Towards an Adaptive Improvement Management Framework (Position Paper)

Andreas Jedlitschka & Stephan Weibelzahl
Fraunhofer Institute Experimental Software Engineering
Sauerwiesen 6, D 67661 Kaiserslautern, Germany
[jedl; weibel]@iese.fraunhofer.de

Abstract

Software process improvement (SPI) knowledge is often brought into organizations from outside, for example, by external experts. The idea of this work is to build up an improvement management framework for supporting managers in their work; finding improvement potentials and making strategic decisions about the implementation of improvement measures. The adaptation of the information delivery to the manager's needs and skills is one aspect of the framework and is solved by employing case-based reasoning. Additionally, we use an organizational model that is put forward as the basis for choosing the improvement action that fits most appropriately within the context and is most promising with respect to the organizational goals

1 Introduction

The aim of managers, especially those responsible for quality is to support their organization in the challenge of improving its maturity and competitiveness. The work of a manager comprises of finding the potential for improvement and preparing strategic decisions. This includes (a) setting strategic goals, (b) defining improvement actions, (c) controlling projects, and (d) analyzing the outcomes. Based on the outcomes, further improvement actions have to be chosen. This represents a closed loop of continuous improvement that is described in various literature, e.g., the Quality Improvement Paradigm (QIP) [BCR01a].

When making strategic decisions two challenges might occur: (a) improvement actions have to be well coordinated and adapted to the organizations' needs, and (b) due to high technological turn-over, managers have the need to keep themselves up-to-date with these developments and trends. In practice it is not possible to overlook all potential technological evolution. In discussions with people from industry we have learned that there is a need for guidelines and templates to support decision-making in the manner mentioned above. To support managers, we would like to offer technical and structural support based on the combination of three information sources, namely education on the job (learning by doing and from experience), interpersonal exchange, and an experience-based information system [AD+01]. Empirical studies show that the technological support allows a broader access to experience [Tau00].

This work addresses the challenges mentioned above by proposing an adaptive improvement management framework that (a) allows the classification of the organization, building the basis for adaptation and coordination of improvement actions to the needs, (b) allows the classification of the solutions for improvement actions available, and (c) is adaptive to the user's context and skills (experience and preferred solutions).

2 Proposed Solution

The case-based reasoning (CBR) approach [AP94] has been shown to be an effective user modeling technique for adaptive sales support [WW99; JA01] as well as adaptive decision support for IT security tasks [Jed02]. Thus, we suggest to model the manager by employing CBR for two general purposes: to learn from users' past behavior (content-based) and to learn in a collaborative way by assuming that users behave in ways similar to those of other users before (collaborative) [ZA01].

Actual work focuses on the improvement management framework [JP03; JB03] and the modeling of the targeted organization. The framework in general will consist of (a) the (strategic) improvement management methodology, which stems from the QIP, (b) the categorization methodology, (c) an internal repository for storing project experience in the form of qualitative and quantitative data, and (d) an external, web-accessible repository[1] containing cross-organizational experience on a higher abstraction level [JN03].

Lessons learned from the Experimental Software Engineering Network[2] (ESERNET) project show that for acquiring useful information about technology applicability on the one hand and reaching a critical mass of content for the repositories on the other hand it is necessary to have a standardized way of reporting results and some kind of quality checks [JP03]. Approaches to tackle these challenges include an explicit orientation of the data acquisition towards the users' needs, as well as the use of executive summaries and other reporting structures [CW03, JP03, JB03].

[1] E.g,. see http://www.esernet.org

[2] ESERNET is funded by the Information Societies Technology Programme of the European Union under grant IST-2000-26754.

The approach itself is somehow similar to other approaches used for product recommendation in eCommerce settings [RS02; UM01; AH02]. The difference is that the model of the "customer" is not based on his own interaction with the system, but the manager models the organization. The manager himself gets adaptive support based on his own context model [Jed02]. This means, for example, that improvement actions, which have been chosen by the manager in other projects (so it is assumed that he has experience with it), but also fitting to the situation at hand, will be proposed with a higher ranking than those not yet deployed. If there are solutions fitting better but not yet deployed by the manager, adequate learning material is proposed. Additionally, by deploying improvement actions, experience in this context is collected, and together with the refined organizational model forms the basis for new experience artifacts. Those artifacts will be presented in a context-sensitive way to other users, taking into account that experienced managers do not need basic information about e.g., maintaining customer satisfaction whereas junior managers might need this.

3 An Example

For clarification we give the following scenario: The goals of company XSoft are to reduce the costs of software development. To achieve this strategic goal different improvement measures can be implemented. Two alternatives are: further education of the software developers or introduction of software inspection techniques. Both of them have shown that they are valid under specific circumstances (in laboratory and industrial settings). The system might now recommend to introduce inspections, because the manager has indicated that Xsoft's software developer are well educated which seems to be a prerequisite for this technique. In case the manager has no experience with this technique the system might also offer guidelines, tips or an tutorial on this topic at the same time. In a pilot project inspections are introduced and the effect is measured. The results can be seen as twofold. Firstly, the result is used to evaluate if the inspection technique had impact on the development costs. This is crucial for further investigations in that direction. Secondly, in both cases, success or failure the organization has gained experience and the context has changed.

4 Modeling the organization

The model of the organization by now contains attributes about the organization's context. Many authors from different research communities have observed, that transferring results from one setting to another requires to consider the context carefully. For instance, an empirical study might have been performed under certain context constraints that invalidate the results in settings with a different context. The same is true for the acquisition of experience that requires to know in which context it was gained or it is taken to be valid. We conclude that the context factors are necessary to support decision-making. At the present time there is no commonly accepted definition of context itself and of what factors build an accurate basis to derive similarity conclusions from. For instance, Birk [Bir00] proposes a hierarchical taxonomy of context factors and for more generic conclusions a step-wise aggregation (for instance, company XSoft with 215 employees can in a more generic stage be described as a company with employees in the range between 200-250). This leads to a bigger amount of comparable contexts paying the price of less accuracy.

Due to the huge number of potential factors and the difficulties in acquiring them it seems useful to find a common minimal set of factors that allows conclusions to be drawn regarding the similarity of contexts. The accepted assumption is that if a technique has been successfully applied in one context it is a potential candidate to also be successfully applied in similar context. Examples of contextual factors are: country (Germany, UK, Australia, US, India, Japan, …), domain (automotive, banking, …), management structure (lean, hierarchical, …), size of the software development department, level of the organizational experience (measurement programmes installed? , …), and level of experience of the software developer with different techniques, methods, tools. We recognize that there might be semantically difficulties due to subjective appraisal, and because of rapid development of new technologies we cannot use a fixed taxonomy. To tackle this problem we are applying human based quality checks. This in itself is effort consuming and has to be improved or replaced by an accurate technical solution.

For the first run the values of the attributes have to be collected in an effort consuming way, missing attributes may be estimated based on existing knowledge, or gained through an assessment (self-assessments like in EFQM[3] can also be considered). The model of the company is then compared to existing "technology experience packages" [Bir00] either in the internal or the external repositories. Based on the information gained in this step, the manager should be able to find potential solutions. With the following steps the attributes will be refined more and more by semi-automatically acquiring them from the processes describing the framework environment (e.g., effort sheets, reports about failures or requirement changes), so that forecasting subsequent actions (e.g., further education of employees, changes in development process, introduction of new technologies) and corresponding outcomes (e.g., improved product quality, less development costs, faster time to market) will be more accurate.

5 Contribution to User Modeling (UM)

Obviously the proposed system is not adaptive in the usual sense that the user's needs, preferences or knowledge are inferred. It rather models the organization and thus the user's context. This work will contribute to the UM community with a new application area by means of context adaptive improvement management support on the one hand and the idea of modeling organizations by human interactions on the other. This means that not a natural person but a body corporate is modeled to find appropriate "products" in terms of applicable technologies supporting strategic improvement. The usage of case-based reasoning (CBR) as technology for decision support based on user and organizational models will be further evaluated. Contributions are also made

[3] European Foundation Quality Management (http://www.efqm.org)

to the software process improvement (SPI) community by defining a framework for improvement management, which supports improved adaptation to the organization's needs and goals, and allows better and more transparent coordination of the different improvement actions.

6 Future Work

Besides research on context, the factors relevant to describe context and set reduction future work will especially address empirical research, that is, research concerning the accuracy of the organization model and the improvement action proposed by the system in comparison to the solution an expert in the field would propose in the same case. Also, the system's utility as perceived by its users will be investigated. If a model of an organization is accurate the suggested set of improvement actions will be relevant. The manager and other experts in the field can state that relevance. If the model is less accurate most of the suggested improvement actions are not relevant or do not fit to the organization's needs. Accuracy of the proposed improvement actions can be measured, using experts of the field on the one hand and managers using the system on the other hand. First the experts will have the chance to interview the managers about their needs, context, etc.. Then the experts will make their conclusion based on their experience and the managers will use the system to find a fitting solution. We assume that the results will be in the range of the results that were reported by Tautz [Tau00] regarding experience management systems. He observed that users found more usable answer per time using the technical system than asking experts. Systems utility as perceived by the user can be measured, for example, by asking for feedback. This will also help to improve the content and the system itself.

Acknowledgements

The work presented has partly been funded by the Information Societies Technology Programme of the European Union under grant IST-2000-26754. The complete list of partners and members can be found at www.esernet.org.

References

[AD+01] Althoff, K.-D., Decker, B., Hartkopf, S., Jedlitschka, A., Nick, M. & Rech, J. (2001). Experience Management: The Fraunhofer IESE Experience Factory. In P. Perner (ed.), Proc. Industrial Conference Data Mining, Leipzig, 24.-25. Juli 2001, Institut für Bildverarbeitung und angewandte Informatik

[AH02] De Bra, P.; Brusilovski, P.; Conejo, R. (Eds): Adaptive Hypermedia and Adaptive Web-Based Systems; Proc. of the 2nd Intern. Conf., AH 2002, Málaga, Spain, May 2002, LNCS 2347, Springer 2002

[AP94] Aamodt, A. and Plaza, E. (1994). Case-Based Reasoning: Foundational Issues, Methodological Variations and System Approaches. *AI Communications Vol. 7*, *No. 1*, 39-59.

[BCR01a] Basili, V.R.; Caldiera, G.; Rombach, H.D., Experience Factory; in: Marciniak JJ (ed.), Encyclopedia of Software Engineering, Vol. 1, pp. 511-519, John Wiley & Sons, 2001.

[BCR01b] Basili, V.R., Caldiera, G., and Rombach, H.D.; Goal Question Metric Paradigm; in: Marciniak JJ (ed.), Encyclopedia of Software Engineering, Vol. 1, pp. 528–532, John Wiley & Sons, 2001.

[Bir00] Birk, A.: A Knowledge Management Infrastructure for Systematic Improvement in Software Engineering; Ph. D. diss., Dept. of Computer Science, University of Kaiserslautern, Germany; 2000; Stuttgart: Fraunhofer IRB Verlag

[CW03] Conradi, R.; Wang, A.I.; Empirical Methods and Studies in Software Engineering – Experiences from ESERNET; Springer LNCS 2765, 2003

[JA01] Jedlitschka, A. & Althoff K.-D.: Using Case-Based Reasoning for User Modeling in an Experience Management System; Proceedings: LLWA01 Tagungsband der GI-Workshopwoche "Lernen-Lehren-Wissen-Adaptivität", Forschungsbericht Nr. 763 Universität Dortmund, October 2001

[Jed02] Jedlitschka, A.: Adaptive Support for IT Security Tasks. In [RS02]

[JB03] Jedlitschka, A. & Bunse, C.: On the Use of Empricial Studies in Software Engineering in Bunse, C. & Jedlitschka, A. (Eds) Proc. of the first Workshop in Workshop Series on Empirical Software Engineering "Empirical Studies in Software Engineering", Rovaniemi, Finland, December 2002, Fraunhofer IRB Verlag; to appear

[JN03] Jedlitschka, A. and Nick, M.; Software Engineering Knowledge Repositories in [CW03]

[JP03] Jedlitschka, A. & Pfahl, D.: Experience-Based Model-Driven Improvement Management with Combined Data Sources from Industry and Academia, accepted for ACM/IEEE International Symposium on Empirical Software Engineering 2003.

[RS02] Ricci, F. & Smyth, B.(Eds.) Proc. of the AH'2002 Workshop on Recommendation and Personalization in eCommerce, Málaga, Spain, May 28th 2002.Universidad de Málaga

[Tau00] Tautz, C.: Customizing Software Engineering Experience Management Systems to Organizational Needs; Ph. D. diss., Dept. of Computer Science, University of Kaiserslautern, Germany; 2000; Stuttgart: Fraunhofer IRB Verlag

[UM01] Bauer, M.; Gmytrasiewicz, P.; Vassileva, J. (Eds): User Modeling; Proc. of the 8th Intern. Conf., UM 2001, Sonthofen, Germany, July 2001, LNCS 2109, Springer 2001

[WW99] Weibelzahl, S. & Weber, G. (1999). Benutzermodellierung von Kundenwünschen durch Fallbasiertes Schließen. In Proc. 7. GI-Workshop "Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen", LWA-1999, Otto-von-Guericke-Universität Magdeburg, Germany, 29.-30. Sep.1999, 295-300

[ZA01] Zukerman, I. & Albrecht D.W. 2001. Predictive Statistical Models for User Modeling. In User Modeling and User-Adapted Interaction (UMUAI) 11(1-2): 5-18 (Ten Year Anniversary Issue), 2001. Kluwer Academic Publisher, Netherlands.