ICEIS 2009

11TH INTERNATIONAL CONFERENCE ON ENTERPRISE INFORMATION SYSTEMS

Proceedings

Information Systems Analysis and Specification

MILAN, ITALY – May 6 - 10, 2009

ORGANIZED BY

INSTICC

TECHNICAL CO-SPONSORSHIP

EIC

SWIM

MEMBER
Workflow Management Coalition
WfMC

IN COOPERATION WITH

aaai

acm
Association for Computing Machinery
(SIGMIS)

ICEIS 2009

Proceedings of the
11th International Conference on
Enterprise Information Systems

Volume ISAS

Milan, Italy

May 6 - 10, 2009

Organized by
INSTICC – Institute for Systems and Technologies of Information, Control and Communication

Technical Co-sponsorship by
IEICE SWIM – Institute of Electronics, Information and Communication Engineers / Special Group on Software Interprise Modelling
WfMC – Workflow Management Coalition

In Cooperation with
ACM SIGMIS – Association for Computing Machinery / Special Interest Group on Management Information Systems
AAAI – Association for the Advancement of Artificial Intelligence

Copyright © 2009 INSTICC – Institute for Systems and Technologies of Information, Control and Communication
All rights reserved

Edited by José Cordeiro and Joaquim Filipe

Printed in Portugal
ISBN: 978-989-8111-86-9
Depósito Legal: 292291/09

http://www.iceis.org
secretariat@iceis.org

BRIEF CONTENTS

FOREWORD

This volume contains the proceedings of the eleventh International Conference on Enterprise Information Systems (ICEIS 2009), organized by the Institute for Systems and Technologies of Information Control and Communication (INSTICC), in cooperation with the Association for Advancement of Artificial Intelligence (AAAI), and the Association for Computation Machinery (ACM), technically co-sponsored by the Institute of Electronics Information and Communication Engineers (IEICE), and by the Workflow Management Coalition (WfMC).

This year ICEIS was held in Milan, Italy. This conference has grown to become a major point of contact between research scientists, engineers and practitioners in the area of business applications of information systems. ICEIS 2009 had five simultaneous tracks, covering different aspects related to enterprise computing, including: "Databases and Information Systems Integration", "Artificial Intelligence and Decision Support Systems", "Information Systems Analysis and Specification", "Software Agents and Internet Computing" and "Human-Computer Interaction". Papers published in each and every track describe state-of-art research work that is often oriented towards real world applications and highlight the benefits of Information Systems and Technology for industry and services, thus making a bridge between the Academia and the Enterprise worlds.

Following the trend of previous editions, ICEIS 2009 also had a number of satellite workshops, related to the field of the conference, including the following nine international workshops: 9th Int'l Workshop on Pattern Recognition in Information Systems; 7th Int'l Workshop on Modelling, Simulation, Verification and Validation of Enterprise Information Systems; 7th Int'l Workshop on Security In Information Systems; 6th Int'l Workshop on Natural Language Processing and Cognitive Science; 3rd Int'l Workshop on RFID Technology - Concepts, Applications, Challenges, 3rd Int'l Workshop on Human Resource Information Systems, 1st Int'l Workshop on Future Trend of Model-Driven Development, 1st Int'l Workshop on Ontology for e-Technologies and the 1st Joint Workshop on Advanced Technologies and Techniques for Enterprise Information Systems.

ICEIS 2009 received 644 paper submissions from 70 countries on all continents. 81 papers were published and presented as full papers, i.e. completed work (8 pages/30' oral presentation) and 171 papers, reflecting work-in-progress, were accepted and orally presented as short papers (6 pages/20' oral presentation). Furthermore, 87 contributions were accepted and presented as posters.

These numbers, leading to a "full-paper" acceptance ratio of 12%, and a total oral acceptance ratio below 40%, show the intention of preserving a high quality forum for the next editions of this conference. Additionally, as usual in the ICEIS conference series, a number of invited talks, presented by internationally recognized specialists in different areas, have positively contributed to reinforce the overall quality of the Conference and to provide a deeper understanding of the Enterprise Information Systems field.

This year, the book of Selected Papers, usually published after the conference by Springer-Verlag in the "Lecture Notes in Business Information Processing" series, has been anticipated and made available at the conference, including all full papers. For this reason the ICEIS full papers are not included in the proceedings.

The program for this conference required the dedicated effort of many people. Firstly, we must thank the authors, whose research and development efforts are recorded here. Secondly, we thank the members of the program committee and the additional reviewers for a valuable help with their expert reviewing of all submitted papers. Thirdly, we thank the invited speakers for their invaluable contribution and for taking the time to synthesise and prepare their talks. Fourthly, we thank the workshop chairs whose collaboration with ICEIS was much appreciated. Finally, special thanks to all the members of the INSTICC organizing committee, especially Vitor Pedrosa, whose diligence and dedication was fundamental for the success of this conference.

Two awards will be given to the best papers presented at the conference: one for the best paper award and one for the best student paper award, mainly based on the classifications provided by the Program Committee members and the oral presentation quality.

We wish you all an exciting conference and an unforgettable stay in Milan. We hope to meet you again next year for the 12th ICEIS, details of which will be readily available at http://www.iceis.org.

Conference Chair

Joaquim Filipe, Polytechnic Institute of Setúbal / INSTICC, Portugal

Program Chair

José Cordeiro, Polytechnic Institute of Setúbal / INSTICC, Portugal

CONTENTS

PROCESS INSTITUTIONALIZATION USING SOFTWARE PROCESS LINES

Tomás Martínez-Ruiz, Félix García and Mario Piattini
Alarcos Research Grou, Department of Information Technologies and Systems
Escuela Superior de Informática, Universidad de Castilla-La Mancha
Paseo de la Universidad, 4, 13071 Ciudad Real, Spain
{tomas.martinez, felix.garcia, mario.piattini}@uclm.es

Keywords: Software process institutionalization, Process lines, Variability, Flexibility, Tailoring, Standardization, Variants, Variation points.

Abstract: Software Process Institutionalization is an important step which must be carried out by organizations if they are to improve their processes, and must take place in a coherent manner in accordance with the organization's policies. However, process institutionalization implies adapting processes from a set of the organization's standard processes, and these standard processes must be continually maintained and updated through the standardization of best practices, since adaptation in itself cannot create capable processes. In this paper we propose using the philosophy of software process lines to design a cycle and specify a set of techniques and practices to institutionalize software processes. The cycle, techniques and practices include both process tailoring and process standardization to offer organizations an infrastructure with which to generate processes that are better fitted to their necessities. The use of our cycle will enable capable processes to be tailored from software process lines, and the analysis of these processes will permit the improvement of the organization's set of standard processes and of the software process line.

1 INTRODUCTION

Software product quality does not depend on its own quality, but on the quality of the processes used to create the products and the capability of the processes (Fuggetta, 2000). In order to guarantee this, process models include several best practices obtained from industries. But, since many different enterprises exist, and each of them has its own characteristics and, as Humphrey states, "*just as there are no two identical projects in the world, there are no two identical processes*"(Humphrey, 1989), processes must be tailored previously if they are to be implemented (Yoon et al., 2001).

Software process lines are an approach through which to manage the variations that appear during process tailoring (Rombach, 2005). Variability must be based on a conceptual process with several *variation points* and a metaprocess (Bayer et al., 2005), similar to that of product lines (Bayer et al., 2006). According to (Lu and Sadiq, 2007) variability allows the propagation of best practices. SPEM (Software Process Engineering Metamodel (OMG, 2007)), which is a standard metamodel to represent processes, may offer support to process lines, by means of its extension with *variants*, *variation points* and other elements which provide variability in processes (Martínez-Ruiz et al., 2008).

Furthermore, when processes are tailored they are more and more fitted to the characteristics and needs of the enterprise, and therefore become institutionalized processes of that organization (Calvo-Manzano et al., 2007). These institutionalized processes are ingrained in the way in which the work is performed and there is commitment and consistency to performing the process (Chrissis et al., 2006). The philosophy of institutionalization to create tailor processes is considered in several improvement standards such as ISO 15504 (ISO, 2004), and CMMI (SEI, 2002). As a matter of fact, the third CMMI capability level is described as "a managed process adapted from a set of standard processes using some organization adaptation guidelines, and which provides work products, measures and other improvement information" (Chrissis et al., 2006), which is an institutionalized process.

Since software process lines permit process tailoring, they facilitate support to institutionalize

processes. According to the previously identified issues, in this paper we describe how processes can be institutionalized using software process lines, by means of a cycle with which to manage an organization's institutionalization process in an iterative manner and several techniques and practices to support the cycle. Section 2 describes the aforementioned cycle and Section 3 presents an example of this proposal. Finally, in Section 4 our conclusions and future work are presented.

2 CYCLE OF SOFTWARE PROCESS INSTITUTIONALIZATION

The proposed cycle contains the basic activities through which to perform process institutionalization. shows the steps included in the institutionalization cycle.

The cycle starts with a standard process, which will be institutionalized within the organization. The cycle must be in a continuous state of execution and each of the executions must be considered as iterations. After each execution we consequently obtain a *standardized process* and new *variants* with which to personalize such process. The cycle is divided into four steps. The first two permit the process tailoring, and the other two are focused on process standardization.

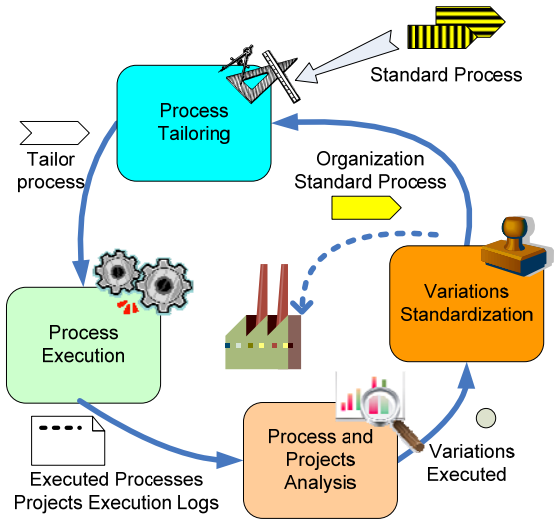

Figure 1: Steps of the Institutionalization Cycle.

- *Process Tailoring*. This step sets off one or more of the organization's standardized process lines. Otherwise, standard process lines or process lines from another organization can be used. In this step, the *variants* are inserted into the *core process* in order to configure the process according to the needs of the project in which the process will be executed.

- *Process Execution*. The tailored process created in the previous step is implemented to develop the project within the organization. Although the process has been tailored, several deviations may occur during the execution of the project, and the process must be readjusted. These adjustments must be registered.

- *Process and Projects Analysis*. The results of the previous steps are analyzed and some process parts are better adjusted than others. This information provides the changes needed in the *core process* and the *variants* which are necessary to generate processes which best fit the necessities of the projects.

- *Process Standardization*. The new *core process* and *variants* defined in the previous step are now standardized, that is they are integrated into the assets of the organization and can be used to tailor new processes (by starting another iteration).

After each iteration, the processes are more and more fitted to the needs of the organization, which is to say they become better and better standardized. In the n-rd iteration of the cycle, the processes are fully standardized to the organization and can be tailored and executed in the project with no variations.

As we can see, during the tailoring part of the cycle, tailoring processes are created by inserting *variation points* into the *core process* of the line and their occupation using *variants*. This corresponds with the top-down approach defined by (Rombach, 2005), while the standardization part describes how to obtain commonalities between processes according to the bottom-up approach.

In order to assist organizations in the use of the cycle, we propose the definition of some techniques and practices based on process lines to support the iterations included in the cycle (to see Figure 2). As can be observed in Figure 2, process tailoring is based on an extension of SPEM to model software process lines (Martínez-Ruiz et al., 2008), which allows the dynamic variation of processes during their execution. By storing and retrieving information in repositories, process mining can be carried out to discover new variations and those which are best. Their use helps us to configure processes according to the characteristics of the organization and project.

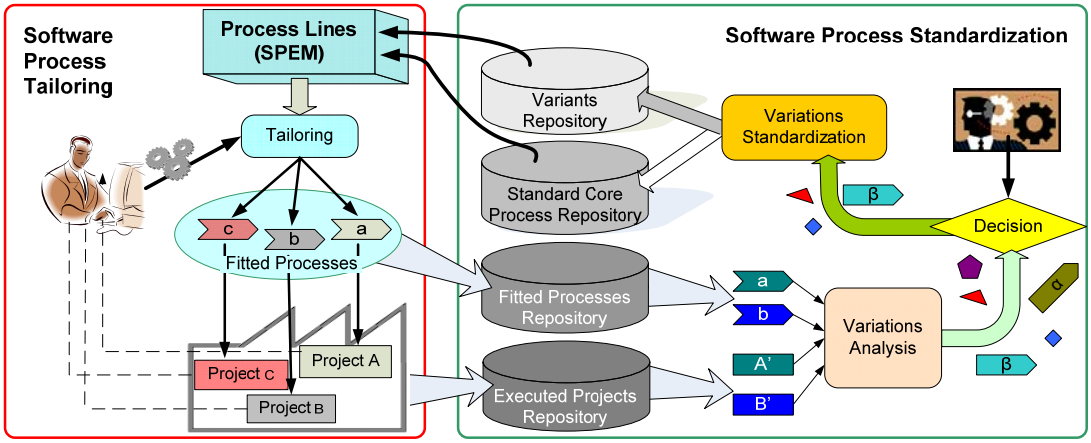

Figure 2: Techniques and Practices for Software Process Institutionalization.

3 APPLICATION GUIDELINES

In order to provide an application example of the proposal presented, we shall illustrate the execution of an iteration of the cycle, by describing how the proposed techniques and practices can be applied. This research is currently being applied to the Development process of the COMPETISOFT process model (Oktaba et al., 2008). This process contains several activities but, depending on the project we are developing, these activities must either be considered separately or can be merged.

To start the cycle, the process line of our development process can be developed by using SPEM with suitable extensions to support process lines, as is described in (Martínez-Ruiz et al., 2008). As is shown in , several variation points can be configured to create different tailored processes.

While processes are being executed, the variants can be reconfigured into the variation points in order to fit the process to the real work.

Both the variations made during tailoring and during execution will be compared to determine how to update the process line in order to support the needs of the variability processes. This information will allow these processes to be standardized within the organization by taking into account the way in which they were executed and by making them more and more similar to the organization's real processes. In the last step of the cycle, new variants and their application in the process are discovered and institutionalized within the organization as simply another component of the process line.

The original Software Development process only considers variability at two points. However, before we execute our cycle the necessity for new variants may be discovered. For example, the inclusion of variability in the construction activity as a result of the programming language and the programming environment used may be necessary. The new process line models this variability by defining new activity variants, and new activity variation points in the Software Development process, as Figure 4 shows.

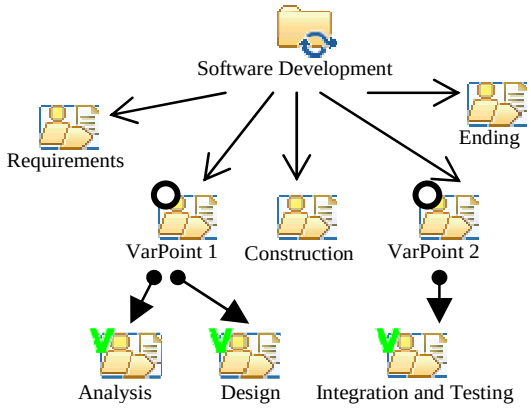

Figure 3: Process line example.

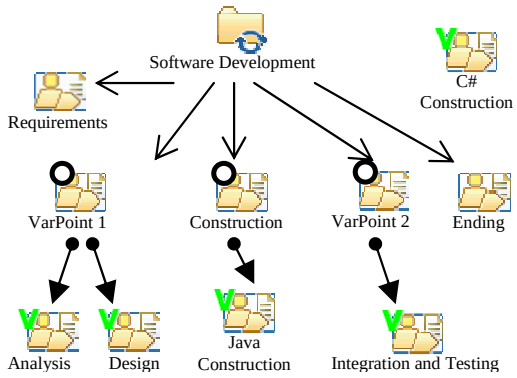

Figure 4: New Process Line with new variability.

4 CONCLUSIONS AND FUTURE WORK

Institutionalization is vital to allow an increase in organizations' maturity levels and to therefore guarantee the quality of software products. Software process lines are a successful mechanism through which to adapt processes. The combination of process lines is an effective means to carry out institutionalization due to their capability of creating personalized processes. As a result of their modularity, they offer a support with which to carry out process mining on the variations, which implies the improvement of both the process line itself and the generated processes. In order to support and to facilitate the implementation of the cycle, we have design several techniques and practices based on software process lines. These offer users the capability to tailor and manage processes, learn about their execution and reuse the information in other processes to facilitate subsequent tailoring.

As future work we wish to design an institutionalization environment which includes the cycle and the techniques and practices described in this paper. The environment will be based on SPEM, with the adequate mechanisms to model process lines. Once techniques and practices have been created, a tool which includes them all and which will permit organizations to use the cycle to institutionalize their processes will be developed.

ACKNOWLEDGEMENTS

This work is partially supported by the research into Software Process Lines sponsored by Sistemas Técnicos de Loterías del Estado S.A. within the framework of the agreement of the Innovación del Entorno Metodológico de Desarrollo y Mantenimiento de Software, the Program FPU of the Spanish MICINN, and by the ESFINGE (financed by the MICINN, TIN2006-15175-C05-05) and INGENIO (financed by the Junta de Comunidades de Castilla-La Mancha, PAC08-0154-9262) projects.

REFERENCES

Bayer, J., Buhl, W., Giese, C., Lehner, T., Ocampo, A., Puhlmann, F., Richter, E., Schnieders, A. and Weiland, J. (2005). Process Family Engineering: Modeling variant-rich processes. PESOA Project, Postdam, Alemania.

Bayer, J., Kose, M. and Ocampo, A. (2006). Improving the Development of e-Business Systems by Introducing Process-Based Software Product Lines. In *7th PROFES 2006*, LNCS 4034 Springer-Verlag, Amsterdam, pp. 348-361.

Calvo-Manzano, J. A., Cuevas, G. and San Feliú, T. (2007). Integración del Modelo de Madurez de la Capacidad-CMMI. In *Fábricas de Software: Experiencias, Tecnologías y Organización* (Eds, Piattini, M. and Garzas, J.) Ra-Ma, Madrid, pp. 235-254.

Chrissis, M. B., Konrad, M. and Shrum, S. (2006). *CMMI: guidelines for process integration and product improvement*, Pearson, Boston.

Fuggetta, A. (2000). Software process: a roadmap. In *ICSE- Future of SE*, Limerick, Ireland, pp. 25-34.

Humphrey, W. (1989). *Managing the Software Process*, Addison-Wesley.

ISO (2004). ISO/IEC 15504-4. Information technology Part 4: Guidance on use for process improvement and process capability determination. International Organization for Standardization, Geneva.

Lu, R. and Sadiq, S. (2007). A Reference Architecture for Managing Business Process Variants, In *9th ICEIS 2007*.

Martínez-Ruiz, T., García, F. and Piattini, M. (2008). Towards a SPEM v2.0 Extension to Define Process Lines Variability Mechanisms. In *Software Engineering Research, Management & Applications*, (Ed, Lee, R.) Springer Verlag. SCI 150, Praga, Czech Republic, pp. 115-130.

Oktaba, H., Piattini, M., Pino, F., Garcia, F., Alquicira, C., Ruiz, F. and Martínez, T. (2008). COMPETISOFT: A Improvement Strategy for Small Latin-American Software Organizations. In *Software Process Improvement for Small and Medium Enterprises: Techniques and Case Studies* (Eds, Oktaba, H. and Piattini, M.) Idea Group Inc. pp. 212-223.

OMG (2007). Software Process Engineering Metamodel Specification. Object Management Group.

Rombach, D. (2005). Integrated Software Process and Product Lines. In *SPW 2005, LNCS 3840*(Eds, Li, M., Boehm, B. and Osterweil, L.) Springer-Verlag, Teddington, UK, pp. 83-90.

SEI (2002). CMMI for Systems Engineering/Software Engineering, Version 1.1. Software Engineering Institute, Pittsburgh.

Yoon, I.-C., Min, S.-Y. and Bae, D.-H. (2001). Tailoring and Verifying Software Process. In *8th APSEC.01*, pp. 202-209.