2003

Transfer of Development Process Knowledge Through Method Adaption and Implementation

Per Backlund
Hogskolen i Skovde, per.backlund@ida.his.se

Christina Hallenborg
Ericsson AB, christina.hallenborg@era.ericsson.se

Guomundur Hallgrimsson
Radar Agency, gudmundurh@iads.is

Transfer of Development Process Knowledge Through Method Adaptation and Implementation

Per Backlund
Department of Computer Science, Högskolan i Skövde
P.O. Box 408, SE-541 28 Skövde, SWEDEN
tel. +46 500 448346 fax +46 500 448399
per.backlund@ida.his.se

Christina Hallenborg
Ericsson AB, Centre for Radio Network Control
Datalinjen 3, SE-583 30 Linköping, SWEDEN
christina.hallenborg@era.ericsson.se

Guðmundur Hallgrimsson
Radar Agency, ISSF, Building 131, Keflavíkurflugvöllur
108 Reykjavik, ICELAND
gudmundurh@iads.is

Abstract

Knowledge transfer is one of the key problem areas in knowledge management. This paper focuses on the transfer of knowledge about the software development process. There is an aim to improve process knowledge and one way of doing it is to introduce commercial development methods. However, doing this is not an easy task. We present a study of how such an introduction of a widespread method was carried out in two large software development organisations. We conclude that a new method has to be adapted and implemented in the organisation in order to make it a part of the organisation specific knowledge base. The new method has to become an integrated part of the existing development process knowledge of the organisation, meaning that it has to be adapted to fit with other organisation specific processes. The ability to do so is the absorptive capacity: the capability to identify and recognise the value of new external knowledge, assimilate it, and apply it.

Keywords

Knowledge management, knowledge transfer, method adaptation, software development process knowledge.

1. Introduction

Knowledge transfer is one of the key problem areas in knowledge management. The problem is frequent since there are a number of different situations in which knowledge is to be transferred within an organisation as well as between organisations. There are a number of different classes of knowledge to consider within the field of information systems engineering (ISE). We claim, in accordance with Iivari (2000), that at least the following areas of knowledge are relevant: a)

Development process knowledge. This includes knowledge about how to run a development project. Development process knowledge also includes the knowledge of how to apply methodologies and technologies for IS development, as well as skills in programming languages. b) *Domain knowledge*. This includes knowledge about the business processes, which the system to be developed is to support, as well as knowledge about how the new information system will affect the organisation as such. c) *Software knowledge*. This includes knowledge about the different aspects of the software product, such as architectural knowledge, design knowledge, knowledge about requirements, and knowledge about other required technological resources. This paper addresses development process knowledge.

Other studies of development processes in software development companies have focused on how methods are utilised, e.g. the method-in-action approach (Fitzgerald, 1998); situational method engineering (Brinkkemper, Saeki & Harmsen 1998; Ralyté & Rolland, 2001) in which methods are constructed by assembling reusable method chunks; and, describing and overcoming the problems of incremental and iterative development (Norbjerg, 2002) and (Madsen & Kautz, 2002). Madsen and Kautz (2002) present an investigation of how the Rational Unified Process (RUP), described in e.g. Kruchten (2000) and Rational (2002), is used at a project level. This paper goes beyond that and describes how RUP is made to fit into the organisation, adding to the understanding of how methods are used in organisations.

In accordance with e.g. Goldkuhl (1994) and Wastell (1999) we view methods as a form of knowledge and it is thus relevant to describe the incorporation of a new software development process in terms of a knowledge transfer. According to Bell, Giordano and Putz (2002) companies work hard to invent and improve processes of their own. There is also an interest in acquiring better process knowledge. If there is a market for process knowledge, it may be considered as a commodity that can be sold and purchased. Indeed, that is what we see when software development companies buy and implement commercial methods, such as RUP.

Acquiring a software development method is an example of knowledge transfer where development process knowledge is to be transferred from a method vendor to a software development organisation. The paper presents a study made at two large software development organisations: Ericsson Radio Systems AB, Centre for Radio Network Control (Ericsson ERA/RNC) and Volvo IT, both in Sweden. The study describes how RUP is adapted to and implemented in the two organisations. We show how the knowledge transfer is made at different levels in order to incorporate the new process knowledge into the organisations.

The remainder of the paper is organised as follows: In the background chapter we elaborate on a model to describe how public development process knowledge is adapted and incorporated into an organisation. In the subsequent chapters we use the model to describe and analyse the adaptation and implementation of a development method in two companies. Finally, we close the paper with a discussion of our results in relation to other work in the area.

2. Background

Leonard (1995) defines core capabilities as the capabilities that embody proprietary knowledge (unavailable from public sources) and are superior to those of the competitors. According to Bell et al. (2002) process knowledge is critical for the performance of a organisation. The product development process is one of the crucial parts of a organisation and hence important to the

competitive advantage. Deriving from this we argue that the development process knowledge is an important core capability of a software development organisation. Since a commercial process such as RUP, is available to any organisation willing to pay for it, RUP may be categorised as public knowledge. Using RUP is a case of what Leonard (1995) describes as importing external knowledge. Although Leonard (1995) refers to technological knowledge we argue that the same reasoning is valid for importing process knowledge. According to Leonard (1995) the relative difficulty of change increases along the four dimensions: physical system, managerial system, skills/knowledge, and values. The physical system is comprised of tools and methodologies and is the 'easiest' dimension to change. Changes at this level are visible but implementation seldom proceeds as planned. Concerning skills and knowledge one issue is how the skills and knowledge base of a organisation can be changed. This depends much on the type of knowledge required. Public knowledge may be obtained by hiring out of formal education programs. Industry specialists can be drafted from competitors. However, organisation specific knowledge cannot be hired. It has to grow over time. Hence, extending the organisation specific knowledge means more than just buying a new development process.

According to Davenport and Prusak (1998) knowledge may be generated through acquisition. In this process originality is less important than usefulness. The most direct and often most effective way to acquire knowledge is to buy it, i.e. to purchase an organisation or to hire individuals who have it. In this paper we also propose that organisations acquire knowledge by buying processes and supporting tools. It is not a question of originality; rather it is a question of successfully implementing commonly available knowledge.

Nonaka and Takeuchi (1995) and Wiig (1993) define two types of knowledge: explicit knowledge and tacit knowledge. Explicit knowledge can be articulated in natural and formal language, which makes it 'easy' to transmit between people via e.g. documents and other types of records. Tacit knowledge has to do with personal knowledge that is embedded in personal experience and is therefore not so easy to formalise and record. Table 1 provides an overview of some characteristics of explicit and tacit knowledge.

Explicit Knowledge	**Tacit Knowledge**
- Can be articulated and stored	- Hard to articulate and communicate
- Generalised knowledge	- Hard to formalise
- Knowledge of rationality	- Individual action and experience
- Knowledge about facts and events	- Subjective and intuitive
- Production knowledge (rules)	- Knowledge of experience
- Procedural knowledge (methods)	- Skills and habits
- Functional and systematic knowledge	- Values and judgements

Table 1 An overview of the characteristics of tacit and explicit knowledge. Based on Nonaka and Takeuchi (1995) and Wiig (1993).

Nonaka and Takeuchi (1995) identify four types of conversions between the two types of knowledge (see Figure 1). For the purpose of this paper we are interested in combination and internalisation.

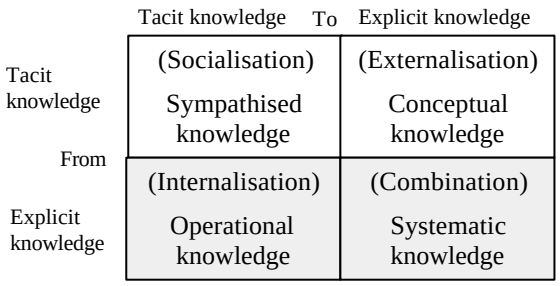

Figure 1 Conversions between tacit and explicit knowledge (Nonaka and Takeuchi, 1995, page 72)

Combination: from explicit to explicit. Combination means that externalised concepts are systemised. In this process different bodies of explicit knowledge may be combined in order to extend the existing knowledge base. The outcome is referred to as *systematic knowledge*.

Internalisation: from explicit to tacit. This is a critical conversion since it is in the process of internalisation that the experiences from socialisation, externalisation, and combination are transformed to new individual knowledge. The outcome is referred to as *operational knowledge*.

Nonaka and Takeuchi (1995) describe the knowledge creation process of an organisation in five steps: sharing tacit knowledge; creating concepts; justifying concepts; building an archetype; and cross-levelling knowledge. It is in this process that the above mentioned knowledge conversions take place. This paper focuses on the fourth step, building an archetype, and the fifth step, cross levelling knowledge.

We refer to RUP as codified and explicit knowledge about systems development. RUP has a number of corner stones, which are referred to as best practices of software development. These practices are: develop iteratively, manage requirements, use component-based architectures, use

visual modelling, continuously verify software quality, and control changes to software. According to our understanding these ideas are well known and most software developing organisations agree on their applicability.

According to Kruchten (2000) RUP may be used in its entirety or in parts. The adaptation may be made at two levels: a corporate level and a project level. The adaptations are documented in a development case. The development case may be kept either as a document or in the form of a web site. Furthermore, RUP prescribes a way of introducing the process into an organisation (Kruchten, 2000). Introducing a new work process means that the knowledge of the co-workers as well as their values are affected.

Based on the views of Nonaka and Takeuchi (1995), Wiig (1993), Leonard (1995), and Davenport and Prusak (1998) we propose a model for describing the incorporation of public process knowledge into a software development organisation (Figure 2). The adaptation of the public knowledge means that the new knowledge has to be aligned with the current processes of the organisation in order to become a part of the process knowledge of the organisation. We refer to this in terms of combination and internalisation as described above.

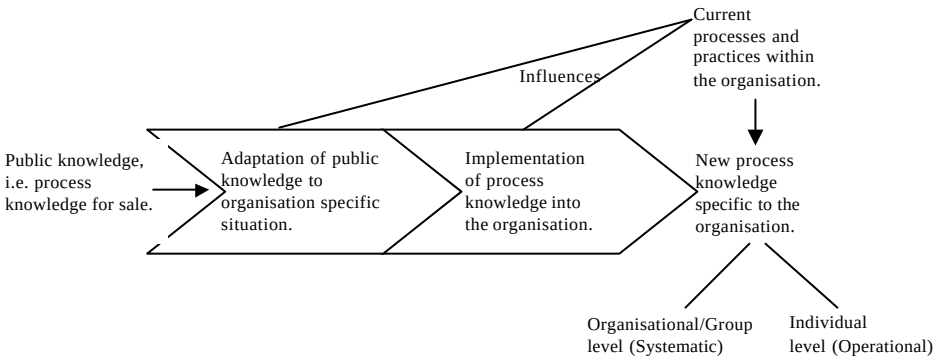

Figure 2 From public process knowledge to organisation specific process knowledge.

In the next step, the adapted version of the process knowledge is implemented in the organisation. This is called internalisation and takes place when new knowledge is made operational in the organisation. These two steps create the process knowledge specific to the organisation. As can be seen from Figure 2 we end up with new systematic organisational knowledge and new operational knowledge.

3. The Case

In this section we describe how RUP was adapted and implemented in two companies, using the model in Figure 2 as a point of departure for our description. The presentation will be structured according to the following:

- The process landscape gives an overview of how RUP fits into the current situation in the company (combination).
- The coverage of the adaptation presents what parts of RUP are used (combination).
- The approach to adaptation presents how the organisations choose to do their adaptations (internalisation).

- The organisational implementation describes the strategy for implementing the new process in the organisation (internalisation).

Data was collected during the spring of 2002.During the document study we had access to internal reports, presentations, and development cases. We looked for indicators of how RUP was adapted and introduced. These indicators helped in forming the questions in order to find out about the individuals' opinions about RUP and its implementation. The interviews involved eight persons at Ericsson ERA/RNC (using field notes and summaries of responses) and four persons at Volvo IT (using tape recordings and interview transcripts). The interviews may be characterised as semi structured and the respondents were given the possibility to read the questions before the interviews.

The major reasons for Ericsson ERA/RNC to introduce RUP are: a perceived need to further develop the incremental way of working; the current in-house process for systems development is no longer maintained; the current in-house process for systems development is based on the waterfall model; and, a need for revision in order to integrate new processes. The projects in Ericsson ERA/RNC are typically large with a duration of more than two years. When Volvo IT decided to adopt the Rational Unified Process as the main method in all new application development the main reasons for change were: a need to deal with the more rapid changes of the customers' operations; a need for a global and common way of work; and, a need for a modifiable development process.

3.1 The Process Landscape

To use the term of Leonard (1995), development process knowledge may be perceived as a core capability for a software development organisation. Few companies can build core capabilities without importing knowledge (Leonard, 1995). The desire to import development process knowledge may indicate that the company has identified a capability gap, i.e. the existing internal processes are not sufficient. However, our study shows that there is a need to map the new process knowledge to the existing one, thus forming a new process landscape. We view this as a way to combine different bodies of knowledge to form new systemised concepts (Nonaka & Takeuchi ,1995).

Ericsson ERA/RNC has previously carried out their development projects using two parallel flows: the project flow and the systems development flow. The project flow comprises the coordination and management of projects, whereas the systems development flow consists of the activities devoted to building the actual system, i.e. analysis, design, implementation, and testing. Since there has to be coordination between the two flows there is a need for integration points between them. These points are referred to as milestones and constitute the mapping between the flows and the various sub processes. Figure 3 gives an overview of the process landscape in Ericsson ERA/RNC.

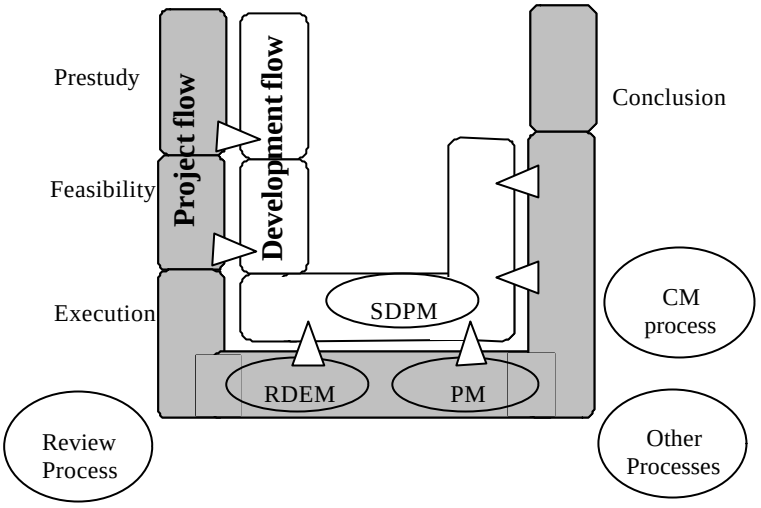

Figure 3 The process landscape in Ericsson ERA/RNC

As can be seen from Figure 3 there is a number of existing processes supporting project management and development activities. SDPM is an in-house process for systems development. It covers activities from pre-study to implementation and testing. However, it does not cater for project administration and configuration management. RDEM is an in-house model for requirements driven incremental development. PM is an in-house process for project management. Apart from the above-mentioned processes there is a number of other supporting processes, including the CM process for configuration and change management.

There is a wide range of methods in use at Volvo IT. Most of which are in-house methods, but some are commercial ones. As is the case in Ericsson ERA/RNC, RUP is to become a part of the method landscape (Figure 4) in the organisation.

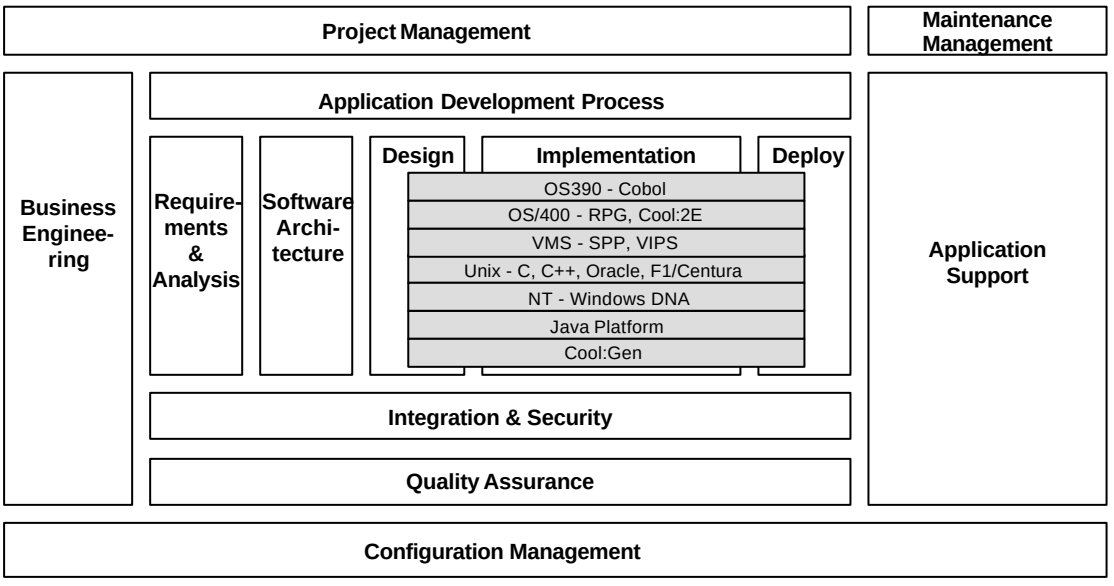

Figure 4 The method area map. From Karlsson Blom (2002)

As can be seen from Figure 4 there are a number of methods complementing RUP. For example there are in-house methods for project management, maintenance and configuration management, application support, quality assurance, and business engineering.

3.2 Coverage of the Adaptation

The concept of combination is also an issue of deciding on the coverage of the adaptation. In accordance with Nonaka and Takeuchi (1995) we describe this phenomenon in terms of a reconfiguration of the new knowledge together with the internal development process knowledge described in the previous section. We studied three adaptations at two different departments at Ericsson ERA/RNC. The first version was aimed to comprise an *introduction* specifying the purpose and scope of the adaptation and a crude description of the development work at the actual department. The *overview of activities* gives a graphical representation of how the different activities in a phase are related to each other. There is also a mapping to the actors carrying them out. The *workflow matrix* shows what activities the adaptation includes and to what artefacts they are related. There are actually four matrices, one for each phase defined in RUP. The adaptation does not only cover the selection of workflows but also how they are modified.

The adaptation made at the other department covers three concepts: *milestones*, *roles*, and *process areas*. The adaptation relates RUP to the organisation specific project management process in which there are checklists for each milestone. The roles are defined in order to make explicit to the project members what is expected from them. Finally, the process areas in the adaptation may be described in terms of a mapping between the organisation specific processes and RUP workflows (Figure 5). According to the interviewees none of the workflows from RUP were implemented as is.

Workflow	***Department 1***	***Department 2***
Business Modelling	**RUP**	Not described
Requirements	**RUP**	**RUP**
Analysis and Design	**Internal/RUP**	**Internal/RUP**
Implementation	Internal	**Internal/RUP**
Test	Internal	**Internal**
Deployment	Internal	**Internal**
Configuration and Change Mgmt.	Internal	Internal
Project Management	Internal	**Internal**
Environment	Not described	Not described

Figure 5 An overview of the processes used in the different workflows. Bold style indicates inclusion in the development case.

The study made at Ericsson ERA/RNC indicates two different approaches of adaptation: use the development case to show how RUP is used in the organisation or use the development case as a framework to refer to all sub processes in a project, no matter if they are new or already existing.

At Volvo IT we found adaptations specific to different environments. The guidelines address for example how to proceed from design to code in a Windows DNA environment (Figure 4). There

are also local adaptations and complements to RUP in order to make it fit better into the specific circumstances of a certain location. This may include work procedures for which RUP may lack support. At the higher organisational level the decision to complement RUP with an in-house method for project management and administration constitutes an example of this approach. Furthermore, Volvo IT has made a mapping of the roles in RUP to nine more abstract roles, considered more suitable for the type and size of projects carried out in the company. In addition to these adaptations each division (location) may make adjustments in order to fit RUP to local purposes.

The development case provides a description of how the core workflows of RUP are employed in the particular project. In the project adaptation studied, all core workflows were used with various modifications. The largest modifications were made to the business modelling workflow. Considering the supporting workflows we have already described how project management is supported by the in-house project management method. We also note that additional artefacts complement RUP: project measurements, risk management plan, project charter, steering committee plan, and standardisation of names. All these artefacts originate from the in-house project management method.

Some of the respondents described that they find it convenient to build on their earlier experiences when carrying out activities prescribed by RUP. They do not follow the steps exactly; they rather use RUP as a checklist in parallel or afterwards. We also found that in situations where the activities in RUP have to undergo major changes, developers tend to do things in the way they are used to work. This also implies that activities not described in RUP are carried out as complements to the method. We attribute this to the existing development practices of the organisation.

3.3 Approach to Adaptation

Nonaka and Takeuchi (1995) use the concept of internalisation when describing the move from explicit to tacit knowledge. We use the term to describe a similar phenomenon when organisations internalise new knowledge. Internalisation is improved when knowledge is verbalised. RUP actually offers a way of doing this by the development case. Thus the creation of a development case may serve other purposes than instantiating the process in a certain project.

At Ericsson ERA/RNC the adaptation of RUP was made at two different departments. The first case, on which we have focused our investigation, experienced a situation where the current processes offered no support. The adaptation was intended to solve the problems experienced and was made within the scope of the actual project. The development case was used during the project, until a new situation arose and the development case was discarded. After this decision a group was appointed to review the initial development case. Their recommendation was to use a more organised approach to adapt and implement RUP in the organisation. In order to achieve this, consultants from Rational were brought in. However, the suggested areas for improvement did not differ significantly from the ones identified by the organisation itself. The greatest advantage perceived was that the work with the chosen areas of improvement became more structured.

The adaptation of RUP may be done in different ways. The two departments have made different decisions regarding how to represent their development cases. One department integrated their adaptation with RUP online. This approach means that the initial RUP structure is used with the new and adapted department specific parts available in the navigation menu. The other department chose to build a separate web structure for their adaptation. The structure is not an integrated part of RUP online, but it is linked to it.

Volvo IT provides an adaptation at an organisational level in order to make better use of resources and competence. This high level adaptation of RUP provides guidelines for implementing and adapting the process, which are made available to other departments of the company. The adjustments made at the different locations may for example concern documented work procedures, specific to a department, for which RUP provides no support. This is done in a bottom up manner so that these guidelines are made accessible to other departments.

There is also a possibility to customise the product itself, i.e. RUP, as has been done at Ericsson ERA/RNC by integrating the adaptations made with RUP online. However, Volvo IT has decided not to make any major modifications to the product itself since such an approach would require much rework for new versions of the product.

3.4 Organisational Implementation

Following the work of adaptation there is a phase of implementing the new process in the organisation. Nonaka and Takeuchi (1995) describe internalisation in terms of learning by doing and creating archetypes.

The implementation in Ericsson ERA/RNC was made in parallel with the evaluation. A workshop was carried out in order to spread the knowledge from the pilot projects throughout the organisation. In spite of the parallelism, the implementation took longer than estimated in the RUP documentation not only due to the fact that the adaptation of the process is included, but also due to the fact that the organisation and its technology is complex and that the projects are large and spanning up to two years (Figure 6). The pilot projects used in the typical approach (Figure 6) may be described as archetypes in that they serve as a means of disseminating knowledge. They also provide a way of learning by doing. The interviewees point out that the integration between RUP and the other processes used in the company is cumbersome since each process has its own terminology and there is a need for translations, something that may be explained in terms of creating new concepts (Nonaka & Takeuchi, 1995). However, the usage of integration points has made process integration viable.

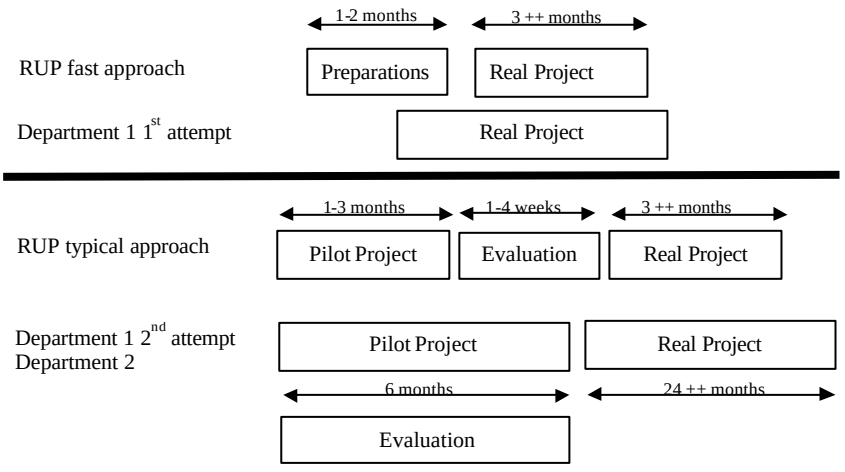

Figure 6 How the implementation approaches were used at Ericsson ERA/RNC

The strategy chosen at Volvo IT was to draw on the experiences from six pilot projects undertaken in 1999 (Figure 7). Management decided on a project-by-project strategy in order to implement RUP incrementally into the organisation. In order to feed the organisation with competent people Volvo IT educated RUP coaches and RUP specialists whose mission was to spread the knowledge throughout the organisation. In the first projects consultants played these roles, but the plan is to replace them with in-house staff.

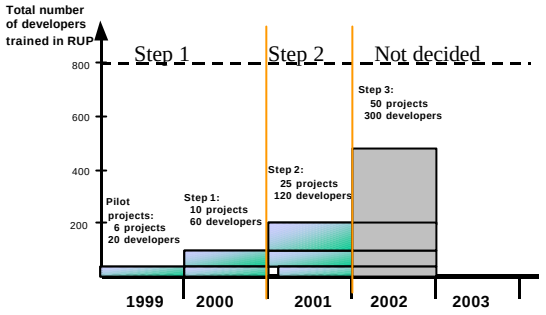

Figure 7 The RUP implementation plan at Volvo IT (Börjesson, 2002)

Another consideration was to identify suitable RUP projects. Such projects are characterised by: a team size of 3-10 persons; a duration of 3-9 months; and an estimated effort of 2000-3000 hours. The next step, to use RUP in larger projects, is planned to take place when half of the developers in the organisation are experienced in using RUP.

4. Discussion and Conclusion

In this paper we have proposed that a newly acquired process has to be adapted and implemented in the organisation in order to make it a part of the organisation specific knowledge base. It is not only a question of adapting the process itself, it also has to become an integrated part of the existing process of the organisation. The ability to do so is the absorptive capacity: the capability to identify and recognise the value of new external knowledge, assimilate it, and apply it to commercial ends. Figure 8 shows how the model presented in Figure 2 may be instantiated in order to describe the adaptation and incorporation of RUP. We note that the public knowledge contained in RUP is made organisation specific in terms of a new development process at an organisational/group level as well as being internalised at an individual level. Furthermore, RUP is combined with current organisation specific process knowledge thus forming new organisation specific knowledge.

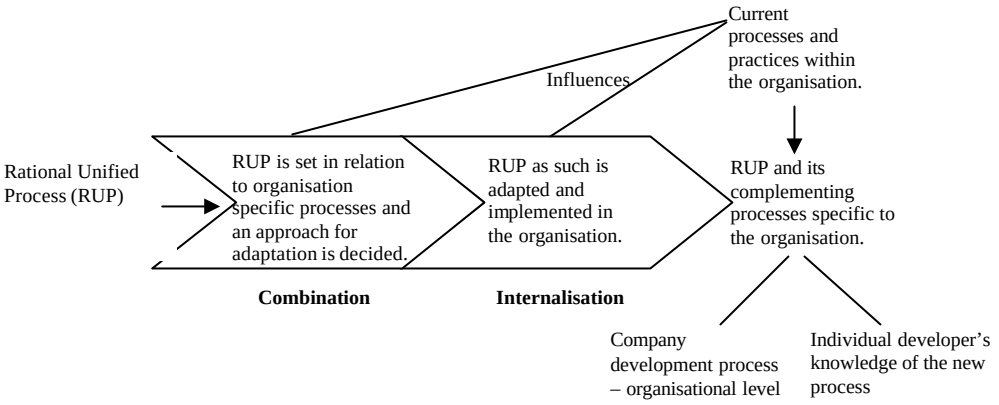

Figure 8 The acquisition of RUP means that it is made a part of the organisation specific development process knowledge.

We found that a commercial method, such as e.g. RUP, is general in order to fit into different organisations. In this sense RUP itself is no guarantee for success. According to Lyytinen and Robey (1999) externally acquired knowledge seems to bring little competitive advantage since it is available to any competitor. We would rather describe the competitive advantage in terms of being able to assimilate the knowledge contained in RUP and making it a part of the organisational knowledge. This in turn means an ability to complement in-house processes with RUP. We argue that it is the combination of external and internal knowledge that brings this advantage.

Our study indicates that RUP is typically replaced or complemented with internal processes when it comes to project management, configuration and change management, testing, and quality assurance. We find this important in that the companies seem to appraise in-house methods as better alternatives in a number of situations. There is hence a need to create mappings between the processes and roles within the own organisation and those prescribed by RUP. Our study shows that this is done in order to relate RUP to the working habits in the organisation.

When comparing our findings with a study made of method tailoring at Motorola (Fitzgerald, Russo & O´Kane, 2002) we discover a group level adaptation of RUP, called ERUP (although not further discussed in the paper), at Ericsson, (Figure 9). We also introduce an intermediate level in our description of Volvo IT reflecting the organisational adaptations made to RUP at the higher level. In the case of Volvo IT we found a process of mutual adaptation between the higher level and the different divisions. We also identified and described an individual level, which is not described in the Motorola case.

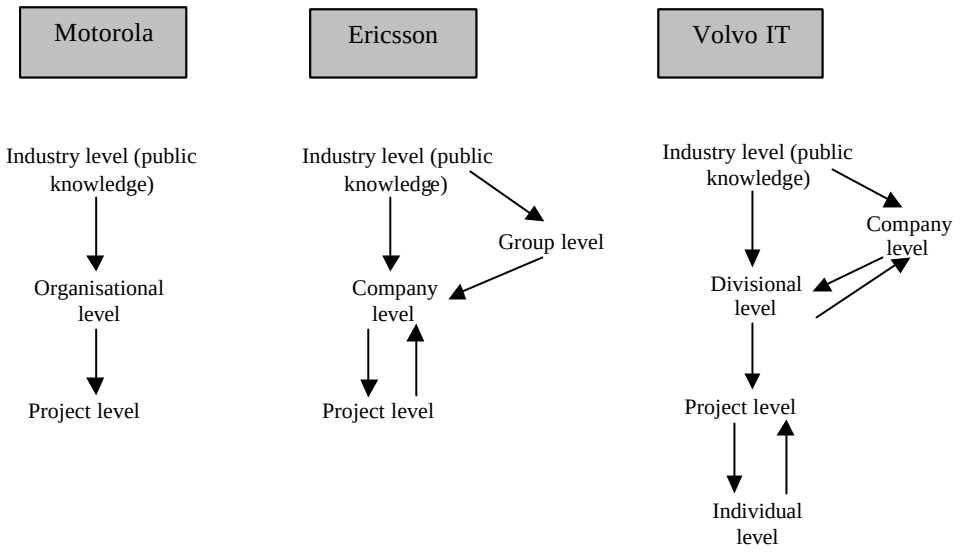

Figure 9 An overview of the adaptations in three different companies

Our study gives a view of how adaptations are communicated between the levels, i.e. adaptations are disseminated throughout the organisation both in a top down and in a bottom up manner.

We conclude that there are advantages in having a web-based method since it makes it easier to reach users in a distributed organisation, which fits well with the argumentation of Brinkkemper (2000). Whether to make the adaptation an integrated part of RUP online or not may be an issue. One potential advantage may be that the user has to become acquainted with the RUP structure in order to find the relevant elements. This may be a good approach when there is a need to make the process as a whole known. The drawback is that the overview of the adaptation is lost. On the other hand, if the aim is to have the adaptation in one place in order to provide good overview the solution may be to build a separate structure. However, other findings indicate unwillingness to do far-reaching adaptations due to potential problems when cooperating with other organisations and in keeping up with new releases of the product.

Our study indicates that there is a need to choose what parts of RUP to introduce. The focus should be put on the areas which need to be improved and where RUP can contribute in doing so. These choices have to be made since the process is both extensive and complex. We distinguish two approaches: a broad adaptation, in the form of a process map, without paying much attention to details; or a deep adaptation which includes more detailed descriptions of activities. Which type of adaptation is suitable depends on the requirements of the organisation. We consider it rather resource demanding to provide both depth and broadness, even though both may be required in order to provide optimal support to method users.

At an individual level we found that RUP is considered to be rather complex and hard to grasp. The novice may spend a large amount of time doing everything prescribed by the method. In fact, Wastell (1999) and Hildreth and Kimble (2002) describe that a method (and organisational procedures) embodies both technical knowledge and authority; and that the authority has a positive value for inexperienced developers. The method does not only prescribe how things should be done, it says that things should be done. After one or two projects the general impression seems to be that developers are more careful to consider what they do in the sense that all actions should bring value to the system being produced. Consequently, the selection of what to use and what to

exclude is an important aspect of using RUP. It seems that the value of RUP actually increases with the adaptations, since they make it possible to preserve and further develop prior organisational knowledge.

We also see that the time spent on introducing RUP is longer than estimated in the RUP documentation. We consider that one reason for the delay, at least in Ericsson ERA/RNC, is the move from a waterfall model to a model of incremental development. Such a change includes adjustments in the value dimension described by Leonard (1995), which is the hardest one to change. These difficulties may imply a more cautious strategy for implementation but there is often a difference in how an organisation would like to do something and what it has the resources and time to actually do.

Finally, we have understood that current development practices, or theories in use to use the term of Lyytinen and Robey (1999), are important for how a new method is accepted and adapted at the level of the individual developer. This is an interesting way to pursue further work in order to understand how methods affect the way of working.

Acknowledgements

This research is sponsored by the Swedish Knowledge Foundation (KK-stiftelsen).

References

Bell, D. G., Giordano, R. and Putz, P. (2002) Inter-Firm Sharing of Process Knowledge: Exploring Knowledge Markets. *Knowledge and Process Management,* 9, 12-22.

Brinkkemper, S. (2000) *Method Engineering with Web-enabled Methods*. In Information Systems Engineering State of the Art and Research Themes. (Brinkkemper, S., Lindencrona, E. and Sölvberg, A. Eds.) Springer, London.

Brinkkemper, S., Saeki, M. and Harmsen, F. (1998) Assembly Techniques for Method Engineering. *In Proceedings of CAiSE'98*, Vol. LNCS 1413 (Pernici, B. and Thanos, C. Eds.), pp. 381-400 ,Springer, Berlin.

Börjesson, G. (2002) Report about the results from implementing RUP at Volvo IT with the aid of the MAPS-2 project. Volvo IT, Internal documentation.

Davenport, T. H. and Prusak, L. (1998) *Working Knowledge: How Organisations Manage What they Know*. Harvard Business school. Boston, Mass.

Fitzgerald, B. (1998) An Empirical Investigation into the Adoption of Systems Development Methodologies. *Information & Management,* 34, pp. 317-328.

Fitzgerald, B., Russo, N. L. and O'Kane, T. (2002) Software Development Method Tailoring in Motorola. Forthcoming in the *Communications of the ACM.* Viewed 2002-10-10 <http://afis.ucc.ie/bfitzgerald/Default.htm>

Goldkuhl, G. (1994) Välgrundad Metodutveckling. Department of Computer Science, University of Linköping.

Hildreth, P. M. and Kimble, C. (2002) The duality of knowledge *Information Research,* 8. Available on-line: Viewed 2003-03-13 <http://informationr.net/ir/8-1/paper142.html>

Iivari, J. (2000) Information Systems Development as Knowledge Work: The body of systems development process knowledge In *Information Modelling and Knowledge Bases XI* (Eds, Kawaguchi, E., Hamid, I. A., Jaakkola, H. and Kangassalo, H.) IOS Press, pp. 41-56.

Karlsson Blom, A. (2002) Metoder i Skövde. Volvo IT, Internal documentation.

Kruchten, P. (2000) *The Rational Unified Process An Introduction* Second Edition. Adison-Wesley. Reading, Massachusetts.

Leonard, D. (1995) *Wellsprings of Knowledge Building and Sustaining the Source of Innovation*. Harvard Business School Press, Boston.

Lyytinen, K. and Robey, D. (1999) Learning failure in information systems development *Information Systems Journal,* 9**,** 85-101.

Madsen, S. and Kautz, K. (2002) Applying System Development Methods in Practice - The RUP Example. *In Information Systems Development (ISD)* (Grundspenkis, J. et al Eds.) Kluwer Press.

Nonaka, I. and Takeuchi, H. (1995) *The knowledge-creating company: How Japanese companies create the dynamics of innovation.* Oxford University Press, New York.

Norbjerg, J. (2002) Managing Incremental Development: Combining Flexibility and Control *In European Conference on Information Systems (ECIS*), Vol. 1 (Wrycza, S. Ed.), pp. 229-239.

Ralyté, J. and Rolland, C. (2001) An Assembly Process for Method Engineering. *In Proceedings of CAiSE '01* (Dittrich, K., Geppert, A. and Norrie, M. Eds.), pp. 267-283, Springer, Berlin.

Rational Corporation (2002) Rational Unified Process Last Accessed 2002 04-08 <www.rational.com>

Wastell, D. G. (1999) Learning Dysfunctions in Information Systems Development: Overcoming the Social Defenses with Transitional Objects. *MIS Quarterly*, 34 (4), 581-600

Wiig, K. M. (1993) *Knowledge Management Foundations*. Schema Press, Arlington.