ARTICLE IN PRESS

SCICO:2059

Science of Computer Programming ●●● (●●●●) ●●●–●●●

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

An ontology-based approach for integrating tools supporting the software measurement process

Vinícius Soares Fonseca, Monalessa Perini Barcellos *, Ricardo de Almeida Falbo

Ontology and Conceptual Modeling Research Group (NEMO), Computer Science Department, Federal University of Espírito Santo, Vitória, ES, Brazil

A R T I C L E   I N F O

Article history:
Received 15 March 2016
Received in revised form 15 September 2016
Accepted 5 October 2016
Available online xxxx

Keywords:
Software measurement
Ontology
Semantic integration
Semantic interoperability

A B S T R A C T

Context: Software measurement is a fundamental practice to support process improvement and project management, since it provides useful data for decision making at both organizational and project levels. Due to the nature of software measurement activities, the use of computational supporting tools is essential. Software measurement can be performed in the context of various software processes and these processes generally have different supporting tools. Thus, it is common to use several tools to allow collecting data regarding the processes. Tools are usually developed at different time, by different teams and without concern for integration. As a result, organizations have to deal with integration issues to enable communication between tools and to properly support the measurement process. A key factor for integration is that tools share a common understanding regarding the meaning of the exchanged terms and services. In other words, it is important to deal with integration not only at the syntactic level, but also at the semantic level. Among the instruments used to address semantics, ontologies have been acknowledged as an important means to address semantic integration. *Objective:* This paper presents the *Ontology-Based Approach for Measurement Systems Integration* (OBA-MSI), an approach that uses ontologies as a basis to integrate tools aiming at supporting the software measurement process. *Method:* OBA-MSI was developed following the Design Science Research paradigm. To evaluate OBA-MSI, we carried out a case study in which we used the approach to integrate tools for a real software development organization. After that, we applied a survey to get feedback from users of the integrated solution. *Results:* The users stated that the integrated solution obtained from applying OBA-MSI properly supported the software measurement process and added value to the organization, providing more benefits than using the tools in isolation. *Conclusions:* The use of OBA-MSI to aid tools integration helps deal with semantic conflicts and contributes to obtain a proper support to the software measurement process.

1. Introduction

Software measurement is a process applied by organizations in several contexts. For instance, in project management, measurement is used to help develop realistic plans, monitor project progress, identify problems and justify decisions [1].

* Corresponding author.
E-mail addresses: vsfonseca@inf.ufes.br (V.S. Fonseca), monalessa@inf.ufes.br (M.P. Barcellos), falbo@inf.ufes.br (R. de Almeida Falbo).

http://dx.doi.org/10.1016/j.scico.2016.10.004
0167-6423/

In process improvement initiatives, measurement supports analyzing process behavior, identifying needs for improvement and predicting if processes will be able to achieve the established goals [2].

Fenton and Pfleeger [3] state that measuring software products, processes and projects is crucial for software organizations because measures quantify the properties of these entities and allow getting relevant information about the work done and to be done. The main purpose of measurement is to provide quantitative information to support decision making [4]. In this sense, measurement should be applied to several software processes (e.g., project management, quality assurance, requirements engineering, coding, testing, etc.) to provide information needed to well-informed decision making at both project and organizational levels.

Typically, organizations use different tools to support different processes. For example, schedule and budget tools are used to support project management; modeling tools are used to support requirements engineering, and development environments and version control systems are used to support coding and source code management. Although these tools are not usually conceived to support software measurement, many times they store useful data related to the supported processes (e.g., number of defects, time and cost spent on activities, number of lines of code, test failure rate, etc.). In order to properly support the software measurement process, these tools must be integrated. However, this is not an easy task. The heterogeneity between systems to be integrated is the major difficulty. In general, each tool runs independently and implements its own data and behavioral models, which are not shared between different tools, leading to several conflicts [5].

Semantic conflicts occur when applications use different meanings to the same information item, i.e., when information items seem to have the same meaning, but they do not [6]. Neglecting semantic conflicts in an integration initiative can lead to integrated solutions that fail in achieving their purposes. To reduce these conflicts, integration initiatives should address semantic issues. In this context, ontologies can be used as an interlingua to map the concepts used by different applications, enabling data and services understanding [7].

In the literature there are some initiatives involving tools integration to support the software measurement process. We carried out a systematic investigation and identified 12 initiatives [8,9]. By analyzing them, we noticed a lack of concern with semantics and failure to consider a holist view of the software measurement process. Besides, none of the found initiatives followed a systematic approach to integrate the tools. Considering that integration is not a trivial task, it is important to adopt an approach that helps dealing with the complexity of the task by providing well-established steps, separation of concerns and reduction of subjectivity.

Taking these gaps into account, we developed the *Ontology-Based Approach for Measurement Systems Integration* (OBA-MSI), a systematic approach that uses the Reference Software Measurement Ontology (RSMO) [10] and the Software Measurement Task Ontology (SMTO) [11] to guide tool[1] integration to support the software measurement process. OBA-MSI specializes the *Ontology-Based Approach for Semantic Integration* (OBA-SI) proposed in [7]. OBA-SI can be applied to carry semantic integration of applications in any domain. However, tool integration in the software measurement domain has some peculiarities that are not properly addressed by OBA-SI. For instance, OBA-SI is more suitable for integrating applications that support the same process. However, tool integration to support the software measurement process typically involves applications designed to support different processes that must be integrated to the software measurement process. Besides, in OBA-SI, integration requirements elicitation is superficially addressed. In OBA-MSI, integration requirements elicitation is detailed following a goal-based approach.

By specializing OBA-SI to the software measurement domain, three main contributions are given: (*i*) OBA-SI general steps are detailed and specialized to the case of software measurement, resulting in more palatable steps to be followed by users; (*ii*) we defined an integration approach more suitable for integrating tools developed with the aim of supporting very different processes, but that produce input for the measurement process; and (*iii*) an ontology framework made up of RSMO and SMTO is provided. One of the most effort-demanding step in OBA-SI is related to select (or develop) and integrate ontologies to be used in the integration initiative. In OBA-MSI, the ontologies to be used are provided and the effort is reduced to understand these ontologies.

In this paper we present OBA-MSI and the main results of its application in a real case study. It is organized as follows: Section 2 presents the main aspects related to software measurement and integration, introduces OBA-SI, the approach from which OBA-MSI was developed, and discusses some related works; Section 3 briefly presents the Reference Software Measurement Ontology (RSMO) and the Software Measurement Task Ontology (SMTO), which are the ontologies used in OBA-MSI to address semantic issues; Section 4 presents the research method followed in this work; Section 5 presents OBA-MSI; Section 6 describes the use of OBA-MSI to integrate tools aiming at supporting the software measurement process in a real software development organization; Section 7 regards some discussions about OBA-MSI and its use; finally, Section 8 presents our final considerations.

[1] In this paper the terms *tool*, *application* and *system* are used as synonyms.