Using Contextual Goal Models for Constructing Situational Methods

Jolita Ralyté[1(✉)] and Xavier Franch[2]

[1] University of Geneva, Geneva, Switzerland
jolita.ralyte@unige.ch
[2] Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
franch@essi.upc.edu

Abstract. Situation and intention are two fundamental notions in situational method engineering (SME). They are used to assess the context of an ISD project and to specify method requirements in this context. They also allow defining the goals of the method chunks and the conditions under which they can be applied. In this way, the selection and assembly of method chunks for a particular ISD project is driven by matching situational method requirements to method chunks' goals and context descriptions. In this paper we propose the use of contextual goal models for supporting all SME steps. Our approach is based on iStar2.0 modeling language that we extend with contextual annotations.

Keywords: Situational Method Engineering · Contextual goal model
iStar2.0

1 Introduction

The mission of situational method engineering (SME) [1] consists in providing concepts and guidance for building situation-specific (i.e., situational) methods taking into account the particular context and requirements of the information systems development (ISD) project at hand. SME is founded on modularity, reusability and flexibility principles [2]: method knowledge is formalized in terms of method chunks, characterized by a set of attributes representing their reuse conditions, and stored in a method repository. Method chunks can be reused in many different method constructions. The quality of a situational method heavily depends on how well the obtained method fits the situation [3]. Arguably, selecting the right method chunks is the most challenging task in SME. It requires understanding not only functional method requirements but also the contextual ones. Functional requirements express how the method is expected to support different system engineering activities of the ISD project [4], while the contextual ones reflect the situation of the project and define the conditions in which the method will be used [5]. The SME literature exposes several different ways to deal with functional and contextual aspects in SME; some of them are discussed in the following section. Still, there is space for innovation. In this paper we propose to represent these two aspects in the same model called contextual goal model. We use these models to express situational method requirements as well as to specify method chunks. Furthermore, we introduce a systematic goal modeling in all steps of SME.

J. C. Trujillo et al. (Eds.): ER 2018, LNCS 11157, pp. 440–448, 2018.
https://doi.org/10.1007/978-3-030-00847-5_31

In particular, we use the iStar2.0 [6] goal modeling language that we extend with contextual elements. This type of models allows representing system engineering goals and annotating them with context criteria. To summarize, the research objective of our work is to exploit the use of contextual goal models in SME.

2 Situational and Intentional Aspects in SME

A detailed overview of the state of the art in SME can be found in [1]. In this section, we briefly present the situational and intentional aspects of SME, and the iStar2.0 goal modeling approach, which is fundamental in our contribution.

The notion of situation is often described by using a set of predefined contingency factors or context criteria that can be used at the level of a particular project, a process in the organization or the whole organization [5, 7–9]. The situation of a project is defined by giving values to a set of the most pertinent criteria, e.g. user involvement = low, time pressure = high, delivery strategy = incremental, etc. In addition, the same criteria are also used for characterising the suitability of method chunks to different situations [5].

The notion of intention or goal is mainly used to define the purpose of a process-driven method component and in particular, method chunk [5]. It allows to formalize the system engineering goal to be achieved by applying this method chunk. Goals are also used to express situational method requirements that are then matched to the method chunks' goals to find the best fitting method chunks. The goal-driven process modeling formalism Map [10] constitutes the foundation of the assembly-based [5] and the method family [11] approaches. It allows expressing methods in terms of intentions and strategies to reach the intentions. However, it has no constructs for relating context criteria to the method intentions and strategies.

Recently, iStar2.0 [6] goal models have been used in SME, mainly to formalize the content of method chunks [12] as reusable goal models. In [9] we have introduced the idea of using contextual iStar2.0 for specifying method requirements. Here we explore their usage in all steps of the assembly-based SME process.

iStar2.0 [6] is a standardized kernel goal modeling language dedicated to represent functional and quality requirements, their dependencies and their refinement. The language includes four *intentional elements*, namely *goals*, *qualities*, *tasks* and *resources*, and the notion of *actor*. Actors are autonomous entities that aim at achieving their goals in collaboration with other actors. They can be subtyped as physical agents or abstract roles and they can be related through specialization (*is-a*) or a general link called *participates-in*. Their collaboration is materialized as dependencies through intentional elements: an actor (depender) is dependent on another actor (dependee) for achieving her goal, satisfying a quality, realizing a task or obtaining a resource. All intentional elements of an actor, their refinement and interrelationships are clustered into the *actor's boundary*. View can be defined over iStar2.0 models, and in this paper we will use the Strategic Dependency (SD) and Strategic Rationale (SR) model coming from the original *i** [13]: SD models only show actors and dependencies, while SR models show the internal structure of actors.

3 SME with Contextual Goal Models

A complete SME approach has two phases [14]: (1) the construction of the contextually situated method chunks for reuse and (2) the construction of situation-specific methods by reusing and combining method chunks from the repository. Due to the space limit, we focus in this paper on the later and only briefly introduce the former.

Most of the SME approaches use the notion of goal to define method chunks and specify new method requirements. However, any of them use contextual goal models to do that. Here, we explore this idea by extending the *method chunk-based* SME approach [5] with contextual goal models, which are iStar2.0 goal models enriched with the contextual information. By relating method goals with context criteria we express not only functional but also contextual method requirements that are then used to select and assemble the most appropriate method chunks. Figure 1 depicts the process of situation method construction where contextual goal modelling is used as an underpinning technique at each of its steps.

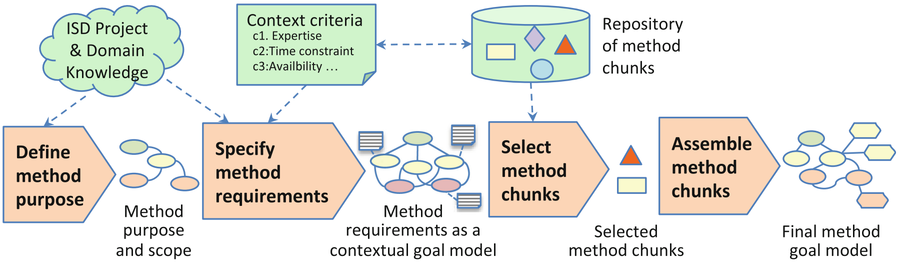

Fig. 1. The process of situational method construction with contextual goal models

3.1 Method Knowledge for Reuse

Because the selection of method chunks consists in matching method requirements with the method chunks' goals and contextual characteristics, the definition of method chunks should also be represented with contextual goal models. Which motivates us to extend the specification of method chunks [5, 14] with contextual goal models reflecting their purpose and contextual fitness. As mentioned above, we use iStar2.0 to create these goal models. The method engineer starts with developing the SD goal model of the method and the context criteria that are pertinent for defining the conditions for method application. Criteria and their values can be defined from scratch based on the related literature review and/or the method engineers' experience, or by refining some existing generic set of context criteria [5, 7, 8]. Next, for each identified method chunk the method engineer develops the SR goal model and extends it with contextual information. In order to integrate the intentional and contextual aspects of method chunks in the same model, we represent the context criteria together with their values as iStar2.0 elements decorated with context information. This representation is an extension of the standard, as introduced in [9].

3.2 Situational Method Construction by Reuse

Defining the Purpose of a Situation-Specific Method. The purpose of a situational method is closely related to the objective of the project it has to serve. For example, if the project aims at developing a web service, a situation-specific method can be required to cover the entire development process or just a part of it (e.g. service design). Therefore, defining the purpose of the method also means delimiting its scope and thus, restricting the elicitation of method requirements and the selection of method chunks. Indeed, the goal of each selected method chunk has to fit in the scope of the method, which means to be a sub-goal of the method goal.

We recommend starting with the SD goal model representing the project as a socio-technical system, where stakeholders, their roles and expectations are revealed. This model provides the context for defining the purpose of the required method, i.e. creating a goal model representing how the intended method is expected to support the achievement of different project actors' goals, resources and tasks and to deliver the expected qualities. Therefore, the intended method has to be modeled as one or several actors each of them reflecting a well-defined and autonomous part of the method (e.g. a particular platform or toolkit, an ISD process step). The different participants of the project are also represented as actors in the goal model. They depend on the method to satisfy their goals, furnish resources, and realize tasks with a given quality. On the other hand, the success of the method usually demands some expectations from the surrounding actors, therefore the actors may also act as *dependees*. The goal model is limited to the SD representation; actors' boundaries (the SR models) are not developed at this step of the approach.

Specifying Situational Method Requirements. The aim of this step is to express the requirements of the project in terms of contextual goal models. It includes two steps: (1) modeling the organizational context of the project as SR models of the actors defined in the method purpose model, and then (2) assessing the project-specific contextual conditions that should be taken into account when constructing the intended method, and in particular, when selecting the method chunks from the method repository. These conditions are formalized in terms of *context criteria,* each of them having a set of predefined values. The criteria and their values are part of the reusable method knowledge as explained in Sect. 3.1. Some criteria are uni-valued (e.g. Time pressure = Low/Medium/High), while others can be multi-valued (e.g. People per requirements elicitation session = {Individual, Group, Mass}. The assessed criteria are then added to the goal models extending them with contextual information. The obtained models express functional and contextual requirements of the project at hand.

Selecting and Assembling the Method Chunks. In this crucial step, an iterative matching process among the organizational context and the repository of method chunks is conducted by the method engineer. During this process, the models grows by: (1) recording the relevant decisions that are necessary to make in order to select the method chunk(s), (2) once selected, reflecting in the model the consequences of adopting such chunk(s).

4 Example: Requirements Elicitation Method

To illustrate our approach we propose an example of situational method construction, namely "Requirements Elicitation Method". Requirements elicitation is one of the key activities in the requirements engineering process [15] for which many different techniques exist. The example will in fact call for the selection of two different method chunks proposing such techniques for two different groups of stakeholders. In this way, we aim to demonstrate the usage of contextual goal models in method requirements specification and how these requirements depend on the contextual conditions.

4.1 Reusable Method Knowledge

Figure 2 shows the iStar2.0 model that specifies the goals of the *Requirements Elicitation Method*. This model acts as a template introducing the main elements that the forthcoming chunks need to refine. Requirements elicitation involves four actors with some mutual dependencies. The *Requirements Engineer* is in charge of delivering a *Requirements Document* to the customer *Organization* by *Conduct*ing the *Elicitation* using a *Requirements Elicitation Method*. The *Stakeholder* provides his/her *Needs* as requested by the method. These needs will be properly elicited only if the *Elicitation* is *Well-fit* and the requirements engineer commits to have the *Method Steps Followed*.

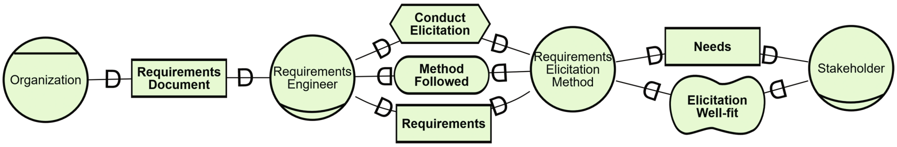

Fig. 2. The generic purpose model of the Requirements Elicitation Method

We take the systematic review and analysis of requirement elicitation techniques conducted by Carrizo et al. [16] as a source for defying the method chunks and the associated context criteria. In this paper, up to 15 different elicitation techniques are compiled and analyzed under the lenses of the context criteria. The criteria are defined in terms of four factors (Elicitor, Informant, Problem Domain, and Elicitation Process) each of them specified with a set of attributes. The fitness of method chunks to different situations is determined by assessing values of the relevant context criteria for each method chunk. In Table 1 we present an excerpt of this analysis for 6 classical elicitation techniques (considered as method chunks) and the set of selected criteria. As a result of this analysis and the representation of the criteria as iStar2.0 elements, we can build the contextual models that define the particular method chunks. Figure 3 shows an excerpt of the model for the *Elicitation by Interviews* method chunk, which is declared as a specialization of *Requirements Elicitation Method* (not shown in the figure for clarity of the drawing).

Table 1. The adequacy of requirements elicitation techniques to the selected context criteria, from [16]. Values: (+) recommended, (–) indifferent, (x) not recommended.

Context criteria		Values	Requirements elicitation techniques					
			Interviews	Questionnaire	Observation	Prototyping	Brainstorming	Scenarios
Elicitor	Elicitation experience (c_2)	High	+	+	+	+	+	+
		Medium	+	+	+	+	–	+
		Low	–	–	+	–	–	–
Informant	People per session (c_5)	Individual	+	+	+	+	x	+
		Group	–	+	+	–	+	–
		Mass	x	+	–	–	–	–
	Consensus (c_6)	High	+	+	–	+	–	+
		Low	x	+	–	+	–	–
	Availability (c_{10})	High	+	+	+	+	+	+
		Low	–	+	+	–	x	–
	Location/accessibility (c_{11})	Near	+	+	+	+	+	+
		Far	–	+	–	x	x	x
Problem Domain	Type of information to be elicited (c_{12})	Strategic	+	+	–	–	+	–
		Tactical	+	+	+	+	+	+
		Basic	–	+	–	+	x	–
Elicitation process	Project time constraint (c_{15})	High	–	+	x	x	x	+
		Medium	+	+	+	+	–	+
		Low	+	+	–	–	+	+
	Process time (c_{16})	Start	+	–	+	+	+	–
		Middle	+	+	–	+	–	+
		End	+	+	x	x	x	–

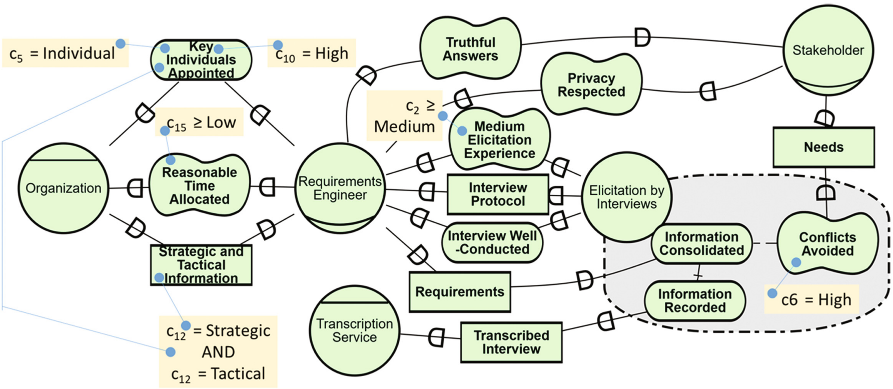

Fig. 3. The contextual goal model of the method chunk *Elicitation by Interviews*

4.2 Constructing a Situation-Specific Method

To illustrate the construction of the *Requirement Elicitation Method* in a specific setting we take the case of the University of Geneva (UniGe) and its ISD project aiming to develop a UniGe mobile application. The application should allow quickly accessing different UniGe information services, like announcement of events, school calendar, course program, access maps to different campuses, etc. The UniGe ISD department

(*UniGe ISDD*) is responsible for developing the application and needs a method for conducting requirements elicitation. *Student* and *Administrator* are two types of *Stakeholders* from whom the needs will be elicited. For the sake of brevity, the SD model of the socio-technical system of the project is not shown here. Figure 4 directly shows the SD model representing the purpose of the required project method.

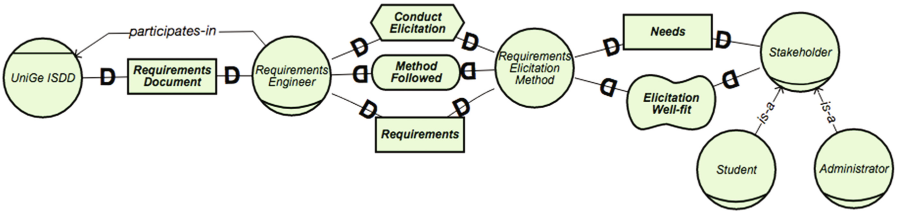

Fig. 4. The purpose of the project method

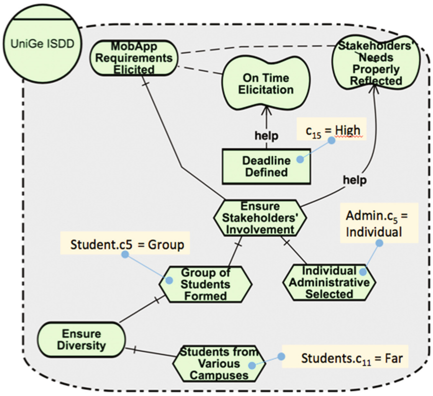
UniGe ISDD
MobApp Requirements Elicited
On Time Elicitation
Stakeholders' Needs Properly Reflected
help
c15 = High
Deadline Defined
help
Ensure Stakeholders' Involvement
Admin.c5 = Individual
Student.c5 = Group
Group of Students Formed
Individual Administrative Selected
Ensure Diversity
Students from Various Campuses
Students.c11 = Far

Fig. 5. Specification of method requirements in the organizational context of the project (SR diagram)

Figure 5 shows the needs of the project for the *Requirements Elicitation Method*. The main goal of *MobApp Requirements Elicited* is reinforced by two important qualities: to ensure an *On Time Elicitation* and *Stakeholders' Needs* are *Properly Reflected*. Remarkably, the university has appointed a critical task for supporting these goals: *Ensure Stakeholders' Involvement*. In the case of students, *Groups of Students* will be *Formed*, whereas in the case of administration, *Individual Administrative* people will be *Selected*. The contextual requirements for the *method* are expressed by assessing the relevant context criteria.

Given the existence of two different types of stakeholders, selecting more than one method chunk is perfectly feasible and in fact, the most likely situation. Indeed, for eliciting the requirements from students, and considering the criteria (c_5 and c_{11}), the *Questionnaires* technique seems to be the most appropriate. The case of administrative staff as stakeholders is a bit more complex. Three techniques (*Observation*, *Prototyping* and *Brainstorming*) can be discarded due to their high time constraints on the project time (c_{15}). However, it is not clear which of the others (*Interviews*, *Questionnaires* or *Scenarios*) could be applied. Therefore, the role of the method engineer as facilitator becomes crucial. Supposing that *Elicitation by Interviews* technique is the most suitable one, the two selected method chunks are then assembled into the project-specific method and applied by the project *Requirements Engineer*.

5 Conclusion

In this paper we explore contextual goal modeling as a way to deal with intentional and contextual aspects in situational method engineering (SME). In particular, we revise the method chunk-driven SME approach by introducing contextual goal models in both SME levels: (1) specification of method knowledge for reuse and (2) situation-specific method construction by reuse. For that we use iSar2.0 goal modeling language. But because this language does not include context elements, we introduce decorations in goal models to represent contextual information.

This work is a result of the OpenReq project, funded by the European Union's H2020 research and innovation programme under the grant agreement No. 732463.

References

1. Henderson-Sellers, B., Ralyté, J., Ågerfalk, P.J., Rossi, M.: Situational Method Engineering. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-41467-1
2. Rolland, C.: Method engineering: towards methods as services. Softw. Process Improve. Pract **14**, 143–164 (2009)
3. McBride, T., Henderson-Sellers, B.: A method assessment framework. In: Ralyté, J., Mirbel, I., Deneckère, R. (eds.) ME 2011. IAICT, vol. 351, pp. 64–76. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19997-4_7
4. Ralyté, J.: Requirements definition for the situational method engineering. In: Rolland, C., Brinkkemper, S., Saeki, M. (eds.) Engineering Information Systems in the Internet Context. ITIFIP, vol. 103, pp. 127–152. Springer, Boston (2002). https://doi.org/10.1007/978-0-387-35614-3_9
5. Mirbel, I., Ralyté, J.: Situational method engineering: combining assembly-based and roadmap-driven approaches. Requir. Eng. J. **11**(1), 58–78 (2006)
6. Dalpiaz, F., Franch, X., Horkoff, J.: iStar 2.0 Language Guide. https://arxiv.org/abs/1605.07767
7. van Slooten, K., Hodes, B.: Characterizing IS development projects. In: Brinkkemper, S., Lyytinen, K., Welke, R.J. (eds.) Method Engineering. ITIFIP, pp. 29–44. Springer, Boston (1996). https://doi.org/10.1007/978-0-387-35080-6_3
8. Kornyshova, E., Deneckère, R., Claudepierre, B.: Contextualization of method components. In: RCIS 2010, pp. 235–246. IEEE Computer Society Press (2010)
9. Franch, X., et al.: A situational approach for the definition and tailoring of a data-driven software evolution method. In: Krogstie, J., Reijers, Hajo A. (eds.) CAiSE 2018. LNCS, vol. 10816, pp. 603–618. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91563-0_37
10. Rolland, C., Prakash, N., Benjamen, A.: A multi-model view of process modelling. Requir. Eng. J. **4**(4), 169–187 (1999)
11. Kornyshova, E., Deneckère, R., Rolland, C.: Method families concept: application to decision-making methods. In: Halpin, T., et al. (eds.) BPMDS/EMMSAD-2011. LNBIP, vol. 81, pp. 413–427. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21759-3_30
12. López, L., Costal, D., Ralyté, J., Franch, X., Méndez, L., Annosi, M.C.: OSSAP – a situational method for defining open source software adoption processes. In: Nurcan, S., Soffer, P., Bajec, M., Eder, J. (eds.) CAiSE 2016. LNCS, vol. 9694, pp. 524–539. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39696-5_32

13. Yu, E.: Modelling strategic relationships for process reengineering. Ph.D. thesis, University of Toronto, Toronto (1995)
14. Ralyté, J., Rolland, C.: An approach for method reengineering. In: Kunii, H.S., Jajodia, S., Sølvberg, A. (eds.) ER 2001. LNCS, vol. 2224, pp. 471–484. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45581-7_35
15. Pohl, K.: Requirements engineering: Fundamentals, Principles, and Techniques. Springer, Heidelberg (2010)
16. Carrizo, D., Dieste, O., Juristo, N.: Systematizing requirements elicitation technique selection. Inf. Softw. Technol. **56**(6), 644–669 (2014)