochoa, aquispe, jhurtado, cecilia}@dcc.uchile.cl

Abstract. In software development it is clear that the projects' context determines which software process is the most appropriate one to support a certain product development. In a previous work the authors have identified the set of dimensions and variables that should be considered in the context specification of small software projects, due they could affect the development process. This report compliments such a previous work by proposing a step-by-step method to define the organizational project context profile that will be used by a software company to tailor its organizational software process. Every time that the company is going to develop a project, it must instantiate of such a profile to reach a computable representation of the context that characterizes to that project. Such a representation will be then used by a set of tailoring transformations to generate an adapted software process, which is suitable to support the development of the product being considered in the project. This method helps to automate the tailoring process, and thus it becomes more simple, fast and predictable. This project context definition method must be then utilized by the end-user through a Web tool that eases this activity.

Keywords. Project context representation, software process tailoring, organizational project context model, project context instantiation, XMI.

1. Introduction

Several researchers have identified the need to specify the project's context in order to determine which software process is the most suitable one to support its developments [Amb11, Cus09, Dor08]. The project's context varies according to different variables such as: product size, project duration, product complexity, team size, application domain knowledge, and familiarity with the involved technology, among others. In a previous work the authors have defined a *Software Project Context Modeling Language* (SPCML), which indicates that the *project context* can be described by using four dimensions (i.e. project, team, product, process) and a set of attributes (i.e. variables) that usually affects the project context definition [Hurt11]. Such a context can be then used as input to tailor the organizational software process as shown in Figure 1. More details of about the tailoring process can be found in [Hurt11b].

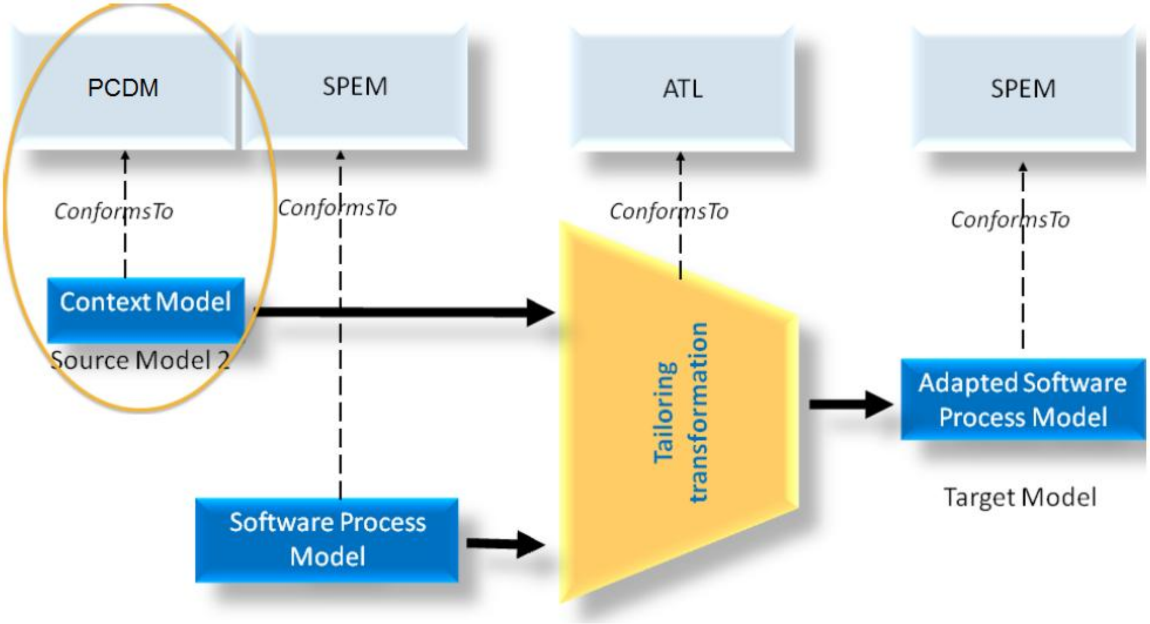


Figure 1. Tailoring process proposed in the ADAPTE project [Adap12, Hurt11b]

The authors have also defined canonical model to represent the software project context, which has been explained in detail in [Hurt11]. Next we present the abstract representation of such model to show the project dimensions and attributes that are considered as part of this context.

Software Development Project Context CanonicalCase

Dimension Project

ProjectType:	{newDevelopment, extension, maintenance}
Duration:	{short, medium, large}
ClientInvolvement:	{high, medium, low, known}
Problem Knowledge:	{clear, ambiguous, unclear}
TimeConstraints:	{veryConstrained, typical, unconstrained}
BudgetConstraints:	{veryConstrained, typical, unconstrained}

Dimension Team

TeamSize:	{veryRestricted, typical, unrestricted}
TeamExpertise:	{high, regular, low}
BussinesKnowledge:	{know, affordable, unknown}
ProductKnowledge:	{know, affordable, unknown}

Dimension Product

TechnicalComplexity:	{high, medium, low}
QualityRelevance:	{high, regular, minimum}

Dimension Process

ProcessFocus:	{finalProduct, everyProduct}

Next section describes the Project Context Definition Method (PCDM), which is proposed to help process engineers and project managers to define both, the organizational project context model and also the specific context instances for a particular project. Then, Section 3 presents the conclusions of this work.

2. The Project Context Definition Method

The set of context variables that is relevant to support the software process adjustments is potentially different from company to company. Therefore these organizations must define their own profile to represent project contexts. This section describes the Project Context Definition Method (PCDM) that helps Small and Medium Enterprise (SME) to perform such activity. Moreover, this section also indicates how to instantiate and validate a particular project context, which will be then used as input for the tailoring process.

The PCDM method involves four phases according to the sequence indicated in Figure 2: (1) Definition of the Organizational Project Context Model, (2) Instantiation of the Project Context, (3) Validation of the Project Context Instantiation, and (4) Generation of the XMI file that will be used to perform the tailoring of the software process.

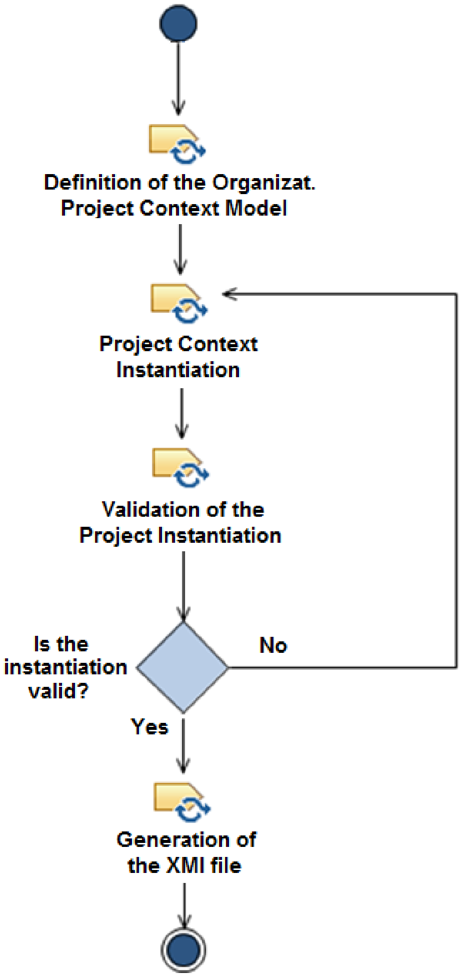


Figure 2. Project Context Definition Method

The definition of the Organizational Project Context Model is an activity similar to the definition of the organizational software process in any software company; i.e. it requires the participation of process engineers, project managers and any other people able to provide/evaluate the context information of the projects performed by the organization. Such activity is done or reviewed once every two or three years; therefore this definition/re-definition becomes affordable by SMEs.

The rest of the activities shown in Fig. 2 (i.e. the last three activities) must be done every time that the organization is going to develop a software project. The project context instantiation and its validation are required to generate the XMI file that describes the project context. Such file is used as input by the tailoring transformation rules to generate the adapted software process (see Fig. 1). The resulting process will then be used by the developers to build the software product. Next we describe each of these phases involved in the Project Context Definition Method (PCDM).

2.1. Definition of the Organizational Project Context Model

The PCDM allows process engineers (or equivalent roles) to determine the context dimensions relevant to perform the tailoring process in a particular organization; i.e. the project context profile for a specific software company. As was mentioned before, the profile considers the dimensions, the attributes and values that can be used for the tailoring process. At this phase the use of historical information is highly recommended.

The definition of this organizational project context model involves three stages as shown in figure 3: (1) context elicitation, (2) context analysis/specification, and (3) context validation. The second stage considers two parallel (but not independent) activities: context analysis and context specification. These activities generate a permanent feedback among them, therefore the resulting project context specification should be completely aligned to the analysis performed on the organizational project contexts. Next we explain each of the activities involved in the definition of this organizational project context model.

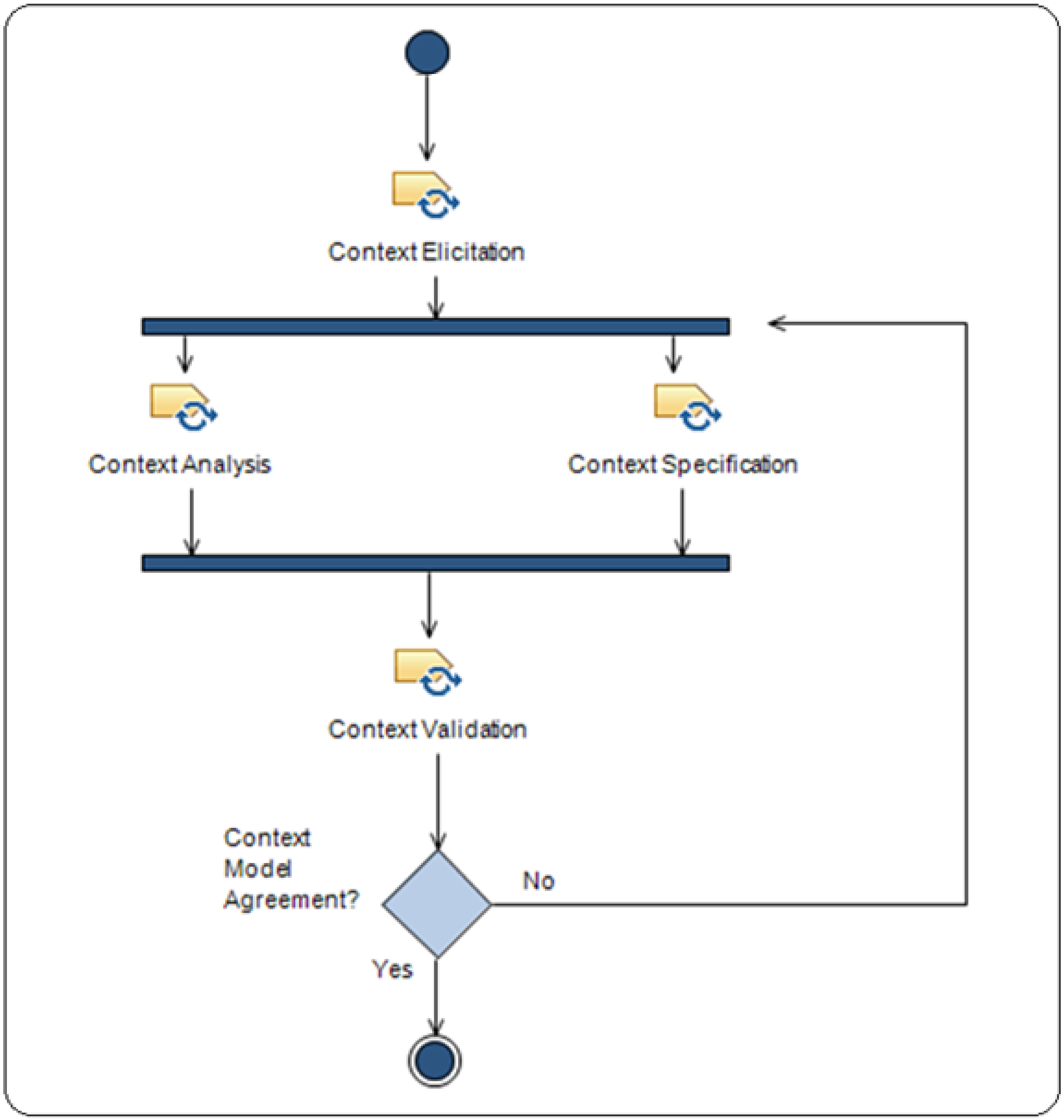


Figure 3. Definition of the Organizational Project Context Model

2.1.1. Context Elicitation

The context elicitation involves the gathering of the information about typical project contexts involved in the previous developments of the organization. This includes identifying the information sources of past projects, experts (or experienced people) in the software process, previous process adaptations, project types, process templates and configurations, and tailoring guidelines (if they exist). Such information must be organized according to the context dimensions shown in the project context canonical model presented in Section 1.

This information gathering process can be done using several types of activities, such as workshop, focus groups, interviews, projects mining, and processes mining. This activity is finished when all common project contexts and most of the optional ones have been recorded and classified properly.

2.1.2. Context Analysis

During this activity the information retrieved and classified in the previous stage is analyzed to determine which dimensions are the most suitable ones for a particular SME. The experience of project managers and process engineers is very useful to make such analysis.

The main goal of this stage is to determine which are the context dimensions, attributes and values that must be considered by a certain SME to perform the tailoring of its organizational software process. This activity should be done using workshops, and it must address the context layers definition following a top-down approach; i.e. defining first the context dimensions, then the context variables and finally the values to be assigned to such variables. It is recommended to use different workshops to analyze each context layer in order to avoid mixing components with different granularity. The output of this activity is a project context structure similar to the project context canonical model, but that considers just the dimensions, attributes and values that are relevant for such organization.

2.1.3. Context Specification

The context specification must formalize the structure defined during the context analysis and provide clear definitions that allow people to perform appropriate project context instantiations. Ambiguity is the main issue to address in this stage; therefore simple examples of the meaning of a certain value for a context attribute can be defined to deal with the ambiguity when such values are used during the project context instantiation. In order to perform the context specification it is recommended to fill a table similar to Table 1.

Table 1. Partial example of an Organizational Project Context Specification

Dimension	Attribute	Value	Examples
Project	ProjectType	newDevelopment	- Projects developed from the scratch. - New systems or subsystems, which are data coupled with other systems.
		Extension	- Development of extra functionalities on products developed by the organization.
		Maintenance	- Adjustments/changes to functionality developed by the organization. - Integration of two or more modules developed by the organization.
	Duration	Short	- Minor to 800 man-month. - Up to 1 month of development.
		Medium	- Between 800 and 2000 man-month. - Between 2 and 4 months of development.

		Large	- Between 2000 and 3500 man-month. - Between 4 and 6 months of development.
	Problem Knowledge	Ambiguous	- The client has an idea about the product to be developed; however the problem to address and its context must be clarified.
	…	…	…

In order to build this table we can use an approach similar to the one defined in the Wideband-Delphi estimation method [Boe81]; i.e. during a meeting the participants (all experienced people) must define individually what they mean by (e.g.) a short project. The individual definitions are then presented and discussed until a consensus is reached among the participants. After that, the specification team passes to discuss the next value for a context attribute. Applying this technique it is possible to reach a set of unambiguous definitions that will be then very useful to perform the tailoring of the organizational software process.

2.1.4. Context Validation

During the context validation, the definition and meaning assigned to the values of the context attributes is checked and adjusted. Depending on the validation results, a new round of context analysis and context specification can be performed.

The validation activity can be done through workshops where the participants must specify the context for several hypothetical projects. In this exercise past projects can also be used. If the evaluation performed by every participant is the same, then we can assume the values for the context attributes were unambiguous. In other case the conflicting scores must be analyzed to identify the cause of such mismatches. If the cause was a difference in the meaning assumed for a certain context value, then such value must be clarified or redefined to remove the ambiguity.

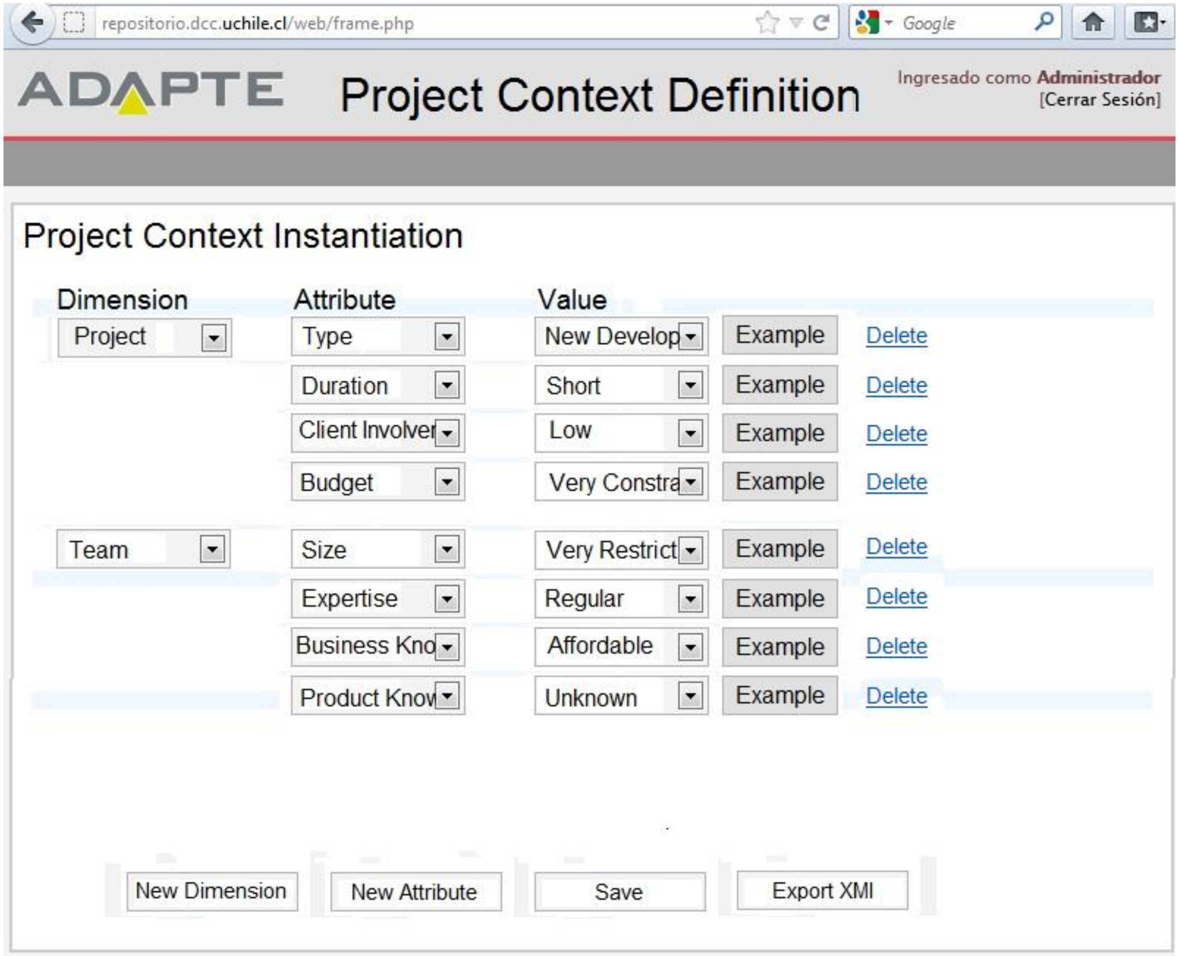
2.2. Instantiation of the Project Context

As mentioned before, every time that an organization is going to develop a software project, it must define the context that characterizes such a project. Thus, it will be possible to perform the tailoring of the software process.

The context characterization involves the project context instantiation and validation, and the generation of the XMI file that specifies the features of the project. These last two activities are explained in sections 2.3 and 2.4 respectively.

The project context instantiation involves just the process that assigns values to each context attribute based on the context of a particular project. It uses the dimensions, attributes and values defined in the Organizational Project Context Specification. It is recommended that this activity is performed using a tool with a user interface, avoiding thus that end-users have to deal with configuration files or low-level information about the project context.

Figure 4 shows the initial design of the interface used to both, define the context profile for the projects of an organization, and also to perform the project contexts instantiation. When the users are defining the context profile of the organization, they can add or remove context dimensions and attributes. However during the context instantiation process the users can utilize just the project context profile (i.e. the context model of the organization) to assign values to the context attributes as shown in Fig. 4. Due the instantiation process can be done asynchronously, the system must allow the user to save the work until the user decides to complete it.



Project Context Definition

Ingresado como Administrador
[Cerrar Sesión]

Project Context Instantiation

Dimension	Attribute	Value
Project	Type	New Develop
	Duration	Short
	Client Involver	Low
	Budget	Very Constra
Team	Size	Very Restrict
	Expertise	Regular
	Business Kno	Affordable
	Product Know	Unknown

Figure 4. User Interface for Project Context Instantiation

2.3. Validation of the Project Context Instantiation

The validation process requires the participation of one or more people that had not participated in the project context instantiation process. These people are going to act as quality engineers by ensuring the context definition represents accurately the project features; therefore it can be used to tailor the software process.

The process is quite simple; for each context attribute the quality engineers must indicate if they agree (or not) with the value assigned by the person in charge of the context instantiation for a project. By clicking on the button with the label “Example” the users can access to examples of typical projects having the same duple `<attribute><value>`.

If the quality engineers agree with all values assigned to the project attributes, then the context instantiation can be considered as valid. Once validated the instantiation process, it is possible to generate the XMI file that represents the context of a specific project.

2.4. Generation of the XMI file

This process is automatic and invisible for end-users. The process reads the information stored in the project context profile, which is usually recorded in a set of database tables, and generates a text file with particular tags. Such tags correspond to the elements participating in the Project Context Canonical Model shown in Section 1. The Annex A shows an example of an XMI file representing the context of a particular project.

In order to generate the XMI file the users have just to click the button with the label “Export XMI” (see Fig. 4) and then save the generated file. Such file is understandable for the tailoring transformation rules that are going to perform the adaptation of the organizational software process, as shown in Fig. 1.

3. Conclusions

This document describes the Project Context Definition Method (PCDM) that was designed to help project managers and process engineers of small and medium software companies (SME) to specify both, (1) the Organizational Project Context Model; i.e. the project context profile for an organization, and (2) each particular project context. A step-by-step explanation of the PCDM phases is also included in this document.

The process is simple and systematic, which ensures that any SME can use it even involving few qualified personnel. This method adheres to the Context Modeling Language defined in [Hurt11], and its use will be supported by a software tool that is currently under development. Such tool will become this method easier to utilize for the end-users. We hope that this method helps SMEs to perform a more accurate, simple and fast project context definitions, and thus to transport those benefits to their software process tailoring.

Acknowledgements

This work has been partly funded by Fondef project (Conicyt, Chile), grant D09I1171.

References

[Adap12] ADAPTE Project. Adaptable Domain And Process Technology Engineering. FONDEF grant D09I1171. URL: www.adapte.cl. Last visit: March 2012.

[Amb11] O. Armbrust, H.D. Rombach. The Right Process for Each Context: Objective Evidence Needed. In (Eds.) Raffo, Pfahl, and Zhang. Proceedings of the

International Conference on Software and Systems Process (ICSSP'11), pages 237–241. Honolulu, HI, USA, May 21-22, 2011.

[Boe81] B. Boehm. Software Engineering Economics. Prentice Hall, 1981.

[Cus09] M.A. Cusumano, A. MacCormack, C.F. Kemerer, W. Crandall. Critical Decisions in Software Development: Updating the State of the Practice. IEEE Software, 26(5):84–87, 2009.

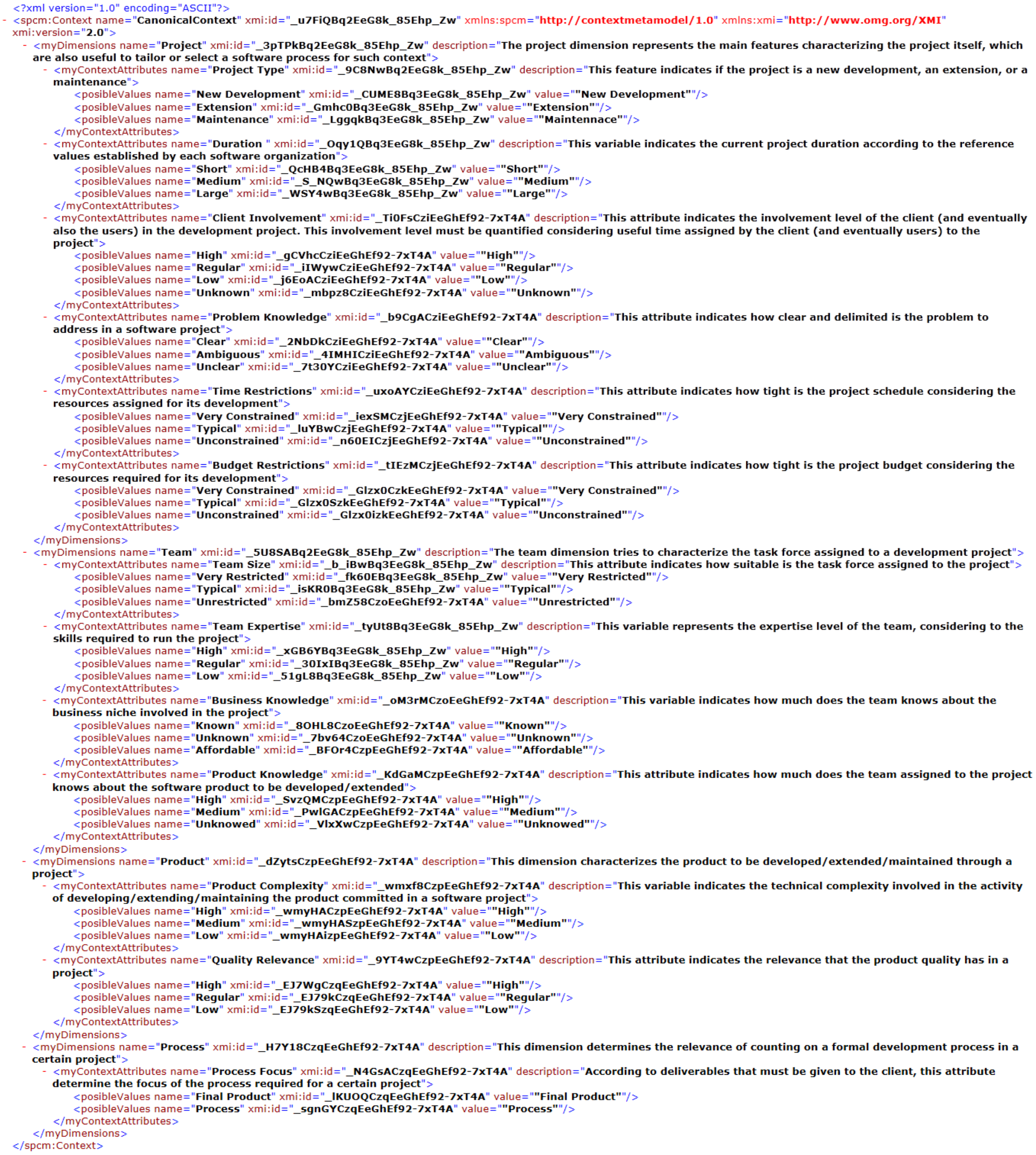
[Dorr08] J. Dorr, S. Adam, M. Eisenbarth, M. Ehresmann. Implementing Requirements Engineering Processes: Using Cooperative Self-Assessment and Improvement. IEEE Software, 25(3):71–77, 2008.

[Hurt11] J.A. Hurtado, S.F. Ochoa, A. Quispe, C. Bastarrica. A Context Modeling Language to Support Tailoring of Software Processes. Technical Report TR/DCC-2011-14. Computer Science Department, University of Chile. December 23, 2011. Available at: http://www.dcc.uchile.cl/TR/2011/TR_DCC-20111223-014.pdf

[Hurt11b] J.A. Hurtado, C. Bastarrica, A. Quispe, S.F. Ochoa. An MDE Approach to Software Process Tailoring. Proc. of International Conference on Software and system Processes (ICSSP'11), co-located with ICSE 2011. Honolulu, Hawaii, USA. May 21-22, 2011.

Annex A. Example of an XMI representation of a software project

Next XMI code identifies project dimensions, attributes belonging to such dimensions and a set of values that can be assigned to the attributes.



```
   <?xml version="1.0" encoding="ASCII"?>
 - <spcm:Context name="CanonicalContext" xmi:id="_u7FiQBq2EeG8k_85Ehp_Zw" xmlns:spcm="http://contextmetamodel/1.0" xmlns:xmi="http://www.omg.org/XMI"
   xmi:version="2.0">
   - <myDimensions name="Project" xmi:id="_3pTPkBq2EeG8k_85Ehp_Zw" description="The project dimension represents the main features characterizing the project itself, which
     are also useful to tailor or select a software process for such context">
     - <myContextAttributes name="Project Type" xmi:id="_9C8NwBq2EeG8k_85Ehp_Zw" description="This feature indicates if the project is a new development, an extension, or a
       maintenance">
         <posibleValues name="New Development" xmi:id="_CUME8Bq3EeG8k_85Ehp_Zw" value=""New Development""/>
         <posibleValues name="Extension" xmi:id="_Gmhc0Bq3EeG8k_85Ehp_Zw" value=""Extension""/>
         <posibleValues name="Maintenance" xmi:id="_LggqkBq3EeG8k_85Ehp_Zw" value=""Maintennace""/>
       </myContextAttributes>
     - <myContextAttributes name="Duration " xmi:id="_Oqy1QBq3EeG8k_85Ehp_Zw" description="This variable indicates the current project duration according to the reference
       values established by each software organization">
         <posibleValues name="Short" xmi:id="_QcHB4Bq3EeG8k_85Ehp_Zw" value=""Short""/>
         <posibleValues name="Medium" xmi:id="_S_NQwBq3EeG8k_85Ehp_Zw" value=""Medium""/>
         <posibleValues name="Large" xmi:id="_WSY4wBq3EeG8k_85Ehp_Zw" value=""Large""/>
       </myContextAttributes>
     - <myContextAttributes name="Client Involvement" xmi:id="_Ti0FsCziEeGhEf92-7xT4A" description="This attribute indicates the involvement level of the client (and eventually
       also the users) in the development project. This involvement level must be quantified considering useful time assigned by the client (and eventually users) to the
       project">
         <posibleValues name="High" xmi:id="_gCVhcCziEeGhEf92-7xT4A" value=""High""/>
         <posibleValues name="Regular" xmi:id="_iIWywCziEeGhEf92-7xT4A" value=""Regular""/>
         <posibleValues name="Low" xmi:id="_j6EoACziEeGhEf92-7xT4A" value=""Low""/>
         <posibleValues name="Unknown" xmi:id="_mbpz8CziEeGhEf92-7xT4A" value=""Unknown""/>
       </myContextAttributes>
     - <myContextAttributes name="Problem Knowledge" xmi:id="_b9CgACziEeGhEf92-7xT4A" description="This attribute indicates how clear and delimited is the problem to
       address in a software project">
         <posibleValues name="Clear" xmi:id="_2NbDkCziEeGhEf92-7xT4A" value=""Clear""/>
         <posibleValues name="Ambiguous" xmi:id="_4IMHICziEeGhEf92-7xT4A" value=""Ambiguous""/>
         <posibleValues name="Unclear" xmi:id="_7t30YCziEeGhEf92-7xT4A" value=""Unclear""/>
       </myContextAttributes>
     - <myContextAttributes name="Time Restrictions" xmi:id="_uxoAYCziEeGhEf92-7xT4A" description="This attribute indicates how tight is the project schedule considering the
       resources assigned for its development">
         <posibleValues name="Very Constrained" xmi:id="_iexSMCzjEeGhEf92-7xT4A" value=""Very Constrained""/>
         <posibleValues name="Typical" xmi:id="_luYBwCzjEeGhEf92-7xT4A" value=""Typical""/>
         <posibleValues name="Unconstrained" xmi:id="_n60EICzjEeGhEf92-7xT4A" value=""Unconstrained""/>
       </myContextAttributes>
     - <myContextAttributes name="Budget Restrictions" xmi:id="_tIEzMCzjEeGhEf92-7xT4A" description="This attribute indicates how tight is the project budget considering the
       resources required for its development">
         <posibleValues name="Very Constrained" xmi:id="_Glzx0CzkEeGhEf92-7xT4A" value=""Very Constrained""/>
         <posibleValues name="Typical" xmi:id="_Glzx0SzkEeGhEf92-7xT4A" value=""Typical""/>
         <posibleValues name="Unconstrained" xmi:id="_Glzx0izkEeGhEf92-7xT4A" value=""Unconstrained""/>
       </myContextAttributes>
     </myDimensions>
   - <myDimensions name="Team" xmi:id="_5U8SABq2EeG8k_85Ehp_Zw" description="The team dimension tries to characterize the task force assigned to a development project">
     - <myContextAttributes name="Team Size" xmi:id="_b_iBwBq3EeG8k_85Ehp_Zw" description="This attribute indicates how suitable is the task force assigned to the project">
         <posibleValues name="Very Restricted" xmi:id="_fk60EBq3EeG8k_85Ehp_Zw" value=""Very Restricted""/>
         <posibleValues name="Typical" xmi:id="_isKR0Bq3EeG8k_85Ehp_Zw" value=""Typical""/>
         <posibleValues name="Unrestricted" xmi:id="_bmZ58CzoEeGhEf92-7xT4A" value=""Unrestricted""/>
       </myContextAttributes>
     - <myContextAttributes name="Team Expertise" xmi:id="_tyUt8Bq3EeG8k_85Ehp_Zw" description="This variable represents the expertise level of the team, considering to the
       skills required to run the project">
         <posibleValues name="High" xmi:id="_xGB6YBq3EeG8k_85Ehp_Zw" value=""High""/>
         <posibleValues name="Regular" xmi:id="_30IxIBq3EeG8k_85Ehp_Zw" value=""Regular""/>
         <posibleValues name="Low" xmi:id="_51gL8Bq3EeG8k_85Ehp_Zw" value=""Low""/>
       </myContextAttributes>
     - <myContextAttributes name="Business Knowledge" xmi:id="_oM3rMCzoEeGhEf92-7xT4A" description="This variable indicates how much does the team knows about the
       business niche involved in the project">
         <posibleValues name="Known" xmi:id="_8OHL8CzoEeGhEf92-7xT4A" value=""Known""/>
         <posibleValues name="Unknown" xmi:id="_7bv64CzoEeGhEf92-7xT4A" value=""Unknown""/>
         <posibleValues name="Affordable" xmi:id="_BFOr4CzpEeGhEf92-7xT4A" value=""Affordable""/>
       </myContextAttributes>
     - <myContextAttributes name="Product Knowledge" xmi:id="_KdGaMCzpEeGhEf92-7xT4A" description="This attribute indicates how much does the team assigned to the project
       knows about the software product to be developed/extended">
         <posibleValues name="High" xmi:id="_SvzQMCzpEeGhEf92-7xT4A" value=""High""/>
         <posibleValues name="Medium" xmi:id="_PwlGACzpEeGhEf92-7xT4A" value=""Medium""/>
         <posibleValues name="Unknowed" xmi:id="_VlxXwCzpEeGhEf92-7xT4A" value=""Unknowed""/>
       </myContextAttributes>
     </myDimensions>
   - <myDimensions name="Product" xmi:id="_dZytsCzpEeGhEf92-7xT4A" description="This dimension characterizes the product to be developed/extended/maintained through a
     project">
     - <myContextAttributes name="Product Complexity" xmi:id="_wmxf8CzpEeGhEf92-7xT4A" description="This variable indicates the technical complexity involved in the activity
       of developing/extending/maintaining the product committed in a software project">
         <posibleValues name="High" xmi:id="_wmyHACzpEeGhEf92-7xT4A" value=""High""/>
         <posibleValues name="Medium" xmi:id="_wmyHASzpEeGhEf92-7xT4A" value=""Medium""/>
         <posibleValues name="Low" xmi:id="_wmyHAizpEeGhEf92-7xT4A" value=""Low""/>
       </myContextAttributes>
     - <myContextAttributes name="Quality Relevance" xmi:id="_9YT4wCzpEeGhEf92-7xT4A" description="This attribute indicates the relevance that the product quality has in a
       project">
         <posibleValues name="High" xmi:id="_EJ7WgCzqEeGhEf92-7xT4A" value=""High""/>
         <posibleValues name="Regular" xmi:id="_EJ79kCzqEeGhEf92-7xT4A" value=""Regular""/>
         <posibleValues name="Low" xmi:id="_EJ79kSzqEeGhEf92-7xT4A" value=""Low""/>
       </myContextAttributes>
     </myDimensions>
   - <myDimensions name="Process" xmi:id="_H7Y18CzqEeGhEf92-7xT4A" description="This dimension determines the relevance of counting on a formal development process in a
     certain project">
     - <myContextAttributes name="Process Focus" xmi:id="_N4GsACzqEeGhEf92-7xT4A" description="According to deliverables that must be given to the client, this attribute
       determine the focus of the process required for a certain project">
         <posibleValues name="Final Product" xmi:id="_lKUOQCzqEeGhEf92-7xT4A" value=""Final Product""/>
         <posibleValues name="Process" xmi:id="_sgnGYCzqEeGhEf92-7xT4A" value=""Process""/>
       </myContextAttributes>
     </myDimensions>
   </spcm:Context>
```