Ontology based Bayesian Software Process Improvement

ICSOFT 2014 - 9th International Joint conference on Software Technologies - Vienna, Austria, 29-31/8/2014

PRINT SMEs PROJECT Archimedes III Research Programme

Stamatia Bibi[1], Vassilis C. Gerogiannis[2], George Kakarontzas[3] & Ioannis Stamelos[1]

[1]Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

[2]Department of Business Administration, Technological Educational Institute of Thessaly, Larissa, Greece

[3]Department of Computer Science and Engineering, Technological Educational Institute of Thessaly, Larissa, Greece

{sbibi, stamelos}@csd.auth.gr,{gerogian,gkakaron}@teilar.gr

Abstract

The paper presents an **ontology based approach that can support small and medium-sized software enterprises (SMEs) to achieve their software process improvement goals**.

To validate the approach parts of this model was instantiated using company specific process data from a telecommunication SME. The resulted process models are further analysed through applying Bayesian analysis.

Introduction

A practical approach for **supporting the improvement of selected software process areas which take place in a software SME** is suggested. The approach is called SPRINT (Software PRocess ImprovemeNT) SMEs and adopts an **ontology-based knowledge representation to capture the relevant data that describe a software process**.

The SPRINT SMEs approach consists of the following steps:

(i) Identification of software process areas of a SME and selection of specific areas which require improvement.

(ii) Definition of a knowledge base that describes the process area under improvement.

(iii) Conceptualization and analysis of an ontology that represents the process domain.

(iv) BN analysis and suggestions for process improvement.

A knowledge based approach for SPI

The **first step** of the **SPRINT SMEs approach** involves the **identification of the defective process area to be improved**. Areas that can be considered are: requirements engineering, design specification programming and development, software testing, software project management etc.

The target of the **second step is to specify and design a knowledge base that consists of information relevant to the knowledge required for improving the area(s) selected in the previous step**. Two questions need to be answered (Bibi et al., 2010): **(i) which metrics can provide useful information for each particular process area? (ii) which projects will be considered to create a process area knowledge base?**

In the **third step**, we adopt an **ontology-based paradigm**. Ontologies formally represent knowledge as sets of concepts within a domain by using a shared vocabulary to denote the types, properties and interrelationships of those concepts. **The analytical ontology for the project planning phase as the target of improvement attempts is presented in Figure 1**. During project planning, the project objectives are defined along with the project schedule and its activities. People to perform the project activities have to be allocated. Also project monitoring and control should be performed.

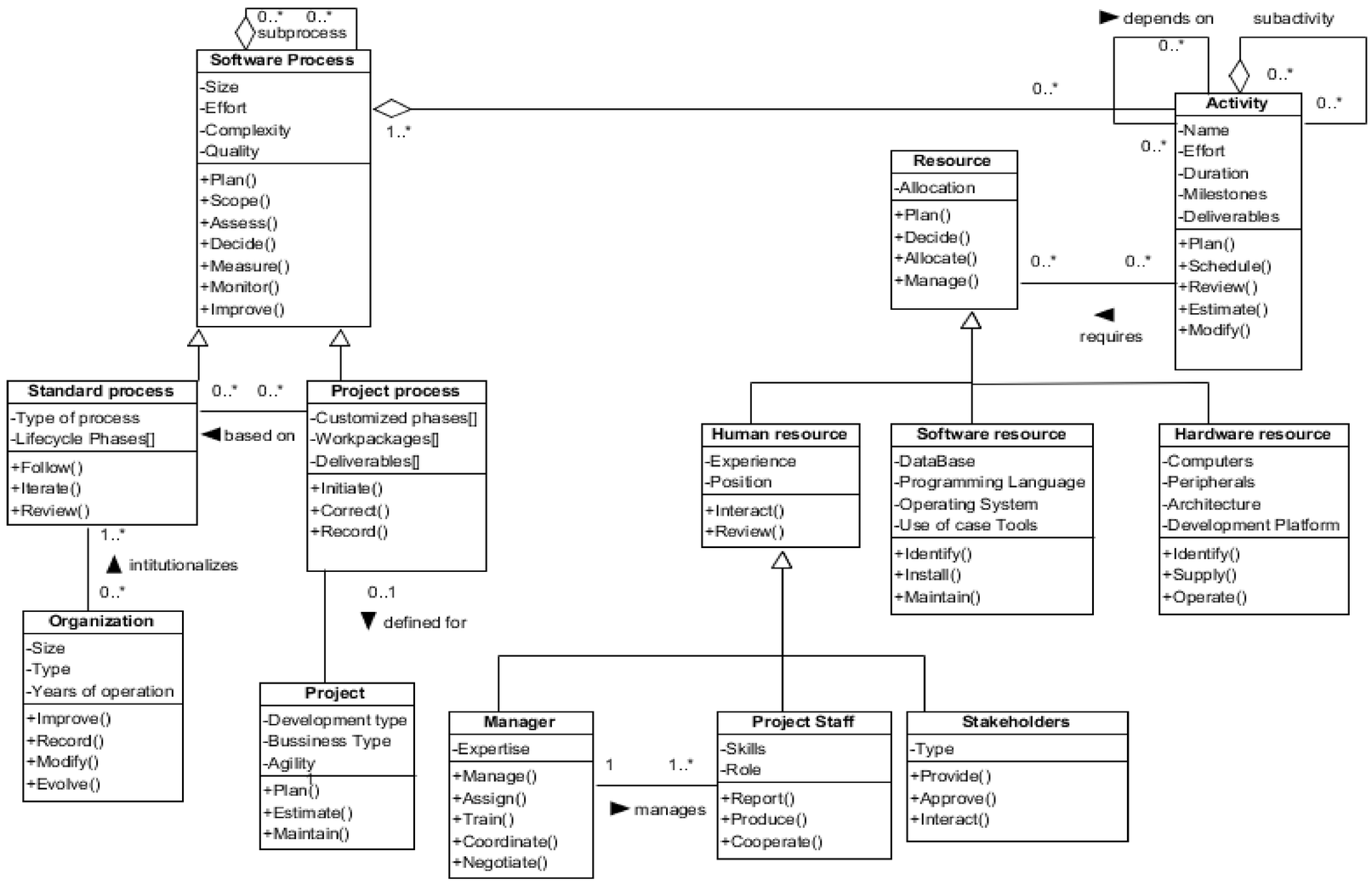

Figure 1. : The extended software process ontology

In the **fourth step**, the approach utilises **Bayesian Networks to experiment with the ontologies defined in the previous step**. A BN is a directed acyclic graph that represents a causal network consisting of a set of nodes and a set of directed links between them, in a way that they do not form a cycle.

Validation

In a Greek telecommunications SME, 35 employees.

- **1st step, defective areas identification:** two areas of improvement, effort/duration estimation and software reuse.
-**2nd step, knowledge base development:** collection of company specific metrics (telecommunication metrics), effort and size metrics from five recent projects.
-- **3rd step, process ontology creation:** Parts of the ontology described in Figure 1 were used.
-- **4th step, BN analysis: Two BN models were created, one involving the effort estimation process (Figure 2) and another one involving the software reuse process**.

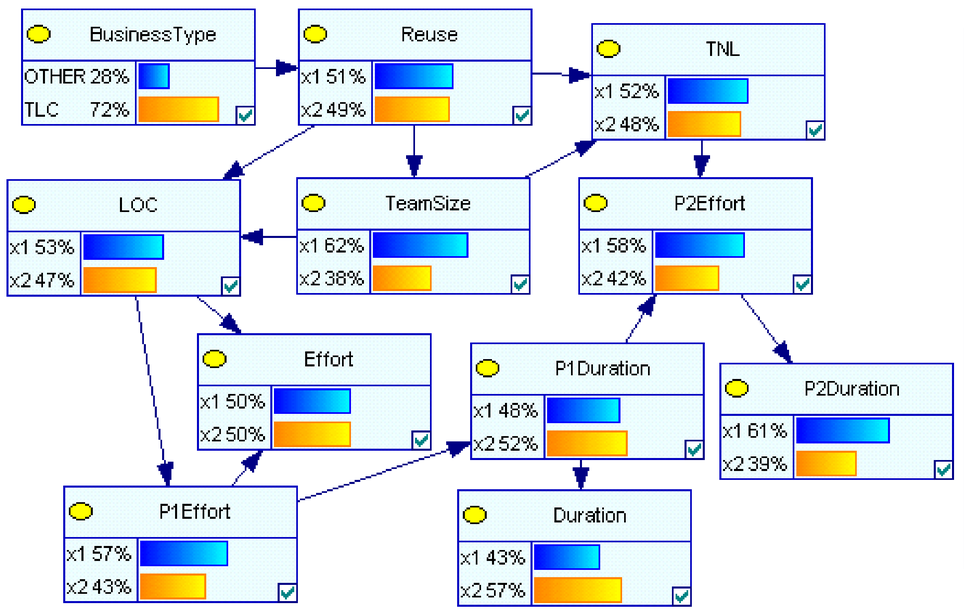

Figure 2. : Software process BN for effort estimation

Useful insights gained:

- **The lower the number of code structure variables the greater the reuse**. It seems that smaller parts of code can be more easily reused.

- Contrary to expectations, **larger teams can produce results in shorter time and they are able to reuse larger percentage of code from previous projects**. The management currently is validating the experimental results on larger teams.

Conclusions

This paper presented an approach to support software process improvement activities for software development SMEs. The approach utilizes **a generic ontology that is tailored to the needs of an SME and applies Bayesian network analysis to make measurable each concept that is represented in the process ontology**. The proposed approach will be further validated at a multiple case study involving Greek SMEs, which show interest in improving their development practices.

Acknowldgements

Research in SPRINT SMEs project is implemented through the Operational Program "Education and Lifelong Learning" and is co-financed by the European Union (European Social Fund) and Greek national funds in the context of the R&D program ARCHIMEDES III

European Union
European Social Fund

OPERATIONAL PROGRAMME
EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION & RELIGIOUS AFFAIRS
MANAGING AUTHORITY

NSRF 2007-2013
programme for development
EUROPEAN SOCIAL FUND

Co-financed by Greece and the European Union

References

- Bibi, S. Stamelos, I., Gerolimos, G., Kollias, V., 2010. BBN based Approach for Improving the Software Development Process of an SME - a Case Study, Journal of Software Maintenance, 22(2).
- Bringuente, A., Falbo, A., Guizzardi, G., 2011. Using a Foundational Ontology for Reengineering a Software Process Ontology, Journal of Information and Data Management, 2(3), 511-526.
- Fenton , N., Neil, M., Marsh, W. , Hearty,P., Marquez,D., Krause,P., Mishra, R., 2007. Predicting Software Defects in varying Development Lifecycles using Bayesian Nets, Information & Software Technology 49(1), 32-43.
- Guizzardi, G., Falbo, R. A., Guizzardi, R. S. S., 2008. Grounding Software Domain Ontologies in the Unified Foundational Ontology (UFO): the Case of the ODE Software Process Ontology. XI Iberoamerican Workshop on Requirements Engineering and Software Environments, pp.244-251..
- Katifori, A., Halatsis, C., Lepouras, G., Vassilakis, C., Giannopoulou, E., 2007. Ontology Visualization Methods - a Survey, ACM Computing Surveys, 39(4).