

SOUTH ASIAN JOURNAL OF MANAGEMENT RESEARCH

(SAJMR)

Volume 1 Number 2

July 2009

Contents

Editorial

- Confluence of Corporate Social Responsibility (CSR) and Strategic Management: A Review** 55
Som Sekhar Bhattacharyya

- Contribution of Location Theories for Regional Development: Why some Technology-Based Firms Choose to be Rural?** 63
João J. Ferreira, Carla S. Marques and Cristina Fernandes

- Rational Unified Process Methodology Frame Work - Tailoring Software in Software Development Process** 76
D.V. Chandra Shekar and O. Ravi Shankar

- Management of Thiocyanate Pollution Using a Novel Low Cost Natural Waste Biomass** 85
Ravindra Y. Thakur and Yogesh B. Patil

- Impacts of Watershed Development Projects Management Through Labour and Machines: A Comparative Study of Two Villages in Maharashtra** 97
Dnyandeve Talule and Sandeep Jadhav

Book Reviews

- Wastewater Treatment for Pollution Control and Reuse** 125
Dhananjay S. Mali

- The Myth of The Rational Market** 128
T.V.G. Sarma



**Chh. Shahu Institute of Business
Education and Research (SIBER)**

(An Autonomous Institute)

Kolhapur - 416 004, Maharashtra State, INDIA

SOUTH ASIAN JOURNAL OF MANAGEMENT RESEARCH (SAJMR)

ISSN 0974-763X



Published by
Chh. Shahu Institute of Business Education & Research (SIBER)
University Road, Kolhapur - 416 004, Maharashtra, India

Contact: 91-231-2535706 / 07 Fax: 91-231-2535708 Website: www.siberindia.co.in, Email: sajmr@siberindia.co.in

■ Patron

Dr. A.D. Shinde

■ Editor

Dr. Babu Thomas
SIBER, Kolhapur, India

■ Editorial Board

Dr. Francisco J.L.S. Diniz
CETRAD, Portugal

Dr. R.V. Kulkarni
SIBER, Kolhapur, India

Dr. R.A. Shinde
SIBER, Kolhapur, India

Dr. Paul B. Carr
Regent University, USA

Dr. M.M. Ali
SIBER, Kolhapur, India

Dr. Lal Das
RSSW, Hyderabad, India

Dr. M. Nand Kumar
Goa University, Goa, India

Dr. Babu Zachariah
SIBER, Kolhapur, India

Dr. Gary Owens
CERAR, Australia

Dr. K. Pradeepkumar
SIBER, Kolhapur, India

Dr. R.M. Bhajracharya
Kathmandu University, Nepal

Dr. P.R. Puranik
NMU, Jalgaon, India

Prof. K.R.R. Mahanama
Colombo University, Sri Lanka

Dr. Yogesh B. Patil
SIBER, Kolhapur, India

Dr. Rajendra Naragundkar
IFIM, Bangalore, India

Dr. K.V.M. Varambally
Manipal Institute of Management, India

Dr. R.L. Hyderabad
Karnataka University, India

Dr. B.U. Dhandra
Gulbarga University, India

The first issue of the first volume of SAJMR received good response from the readers. The feedback received from the readers made us to improve the present issue.

The present issue has broader scope than the earlier, yet we have kept the true spirit of the journal. In this issue we have included articles from the Computer Studies and Environmental Management field as well. As we have stated earlier, the objective of the journal is to provide a common platform for the practicing managers and academicians to share their research knowledge through this journal.

In future, we also welcome articles related to different pedagogical approach in management teaching. Many courses including management discipline everywhere use more of a traditional approach of lecturing to share the knowledge. Lecturing method is more passive in nature. Case study comes next to lecturing method in imparting knowledge. But not much has been done in developing experiential approach as a pedagogy of teaching in management field, particularly in India.

Experiential learning is more active rather than passive. The readers can contribute case studies and teaching material in experiential learning approach in different management fields such as marketing, human resources, organization behaviour, organizational change and development, strategic management, etc.

I am grateful to all the authors, reviewers and editorial members of the journal for their contribution and support in bringing out the second issue of first volume of the journal successfully.

Dr. Babu Thomas
Editor

Rational Unified Process Methodology Frame Work - Tailoring Software in Software Development Process

D.V. Chandra Shekar^{1*} and O. Ravi Shankar²

^{1*} Department of Computer Science, T.J.P.S. College, Guntur-522 006, Andhra Pradesh, INDIA
Email: Chand.info@gmail.com

² Department of Business Management, Gates Engineering College, Gooty, Andhra Pradesh INDIA
Email: oravisankar@gmail.com

Abstract

Rational Unified Process (RUP) is a comprehensive software development process framework that has gained a lot of interest in industry. One major challenge in adopting RUP into use is to tailor it to specific needs. This study presents a review and a systematic assembly of existing studies. We have found that tailoring RUP is a considerable challenge by itself and that tendency is turning from large complete process frameworks towards smaller and more lightweight processes.

Keywords: Method Adoption; Method Tailoring; Rational Unified Process; Software Development Process

1. Introduction

As software development is a highly complex process; methodology support is a prerequisite for the completion of a successful software development project. There exist a wide variety of software development methodologies, spanning from heavy and bureaucratic processes to light-weight and dynamic processes. Lately agile processes have gained a lot of interest both in the industry and academia. A more mature direction within software development methodologies is the Unified Process (UP) (Jacobson, *et al.* 1999) and its commercial variant Rational Unified Process (RUP). There exist no exact figures on how many organizations have tried and used (R)UP – in any variant. However, an overview of experience reports from software engineering conferences, books and magazine publications indicate a considerable interest in UP and RUP. RUP is an extensive framework of collection of best practices described as a structured collection of process components; activities (what to do and how to do it), roles (by whom) and artifacts (what are the input and/or result of the activities). RUP contains detailed descriptions of these components and how they relate to each other. To establish structure, these components are organized in two dimensions; first by phases from inception to elaboration and then by a set of disciplines adhering to common SE activities. In addition, RUP is based on a few basic values; it is

architecture centric, it is use-case driven and it is an iterative and incremental process. Having this completeness and complexity it is not intended to be a silver bullet process for all development project situations – RUP is a frameworks that must be tailored to the situation of use. It is an absolute necessity to do so to get the intentional value from using RUP.

Despite this indisputable interest, the total amount of empirical studies on the *adoption* and *introduction* of RUP is surprisingly low. A search for empirical studies identified only five studies that to some extent explain tailoring and introduction of RUP. We clearly delineate between simple lessons-learned reports that do not present information on context and study method and those that present these details as well as findings, analysis and conclusions. This leads to the aim of this paper: What do the software industry and the research community knows of the limitations, benefits, prerequisites and costs of tailoring and introducing Rational Unified Process? Thus, cost and benefit of RUP *in use* is outside the scope of this paper.

As RUP covers more or less all aspects of SE it may seem easy to take it into use. However there are many challenges in doing so successfully: how to know which parts to keep, exclude or alter, who should get involved in the process, how much time does it take, how is the result to be taken into use, how does one know that the result was good? To be able to answer

such questions and to pinpoint further research needs, at least in part, we have done a literature review of all existing relevant studies on tailoring and introducing RUP - holding a minimum of methodological quality. In addition, the compiled overview is extended with three case studies on the introduction and use of RUP that we have done over the past few years (Hanssen *et al.* 2005a; Hanssen *et al.* 2005b; Westerheim and Hanssen, 2005; Westerheim and Hanssen, 2006) thus bringing together all available empirical experience on the topic.

This paper first describes the research method, both for the literature review and for the case studies. Results then are presented giving an overview of identified experience reports. A discussion summarizes findings from the literature review and our experiences giving a conclusion addressing the research aim of this paper.

2. Background: method tailoring

There exists a set of guidelines for tailoring and adoption of RUP; that specifically targets the issue (Bergström and Råberg, 2004) and covers the issue in some detail (Kroll and Kruchten, 2003). Additionally there exists a guideline documented through a website (Rational PEP. <http://www1.ibm.com/support/docview.wss?uid=swg21158199>). Further there is some guidance in the RUP documentation itself (Westerheim and Hanssen, 2005) or RUP-related books, (however these guidelines tends to be superficial). Despite the existence of these guidelines the authors have not been able to find any experience reports evaluating their outcome and suitability. On the other hand, there exist a set of experience reports addressing tailoring and adoption of RUP done in other ways. These experience reports are summarized and analyzed later in this paper.

The term methodology is defined as "A body of methods, rules, and postulates employed by a discipline: a particular procedure or set of procedures" by the Merriam- Webster dictionary. Basically, a methodology describes how someone, e.g. an organization, performs a task, e.g. software development. In our context we talk about methodologies for running projects with a given customer having more or less defined goals initially.

The process of adapting RUP can take many possible forms. IBM Rational, the provider of RUP has defined the Process Engineering Process (PEP) (<http://www1.ibm.com/support/docview.wss?uid=swg21158199>). This is a comprehensive adaptation process requiring a fairly large amount of resources (people and time). This may very well be appropriate for larger companies, but for the small ones this process may be too expensive. Adaptation of a framework, such as RUP, can take one of (at least) three approaches. The first is to do it in one step, for each project, thus representing a heavy job in each case. This can be justified for large projects. This approach may be called situational method engineering, as defined by ter Hoefstede and Verhoef (1997). The second approach is to do an up-front adaptation producing a subset of the framework, still being a framework, but now tuned to the organizations general characteristics (technology, customers, domain, traditions etc.). This is the intentional process of PEP and may be called method engineering, as defined by Brinkkemper (1996). The third approach is to first identify and describe a set of recurring project types. Having knowledge of characteristics and differences of these types, an adaptation is done for each type. No matter which approach is used; in the last step, a final adaptation is done in each case (project).

Adapting RUP in practice means to decide on which process elements to keep, remove, alter, add or merge. These decisions can be based on assumptions, experience, goals and visions. It is the quality of this underlying knowledge and experience that determines how good these decisions are. Having decided the content and principles of a process it must be made available to the users – the project team(s). Traditionally process descriptions have taken the form of voluminous printed descriptions. Today the most common form is through web-based process guides, RUP Online is such an example. In the case of RUP, IBM Rational provides a set of software tools to assist the reengineering of the process elements of RUP to build a coherent web based presentation of the result. Edwards *et al.* (1998) emphasized the importance of actively involving stakeholders in the process of tailoring situational specific methods. This will

both ensure that necessary detailed information becomes available and affects the tailoring process and that the resulting process actually is taken into use due to ownership and relevance. Various acceptance models such as TAM, TAM2, PCI and others (Riemenschneider *et al.* 2002) may help to explain and underline the importance of involving stakeholders that, after the tailoring, are going to use or be affected by the resulting process. For example, stakeholder participation may affect the *Usefulness*-construct (the extent to which the person thinks using the system will enhance his or her job performance) and the *Ease-of-use*-construct (the extent to which the person perceives that using the system will be free of effort).

3. Method

In this we first describe the study methods used in our three studies – each description is based on four parts: 1) a brief overview of the study context, 2) study aim, 3) data collection procedures and 4) method for data analysis and finally, in the last part of the chapter we present the method used to perform the literature review.

3.1. Case study A

3.1.1. Context

Company A is Deloitte Software Consultancy Company with 50 employees mainly developing software systems with heavy back-end logic and often with a web front-end, typically portals. However, they also develop lighter solutions with most emphasis on the front-end. All development is done in the form of projects. The authors have followed A for a period of five years - having a varying focus over these years; First we studied how A initially used RUP, out-of-the-box, with no restrictions or guidelines. The study is reported in (Hanssen *et al.* 2005b). Secondly, we carried out an action research project to follow A in an attempt to tailor RUP to a predefined project type. The study is reported in (Hanssen *et al.* 2005a). Thirdly, and finally, we have carried out a case study of a pilot project at A using a heavily downscaled variant of RUP documented in the form of an internal Wiki-web. The results from this study are still not published, but are reported in this article.

3.1.2. Study aim

For the three studies, the study aims were respectively; to present an industry case to provide lessons learned and answers with respect to process uptake and effect. The second study aimed to provide others considering remodeling and adapting a process framework in general, and RUP in particular, an insight in how this has been done in a small software company. The third study aimed to study the use and effects of an extensively downscaled variant of RUP documented in the form of a Wiki- web.

3.1.3. Data collection

For the first study we first interviewed four project managers (claiming to be using RUP in four projects) to make a usage map per project to see what parts of RUP actually was being used. Then, we arranged semi structured interviews with five employees with varying roles to document main experiences and find potential explanations for use/no-use of RUP. For the second study we took an action research approach (Avison *et al.* 1999) following A in the whole process of tailoring RUP, as a group-process, to a defined project-type. In the third study we have interviewed the project manager and analyzed internal mid term- and end- PMA-evaluations (Birk *et al.* 2002) of the pilot project being studied.

3.1.4. Analysis

As all three studies have been descriptive with no hypothesis to validate we have done a qualitative analysis. For the first study, interviews were documented on- the-fly in a usage-map (excel spreadsheet) showing which RUP process components had been used or not with potential explanations from the interviewees. Further on, the interviews were transcribed and analyzed using the constant comparison technique (Seaman, 1999). In the second study which was organized according to the principles of action research our report (Hanssen *et al.* 2005a) contains a discussion that extracts and summarizes key learning's. In the third study we also used the constant comparison technique to extract key learning's from the transcribed interview and the internal project evaluations.

3.2. Case study B

3.2.1. Context

Company B Infotech Company is the software development department (300 persons) within a company with a total of 2000 employees. B is focused at both software development and consulting services within the domain of banking and transportation services. The authors have followed B over a period of two years, entering the scene about a year after the company's RUP specialization had been taken into use by projects. This study is reported in Westerheim and Hanssen (2005).

3.2.2. Study aim

The aim of the study was to investigate the level of use of a large-scale RUP specialization, explaining positive and negative experiences using the tailored process and reasons for use/no-use.

3.2.3. Data collection

In this case study we used three main sources of information; 1) a main contact person which was the leader of the tailoring of RUP prior to our study, 2) the process advisory board responsible of the tailoring and the introduction for the new process in the organization and 3) project managers and software developers. Our main method of data collection was workshops and semi structured interviews with these roles. We had three workshops with the project advisory board; information was recorded on-the-fly using mind-maps. We did two rounds of interviews, the first – interviewing representatives from eight projects face-to-face, mainly project managers. The second round of interviews was carried out one year later with the same eight interviewees, this time over telephone. All 16 interviews were recorded and transcribed for later analysis. The aim of the interviews was to document experiences from the introduction of the tailored RUP, find effects – both positive and negative, and to investigate the level of use and corresponding explanations.

3.2.4. Analysis

All transcribed interviews were analyzed using the constant comparison technique, the first eight interviews were coded and analyzed using the NVivo™-tool, the last eight were coded

manually by two researchers in pair using a whiteboard. Lessons learned and experiences were counted across the interviews to find key learning of most significance.

3.3. Case study C

3.3.1. Context

Company C was a company specializing in the development of web applications with a high emphasis on the user experience of the web sites. The company had employed software developers and psychologists. The latter ones worked as producers, specifying the look and feel of the web sites, as well as the logical aspects of the use of the web pages. The company did develop both ecommerce applications and more entertainment types of sites. This study is reported by Westerheim and Hanssen (2006) and Westerheim *et al.* (2002).

3.3.2. Study aim

The aim of the study was to investigate how RUP could support the specifications and development of non-functional parts of a web site. The company had its own tailored RUP, where the original disciplines and the structure of RUP were not changed. The tailoring was a new user experience discipline, with dedicated activities to be performed by new roles.

3.3.3. Data collection

In this case study the main data source was the conducted Postmortem (Birk *et al.* 2002) analyses. Data from six different projects is included in the case study. The tailoring of RUP was already in place when the researchers started to cooperate with the company.

3.3.4. Analysis

The data in the PMA reports was analyzed using constant comparison.

3.4. Literature review method

A systematic review is a strategy for gathering and systematizing results from several independent studies sharing more or less the same thematic focus. The intention is to establish a compiled overview of all relevant experiences

and to identify gaps in existing knowledge, thus indicating the directions for further research. In this case we did a simplified review inspired by the guidelines described by Kitchenham (2004), hence we call it a literature review.

Systematic reviews have traditionally been used to systematize quantitative research, typically statistical meta-analysis. However, most software engineering method-focused experience reports so far are qualitative single-case studies. We therefore needed to adapt practices to be able to systematize qualitative data. This resulted in a review-protocol that we used to 1) define a common research question, 2) search for relevant literature, 3) select studies to include in an analysis and 4) systematize findings and lessons learned.

Step 1: A common research question

We defined the following question for the review: *What are the challenges, prerequisites and success criteria's for tailoring, introducing and using a software development method, e.g. RUP?*

Step 2: Finding relevant literature

The following SE index databases; ISI Web of science, Compendex and ACM Digital Library were searched using the phrase *unified process AND software*.

Step 3: Select studies to keep

All three authors participated in the evaluation of the search results using the following routine:

© **Deselect on title:** a coarse deselection of studies was done based on title, removing studies with an obvious wrong focus. The exclusions and inclusions were based on a few simple selection criteria: The study aim or topic had to be within the frames of *tailoring / adopting / specializing / introducing the Unified Process or Rational Unified Process*. This is resulted in 100 unique studies.

© **Deselect on title and abstract:** The second selection criterion was: the study must present empirical data. This left 36 studies.

© **Deselect on full text:** Studies were excluded if they had insufficient quality with respect to 1) a well defined and limited study aim, 2) an

adequate description of the study method, 3) a sufficient description of the study context, 4) a presentation of the study results, 5) a thorough analysis of the results and 6) giving conclusions or answers with respect to the defined study aim. This left 5 studies.

© **Final, group based selection:** Each resulting study was reviewed by each of the three authors discussing the six quality criterions defined above. This final step left 2 studies.

Step 4: Systematize findings and lessons learned

The main learning's or conclusions from the resulting studies were identified and expressed as claims. A claim can be seen as a hypothesis supported by at least one study.

4. Results

4.1. Case study A

The first part of the study, addressing RUP-use out-of-the-box concludes that a direct use of a framework, such as RUP, with no assistance, tailoring or guidelines results in low use. Introducing a methodology such as RUP is an investment beyond the license fee. In this case the outcome could have been better if the introduction of RUP was carefully managed and not left as an autonomous effort in each project. The second part of the study concludes that a success factor in tailoring RUP to a defined project type is to have focus on the features of the defined process and that a tailoring workshop should consist of persons with proper experience from case projects of the defined type. In the third study we saw that the main objection with the use of the small footprint process guide was lack of content, the project manager typically had a demand for more and better check lists. However, the content was still under development. The project manager commented that it has to be a balance between content size and the lightness as one of the main positive experiences was the simplicity of the guide – it was easy to find relevant guidance. As the process guide is a Wiki-web the project manager clearly saw a need of defining an editor role as editing is free to all and may compromise the content. The content which basically is a collection of activity descriptions organized

over the four RUP phases seemed appropriate for the case project; only four new activity descriptions were suggested. Beyond task guidance the project manager strongly demanded practical process support tools such as estimation models, project follow-up support, a testing framework etc. When asked to comment the difference between this light process guide and the complete RUP, the project manager emphasized the ease of use and clear relevance of the new guide as opposed to RUP's well of information that may be hard to find one's way through. However, interestingly, a definite premise of using such a minimum version of RUP is that the user must have a good understanding of the principles of RUP.

Claim A.1: RUP, out-of-the-box is over-comprehensive and will provide more confusion than guidance and consequently low uptake and use.

Claim A.2: Tailoring RUP efficiently must be based on best practice from the native organization and relevant project cases.

Claim A.3: RUP may be downscaled extensively to increase relevance and ease of use, however, a successful use requires a good knowledge of RUP principles.

4.2. Case study B

The findings resemble known models of technology acceptance (Riemenschneider *et al.* 2002); little knowledge of RUP and thereby low motivation results in low or no use. On the other hand, knowledge and motivation for RUP results in medium/extensive use. In relation, education seems to be an important factor, not only prior to the process but also continuously through the use. Further, we found that management support seemed to be an important factor with respect to uptake and to continuously improve the process during use; this also resembles other similar studies (Dybå *et al.* 2004).

Claim B.1: Low knowledge of RUP creates low motivation and further low uptake and use.

Claim B.2: Management support is a success factor in tailoring and using RUP efficiently.

4.3. Case study C

The main result, when it comes to introduction of

RUP, is that formalization of roles makes them more visible and understandable to others in a project. In this case, new roles related to graphical design were added to the RUP process resulting in a higher acceptance from more technical roles which consequently increased the uptake and use of RUP in the project.

Claim C.1: Explicit definition of roles makes them visible to other project members and thus positively affects the use of the process. Our search for empirically justified claims on RUP tailoring and adaptation resulted in only two study reports; a clear signal that more research is needed in this area. Here we summarize the claims these papers add to the research community. To assess the validity of these claims, we also include a short summary of the setting and research method described in each of the papers. The papers we identified were by Folkestad *et al.* (2004) and Bygstad (2004).

Folkestad *et al.* (2004):

Context: The specific case being studied was a project to transfer an existing system from mainframe architecture to a client-server based architecture. The company saw the project as an opportunity to rebuild and enhance the competence of their staff and was willing to spend resources on this. They chose to use a version of Unified Process as their software development approach. The size of the project was about 30 man-years and lasted three years.

Study aim: The study aims are clearly stated as 1) Identify the effects of changing to a new process. 2) Identify the causes for these changes. 3) Identify properties of the new work process that were instrumental in the change.

Data collection: The data was gathered after the project had been running for one year. The main sources were seven semi-structured depth interviews with members of the software developer group. In addition some data was gathered through informal discussions and from the business documents regarding the development process and the project.

Analysis: The data was analyzed qualitatively using a method called Activity Theory, which can be considered "a framework for the understanding of human activity".

Limitations: Openly discussed in the paper.

Since it is a single case study, it is not easy to generalize the results. Factors like openness, flat hierarchy, and confident staff may be the cause behind the results, just as much as UP itself.

Findings: We have extracted the following findings based on this paper:

Claim R.1: The iterative approach of Unified Process will ensure large effects in terms of learning.

Claim R.2: Unified Process will improve on communication and work distribution in a company.

Claim R.3: Unified Process helps constrain activities and leads to developers being more focused on their tasks, and hence it has a positive influence on productivity and quality.

Claim R.4: As a project develops, elements of Unified Process will become internalized and become tools for the developers. Or in other words, the developers will focus less and less on UP in itself, but focus more on following the practices that they decide to adopt.

Bygstad (2004):

Context: A RUP development project at Scandinavian Airline System (SAS), carried out by the Wipro IT Group (SIG) (owned by SAS). The goal of the project was to establish a web based marketing channel, enable easy publishing and integrating it with the existing booking systems. SAS had chosen RUP as their standard software methodology two years prior to this project. RUP was tailored to the project, and was linked to established practices in SIG.

Study aim: The research questions are 1) how can the project manager control the integration challenge? And 2) what support is there in the software engineering frameworks like RUP?

Data collection: The case was followed for 18 months. Interviews were conducted over three intervals, project meetings were observed and project documentation analyzed.

Analysis: All data was coded with in-vivo codes, using only domain (project) terms. Then each iteration of the project was analyzed qualitatively using constant comparison methods.

Limitations: There is no discussion concerning external validity, but since it is a single case

study, the results may not be easy to generalize. The internal validity is discussed in the paper with emphasis on how they addressed the principles of dialogical reasoning, multiple interpretations and member verification in their analysis.

Findings:

Claim R.5: RUP provides good support for internal technical integration and poor support for external technical integration.

Claim R.6: RUP provides weak support for internal stakeholder integration throughout a project.

Claim R.7: RUP provides strong support for external stakeholder integration in the early phases, but weak support in the later phases.

Claim R.8: RUP gives strong declinational support to step-wise external integration, but too little practical support.

Claim R.11: Using RUP as a basis, linking it to existing best practices results in a process that is actually used.

5. Discussion

The search for relevant empirical studies, with sufficient quality, on tailoring and introduction of RUP resulted in only two study reports. In addition to our three own studies this forms a very small experience base and it has shown to be hard to see any trends across these studies.

From the studies we see that RUP initially is too complex to be used without any tailoring which in practice means that the project manager must make more or less ad-hoc decisions. This becomes an error prone process if the knowledge of the content of RUP is low and thus makes it hard to decide upon which elements to keep, alter or avoid (Hanssen *et al.* 2005b). The RUP-online documentation is a comprehensive collection of process elements and their relations containing about 3700 web pages – which makes it necessary to have a detailed knowledge about the content to be able to select a consistent subset suitable for a given context of use. In the first attempt to deselect RUP elements in case study A we saw that insufficient knowledge of such details quickly became a problem. In case study B a dedicated team needed to get assistance from a trained RUP mentor to be able to accomplish a

successful tailoring. In the second attempt in case study A, a bottom-up approach was used – building a small process guide based on existing best practices using RUP merely as inspiration rather than a commodity. This approach made it at least possible to accomplish the task and resulted in a complete process guide that was taken into use by project teams. In this case, almost all users of this heavily downscaled RUP-process had very high knowledge of RUP through training. This made it possible to use simplistic guidelines as the users knew the details or at least where to find them when needed. The resulting process guide itself in case A was a simple overview of the most important high-level tasks to perform in a development project – no templates or process maps were included. So, the resulting process and its web-based representation can be characterized as minimalist, thus raising the question to what RUP was; how much do you have to keep unaltered to still call it RUP and when is it merely inspired by RUP that by itself is a collection of already existing best practices and guidelines? As a contrast to case A where the basic knowledge of RUP was high we saw in case B that the intended users had little knowledge which clearly affected their motivation for use which consequently also resulted in low uptake of the new process - even though it in this case was tailored to their project characteristics by a dedicated tailoring team. Other studies also support this in the case of acceptance and uptake of electronic process guides (Dybå *et al.* 2004). It is reasonable to believe that low knowledge negatively affects these motivational factors. Further on, in case B, we found that management support was a success factor – one project in this case study was found to actually use RUP and report a certain level of success of doing so. In this case the management had been clear in their expectations that the project should use RUP and supported this. In other projects in the same case study, management was more absent which made the project members use their own varying best practices, thus hampering the goal of establishing a corporate unified development process. Another potential success factor for uptake was found in case study C. As RUP clearly defines roles it became evident how each role was needed and how they related to each other through joint activities and shared facts. This increased the acceptance of existing roles

that was not documented to be a part of the total development process. We have not followed our own cases to assess the use of RUP over time, however Folkestad *et al.* (2004) found that developers, over time, will focus less and less on the process in itself, but focus more on following the practices that they decide to adopt. Thus, the value of introducing RUP may have important effects when it comes to learning a new shared process.

An interesting note in the context of RUP and the challenge of making it fit to local needs and context is the recent spirited development of agile processes (Cockburn, 2002). Ivar Jacobson, one of the contributors to RUP has recently initiated a total remake of RUP, resulting in something called the Essential Unified Process (EssUP). This is intended to be a great improvement of RUP and Jacobson says in a whitepaper (Jacobson *et al.* 2006): “The Unified Process became too heavy, the process improvement programs required too much boring work...”. This is interesting since RUP for years has been marketed as a framework that could help most software organizations in professionalizing software development effectively. EssUP can simply be described as a combination of RUP – which may be seen as a heavy type of process – and agile software development principles (Agile Manifesto: <http://www.agilemanifesto.org>). Our findings, both from our own studies and others support the view that RUP is too heavy and that it may require too much tedious and difficult work. The question is; will a combination of RUP and agile be a better approach? Others as well has addressed the challenge of making RUP simpler and agile which, in sum, can be seen as a shared opinion that RUP have its limitations despite its comprehensiveness. This adds to our findings summarized in this paper.

RUP has, since its creation, gone through several transformations, all leading towards a more agile approach of designing and developing software. This has resulted in various variants and spin-offs of the process, followed by numerous books and even more presentations, speeches, courses and consultant services. It is hard to predict where this will end; however, based on our findings we see a clear need of simplifying RUP (and other processes) to ensure uptake and efficient use. The

development turns clearly towards the agile side of the spectrum – perhaps in search for a balance between discipline and agility (Boehm and Turner, 2004).

6. Conclusion

Based on our, and a few other empirical studies on tailoring and introduction of RUP into development organizations we found that there

exist few or none (reported) direct success stories. All experiences pull in the same direction; RUP is, out of the box, too complex, however, tailoring it to specific needs is also too complex. Looking at the evolution of RUP itself over the past years and the cases we summarize here we see a clear need for, and movement towards, a more agile process that can be tailored with less effort.

References

- Avison, D. *et al.* (1999) Action Research. *Communications of the ACM*, 42(1): 94 (4).
- Bergström, S. and Råberg, L. (2004) Adopting the Rational Unified Process, Addison-Wesley, pp. 165-182.
- Birk, A., Dingsøyr, T. and Stålhane, T. (2002) Postmortem: Never Leave a Project without It. *IEEE Software* 19(3): 43-45.
- Boehm, B. and Turner, R. (2004) Balancing Agility and Discipline - A Guide for the Perplexed: Addison-Wesley.
- Brinkkemper, S. (1996) Method engineering: Engineering of information systems development methods and tools. *Information and Software Technology* 38(4): 275-280.
- Bygstad, B. (2004) Controlling Iterative Software Development Projects: The Challenge of Stakeholder and Technical Integration, In: Hawaii International Conference on System Sciences, Hawaii, USA.
- Cockburn, A. (2002) Agile Software Development. The Agile Software Development Series, Ed. H.J. Cockburn A.: Addison-Wesley.
- Dybå, T., Moe, N.B. and Mikkelsen, E.M. (2004) An Empirical Investigation on Factors Affecting Software Developer Acceptance and Utilization of Electronic Process Guides. In: METRICS, Chicago, USA.
- Edwards, H.M., Barrie Thompson, J. and Hardy, C.J. (1998) Developing situationally specific methods through stakeholder collaboration. In: Computer Software and Applications Conference (COMPSAC).
- Folkestad, H., Pilskog, E. and B. Tessem (2004) Effects of Software Process in Organization Development – A Case Study. In: Learning Software Organizations (LSO).
- Hanssen, G.K., Westerheim, H., Bjørnson, F.O. (2005a) Tailoring RUP to a defined project type: A case study. In: Profes, Oulo.
- Hanssen, G.K., Westerheim, H., Bjørnson, F.O. (2005b) Using Rational Unified Process in an SME - A Case Study. In: EuroSPI, Budapest.
- Jacobson, I., Booch, G. and Rumbaugh, J. (1999) The Unified Software Development Process. Object Technology Series, Ed. G. Booch, I. Jacobson, and J. Rumbaugh., Reading, Massachusetts: Addison Wesley Longman Inc. p. 463.
- Jacobson, I., Ng, P.W. and Spence, I. (2006) The Essential Unified Process – A Fresh New Start.
- Kitchenham, B. (2004) Procedures for Performing Systematic Reviews, Keele University and Empirical Software Engineering National ICT Australia Ltd. p. 33.
- Kroll, P. and Kruchten, P. (2003) The Rational Unified Process Made Easy - A Practitioner's Guide to the RUP, Ed. O.T. Series: Addison Wesley.
- Riemenschneider, C.K., Hardgrave, B.C. and Davis, F.D. (2002) Explaining software developer acceptance of methodologies: A comparison of five theoretical models. *IEEE Transactions on Software Engineering* 28(12): 1110-1135.
- Seaman, C.B. (1999) Qualitative methods in empirical studies in software engineering. *IEEE Transactions on Software Engineering* 25(4): 557-572.
- ter Hofstede, A.H.M. and Verhoef, T.F. (1997) On the feasibility of situational method engineering. *Information Systems Journal* 22(6): 401-422.
- Westerheim, H. and Hanssen, G.K. (2006) Extending the Rational Unified Process with a User Experience Discipline: A Case Study. In: EuroSPI, Joensuu, Finland.
- Westerheim, H., Hanssen, G.K. (2005) The Introduction and Use of a Tailored Unified Process - A Case Study. In: Euromicro Porto, Portugal.
- Westerheim, H., T. Dingsøyr, and G.K. Hanssen, (2002) Studying the User Experience Discipline extension of the Rational Unified Process and its effects on Usability - The design of a case study. In: Empirical Studies in Software Engineering: Proceedings of the First International Workshop, December 2002, C. Bunse and A. Jedlitschka, Editors, Fraunhofer IRB Verlag: Rovaniemi, Finland pp. 69-74.