

PRACTICE SELECTION FRAMEWORK

MARTIN IVARSSON

*Department of Computer Science and Engineering
Chalmers University of Technology
Gothenburg, SE-412 96, Sweden
martin.ivarsson@chalmers.se*

TONY GORSCHKEK

*School of Computing, Blekinge Institute of Technology
371 79 Karlskrona, Sweden
tony.gorschkek@bth.se*

Received 3 May 2010

Revised 5 April 2011

Accepted 3 July 2011

Knowledge management (KM) in software engineering and software process improvement (SPI) are challenging. Most existing KM and SPI frameworks are too expensive to deploy or do not take an organization's specific needs or knowledge into consideration. There is thus a need for scalable improvement approaches that leverage knowledge already residing in the organizations.

This paper presents the Practice Selection Framework (PSF), an Experience Factory approach, enabling lightweight experience capture and use by utilizing postmortem reviews. Experiences gathered concern performance and applicability of practices used in the organization, gained from concluded projects. Project managers use these as decision support for selecting practices to use in future projects, enabling explicit knowledge transfer across projects and the development organization as a whole. Process managers use the experiences to determine if there is potential for improvement of practices used in the organization. This framework was developed and subsequently validated in industry to get feedback on usability and usefulness from practitioners. The validation consisted of tailoring and testing the framework using real data from the organization and comparing it to current practices used in the organization to ensure that the approach meets industry needs. The results from the validation are encouraging and the participants' assessment of PSF and particularly the tailoring developed was positive.

Keywords: Postmortem review; knowledge management; software process improvement; software engineering; taxonomy; SPI.

1. Introduction

To maintain and increase competitive advantages, software organizations must continually strive to maximize the utilization of the knowledge and experience

internal to the organization, refining and improving their development practices. Process assessment and improvement are often the means by which development organizations assure and improve their processes and tools to support their specific needs. For improvement, several well-known software process improvement (SPI) frameworks exist, both prescriptive (see, e.g. CMMI [52] and SPICE [53]) and inductive [54, 40, 27]. Prescriptive, or model based, frameworks take an approach based on a set of best practices that have proven successful in other organizations. The improvements are implemented through the benchmarking of the organization to the predefined one-size-fits-most best practices. Inductive methods take their starting point in a thorough understanding of the current situation, basing improvement efforts on the issues most critical to the specific organization.

Independent of the framework chosen, there is a risk that several things will be omitted when an organization turns directly to process improvement as a way to assure and improve its processes. First, process improvement often happens as a reaction to a problem, and an SPI effort can take up to 24 months or more to realize [55]. Second, cost in terms of initiation threshold (training, tools, doing it the first time and so on) and potential risks associated with unproven practices can also be important factors [4–6]. Third, management and practitioner commitment is crucial, and change might be resisted if not very well motivated and supported by both practitioners and managers [20, 56–60]. Fourth, but not least, looking directly towards the new, e.g. introducing new practices such as use-cases, modeling practices and so on, can result in knowledge and experience already owned by the organization being lost or overlooked [38].

The Experience Factory (EF) [19] approach have been central in leveraging knowledge internal to software organizations and to learn from past success and failures in order to improve development processes. EF is an infrastructure for sharing and reusing experiences gathered in projects carried out in an organization. Traditionally, objective metrics have been used to evaluate projects and then to package and disseminate these experiences in the organization [20]. In comparison to subjective metrics, objective metrics have the advantage of not being influenced by judgment of an expert. Still, commitment and intimation threshold remains a problem as EF is a major undertaking for an organization [20, 22].

An alternative is to use postmortem reviews to gather subjective metrics in EF to lower cost and circumvent the need for only relying on objective metrics. Postmortems can have a learning effect on an individual level, team-level and organizational level. However, few studies have focused on the organizational level [61].

Postmortems introduce a new problem as even if experiences are captured and stored, they are not always reusable or even usable by their intended audience [47]. This becomes evident when looking at state-of-practice as reports of software engineers not learning from past mistakes [10, 25] and experiences owned by the organization being overlooked [25]. In addition, the effort for analyzing postmortem data to enable learning on the organizational level can be high [13].

This paper proposes an approach towards quality assurance and improvement of processes, utilizing the traditional SPI mindset, but building upon it and changing the focus to a bottom-up approach that starts with both organizational needs and organizational knowledge. The approach is presented as a framework, namely the Practice Selection Framework (PSF). PSF is an Experience Factory approach that utilizes postmortem reviews for capturing experience in order to lower the initiation threshold for improvement and learning.

PSF is a framework for sharing experiences regarding practices used in an organization and was developed in close cooperation and collaboration with a large multinational company developing software intensive products, namely Volvo Car Corporation. In PSF, experiences are stored and structured in the practice repository that contains the practices of an organization, and they are tagged with meta-data, primarily collected from the organization itself. The meta-data include information such as applicability, experiences, tips, and pros and cons associated with the practice in question. One of the overall goals is to leverage the inherent knowledge in the repository to equip projects with practices and support project managers when starting projects.

A crucial part of PSF is enabling the feedback loop. Practitioners perform a sort of mandatory postmortem, adding meta-data (or even practices) to the repository after every project. The idea is that the best and worst experiences regarding practice usage are collected in one place, improving the practice selection activity continuously as it offers iteratively improved decision support material to practitioners searching the repository for practices appropriate for their specific situation. This enables sharing practices that have been shown to lead to success in the entire organization and providing continuous evaluation of practice performance. Keeping track of the performance of practices used provides a basis for (and can be seen as) continuous process improvement.

Major benefits of this way of working are that good experiences are shared across projects and organizational specific meta-data are collected with regard to each individual practice instead of only having general one-size-fits-most information prescribed by, e.g. an SPI framework. This is not to say that traditional SPI is not relevant; rather it is an integral part of PSF. As practice experiences from projects are reported back, the usefulness and fit of practices can be judged. If the judgment in any instance is that the present practices are inadequate, a traditional SPI effort can and should be initiated but should be aimed largely at PSF, i.e. improving/adding to the collection of practices and the meta-data in the repository.

This paper presents the PSF framework and its constituents as well as a static validation [62] performed in industry. The validation take the form of a workshop aimed at tailoring and validating the concepts used in PSF, as well as collecting feedback on the framework itself. The results show that PSF is a promising approach for sharing and improving practices used in an organization.

The paper is structured as follows. Section 2 provides an overview of knowledge management for process improvement in software engineering and discusses their

characteristics and possible limitations. In Sec. 3, an overview of PSF is given and illustrated with generic examples. Section 4 presents the design of the static validation, including research questions to be answered, the context in which it was performed, and the actual design of the validation. The results from the validation is presented in Sec. 5 and discussed in Sec. 6. Finally, conclusions are presented in Sec. 7.

2. Background and Related Work

There are many methods and frameworks that take organizational learning into account. Knowledge Management in general, but also in relation to software engineering in particular, has discussed the importance of managing and utilizing knowledge and of spreading tacit and explicit knowledge in an organization [1, 2]. The recognition that practitioners themselves are the main company assets makes this even more evident. The challenges identified point towards the difficulty in valuing tacit knowledge and deciding what knowledge should be made explicit. The cost of making knowledge explicit and the use of training as one major vessel for knowledge transfer are high-cost, and accuracy can be hard to gauge: i.e. what knowledge should be made explicit, packaged and transferred? The actual transference is also a challenge. For a project manager initiating a project, the choice of what practices to use is many times based on a “gut feeling” derived from experience and tacit knowledge [2]. Using e.g. predictive or process models that use large amounts of project data as input or using simulation can support practitioners [3], but transparency and, ultimately, trust can be an issue. In addition, the cost of keeping high-quality/high-accuracy data for every project up-to-date, making it suitable for input to the prediction simulation, again becomes a problem, as high initiation and maintenance costs increase the initiation threshold of any process improvement activity [4–6].

The concept of performing postmortems upon development project conclusion is closely related to the concept of supporting organizational learning, making experiences explicit, and can be seen as vital for process improvement activities [7–9]. Performing postmortems is both practical and low cost, offering clear benefits, while it also has a low initiation threshold, i.e. suitable for organizations of all sizes. The main problem is that organizations seldom perform postmortems, even if they are in the official process charter, as pressures to start the next project overshadow the good intentions of learning. Verner and Evanco report that, of 42 projects studied, only 33% had postmortem reviews [8], even though performing postmortems was associated with the production of high quality development artifacts and the ability to manage risk more efficiently throughout the development process [8]. As a result, the same mistakes propagated across projects over time [8].

Another problem is that, even if postmortems are performed, information is seldom shared across project boundaries [10–12], resulting in situation that the only

vessel for knowledge transfer is the individual practitioner moving between projects. Methods for analyzing postmortem data for use on an organizational level exist but carry high costs [13]. The project managers, who are the driving force behind selecting the practices to be used in a specific project, might or might not be involved in the learning experience of another project. This problem is compounded by the fact that few organizations have procedures or structures in place to enable sharing of information [7, 8, 11]. This can be devastating, as good practices are highly dependent on project characteristics, i.e. what works for one project might not be suitable for another [14–18]. Relying on only word-of-mouth rumors, good practices conveyed out of context with little or no additional information can create more problems than they solve. Looking at SPI frameworks, such as CMMI, most of the data gathered during process assessment is project based, i.e. the realization phase is often the subject of study. One potential problem is that most SPI frameworks adopt a one-size-fits-all view, not only across companies but also across projects.

Experience Factory (EF) [19, 20] is very closely related to the same principles, i.e. learning and knowledge transfer. EF is an infrastructure for reusing life cycle experiences and products for software development. Experiences are collected from development projects and are packaged and stored in an experience base. Packaging entail generalizing, formalizing and tailoring the information collected to be easy to reuse. The idea is that software development projects can improve their performance through the utilization of experiences from previous projects. The classical usage of EF utilizes the Quality Improvement Paradigm (QIP) [21] for software process improvement supported by the Goal Question Metric (GQM) [21] for establishing project and corporate measurements. The instantiation of EF requires the creation of a new experience factory organization to be responsible for the analysis and packaging of the experience collected. Packaging is a difficult task in itself [22], and the cost of the EF organization is about 10% of the total organizational project budget [20]. This is not to say that EF has negative return on investment but rather that the commitment and thus the initiation threshold for such an undertaking is considerable, especially for small and medium sized organizations (SMEs). The most prominent use of EF as described above is the NASA SEL [20] (for a review of research on EF see [23]).

PSF was strongly inspired by the idea of postmortems, while aiming to remedy some of the problems presented by combining it with an EF like approach for supporting organizational learning. Building on state-of-practice, the realization that gut feeling was dominant in practice selection, PSF is intended to allow practitioners to share information and experiences associated with a given practice. In essence, by using the PSF repository, it is possible to “browse” through other colleagues’ gut feelings and experiences, while maintaining the context of a specific project. The context of the experiences gathered in the repository is maintained as project characteristics and saved in relation to the practices and experiences. Traceability to originator is also saved as it enables face-to-face communication

transferring tacit knowledge, which is crucial for keeping PSF light-weight, making it accessible to SMEs as well as larger organizations [24, 25]. As the focus of PSF is the practices, everything associated is seen as meta-data, and the main work is connected to the gathering of this meta-data from practitioners as projects are concluded.

PSF builds on the concepts of EF and related research. The main differences from the classic EF are highlighted below:

- The classic usage of EF builds on that measurements are in place to evaluate the technologies used while PSF focuses on qualitative feedback through postmortems. Using qualitative measurements enables more organizations to utilize EF as few have large measurement programs, as the instantiation can be very costly [20]. Still, using postmortems integrates feedback into the engineering work and thus assures that experiences are collected.
- In EF, experiences, processes and technologies are packaged, i.e. generalized, formalized and tailored to be easy to reuse. Packaging is a difficult task [22] and in PSF this is circumvented by using already existing packages, practices. PSF builds on the current situation, the practices used in the organization, and uses experiences to gradually refine these and transfer successful ones to other projects. In contrast to the classic use of EF, PSF stores experiences that are already analyzed in a postmortem analysis meeting. This means that there are no explicit packaging and generalization in PSF, but rather that experiences are pre-packaged as they concern some already existing practice, which it is connected to. Generalization is not necessarily sought in PSF, but rather to let projects decide what would work in their project. PSF provides projects with decision support for this decision in giving them access to practices used in the organization and experiences from them.
- EF has a separate organization for processing the information collected from the development organization. This organization is concerned with establishing measurement goals, analyzing and packing the information collected. Since there are no or few quantitative measurements and no explicit packaging in PSF, this burden is limited and there is not necessarily a need for a separate organization. Instead, PSF can be seen as a tool for process and project managers to use for choosing among and improving the practices used in the organization. This means that the use of PSF is distributed over already existing roles.
- As there is no packaging of experiences and the information in PSF are in most cases subjective (not as accurate as quantitative measurements) the focus shifts towards enabling sharing tacit knowledge to overcome the ambiguities, e.g. enabling locating persons to get more information from.

In comparison to postmortem reports, PSF adds structure to experiences captured. This is achieved by focusing the type of experiences captured, i.e. experiences related to practice performance. In addition, to facilitate locating

relevant information captured, practices are arranged according to the process model used [22, 26].

3. Practice Selection Framework — An Overview

This section gives an overview of PSF and a detailed account on how to use PSF to share practices, coupled with experiences within an organization. An overview of PSF can be found in Fig. 1. PSF is an approach for leveraging the knowledge inherent to an organization regarding practices used. People in projects often already know what practices work and what does not [27–29].

In PSF, practices are the primary driver for transferring experiences in an organization. This is motivated by the fact that practices are often pre-packaged in the form of process or method descriptions, often including templates and guidelines, which are already available. This natural state should decrease the overhead of documenting experiences. In addition, practices are a primary candidate for conveying knowledge, as most process improvement efforts revolve around introducing new practices to tackle problems divulged during process assessment. Thus, practitioners are used to practices being the main artifact of a change.

Practices together with meta-data collected from the organization are stored in a practice repository (labeled A in Fig. 1), which is shared across the organization. Meta-data include attributes describing practices (labeled A1 in Fig. 1), experiences from using the practice (labeled A2 in Fig. 1) and the characteristics of projects in

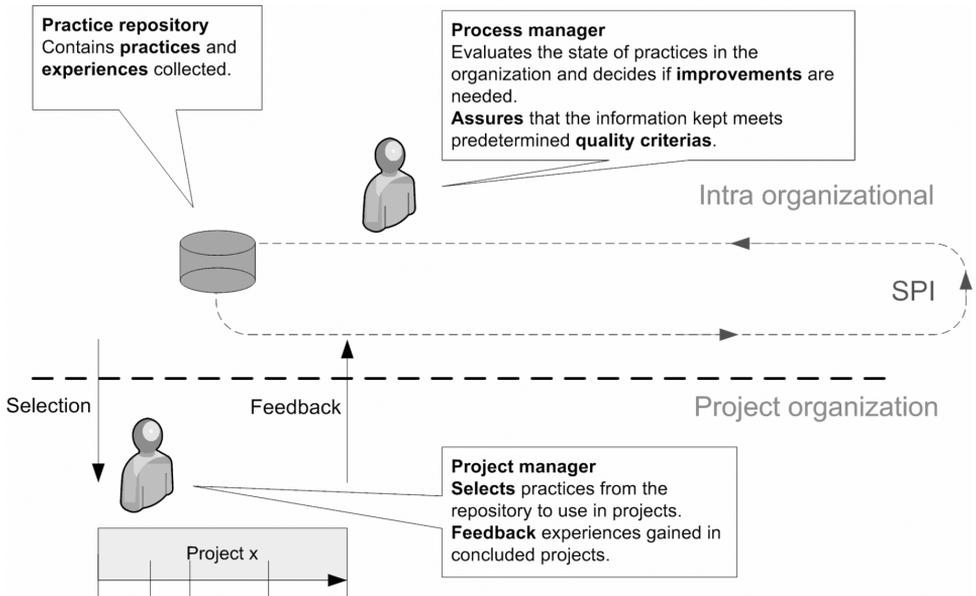


Fig. 1. Overview of the practice selection framework.

which the practice have been used (labeled A3 in Fig. 1). The practice repository and its constituents are described in Sec. 3.1 and the structure used to organize practices is discussed in Sec. 3.1.4.

Practices as well as meta-data are collected through postmortem reviews when projects conclude (labeled B in Fig. 1) which is detailed in Sec. 3.2. The meta-data include information such as applicability, experiences, tips, and pros and cons associated with the practice in question. One of the overall goals is to leverage the inherent experience stored in the repository to equip projects with practices and support project managers when starting projects (labeled C in Fig. 1). Experiences collected regarding practice performance aid other parts of the organization i.e. other projects to make informed decisions regarding adopting practices already used in the organization (see Sec. 3.3). This enables projects learning from each other in addition to continuous process assessment as experiences can indicate if there are practices that need to be improved. How to use PSF to assess practices used in the organization and support software process improvement (labeled D in Fig. 1) is described in Sec. 3.4.

Finally, different organizations looking to adopt PSF will most likely have different needs and constraints. Acknowledging that there is not a one-size that fits all organizations, Sec. 3.5 discusses how PSF can be tailored and implemented.

3.1. A — *Practice repository*

Practices and meta-data collected from the organization are stored in the practice repository (Experience base in EF). As the focus of PSF is the practices, everything associated is seen as meta-data. Meta-data take the form of characteristics of projects where practices have been used, experiences from using practices as well as attributes of the practice itself e.g. pre-requisites for usage (see Fig. 2). The characteristics of projects set the context for the experiences. This means that project managers that use the information in the repository can evaluate how relevant the experience is for his/her project [21]. It is up to the organization adopting PSF to decide how the entities are tailored with respect to what information should be collected for each.

The next sub-sections give an account of information that can be used for each of the entities in the practice repository. In addition, to facilitate easier finding practices used in the organization, the repository needs to be structured. How the repository can be structured is discussed in Sec. 3.1.4.

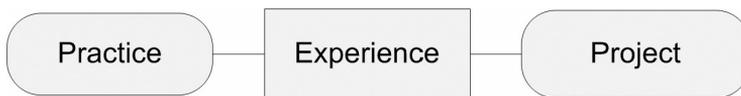


Fig. 2. Entities in the practice repository.

3.1.1. A1 — Practice

Keeping descriptions of practices is necessary to be able to facilitate their adoption in other projects. To achieve this, the definition of a practice must be specific to the organization, e.g. if use cases [30] are a practice used in the organization, it is the organization’s definition of a use case and its experience related to it that are important to share within the organization. The definition of the practice in the literature is irrelevant as it is not the organization’s own experience. Guiding practice documentation by stipulating what attributes are important ensures a minimum level of information completeness and uniformity in the repository, making it easier to use. A number of attributes suitable for describing practices are summarized in Table 1.

Practices are often described in the form of process descriptions, templates used, examples, checklists and guidelines. In addition to describing the practice, certain attributes have been included to support practitioners in selecting practices from the repository. Purpose provides an initial screening of practices when selecting from the repository through abstraction of what the practice is used for. If the practice’s purpose does not fit the needs of the project, the search can quickly move on to the next practice while, if the purpose fits, a more thorough evaluation of the practice can be made. Practice dependencies capture what is needed in terms of other practices to take advantage of a practice at hand, e.g. capitalizing on the use of an object oriented programming language is dependent on having an object oriented design in place. Process connection indicates in what part of the development process the practice is meant to be used (see Sec. 3.1.4). Finally, prerequisites are used to describe what in addition to other practices is needed. This includes

Table 1. Practice attributes.

Practice	
Attribute	Description
Title	A descriptive title for the practice.
Purpose	Purpose should be a concisely capture what the practice is used for. This attribute is used to screen practices quickly.
Description	The description should convey how the practice is used. In addition, other sources can be used to facilitate usage such as: <ul style="list-style-type: none"> ● Templates ● Examples ● Checklists
Process connection	Process connection traces the practice to the governing process. The process connection shows where in the development the practice is meant to be used.
Dependencies	Dependencies captures what other practice are needed to make this practice work.
Pre-requisites	Pre-requisites describes all other things that need to be in place to use the practice. This includes roles, tools etc.

roles, life cycle models etc. that have to be in place to make use of a practice. An example of a prerequisite is the need of having the role tester present if test-case driven inspections are to be used [31].

3.1.2. *A2 — Experience*

Postmortem analysis is used to assess practices when projects are concluded (see Sec. 3.2). The experiences collected from the analysis are documented in PSF to enable other projects to learn from projects' failures and successes and thus pick up practices that have been shown to be successful elsewhere in the organization. To control what information is collected, attributes in PSF are tailored to fit the information prerequisites of the organization. In addition, rules control the extent to which experiences are collected. The rules stipulate what attributes is mandatory for projects to supply and the required documentation. Ideally, experiences are rooted in quantitative measures, e.g. effectiveness and efficiency of the practice in the project. However, as measurement programs are expensive to set up and sustain, the attributes presented here focuses on a few qualitative measures presented in Table 2.

PSF supports both codification and personalization strategy of knowledge management. Codification is achieved by describing practices, projects and experiences and storing them in the practice repository. In addition, PSF enable personalization by traceability to the person that has experience from using the practice as an experience (named "Responsible" in Table 2). This is important as the information related to practices often is incomplete or does not reflect their actual use. This is in part because extensive documentation is considered too resource demanding or that the knowledge is hard to make explicit. To manage this,

Table 2. Experience attributes.

Experience	
Attribute	Description
Traceability to project	Provides a link to the project in which the experience has been captured.
Traceability to practice	Provides a link to the practice that the experience relates to.
Responsible	Provides traceability to the one responsible for the practice in this particular project. This attribute can be used to keep track of persons with experience from using the practice within the organization.
Performance	Performance reflects how the practice have performed in this project. Performance can be actual measures of effectiveness or efficiency or a subjective measure of how well project participants think that the practice have fulfilled its purpose e.g. on a 100 point scale.
Cost/bang	Essentially bang-for-buck or return on investment for the practice. This reflects how the participants in the project perceive the benefits from the practice weights against the cost of using it.
Recommendations	Recommendations are a way to share additional experiences from using the practice. Recommendations can for example be concrete advice on how to improve the practice.

organizations can opt for a lightweight level of documentation and instead use personalization of the knowledge. This can be made possible by providing traceability to experts owning tacit knowledge, e.g. “Charlie knows use-cases, talk to him”. This makes it possible for practices and experiences to be personalized, and PSF can be used as a catalog with links to experts. This aids in transferring tacit knowledge, which compensates for shortcomings in documentation. In addition, this can be used to keep track of experts or employees with particular experiences in the organization, which can be used when staffing projects or identifying resources to be used in tutoring and training new employees or when a project adopts new practices.

All the attributes in Table 2 focus on capturing subjective experiences of practice use and initiation threshold. Collecting experiences of the initiation threshold, such as risk and initiation cost, alleviate adopting practices in other projects. Attributes related to practice use include performance, who is responsible, cost/bang, pros and cons. Performance consists of an evaluation made by practitioners of how the practice has performed, which is used by others as an indication of whether it is worth adopting. Even though a practice is perceived as being advantageous, certain aspects of the practice might bring negative effects or warrant changes in the practice. The recommendations attribute are a way for practitioners to convey these aspects of practice usage experienced. Cost/bang is a way to capture the relative cost effectiveness of a practice. If, for example, a requirements specification practice is to be evaluated, the cost/bang captures how expensive each requirement was to specify on average using the practice in question.

3.1.3. *A3 — Project*

The context in which the experience is acquired is saved to enable project managers to gauge the relevance of the experience in relation to their own challenges and projects. For experiences to be usable in a particular project they need to be valid in that context, e.g. an experience derived in a small research project might not be applicable in a large production project. To evaluate the transferability of experiences, the characteristics of the project in which the experience was derived are stored in connection with the experiences. Project characteristics are also used to catalyze the diffusion of practices in the organization, as other projects can see what practices are used in other projects together with how those projects compare to the one at hand. A summary of possible project characteristics is given in Table 3.

Project characteristics should be tailored to the organization at hand to avoid keeping characteristics that are not considered important for the organization or the ones which are redundant in nature, e.g. keeping development mode as a attribute when all projects executed in the organization are bespoke projects. This builds a frame of reference for project managers reasoning regarding the suitability of a practice for projects. Keeping experiences and context information in the form of project characteristics is important, as practitioners consider case studies, field studies and legacy data as most valuable when selecting practices [32], where it is

Table 3. Project attributes.

Project	
Attribute	Description
Lead-time	Captures the lead-time for the project.
Personnel	The number of people involved in the project.
Project size in man-hours	The resources in man-hours used in the project.
Requirements stability	The volatility of the requirements. Captures some of the conditions (level of change) under which the project was performed.
Development mode	The development mode of the project including for example customer contract, innovative, market driven etc.
Development type	The project type including for example new product development, maintenance, evolution of existing product etc.

vital to be able to evaluate whether the context of the experience at hand is applicable in a future project.

3.1.4. *A4 — Support structure of repository*

Structuring the repository by decomposing the practices into several levels of abstraction helps to keep an inventory of what practices are present in different parts of the process. In addition, this eases the effort required to search for and increases the probability of successful identification of practices when selecting from the repository [22, 26]. Practices are decomposed to structure the information into several process abstraction levels instead of flattening the information by only providing one level in the structure of the repository. In Fig. 3, a decomposition based on CMMI [33] comprising four levels of abstraction is shown. The top three levels structuring the process consist of Process area, Specific goals and Specific practice. These levels are used to structure the practices used in the organization, which are the bottom level of abstraction. The top level, the Process area, divides

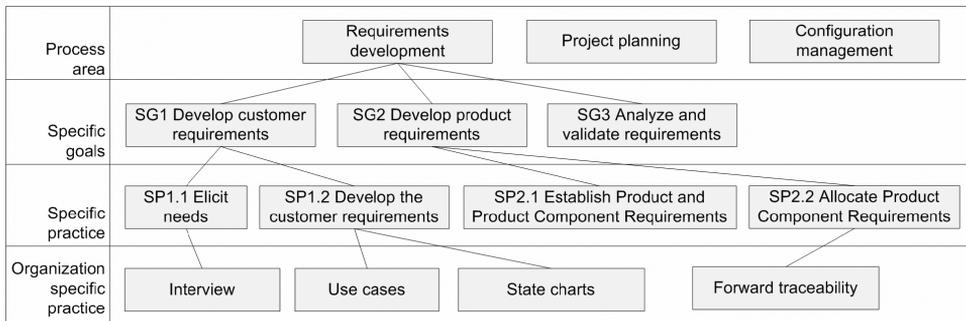


Fig. 3. Example of practice decomposition.

the process into clusters of related practices. CMMI comprises 22 process areas of which three are shown in the top part of Fig. 3. A project manager searching for a practice related to requirements engineering would only need to consider the practices found under the respective abstraction. The next abstraction level further refine the division into Specific goals needed to satisfy the process area. Figure 3 shows the goals needed to satisfy the Requirements development process area, which include Develop customer requirements, Develop product requirements, and Analyze and validate requirements. The next abstraction level encompasses practices used to achieve the goals on the level above. Figure 3 shows an excerpt of practices used to fulfill the goals for the Requirements development process area. The last level holds the concrete practices used in the organization that constitute one way of performing these practices. Developing customer requirements can for instance be performed with a wide array of practices for specifying requirements (see Fig. 3) e.g. Use cases and State-charts. How the decomposition is arranged in terms of how many levels and what these consist of is specific to the organization and needs to be tailored when instantiating PSF. Organizations that already have process descriptions or guides can use these to connect practices to, or connect experience attributes to practices already documented. Structuring the repository according to a process has shown to be beneficial in other knowledge management efforts. The tailoring of abstraction levels can be done in a way similar to that when working with requirements on different abstraction levels [34, 35].

3.2. B — Feedback

Postmortems reviews are a practical method for learning from concluded software projects [36]. Postmortems enable projects reflecting on lessons learnt, what worked and what can be improved in the future. In PSF, a postmortem is made when a project has concluded to assess the performance and uncover what practices were used successfully and which constitute an improvement potential. The results are documented in the practice repository according to the rules and attributes established. This makes experiences regarding practice performance transcend project borders as the repository is shared within the organization. It is up to the organization how to organize the postmortem regarding only supplying experiences for the most successful and unsuccessful practices to all used in the project. Only providing experiences of practices featuring some particular characteristic offers a less resource intensive option to documenting experiences related to all practices in the project. Using postmortems for feedback of experiences integrates knowledge management into the engineering work and enables management to assure that a minimum level of experiences is collected. This is important, as it is often hard to get workers to share their experience [37]. An example of feeding back experiences is given in Fig. 4 where two practices, use cases and a cost estimation practice has been used in a maintenance project. In the example it can be seen that use cases have been perceived as working well while the cost estimation practice could need improvement.

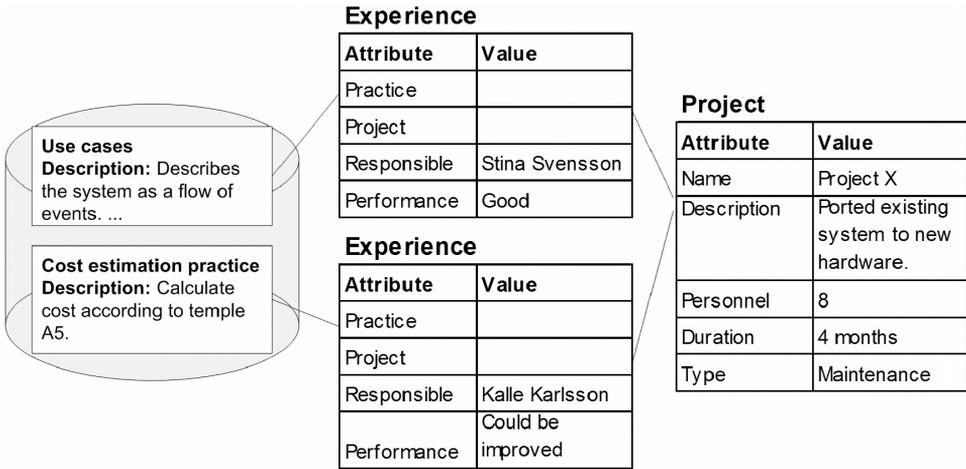


Fig. 4. Feedback of experience.

3.3. C — Selection of practices

Selection involves utilizing the information stored in the practice repository to equip a project with practices. This is usually carried out at project start-up but can also be used when a project needs new practices, e.g. mid-project. In an organization where projects and processes are diverse, the project manager who has knowledge about the project in question preferably performs this task. To support project managers in selecting practices, meta-data including experience and the context in which they were derived are used as decision support. PSF offers different types of decision support material for project managers looking to adopt practices. First, the practice itself together with its meta-data e.g. purpose, pre-requisites etc. offer an initial description of the practice itself. Second, meta-data concerning projects can show if and in what projects the practice have been used before and the characteristics of these. Finally, experience collected from these projects indicate how the practice has performed. To get more information, experiences offer traceability to persons that have experience from using the practice. Example 1 illustrates how practice selection can be performed.

Example 1. An example of practice selection.

Previous projects that the project manager has participated in have all received a complete requirements specification from the client. Now, a new project includes finding out and specifying the requirements for the client. The project manager thus need to, among other things, adopt a new practice for writing down the requirements for the system. Browsing through the repository (see Fig. 5) he finds the *Develop customer requirements*

(Continued)

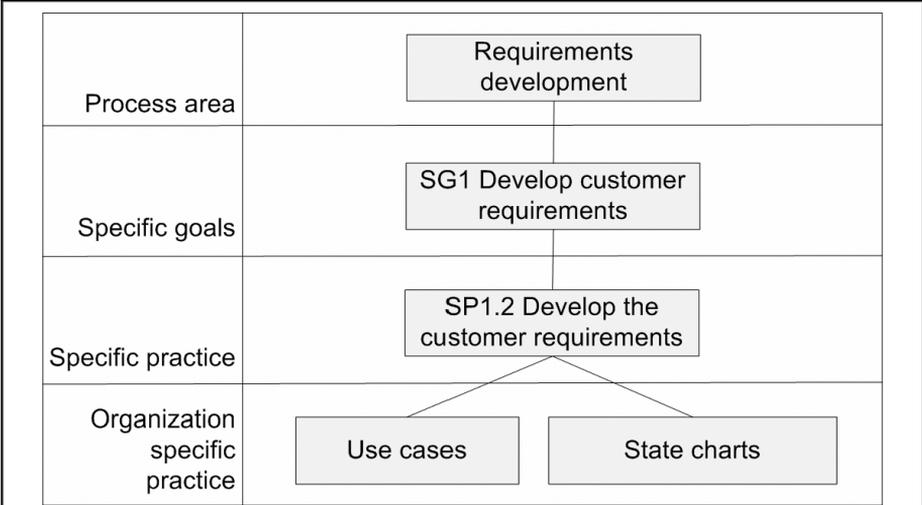


Fig. 5. Except of the practice repository.

practice which essentially entail writing down requirements, something that the organization in this example has experience from using Use cases and State charts. The project manager thus has the choice of trying one or both of these practices or try something not previously used in the organization. In order to make an informed decision he examines the experiences collected from previous projects shown in Fig. 6. Here he can see that Use cases have

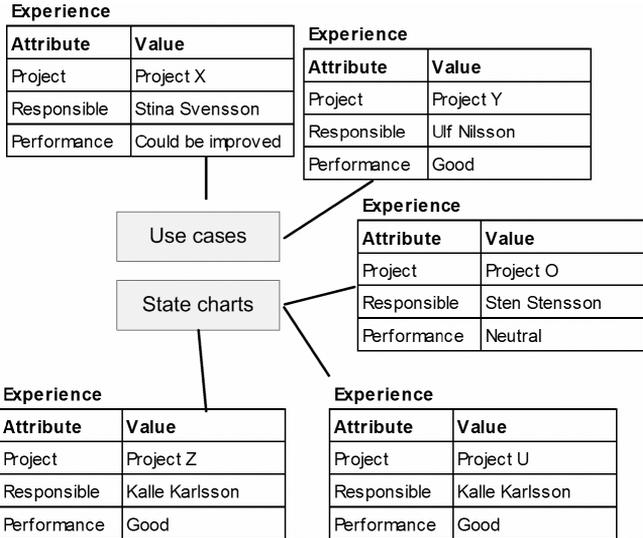


Fig. 6. Experiences stored in the practice repository.

(Continued)

opinions from projects while State charts have mostly received favorable received mixed opinions. To make the decision he compares the projects that have previously used the practices to the one in hand and sees that projects X, O and U are comparable to the project in hand, i.e. similar project characteristics and similar products. As State charts seem to have performed better than Use cases for similar projects he decides that State charts are a candidate to use in the new project and proceeds by talking to persons who have hands-on experience in using the practice, which is also stored in the repository. He might also use this information to try to staff the project with staff that have experience from specifying requirements using the practices chosen.

3.4. D — *Identification of improvement potential*

In addition to alleviating sharing practices within the organization, PSF also supports process improvement as the repository provides a snapshot of how the organizations practices perform. This can be used to assess the need for improvement in the organization. The issues in Table 4 summarize how PSF is used to unveil a need for improvement. An improvement effort is initiated as a response to a careful consideration of current practices and associated experiences. The initiation of an external SPI activity can be mapped to three activities in PSF. Selection of practices from the repository not only supports project managers choosing working practices

Table 4. Process issues with relation to PSF.

Relation to PSF	Description	Action
Missing practice	There are no practices present in the repository matching the needs of the project.	Use SPI efforts to identify and include new practices in the repository.
Practice selection failure	The selection of practices performed has resulted in choosing the “wrong” practice for the project at hand. Projects abandon the practice.	Even though the practice has not been used in the project, the negative experience should be fed back into the practice repository to warn others making the same selection.
Practice deficient	A practice has received negative feedback indicating that there is a potential for improvement in selecting another practice for the task.	Inventory the repository for other suitable practices. If none is found, initiate an SPI effort to rectify the deficiency.
External issues	Issues negatively influencing project performance that do not relate directly to PSF or practices contained therein.	Initiate an SPI effort to identify and rectify the problem. If the improvement involves new practices, include them in the repository.

for a project but also doubles as an inventory of practices and experiences owned by the organization. A need for improvement is detected either as a lack of practices suiting the need of the project or as suitable practices having received mostly negative feedback. Should an adequate practice (or solution to a problem) be missing from the repository, an external process improvement activity should be initiated. The selection of an SPI framework to use for improvement is not relevant; rather, it is up to the organization to decide and a task to be solved. The important thing is that the SPI effort be focused on the practice repository. This implies that the process assessment originates with the practices in the repository, and any change or addition to the process (for example adding a new practice) is mapped and included into the repository, which in effect causes the improvement effort to transcend project boundaries making the results accessible to the whole organization.

However, relying only on the experience originating from inside the organization can mean in the long run that the organization becomes increasingly good at using obsolete practices [38]. To limit this risk, and for the repository to be up-to-date and optimized, regular SPI efforts should also be performed. These can be seen as maintenance overhauls and can be triggered by, for example, practices receiving mostly negative experiences or by a general conception that the process could be improved. In an optimal case this is based on quantitative data obtained from an extensive measurement program collecting metrics on, for example, defects (quality) and time and cost per production unit (efficiency). In reality very few organizations have extensive measurement programs as the cost of keeping and updating detailed records and collecting metrics for analysis is very high [20, 39]. In practice, maintenance can be initiated when the need arises, but it could be prudent to consider it as ongoing quality assurance.

3.5. Tailoring and implementing PSF

PSF is a framework that needs tailoring to fit an organization's needs, conditions and process. This means that the parts presented thus far need to be adapted to reflect needs in the organization. This includes choosing what attributes to use in the practice repository to describe practices, experiences and projects. In addition, how selection of practices, feedback of experience and assessment of practices used need to be established and responsibilities assigned to roles in the organization. An initial tailoring can be performed as described in the static validation presented in Sec. 4.

When tailored, PSF is introduced in the organization using an appropriate process assessment methodology such as iFLAP [27, 40] or SCAMPI [41]. During an initial process assessment of the organization, practices already in use are documented and discussed. This is the initial documentation of the repository. In addition, it is important to collect meta-data in relation to the practices discussed, collecting experiences and, especially, noting inadequacies. Initially seeding the

repository is important [22] as a repository that have no usable information is likely to be abandoned and provides no incentive to supply new information.

4. Static Validation of PSF in Industry

Static validation of PSF was performed at Volvo Car Corporation to get initial feedback on how the framework addresses needs in industry. This section presents the design of the static validation. First, the research methodology is introduced in Sec. 4.1. The context in which the study is conducted and the participants are presented in Secs. 4.2 and 4.3 respectively. Section 4.4 describes the workshop used to tailor and introduce the participants to PSF while Sec. 4.4.1 introduces the questionnaire used to measure the participants view of how PSF would work at VCC. Finally, the research questions aimed to be answered in the validation are presented in Sec. 4.5.

4.1. Research methodology

The study presented in this paper is part of a technology transfer cycle consisting of innovation, static validation, dynamic validation and finally release of the technology for wider use. This general model for technology transfer developed for software engineering is shown in Fig. 7 (see Gorschek *et al.* [42, 43]).

The innovation step is where the idea is born and developed into a technology. The technology is then tested and validated in different settings, from laboratory to industry, utilizing different types of research methods as needed. Static validation of technologies often involves experimentation in academia to investigate the basic concepts before the technology is tested in industry. Static validations in industry sort out teething problems related to transferring a technology to industry and provide a

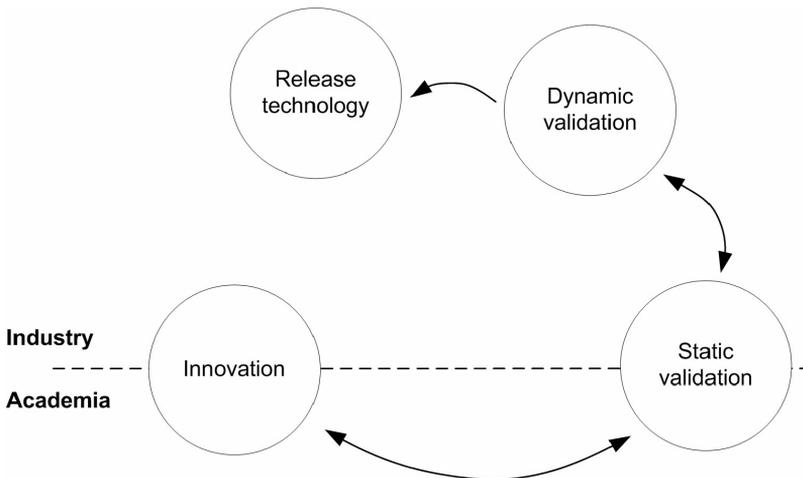


Fig. 7. Technology transfer process (adapted from [42]).

way to get initial feedback on scalability and usability without incurring the risks of testing it in production projects. Given the results of the static validation, one can either move back to the innovation step and refine the idea or move on to dynamic validation to test the idea in a real life setting. Dynamic validation is carried out in case studies in either “real” software projects or a smaller pilot project aimed at evaluating the technology in question. The last step in the technology transfer process is to release the technology for wider use when it has been shown to be useful and usable.

This paper presents a static validation of PSF in industry. The validation is set up as a workshop (described in Sec. 4.4) aimed at tailoring PSF to fit the organization, while also validating the concepts used in PSF. During the workshop PSF is tailored to fit the organization and usage of PSF is simulated by applying it to examples from the organization. Validation is achieved through collecting the workshop participants’ opinion on how this tailoring would work in a real setting. This gives initial feedback on what concepts work and what needs to be changed before actual piloting and thus build commitment to the solution.

4.2. Research context

The evaluation presented was conducted at the Swedish automotive manufacturer Volvo Car Corporation (VCC) currently owned by Ford Motor Company. Process descriptions at VCC are documented in a Business Management System (BMS). Instructions in the BMS guides what to do in projects and technologies are thus already chosen when commencing a project. Differences between projects stem from differences in system characteristics, i.e. safety critical systems need additional practices. Concerning knowledge management and retaining experiences gained from projects, postmortems are carried out on project conclusion and experiences documented in white books and provide feedback to instructions in the BMS. Experiences from using practices are also disseminated by face-to-face communication and rotation of workforce. In addition, cross-function teams and reviews are also used to utilize the knowledge and experiences. Process documentation and knowledge management activities at VCC are established and successful albeit a potential for improvement was uncovered in the questionnaire used during the workshop in this evaluation. The improvement potential relate to how to effectively communicate and use the experiences gained from concluded projects. For more background information on the case setting see [44, 45].

4.3. Participants

The workshop was run with four participants selected based on expert knowledge in relation to the organization and working practices. Three are employed at VCC and the fourth participant is a PhD student tied to VCC with extensive knowledge from research projects carried out in the organization.

4.4. Workshop description

The static validation is organized as a workshop with participants all actively participating. During the workshop, PSF was tailored to fit the organization, and real examples of practices, projects and experiences from the organization were used as the main input. In essence, this was equivalent to actual use of PSF.

At the end of the workshop, a questionnaire designed to measure the participants view of PSF was filled in (see Sec. 4.4.1). The workshop consists of six steps described below. All the steps are driven by the participants' knowledge of how work is carried out in the organization. This means that all examples used during the workshop comes from real projects, experiences and practices.

1. Overview of PSF

First a general introduction to PSF describing the goals and purpose of the framework is given.

2. Definition of practice and relation to used processes

The second step in the workshop aims at establishing an organization specific definition of a practice. This is important so that all participants have the same terminology and perception of what is being discussed.

3. Choosing what meta-data to collect

What attributes to collect for the entities in the practice repository is then established. To avoid missing important attributes or choosing ones that are redundant or not relevant, these steps are example-driven. For each of the entities, real examples from the organization are used to test the selected attributes. This involves specifying practices, experiences and projects according to the attributes chosen.

3.1. Practice

3.2. Project

3.3. Experience

4. Using PSF for selection and improvement

When all attributes are established, more examples, including all entities are developed. During this step, the participants are asked to reflect and discuss how the tailoring would perform with regard to usage, i.e. finding improvement potential and supporting practice selection. The tailoring is also refined during this step to reflect issues uncovered.

5. PSF support structure

The roles and responsibilities as well as initial ideas on how to organize the feedback of experiences are discussed on the fifth step of the workshop. In addition, the structure used to organize practices in the repository as well as how to connect it to processes already used in the organization is established.

6. Questionnaire

The last step of the workshop consists of a questionnaire aimed to measure the participants' view of PSF in this particular organization. The questionnaire is described in Sec. 4.4.1.

4.4.1. Questionnaire

A questionnaire is used at the end of the workshop to measure the participants' views on the tailoring of PSF developed. A summary of the questionnaire can be found in Appendix A. The questionnaire is divided into two parts. The first part asks for information on the current situation relevant to PSF. This is used as a benchmark for the answers given in respect to PSF. The next part collect opinions on how the participants think that this situation would change if PSF, as tailored during the workshop, were implemented all out. Most questions are designed to be answered on the scales shown in Fig. 8. The figure shows how the views are converted into numbers. For example, if the subject thinks that there will be “less” effort this answer is given the value -2 . The same goes for the other scales, if a subject thinks that the quality will be “better” the answer is converted into a 2, and 4 for “much better”, and so on. For the first part of the questionnaire, the participants are asked to give point estimates while on the second part they are asked to provide three estimates, worst case, likely case and best case scenarios. The worst and best case are meant to capture the real case with 90% confidence. This makes it possible to assess the level of uncertainty the subjects have on the particular question.

4.5. Research questions

This section presents the research questions that the validation addresses. Each research question is listed in Table 5 with a description/motivation. The research questions drive the design of the validation and a mapping of the research questions

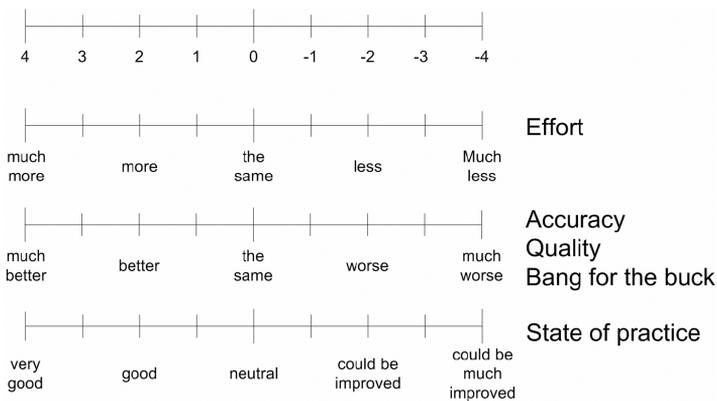


Fig. 8. Conversion of scales used in questionnaire.

Table 5. Research questions.

Research question	Description	WS	Mapping to	
			Part 1	Part 2
(RQ 1) How can PSF be tailored to fit an organization.	To enable successful technology transfer and knowledge management the organization need be able to tailor PSF to fit their needs and constraints.	All	—	All
(RQ 2) Does using PSF improve the quality of the documentation of practices, projects and experiences?	PSF supports both codification and personalization. For successful codification the quality of the resulting documentation need to good-enough to use for sharing and improving practices.	3.1, 3.2, 3.3	7, 8, 11, 12	2.2, 3.2, 4.2
(RQ 3) Is PSF easy and intuitive to use?	This research question concerns the usability of PSF. If PSF is not useable, there is a significant risk of abandoning the approach. Different aspects of usability are detailed in the sub-research questions.	—	—	—
(RQ 3.1) Does PSF improve on practices related to finding out and evaluating practices used in the organization?	To enable sharing and improving practices used in the organization, practitioners need to be able to find suitable practices and evaluate these.	—	9, 10, 13, 14, 15, 16, 18	5.2, 6.2, 7.2
(RQ 3.2) What decision support material in PSF is most important?	PSF provides different decision support material used in improvement and evaluation. This research question concerns what aspects of PSF are most valuable for supporting decisions.	3.1, 3.2, 3.3	—	8
(RQ 3.3) How is the effort for using PSF perceived?	For PSF to be easy to use, the effort for using it cannot be deterring for users.	4	—	2.1, 3.1, 4.1, 5.1, 6.1, 7.1
(RQ 3.4) How is the initiation threshold to using PSF perceived?	To successfully implement PSF, the cost of implementation and for starting using the approach cannot be perceived as a hinder. This would risk failing the implementation.	All	—	1

Table 5. (Continued)

Research question	Description	WS	Mapping to	
			Part 1	Questionnaire Part 2
(RQ 4) Is the decision material provided by PSF, experiences and project information collected, suitable for decision making regarding evaluation and adoption?	For PSF to be suitable for making decisions regarding adopting and improving practices used in the organization, the benefits of using the approach need to outweigh the cost of using it. This research question thus essentially concerns the usefulness, bang-for-buck, of using PSF. Different aspects of bang-for-buck for PSF is detailed in the sub-research questions.	—	—	—
(RQ 4.1) Is gathering and structuring practices in one place, a repository, useful for finding out what practices are used in the organization?	Does the benefit of being able to find practice used in the organization outweigh the cost of establishing and maintaining a practice repository?	5	9, 10	9.1
(RQ 4.2) Are experiences collected from a project useful material to evaluate the project?	For experiences to be useful as decision support material for evaluating a project, the costs of collecting and analyzing them need to be lower than the expected benefits.	3.2	15, 16	9.2
(RQ 4.3) Are experiences and project information collected from the organization useful material for evaluating practices used in the organization to identify potential improvements?	For experiences and project information to be useful as decision support material for identifying improvements, the costs of collecting and analyzing them need to be lower than the expected benefits.	3.1, 3.2, 3.3, 4	13, 14, 15, 16	9.4
(RQ 4.4) Are experiences and project information collected useful material for evaluating practices used in other project to decide on adoption?	For experiences and project information to be useful as decision support material for evaluating practices for adoption, the costs of collecting and analyzing them need to be lower than the expected benefits.	3.1, 3.2, 3.3, 4	9, 10, 13, 14, 15, 16, 17, 18	9.3

to steps in the workshop (see Sec. 4.4) and questions in the questionnaire (see Sec. 4.4.1) are given in Table 5. For example, it can be seen in Table 5 that in order to answer Research Question 2, regarding the quality of documentation, three steps in the workshop are performed, Steps 3.1–3.3, all related to tailoring the practice repository. In addition, the questionnaire is set up to inquire about quality of the current documentation, questions 7, 8, 11 and 12, and how this is expected to change when introducing PSF, questions 2.2, 3.2 and 4.2.

4.6. *Validity*

This section discusses the threats to the validity of the study described in this paper. As described by Wohlin *et al.* [46] validity can be discussed in terms of construct, internal, external and conclusion validity.

4.6.1. *External validity*

The resulting tailoring developed during the workshop cannot be generalized to other environments. Other organizations may face different needs and have other constraints. However, this is not a threat to the validity as generalizing the actual tailoring is not the objective of the study. The generalization that is important to be able to make here is for PSF and the workshop used for the initial tailoring of PSF. As the study only includes one organization and a few participants the ability to generalize the results are limited. However, this is an initial study and the results warrants further investigation to confirm or dismiss the results presented.

4.6.2. *Conclusion validity*

The largest threat to conclusion validity in this study is to ensure reliability of the data gathered. For the workshop, the threat concern eliciting all participants views instead of an obfuscated view or only the view of one strong person. During the workshop, every participant actively participated in all steps and discussions. Even though there was not always an immediate consensus on issues, discussions usually resolved these and thus this threat is considered limited.

The questionnaire used to measure the participants' view of PSF also poses a threat to the reliability of measurements. To limit this threat the questionnaire was extensively reviewed prior to use to ensure that there were no erroneous or leading questions. The questions used are also derived from the research questions, which should also limit this threat.

The participants in the workshop pose a threat to validity as there are few of them and might only represent one view of PSF. However, the participants from VCC are all experts with good knowledge of how development is carried out and the state of knowledge management within the organization and the PhD student have extensive knowledge of working practices used and the organization. This threat is thus limited but the number of participants is still too small to draw any general conclusions.

4.6.3. *Internal validity*

Participants in the workshop not expressing their real opinions, because they feel restricted by the workshop being audio taped and the tailoring and examples used in the workshop photo documented, is a threat to the internal validity. The participants were assured anonymity and the researchers assured not to divulge any sensitive information in publications, which limits this threat.

4.6.4. *Construct validity*

Only relying on one single data source is a threat to construct validity. The results presented for the PSF validation relies on both data gathered during the workshop and from the questionnaire. Relying on several data sources should limit this threat to validity.

5. Results

The workshop ran for an entire working day. This section presents the results from the workshop answers the research questions posed in Sec. 4.5. First, the tailoring developed at VCC is presented and exemplified in Sec. 5.1 and then answers to the research questions are given in the subsequent sections structured according to the presentation in Sec. 4.5.

5.1. *Tailoring of the practice repository at VCC*

Tailoring is important for knowledge management and process improvement [47] as there is not a one-size that fits the needs and restrictions of all organizations. The PSF workshop tailoring entails choosing what information (meta-data) to collect, how to collect it and how to use it. To try the tailoring, examples of practices, projects and experiences from the organization is used to illustrate and try out actual use of PSF. Deciding what meta-data to collect at VCC during the workshop often started with an initial proposal, later refined as examples were used. The next sections presents the tailoring of the resulting practice repository developed during the workshop. Section 5.1.1 presents the tailoring of attributes used to describe practices, Sec. 5.1.2 projects and finally Sec. 5.1.3 describes the attributes chosen to capture experiences. In addition, Sec. 5.1.4 presents an example from the workshop, where the tailoring has been used to describe a practice and an experience according to the tailoring developed.

5.1.1. *Tailoring of practices*

At the beginning of the workshop, it was recognized that practices, as described in the generic version of PSF, correspond to instructions in the Business Management System (BMS) used at VCC to some degree. The difference is that instructions are more generic, a description of what needs to be done, while practices are more

concrete, i.e. how things are done. It was decided that keeping traceability to the BMS and adding some practice specific description would suffice for using PSF. An example of this from the workshop is the “MISRA C compliance” practice which is an activity performed to check if C code adhere to MISRA C [48]. This can be done manually or by using a tool, which means that there are two different practices that can be used for this activity. The practices thus have different values for the attributes. Using a tool scales well and thus have high return on investment, however, the initiation threshold is higher than using a manual check if you do not already have the tool (procurement time, license cost, training, tool setup, etc.). Table 6 gives an overview of attributes considered important when describing practices at VCC.

Table 6. Practice attributes.

Practice	
Attribute	Description
Title	A descriptive title for the practice.
Purpose	A short concise description of the purpose of the practice.
Short description	A short description of the practice. The full description can be found in the BMS. Additionally, specifics for the practice not described in the BMS are described here.
Source reference	Source reference provides traceability where more information can be found about the practice. This includes traceability to the original description in the BMS, literature and courses.
BMS instruction	Traceability to the BMS instruction describing the practice.
Literature	Traceability to selected literature concerning the practice.
Course	A list of internal and external courses that are given concerning this practice.
Process connection	Process connection indicates where in the development process the practice is meant to be used.
Owner	Owner is the person responsible for maintaining the practice.
Prerequisites	Prerequisites lists everything that need to be in place for using the practice.
Output	Output indicates what kind of result is produced by using the practice. Output can be a decision, analysis or document.
Required competence	Required competence lists the needed competence for successfully using the practice. Competence can be experience, knowledge, courses or certificates.
Tool support	This attribute indicates the tool support the organization has available for this practice.
Initiation threshold	Initiation threshold essentially captures the time to return on investment. This is given on a subjective scale ranging from very low to very high. This attribute indicates how easy it is to start using the practice.
Estimated return on investment	Estimated return on investment gives an estimate of the relationship between benefit and cost for the practice. This is given on a subjective scale ranging from very low to very high.
Pro	Pros indicates the generally accepted positive aspects of the practice in the organization.
Cons	Cons captures the generally accepted negative aspects of the practice in the organization.

5.1.2. Tailoring of projects

To characterize projects, attributes describing the size, as well as the level of change, in the project were chosen. The size is described using the number of personnel working in the project, the man hours used in the project as well as the lead time of the project. Level of change captures what type of project is carried out. Low level of change implies a maintenance project with little added functionality while high level of change implies new development projects. To characterize the outcome of projects four attributes were chosen, cost, time and quality deviation from plan and content fulfillment. To characterize the changes in pre-requisites during the project it was decided to add an attribute to capture the requirements stability. An overview of attributes to characterize projects at VCC is given in Table 7.

5.1.3. Tailoring of experiences

In PSF, project postmortems are performed on project conclusion to evaluate the performance of practices used in the project. The results are documented in the practice repository as experiences. An overview of attributes capturing experiences in at VCC is shown in Table 8. Experiences in PSF are used to inform other projects of how a specific practice (or combination) has performed. The goal is to aid project stakeholders in deciding what practices to adopt to facilitate

Table 7. Project attributes.

Project	
Attribute	Description
Name	The project name.
Description	A short description of the project.
Lead time	The projects actual lead time.
Staffing	How the project was staffed. This attribute is divided into the two attributes given below.
Personnel size	The number of personnel that was involved in the project.
Man hours	The number of man hours actually used in the project.
Level of change	Level of change indicates the type of project. A level of change of 1 indicates a maintenance project while a level of change of 6 indicates that the project was concerned with only new development.
White book	Provides traceability to white books documenting experiences and outcomes for the project. The attribute is optional and only used if a white book exists.
Supplier	Indicates if and what supplier was used in the project.
HW/SW or mix	Indicates if the product developed in the project is pure software, hardware or a mix of both.
Cost deviation	Capture the deviation between planned and actual cost for the project.
Time deviation	Capture the deviation between planned and actual time for the project.
Content fulfillment	Capture the deviation between planned and actual content for the project.
Quality deviation	Capture the deviation between planned and actual quality for the project.
Requirements stability	Captures the volatility of requirements in the project.
Comment	The comment field is used to supply additional details and explanations to why the classification looks like it does.

Table 8. Experience attributes.

Experience	
Attribute	Description
Traceability to practice	Traceability to the practice which the experience concern.
Traceability to project	Traceability to the project in which the experience was collected.
Person responsible	The person in the project responsible for the practice.
Organizational belonging	The organizational unit the person responsible belong to.
Return on investment	The relationship between benefit and cost as perceived in the project. The attribute is scored on a subjective scale ranging from very high to very low.
Performance	How the practice has performed in the project. The performance attribute is divided into the three attributes presented below.
Success	Success of the practice unveil how well the practice has fulfilled its purpose. The success is scored on a 100 point subjective scale where 50 indicate average performance.
Cost	Cost is all fixed costs associated with using the practice in this project. This include license costs etc. incurred on the project.
Time	How many hours have been devoted to using the practice in this project.
Pro	The advantages from using the practice as experienced in this project.
Cons	The disadvantages from using the practice as experienced in this project.
Recommendations	Recommendations are a way to share additional experiences from using the practice. Recommendations can for example be concrete advice on how to improve the practice.
Comment	The comment field is used to supply additional details and explanations to why the classification looks like it does.

improvement. Improvement can come in the form of refining the practice to better suit the project, refine the practice description to better reflect actual use and/or uncovering practices that perform poorly so that replacement practices can be found. During the workshop it was decided to use experiences to keep traceability between projects and practices to be able to distinguish what practices are used in what projects. Traceability to the person responsible for the practice in the particular project is also stored to facilitate face-to-face communication. Keeping track of people with experience can alleviate project staffing and gives an account of which people to contact if one considers adopting practices.

To document practice performance evaluations three attributes were established during the workshop; return on investment, performance and resource use. Return on investment is an estimate of how much benefit, for the project, a practice brings in relation to resources and time invested. Performance reflects how well the practice fulfilled its purpose. Resource use is divided into two parts, first cost which is all fixed costs that are incurred on the project for using the practice, and then the variable cost in time. Experiences are not mandatory to document for all practices. Only practices that perform well or have a potential for improvement must be documented. To capture concrete suggestions for improvements, three attributes were added. These include pros and cons as perceived in the project, as well as recommendations. Recommendations are concrete remarks of what could be done to improve on the situation.

Most of the attributes presented for experiences, projects and practices are open ended in nature. This was decided as there are few quantitative measurements in place that directly map to the attributes presented. However, if measurements are in place, they can be used for experiences or projects, and these should be used to complement open ended attributes or estimates based on expert opinion in order to increase the reliability.

5.1.4. Examples of use

To illustrate the attributes chosen, an example used during the workshop is given below. The example is from the use of a review practice for System Requirements Descriptions (SRD). The description of the practice is given in Table 9. The description shows that the review is generally considered to have a low initiation threshold, e.g. there is little time and cost for starting to use it. It is also considered to give high return on investment. These attributes are estimates from how the practice generally performs. Ideally these attributes in the practice description change over time to correspond to what the projects actually feedback from using the practice in projects.

This review practice has been used in project Alpha (obfuscated for reasons of anonymity) and the experience from that project is shown in Table 10. The use of the review was estimated to have high return on investment but was still not considered to be highly successful. The reason for this was that the use of reviews was considered rather low cost even if they did not uncover many defects. The reason for

Table 9. SRD review practice.

Practice	
Title	Review of SRD
Source reference	
BMS instruction	BMS instruction
Literature	—
Course	—
Process connection	Create system solution
Owner	QR responsible
Prerequisites	SRD released for review
Output	Review protocol and decision
Required competence	Some knowledge about reviews Knowledge about the system construction Needed expert knowledge
Tool support	—
Initiation threshold	0 given competence
Estimated resource consumption	Low
Estimated return on investment	High
Short description	Review document from different perspectives
Purpose	Increase product quality Increase document quality
Pro	No investments
Cons	Very dependent on participants

Table 10. Experience from SRD review in Project Alpha.

Experience	
Attribute	Description
Traceability to practice	Review of SRD
Traceability to project	Project Alpha
Person responsible	John Doe
Organizational belonging	92000
Return on investment	High
Performance	
Success	55
Cost	0
Time	48 + 40 (meeting and preparation time)
Pro	—
Cons	—
Recommendations	Used reviewers that were too similar to the document writers. Should be more careful when staffing review teams.
Comment	Didn't find many crucial comments about the solution

this shortcoming is given in the comment field. To remedy this and improve the practice for future projects, the recommendation given was to use reviewers with different backgrounds in relation to the document authors. Another possibility was to use a different reading technique, e.g. perspective based reading.

It should be observed, that a seemingly trivial observation formulated through PSF can have larger implications. A seemingly low-cost, but low yield, practice like the reviews used can be formally recommended for update/improvement and chances increase as the next project reads experiences of the per-default used review practice.

5.2. RQ1: How can PSF be tailored to fit an organization?

The tailoring presented in Sec. 5.1 shows that PSF can be tailored to reflect the needs of the organization and that data from the organization offer valuable testing during the workshop. The company data uncovered what is realistic to collect and how useful the information is. Often it was noted that having simple numbered attributes could not capture the finer details and it was decided to add comment fields that can be used to enrich the descriptions where appropriate. However, there is no guarantee that the initial tailoring of PSF will fit future needs. This means that the information in the repository should be continuously scrutinized to find ways to refine what meta-data to collect.

5.3. RQ2: Does using PSF improve the quality of the documentation of practices, projects and experiences?

At the time of the workshop, VCC had a defined and documented process in the BMS, which acted as an electronic process guide with all instructions and activity

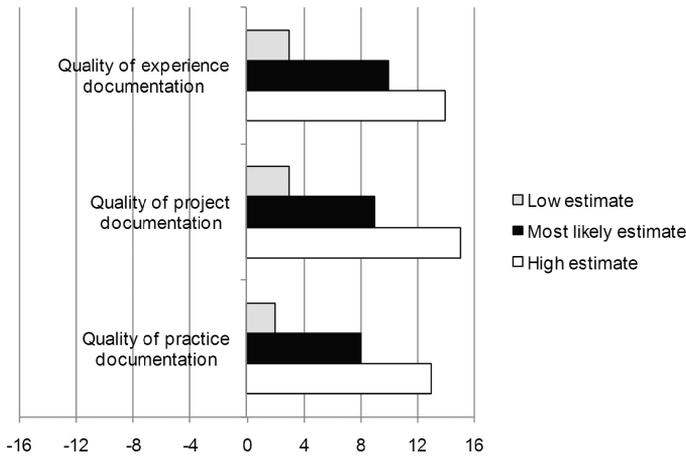


Fig. 9. Quality of documentation.

descriptions accessible through the intranet. During the workshop it was argued that using PSF would provide a feedback mechanism that enables the BMS to be updated and refined to better reflect what is actually done in projects and to more closely correspond to developers' needs. Figure 9 shows how PSF was estimated to influence the quality of the various kinds of documentation. The bars are summarized for all participants in the workshop. The black bar shows the most likely outcome while the grey and white bars show the high and low estimates. The high and low estimates intend to capture the actual outcome with 90% certainty. From Fig. 9 it can be seen that the quality of practices, experiences and projects is estimated to be better using PSF. The high and low estimates show that there is some uncertainty in the estimates; yet, quality is expected to improve even in the worst case. The hesitation regarding quality of practice documentation is explained by how PSF would be used. The prime way of using PSF was discussed to be gathering feedback from projects and letting staff close to projects document or change practices. If process documentation is still isolated from projects, there is not a huge advantage of using PSF. Quality of experience documentation, on the other hand, was expected to improve more than process documentation. At the time of the workshop, experiences from practices used in projects were seldom documented, and the use of attribute driven experience documentation provided in PSF would improve experience documentation. Using the attributes gives both structure aiding the documentation and assure a minimum level needed from the project. Rules can be set up enforcing that at least some experiences are captured from a project, e.g. a project should document three good experiences and three that warrants improvement. Project documentation would also improve using PSF, much due to a lack of a structured documentation at the time of the workshop.

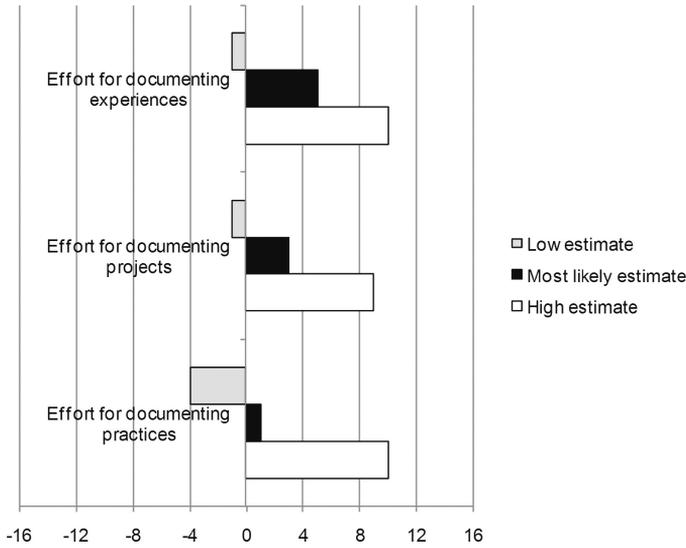


Fig. 10. Documentation effort.

5.4. RQ3: *Is PSF easy and intuitive to use?*

Using PSF includes documenting projects, experiences and practices, and later using this information for transferring practices between projects, learning from concluded projects and identifying improvements. To be easy to use, PSF should not incur too much added effort while still providing improved decision support. During the workshop, the PSF documentation effort was estimated, as can be seen in Fig. 10. The effort for documenting experiences, projects and practices were expected to increase slightly (black bars in Fig. 10) as there were seldom documentation of these entities at the time of the workshop. The figures reflect the expected frequency of documentation; a project is documented once per project while several experiences are collected and documented each project. The estimation of effort for documenting practices show the highest uncertainty (white and grey bars in Fig. 10) due to uncertainties in how PSF would actually be used. If experiences are gathered and practices are refined to reflect how practices are actually used in projects the effort was expected to be low or even less than at the time of the workshop due to that actual working practices do not change that much. If practices are continuously refined to reflect each new experience, then the effort was expected to be higher.

Regarding using PSF for locating and evaluating practices for adoption, and evaluating projects, the results from the workshop are inconclusive (see Figs. 11 and 12). The white bars in Fig. 11 shows that the effort for using PSF for evaluating practices and projects was estimated to stay the same while the effort for finding practices used in the organization was expected to decrease slightly. However, the uncertainties in the estimates are high as shown by the black and grey bars showing

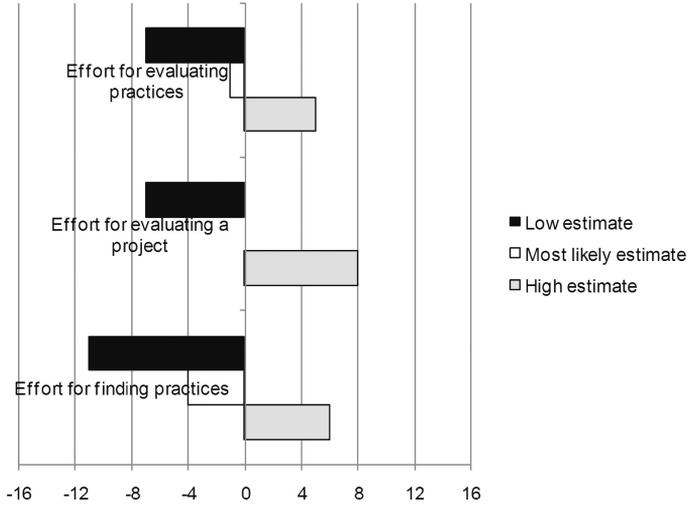


Fig. 11. Effort of use.

the high and low estimates. The black bars in Fig. 12 shows that the precision in evaluating projects and finding practices was estimated to improve using PSF while the precision for evaluating practices was estimated to stay the same as at the time of the workshop. The white and grey bars shows that the uncertainty in the estimate is high. During the workshop, it was discussed that the effort and precision of using PSF is dependent on how the process around PSF is set up and managed. Using a new practice always account for an initial cost and the change also needs to be

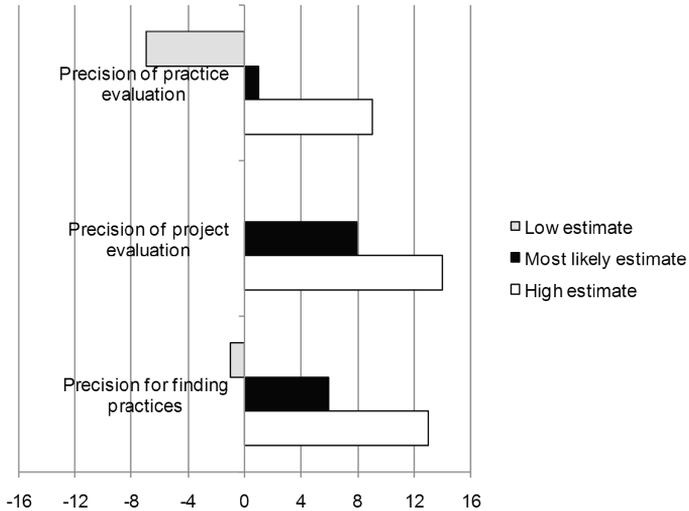


Fig. 12. Precision of use.

compliant with internal and external process standards. This means that people with sufficient knowledge about these constraints have to make decisions about what practices to use. In addition, at the time of the workshop there were few occasions where project managers could choose what practices to use at VCC. This was stipulated by the process documentation in the BMS. Thus, using PSF at VCC would be focused on improvement rather than selection i.e. using experiences gathered to improve existing practices and to identify practices that need replacements. Other organizations might have other constraints and less defined processes, thus enable more focus on using PSF for selecting practices, and thus transfer practices, together with experience, between projects.

Regarding the effort needed to implement and start running PSF the opinions during the workshop differed. Half of the participants considered implementing PSF a risky endeavor while half thought it reasonable to implement. Part of this ambivalence is rooted in that the intended scope of PSF at VCC was not thoroughly established during the workshop. Implementing PSF at VCC would essentially involve creating all practice entities and linking these to the correct BMS instruction. The number of BMS instructions included in the initial PSF effort (the number of processes included) determine the implementation cost. In recognition of this, it is prudent to start small with a limited scope and try PSF before extending the effort. This should limit the risks associated with implementing PSF and minimize the risk of failing before starting to collect experiences from development projects.

The estimates for how PSF would influence precision in decisions was mostly positive (see Fig. 12). The uncertainty in the estimates for evaluating practices is rooted in that quality of decisions depends on the person making them. As with documenting practices, it was noted during the workshop that practitioners close to projects should evaluate practices for improvement to better correspond to needs in projects. This was explained by the fact that project staff, in contrast to line or managerial staff, has recent hands on experience and tacit knowledge from using practices. Finding suitable practices and practices that need improvement was estimated to improve by organizing the information in a few attributes that can be compared and structured. Organizing all practices in one place and connecting experiences to them was valued during the workshop as the most important decision support included in PSF. However, this also depend on appropriate tools that support comparing attributes between experiences and sorting out experiences that have interesting traits.

5.5. RQ4: *Is the decision material provided by PSF suitable for decision making regarding evaluation and adoption?*

Thus far, the results have shown that PSF was estimated to add some cost from collecting experience, refining practices and documenting projects. In return, PSF was estimated to improve quality of documentation and could improve the precision in decisions regarding what practices to improve and transfer between projects. To provide a reality check of these costs and benefits, a valuation of the relation



Fig. 13. Perceived value for the different aspects of PSF.

between these, essentially return on investment of using PSF, summarized the questionnaire. This valuation reveals if PSF is actually worth using, i.e. if the benefits are expected to outweigh the costs. The results are shown in Fig. 13. The black bars shows that all activities was estimated to provide benefits that outweigh the costs using them. The grey and white bars in Fig. 13 unveils that using PSF for identifying improvements to practices is the aspect that was estimated as most likely to provide high value. Other aspects show higher uncertainty. This is expected, as noted earlier, since selection is hard to implement at VCC, as there is little room for alternating practices used in projects. Still, selection of practices was estimated as having high potential, which was recognized during the workshop, and the difficulties discussed. To enable successful selection at VCC, strict rules for what can be selected and who makes these decisions needs to be established and managed. Improvement was discussed to be easier as PSF only high-light practices that need to be improved or changed. This is less intrusive and requires less management as it is only an assessment and do not directly influence on project performance. The effect comes later as a change to practices.

Gathering all practices in one place and adding attributes to them was estimated to be beneficial. However, the estimate has a high degree of uncertainty. The positive parts have been mentioned before (see Sec. 5.4) as enabling finding practices easier. The possible downside of gathering all practices in one place is that it might lead to more formalized routines and that local practices needed to cope with local problems becomes overlooked [49]. This is not something inherent in PSF but rather an

implementation issue and something that needs to be acknowledged. PSF aims at allowing for some room for projects to work as best suited, but can also be used to make practices more rigid. The latter case was valued less during the workshop.

Finally, the least valued aspect, still expected to provide positive return on investment, is using PSF for evaluating projects. Evaluating projects are an added benefit from using PSF in addition to selection and assessment of practices. The main advantage comes from providing different pieces of information in one place through traceability, e.g. white books, characteristics, etc. However, the information collected regarding projects are first and foremost meant to be used to distinguish the context in which the experiences have been collected.

6. Discussion

This initial validation in industry showed that PSF has the potential for providing real benefits. The validation at VCC has naturally been focused on the parts of PSF that best suit the organization in question, which proved to be using it for refining practice descriptions and for identifying improvement potential. This means that the validation and results is mostly generalizable to organizations with similar disposition.

The inclusion of a “recommendation” attribute in the experience entity was valued during the workshop as a good way of capturing concrete advice for how to refine practices and to add descriptions to practices. One can browse the repository looking at the recommendations collected for each. This also provides a foundation for small step-wise refinements of practices.

Even though it was thought that selection and transfer of practices between projects was hard to achieve at VCC, the potential benefit of this way of working was acknowledged. All projects are to some extent different in that it develops different products, has different staff, lead-times, etc. This means that using different practices in different projects and using PSF to share experiences is possible but not as easy to achieve as using it for improvement. Selection of practices is a change that impact project management to a larger extent and might be an option in a later stage of implementation.

The results also indicate that for successful knowledge management it is not only a matter of choosing what information to capture and disseminate. The process and organizational issues impact the endeavor to a large extent. At VCC, this includes roles and responsibilities for documenting practices and selection of practices as well as for whom the information in the repository is aimed. This includes if the information, practice descriptions, are going to be used as an instrument for control or for documenting and disseminating how work is actually performed.

6.1. Further work

Even though this validation is limited, the results show potential and warrant further investigation. A larger dynamic validation, pilot projects, is needed to

further understand how PSF would work in industry over time and how to overcome the difficulties uncovered in this initial validation.

7. Conclusions

This paper presents PSF, an Experience Factory approach, enabling lightweight experience capture and use by utilizing postmortem reviews. Experiences gathered concern performance and applicability of practices used in the organization gained from concluded projects. Experiences are used by project managers to determine if practices used in other projects are useful for them. Process managers use the experiences to determine if there are a potential for improvement of practices used in the organization. This way of working effectively lowers the initiation threshold for working with an Experience Factory approach:

- Data collection with postmortem reviews are a practical approach for gathering experiences [36] and incur relatively low costs on the development organization.
- The effort for analysis and packaging experiences are limited by using data that is already analyzed and packaged (connected to specific practices).
- PSF builds on state-of-practice in practice selection. Professionals are likely to rely on evidence from trusted sources (colleagues) and from settings similar to their own over that of researchers or consultants [32].

PSF was developed in collaboration with industry and part of testing and transferring PSF into industry use is validation to get feedback from practitioners. Validation was performed to tailor and assure that the approach meets industry needs. The validation was set up as a full day workshop where PSF was used on real data from the organization. The use of PSF was then compared to current practices used in the organization, implying that PSF is validated in relation to successful state-of-practice approaches for knowledge management and process improvement. Overall, the results from the workshop were encouraging and the participants' assessment of PSF and particularly the tailoring developed during the workshop was positive. Using PSF to select what practices to use in projects also shows promise. However, selection introduces risks in projects as what practices are used needs to be managed. Introducing new practices also introduces risks. The most promising aspect was using PSF to uncover improvement potential and grounding this in the development organization.

Acknowledgments

This research has been partly funded by the Swedish industry and government joint research program IVSS — Intelligent Vehicle Safety Systems — through the CEDES project. We would also like to thank the participants in the workshop.

Appendix A

Questionnaire

PART 1. Background information

1. Name
2. Role
3. Department
4. Project
5. **Experience**
6. What practices do you have experiences from using?
7. How do you document practices today?
8. How do you value the quality of the practice documentation today?
9. How do you communicate knowledge about practices today?
10. How do you value the quality of practice communication today?
11. How do you document knowledge about concluded projects today?
12. How do you value the quality of the documentation of concluded projects today?
13. How do you communicate knowledge about concluded projects today?
14. How do you value the quality of the communication of concluded projects today?
15. How do you use knowledge gained from concluded projects today?
16. How do you value the use of knowledge from concluded projects today?
17. Do you look at previous projects as input to how to do your next project?
18. How do you select practices to use today? What aspects are important?
19. How would you value the success of the project on these attributes where 3 is a total success and -3 is a total failure.
20. If you feel that there are important aspects missed in the previous questions, please provide them here.

PART 2: PSF estimates

PART 2.1 Effort and resources

1. How do you perceive the effort and resources required for setting up TSF?
2. **Assume that practices are documented in the way established during the workshop. How do you value documenting practices in TSF on the aspects given below?**
 - 2.1. The effort need for documenting practices compared to today?
 - 2.2. The quality of the resulting documentation of the practices compared to today?
3. **Assume feedback of project information as established in the workshop. How do you value documenting projects this way on the aspects given below?**
 - 3.1. The effort need for documenting projects compared to today?
 - 3.2. The quality of the resulting documentation of projects compared to today?
4. **Assume feedback of experiences from using practices in projects as established in the workshop. How do you value documenting experience this way on the aspects given below?**
 - 4.1. The effort need for documenting experiences compared to today?
 - 4.2. The quality of the resulting documentation of experiences compared to today?

PART 2.2. Using TSF

5. How do you perceive that finding out what practices are used in the organization would change on the aspects given below?
 - 5.1. The effort needed for finding out what practices are used in the organization compared to today?
 - 5.2. The precision of finding out what practices are used in the organization compared to today?
 6. **How do you perceive that evaluating your project would change on the aspects given below?**
 - 6.1. The effort needed for evaluating a project compared to today?
 - 6.2. The precision in evaluating the project compared to today?
-

Questionnaire (Continued)

-
7. **How do perceive that evaluating practices used in the organization would change on the aspects given below? Evaluating practices used in the organization is either carried out to distinguish practices that can be adopted to a project or to find improvement potentials in the organization.**
 - 7.1. The effort needed for evaluating the practices in the organization compared to today?
 - 7.2. The precision in evaluating the organization's practices compared to today?
 8. What aspects are most important when evaluating practices in TSF?

PART 2.3. The value of using TSF

9. **This part of the questionnaire concerns the value of using TSF. The value in essence concerns the return on investment or bang for the buck of using TSF. That is, the value gained from using TSF compared to the added overhead. How do you value the activities given below?**
 - 9.1. Gathering practices in one place to facilitate finding them easily?
 - 9.2. Evaluating a project?
 - 9.3. Evaluating practices used in the organization in order to identify ones that can be used in a new project?
 - 9.4. Evaluating practices used in the organization to identify areas that potentially could be improved?
 10. **Would enabling evaluating practices used in other projects affect your decisions regarding what practices to use in your project?**
-

References

1. I. Aaen, Software process improvement: Blueprints versus recipes, *IEEE Software* **20** (2003) 86–93.
2. I. Rus and M. Lindvall, Knowledge management in software engineering, *IEEE Software* **19** (2002) 26.
3. T. K. Abdel-Hamid and S. E. Madnick, *Software Project Dynamics: An Integrated Approach* (Prentice Hall, Englewood Cliffs, NJ, 1991).
4. J. A. C.-M. Villalón, G. C. Agustín, T. S. F. Gilabert, A. D. A. Seco, L. G. Sánchez and M. P. Cota, Experiences in the application of software process improvement in SMEs, *Software Quality Journal* **10** (2002) 261–273.
5. J. P. Kuilboer and N. Ashrafi, Software process and product improvement: An empirical assessment, *Information and Software Technology* **42** (2000) 27–34.
6. D. J. Reifer, The CMMI: It's formidable, *Journal of Systems and Software* **50** (2000) 97–98.
7. R. L. Glass, Frequently forgotten fundamental facts about software engineering, *IEEE Software* **18** (2001) 111–112.
8. J. M. Verner and W. M. Evanco, In-house software development: What project management practices lead to success? *IEEE Software* **22** (2005) 86.
9. A. J. Nolan, Learning from success, *IEEE Software* **16** (1999) 97–105.
10. A. Birk, T. Dingsøyr and T. Stalhane, Postmortem: Never leave a project without it, *IEEE Software* **19** (2002) 43–45.
11. M. von Zedtwitz, Organizational learning through post-project reviews in R&D, *R&D Management* **32** (2002) 255–268.
12. R. L. Glass, Project retrospectives, and why they never happen, *IEEE Software* **19** (2002) 111–112.
13. T. Dingsøyr, N. Moe, J. Schalken and T. Stålhane, Organizational learning through project postmortem reviews — An explorative case study, in *14th European Software Process Improvement Conference (EuroSPI 2007)*, 2007, pp. 136–147.

14. R. L. Glass, Searching for the holy grail of software engineering, *Communications of the ACM* **45** (2002) 15–16.
15. A. M. Davis and D. Zowghi, Good requirements practices are neither necessary nor sufficient, *Requirements Engineering* **11** (2004) 1–3.
16. S. Fricker, T. Gorschek, C. Byman and A. Schmidle, Handshaking with implementation proposals: Negotiating requirements understanding, *IEEE Software* **27**(2) (2010) 72–80.
17. S. Fricker, T. Gorschek and M. Glintz, Goal-oriented requirements communication in new product development, in *2nd International Workshop on Software Product Management held in conjunction with the 16th IEEE International Conference on Requirements Engineering*, Barcelona, 2008.
18. S. Fricker, T. Gorschek and P. Myllyperkiö, Handshaking between software projects and stakeholders using implementation proposals, in *International Workshop on Requirements Engineering: Foundation for Software Quality*, 2007, pp. 144–159.
19. V. Basili and S. Green, Software process evolution at the SEL, *IEEE Software* **11** (1994) 58–66.
20. V. R. Basili, F. E. McGarry, R. Pajerski and M. V. Zelkowitz, Lessons learned from 25 years of process improvement: The rise and fall of the NASA software engineering laboratory, in *Proceedings of the 24th International Conference on Software Engineering (ICSE02)*, Orlando, 2002, pp. 69–79.
21. V. R. Basili, G. Caldiera and H. D. Rombach, Experience Factory, in *Encyclopedia of Software Engineering*, ed. J. J. Marciniak (New York, Wiley, 1994), pp. 469–476.
22. K. Schneider, J.-P. V. Hunnius and V. R. Basili, Experience in implementing a learning software organization, *IEEE Software*, 2002, 46–49.
23. T. Dingsøy, An evaluation of research on experience factory, in *Workshop on Learning Software Organisations at the International Conference on Product-Focused Software Process Improvement*, Oulo, Finland, 2000, pp. 55–66.
24. C. Johansson, P. Hall and M. Coquard, Talk to Paula and Peter — They are experienced — The experience engine in a nutshell, in *Learning Software Organizations: Methodology and Applications; Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering, SEKE'99*, Kaiserslautern, Germany, 1999, pp. 171–186.
25. P. Broessler, Knowledge management at a software house — A progress report, in *Workshop on 'Learning Software Organizations', SEKE 99*, Kaiserslautern, Germany, 1999.
26. F. Maurer and H. Holz, Process-oriented knowledge management for learning software organizations, in *Proceedings of the 12th Knowledge Acquisition Workshop (KAW'99)*, 1999.
27. F. Pettersson, M. Ivarsson, T. Gorschek and P. Öhman, A practitioner's guide to lightweight software process assessment and improvement planning, *Journal of Systems and Software* **81** (2008) 972–995.
28. T. Gorschek, M. Svahnberg and K. Tejle, Introduction and application of a lightweight requirements engineering process evaluation method, in *Proceedings of the Ninth International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'03)*, Essen, Germany, 2003, pp. 101–112.
29. T. Gorschek and C. Wohlin, Identification of improvement issues using a lightweight triangulation approach, in *Proceedings of the European Software Process Improvement Conference (EuroSPI'2003)*, Graz, Austria, 2003, pp. VI.1–VI.14.
30. I. Jacobson, M. Christerson, P. Jonsson and G. Övergaard, *Object-Oriented Software Engineering — A Use Case Driven Approach* (Addison-Wesley, 1992).
31. T. Gorschek and N. Dzamashvili-Fogelström, Test-case driven inspection of pre-project requirements — Process proposal and industry experience report, in *the Requirements*

- Engineering Decision Support Workshop Held in Conjunction with the 13th IEEE International Conference on Requirements Engineering*, Paris, 2005, pp. 31–52.
32. M. V. Zelkowitz, D. R. Wallace and D. W. Binkley, Culture conflicts in software engineering technology transfer, in *NASA Goddard Software Engineering Workshop*, 1998.
 33. CMMI Product Team, Capability maturity model integration (CMMI), Version 1.1, in *CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing Version 1.1 (CMMI-SE/SW/IPPD/SS, V1.1)*, Pittsburgh, 2002.
 34. T. Gorschek, P. Garre, S. Larsson and C. Wohlin, Industry evaluation of the requirements abstraction model, *Requirements Engineering* **12** (2007) 163–190.
 35. T. Gorschek and C. Wohlin, Requirements abstraction model, *Requirements Engineering* **11** (2006) 79–101.
 36. T. Dingsøyr, Postmortem reviews: Purpose and approaches in software engineering, *Information and Software Technology* **47** (2005) 293–303.
 37. K. Schneider, What to expect from software experience exploitation, *Journal of Universal Computer Science* **8** (2002) 570–580.
 38. T. Dybå, Improvisation in small software organizations, *IEEE Software* **17** (2000) 82–87.
 39. W. Afzal and R. Torkar, Incorporating metrics in an organizational test strategy, in *IEEE International Conference on Software Testing Verification and Validation Workshop*, 2008, pp. 305–315.
 40. T. Gorschek and C. Wohlin, Packaging software process improvement issues — A method and a case study, *Software: Practice & Experience* **34** (2004) 1311–1344.
 41. SCAMPI Upgrade Team, Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A, Version 1.2: Method Definition Document, Carnegie Mellon, Software Engineering Institute, Pittsburgh, PA CMU/SEI-2006-HB-002, 2006.
 42. T. Gorschek, C. Wohlin, P. Garre and S. Larsson, A model for technology transfer in practice, *IEEE Software* **23** (2006) 88–95.
 43. T. Gorschek and S. Fricker, Star search — A lightweight innovation process for software intensive product development, *IEEE Software* **27** (2010) 37–45.
 44. L. Almfelt, F. Berglund, P. Nilsson and J. Malmqvist, Requirements management in practice: Findings from an empirical study in the automotive industry, *Research in Engineering Design* **17** (2006) 113–134.
 45. F. Törner, M. Ivarsson, F. Pettersson and P. Öhman, An empirical quality assessment of automotive use cases, in *14th IEEE International Requirements Engineering Conference (RE'06)*, 2006, pp. 89–98.
 46. C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell and A. Wesslén, *Experimentation in Software Engineering* (Kluwer Academic Publishers, Boston, MA, 2000).
 47. R. McDermott, Why information technology inspired but cannot deliver knowledge management, *California Management Review* **41** (1999) 103–117.
 48. MIRA Ltd, *MISRA-C:2004 — Guidelines for the use of the C Language in critical systems*, 2004.
 49. J. Swan, H. Scarbrough and J. Preston, Knowledge management — The next fad to forget people? in *Seventh European Conference on Information Systems*, 1999, pp. 668–678.
 50. M. Ivarsson and T. Gorschek, Technology transfer decision support in requirements engineering research: A systematic review of REj, *Requirements Engineering Journal* **14**(3) (2009) 155–175.
 51. T. Gorschek, S. Fricker and K. Palm, A lightweight innovation process for software-intensive product development, *IEEE Software* **27**(1) (2010) 37–45.

52. CMMI Product Team, Capability Maturity Model Integration (CMMI), Version 1.1, in CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing Version 1.1 (CMMI-SE/SW/IPPD/SS, V1.1), Pittsburgh, PA, 2002.
53. SO/IEC, ISO/IEC 15504 Information Technology — Process Assessment (Part 1-5), 2003–2006.
54. R. Basili, Quantitative Evaluation of Software Methodology, University of Maryland, College Park, Maryland, July 1985.
55. S. Zahran, *Software Process Improvement: Practical Guidelines for Business Success* (Addison-Wesley, Reading, MA, 1998).
56. R. Conradi and A. Fuggetta, Improving software process improvement, *IEEE Software* **19** (2002) 92–100.
57. K. El Emam, D. Goldenson, J. McCurley and J. Herbsleb, Modeling the likelihood of software process improvement: An exploratory study, *Empirical Software Engineering* **6** (2001) 207–229.
58. J. Herbsleb, D. Zubrow, D. Goldenson, W. Hayes and M. Paulk, Software quality and the capability maturity model, *Communications of the ACM* **40** (1997) 30–40.
59. J. D. Herbsleb and D. R. Goldenson, A systematic survey of CMM experience and results, in *Proceedings of the 18th International Conference on Software Engineering*, Los Alamitos, California, 1996, pp. 323–330.
60. A. Rainer and T. Hall, A quantitative and qualitative analysis of factors affecting software processes, *Journal of Systems and Software* **66** (2003) 7–21.
61. T. Dingsøy and R. Conradi, A survey of case studies of the use of knowledge management in software engineering, *Int. J. of Software Engineering and Knowledge Engineering* **12** (2002) 391–414.
62. T. Gorschek, Requirements Engineering Supporting Technical Product Management, Ronneby: Department of Systems and Software Engineering, Blekinge Institute of Technology, 2006.