Determining Organization-specific Process Suitability

Ove Armbrust

Fraunhofer IESE, Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany
ove.armbrust@iese.fraunhofer.de

Abstract. Having software processes that fit technological, project, and business demands is one important prerequisite for software-developing organizations to operate successfully in a sustainable way. However, many such organizations suffer from processes that do not fit their demands, either because they do not provide the necessary support, or because they provide features that are no longer necessary. This leads to unnecessary costs during the development cycle, a phenomenon that worsens over time. This paper presents the SCOPE approach for systematically determining the process demands of current and future products and projects, for analyzing existing processes aimed at satisfying these demands, and for subsequently selecting those processes that provide the most benefit for the organization. The validation showed that SCOPE is capable of adjusting an organization's process scope in such a way that the most suitable processes are kept and the least suitable ones can be discarded.

1 Introduction

Many facets of process technology and standards are available in industry and academia, but in practice, significant problems with processes and process management remain. Specifically, an organization's process landscape often does not contain the processes that are required to support its current activities. Typically, a number of outdated processes exist that are not or hardly used any more, yet they still are presented as a possible choice for projects, possibly even maintained. Complementary to this, there are often conditions for which no suitable processes exist within the organization, so whenever such a condition appears, the organization's employees need to improvise due to a lack of guidance. Both cases are aggravated when it comes to future projects: There is often no pro-active preparation of an organization's processes for future demands. This leads to the following question: *How can an organization's processes be managed so that they support all of the organization's activities, current and future, while keeping the maintenance effort on an adequate level?*

This paper presents the SCOPE approach for systematically determining the process demands of current and future products and projects, for analyzing existing processes aimed at satisfying these demands, and for subsequently selecting those processes that provide the most benefit for the organization. The paper is structured as follows: Section 2 presents related work. Section 3 sheds some light on current industrial practice with respect to process scoping. Section 4 presents the SCOPE approach, and Section 5 summarizes the validation results. Finally, Section 6 discusses the approach and gives an outlook on possible future work.

2 Related Work

There is a variety of related work connected to identifying suitable processes for an organization. In this section, we distinguish product scoping approaches, technique selection approaches, and process-aware approaches.

Product Scoping Approaches. Schmid [1] describes an approach for systematically determining the scope for a software product line. While this approach explicitly considers future products, it mostly ignores projects and processes. Bayer et al. [2] transfer the concept of software product line scoping to (business) workflows, which are by their very nature somewhat similar to software processes. However, they also only consider products, not projects or processes, and include future development only implicitly.

Quality Function Deployment (QFD) is an approach for directing product capabilities based on customer needs. Cohen [3] defines it as "…a method for structured product planning and development that enables a development team to specify clearly the customer's wants and needs, and then to evaluate each proposed product or service capability systematically in terms of its impact on meeting those needs." This approach explicitly considers the anticipated future of products, but again neglects projects and processes.

To summarize, product scoping approaches assist software engineers in building the product that best supports their customers' requirements. However, the ones reviewed do not consider processes and cannot be transferred easily. For example, while for a product, it is typically clear how to provide a certain functionality, for a process, it is much less known whether a specific process can provide the required features at all.

Technique Selection Approaches. Biffl and Halling [4] provide a framework for supporting Fagan inspections. The approach is very detailed and provides decision models based on a literature survey; however, it does not consider the anticipated future and is limited to Fagan inspections. Schweikhard [5] describes a framework for supporting the decision-making process in inspections. It provides a classification scheme for context and variation factors and uses historic and empirical knowledge; however, it also does not consider the anticipated future and is limited to products.

Vegas and Basili [6] provide a characterization scheme for supporting the selection of testing techniques. They also provide a decision model and integrate existing knowledge; however, they neglect the anticipated future as did the previous two approaches, and support projects only, but no products or processes. Madachy et al. [7] developed a simulation model predicting the impact of quality strategies on defect profiles, cost, and risk, using COCOMO II [8] for cost estimation, as well as inputs on introduced defects. It considers products in a very detailed manner; however, it also does not consider the anticipated future, and is designed for products only, neglecting projects and processes.

In [9], Denger et al. analyze a number of approaches to customizing quality assurance techniques for different parts of the software lifecycle. They provide decision models for quality assurance techniques, but also do not consider the anticipated future, and they neglect projects and processes. Rus and Collofello [10] investigate the use of an expert system for making selection decisions for a reliability engineering strategy. They also provide a decision model for achieving reliability, yet again ignore the anticipated future, products, and processes. In addition to this, they focus on reliability only. In [11], the authors describe a vision for comprehensive software engineering decision support regarding techniques. They provide decision models for individual projects, but do not support products or processes. In addition, they also consider the next project, but do not look any further into the future.

To summarize, the technique selection approaches described support software engineers by providing help for decision-making. Strongly simplified, they assume that a certain quality factor is important (e.g., low defect density in the final product, or reliability of the final product) and assist decision makers in selecting appropriate techniques for achieving this goal. However, they typically investigate only either products or projects, but not both. In general, they also neglect processes. They also largely ignore the anticipated future.

Process-aware Approaches. Becker-Kornstaedt [12] describes an 8-step approach to systematic descriptive process modeling. The approach defines the scope of the process model, but considers the anticipated future use of the process model only implicitly. It does not describe how scoping should be performed.

Avison and Wood-Harper [13] investigated the problem of choosing the right development approach for information systems already very early. In the year 1991, they stated that the number of development methodologies is very large, yet there is no single methodology that is optimal for all contexts. Therefore, for every single context, a suitable methodology (or, as it would be called today, process) has to be chosen. Since an organization cannot excel at every methodology, a reduced set must be provided from which developers can choose. They propose a contingency approach and present Multiview, a framework representing a structure to help developers choose procedures, techniques, and tools from a fixed portfolio. Multiview characterizes techniques based on historical knowledge and provides decision models for some techniques, but it does not consider the anticipated future of an organization beyond the next project. It also does not support products.

Becker et al. [14] discuss the application of Quality Function Deployment (QFD) [3] for strategically planning software process improvement (SPI) programs to support an organization's business goals. Their main idea is to regard SPI as the organization's product that is to be optimized in order to support the business goals. They use the House-of-Quality matrices subset of QFD to operationalize this idea. The approach actively considers the anticipated future through the organization's business goals, yet it does not investigate products or projects, but focuses on business goals and identified problems. The recommendations for the decision model remain on a very high level of abstraction (CMMI process areas).

In summary, the product scoping approaches focus on scoping products, i.e., determining the features a number of products should have. They do not consider processes. However, they typically consider the anticipated future explicitly. The technique selection approaches mostly focus on selecting one out of very few specific techniques. Fagan inspections are a very popular subject in this community. The focus of these approaches is typically very narrow, and adapting them to support other techniques, possibly from other categories (e.g., extending a Fagan variant selection approach to support quality assurance techniques in general) requires enormous effort.

The process-aware approaches consider the processes of an organization in their entirety, instead of focusing on small parts of it. However, the approaches described mostly do not support process engineers when it comes to scoping and selecting processes.

3 Industry Approaches

This section introduces some process management approaches that can be found in today's industrial practice.

Fitzgerald et al. [15] report on an approach to provide a Motorola plant in Cork, Ireland with a software development process. Unfortunately, no information is given on how the process was constructed, apart from the reference to industry standards. In addition, continued management of the process is not detailed. A CMMI Level 5-certified IT supplier from India (2008: <10,000 employees) that the author of this paper has worked with pursues a very strict process management regime. The organization's process design team collects comments, recommendations, and requests for changes from all employees, processes them, and provides new releases of the company standard processes every three months based on the information collected. Every release acknowledges about 100 requests from employees. While process management is very strictly organized and responds systematically to employee feedback, there is no strategic process planning or suitability analysis. All modifications to the organization's processes are based on past experience of the employees and thus retrospective. Anticipated future developments are not used when the processes are adapted. A very similar approach has been taken by Josef Witt GmbH, a medium-sized (2,200 employees) mail order business in the clothing domain within the Otto group (123 companies, 55,000 employees).

ESOC (European Space Operations Centre), the European Space Agency's (ESA) ground segment, provides a ready-to-use implementation of the mandatory ESA process standards (ECSS series [16]) for its suppliers, called SETG (Tailoring of ECSS Software Engineering Standards for Ground Segments in ESA [17]). The main driver for adapting and modifying the SETG standards are changes within the superior ECSS standards. ESOC normally does not modify the SETG standards otherwise, for example to reflect changed project contexts. In particular, ESOC does not utilize their knowledge on the anticipated future when changing the SETG standards.

Except for the Motorola report, industrial case studies and the author's experience do not suggest that the software industry performs systematic strategic process management. Many organizations, for example the Indian IT supplier, are driven by standards such as CMMI or SPICE, which are demanded by their customers. Others, such as Josef Witt GmbH, react to problems or events that occurred in the past, but do not consider the anticipated future in their actions. Organizations with highly safetycritical applications such as ESA, finally, are mostly driven by other standards and not so much by actual problems.

All the case studies have in common that there is no systematic analysis as to whether and how much the application of the individual standards actually contributes to achieving the respective organization's business goals, and how such standards must be adapted to achieve these goals better in the future. The Indian IT supplier example shows that even organizations with high process maturity might not manage their processes strategically, considering the anticipated future.

4 Organization-specific Process Suitability

This section introduces the SCOPE approach for determining the suitability of an organization's processes and subsequently selecting a subset thereof, thus adjusting their scope. It requires an organization to determine which kinds of products and projects it typically pursues and is likely to pursue in the future. The organization can then identify the support these products and projects demand with respect to processes. For example, one kind of project may require processes that are able to cope with frequently changing requirements, whereas another one may require processes that specifically support distributed development. Process demands can be recorded along such attributes (e.g., "distributed development"), reflecting each individual product's and project's characteristics. The process demands of products and projects are weighted according to the probability of their realization, taking into account that products or projects sketched for the far future have lower probability than those ready to start. Using the same attributes used for product and project analysis, a process analysis determines the suitability of the organization's processes with respect to each attribute.

A more detailed description of the analysis steps can be found in [18] and will not be repeated here due to space restrictions. The result of the product analysis is, for every product i, its process demand $P(p_i, a_j)$ with respect to attribute j. The project analysis similarly results in a process demand $J(j_i, a_j)$ for every project i with respect to attribute j. In [18], the values for P and J range from 1 to 3; however, other (interval or rational) scales may also be used.

In order to determine the process demand D for a single attribute across all products p (projects j), the arithmetic mean of the sum of all P (J) values is used (Eqn. 1). D_p and D_j here consider how often a specific capability (represented by the appropriate attribute) is required relative to all other capabilities.

$$D_p(a_j) = \frac{1}{n}\sum_{i=1}^{n} P(p_i, a_j) \qquad \textbf{(1)}$$

$$D_j(a_j) = \frac{1}{n}\sum_{i=1}^{n} J(j_i, a_j)$$

The organization-wide process demand D across all products p and projects j is determined by unifying their respective demands (Eqn. 2). D thereafter contains, for every attribute, the organization's process demand with respect to this attribute. *D,* like D_p and D_j, considers how often a specific capability (represented by the appropriate attribute) is required relative to all other capabilities.

$$D = D_p \cup D_j \qquad \textbf{(2)}$$

While *D* reflects what an organization needs in terms of processes for its products and projects, our next step is to determine how suitable its processes are with respect to these demands. This suitability of the organization's processes is determined for each demand, i.e., each attribute from the product and project analysis, using expert estimation, empirical knowledge, or a combination of both. The result is a value $S(p_i, a_j)$ for the suitability of each process for each attribute. The sum of the values of all attributes per process indicates its suitability $S(p_i)$ for the organization (Eqn. 3).

$$S(p_i) = \sum_j S(p_i, a_j) \qquad \textbf{(3)}$$

Please note that $S(p_i)$ so far only reflects the processes' general suitability for the demands stated by the attributes, but does not consider how much each attribute contributes to the organization's success, and how often the respective capability is actually required for the organization's business. The former can be determined by prioritizing the attributes, e.g., through pair-wise comparison [19]; the latter is considered through the product and project analyses. Prioritizing the attributes orders them by importance. This order can be projected on any scale, reflecting each attribute's relative importance. A projection on 50%...100%, for example, would mean that the least important attribute is considered half as important as the most important attribute.

We reflect the variance in attribute importance by adjusting the generic process suitability through the introduction of an organizational factor. The organizational factor $O(a_j)$ for attribute *j* is determined by multiplying the process demand *D* for a specific attribute *j* with the relative importance *I* of this attribute for the organization's success, as determined by the attribute prioritization (Eqn. 4).

$$O(a_j) = D(a_j) \cdot I(a_j) \qquad \textbf{(4)}$$

The organizational factor $O(a_j)$ is then applied to the process suitability $S(p_i, a_j)$ for process *i* and attribute *j,* resulting in an organization-specific process suitability index $S_o(p_i, a_j)$. This adjusts the generic suitability according to the organization's business. The result indicates the suitability of the analyzed processes with respect to the process demands of products and projects and each analysis attribute's contribution to the organization's success (Eqn. 5).

$$S_o(p_i, a_j) = S(p_i, a_j) \cdot O(a_j) \qquad \textbf{(5)}$$

Finally, the organization-specific process suitability $S_o(p_i)$ of an individual process *i* can be determined by summing up the individual organization-specific process suitability values of all attributes *j* for this process (Eqn. 6).

$$S_o(p_i) = \sum_j S_o(p_i, a_j) \tag{6}$$

The suitability index $S_o(p_i)$ describes how well an organization's processes fit all of its demands, i.e., how well they support the product and project characteristics that were identified in total. Fig. 1 shows a graphical rendition of the suitability index of a (fictional) company for its five requirements processes and four design processes. This information can be used, for instance, to focus training: Teaching the example organization's employees *Delphi* will benefit it more than, for example, *Storyboards,* because the *Delphi* process is far more suitable (i.e., fulfills more and more often requested demands) than the *Storyboards* process.

Based on the suitability index $S_o(p_i)$, an organization may also adjust the scope of its processes. We will introduce two possible scenarios here: (1) an input-constrained scenario, and (2) an output-constrained scenario.

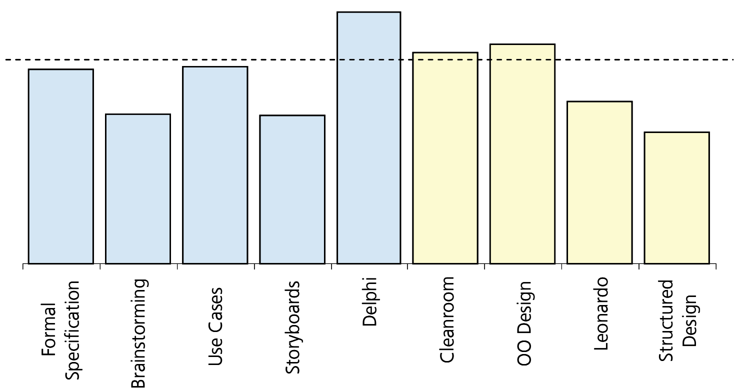

Fig. 1. Example of an organization-specific suitability index (graphical rendition)

In the input-constrained scenario, an organization has a limited amount of resources (typically effort or budget) available, and wants to put them to their best use. Using the suitability index $S_o(p_i)$ as an ordering criterion, the processes can be ordered according to their suitability for the organization. Assuming that the organization can provide a budget for maintaining and evolving *n* processes, it can simply choose the first *n* processes from the ordered list, which provide the best support for the entirety of the organization's products and projects. This means that the *globally best* processes are selected. For example, when our example company's budget is sufficient for two processes each for requirements and design, it would choose *Delphi* and *Use Cases* for the former and *OO Design* and *Cleanroom* for the latter.

In the output-constrained scenario, an organization requires processes to achieve a certain minimum suitability in order to be utilized and maintained. Again using the suitability index $S_o(p_i)$, the organization can define thresholds for the requirements and the design processes, based on experience from past projects. Processes with a suitability index below these thresholds will be discarded; only those with equal or higher suitability indices will be kept. The threshold that was set in our example company is depicted by the dashed line in Fig. 1. As can be seen, from the requirements processes, only *Delphi* is accepted, and from the design processes, *Cleanroom* and *OO Design* are accepted. The other processes are deemed unsuitable and should either be discarded or improved in order to reach the threshold.

5 Validation

The SCOPE approach was validated by means of a controlled experiment and an industrial case study. The validation was aimed at showing that SCOPE (a) allows for a greater reduction in unnecessary process variability than ad-hoc selection (hypothesis H1); (b) allows for selecting processes that cover a broader range of demands than ad-hoc selection (H2); (c) allows for reducing process management effort compared to ad-hoc methods (H3); and (d) is fit for industrial application (H4).

The experiment with master students at the University of Kaiserslautern evaluated hypotheses H1 through H4. During the experiment, the students were provided with information describing an organization’s projects, its demands with respect to processes, and the processes of the organization itself. They were asked to perform typical process management tasks, e.g., identify suitable and unsuitable processes and identify improvement opportunities.

The application of SCOPE at the Japan Aerospace Exploration Agency (JAXA) tested H1, H3, and H4 in an industrial environment. This included the performance of product and project analyses, attribute prioritization, and a process analysis, and followed the output-constrained scenario, where the goal was to fully support all projects, i.e., to keep all identified processes that were used in the analyzed projects.

5.1 Controlled Experiment

In order to be able to control a higher number of environmental influence factors than is possible in industrial case studies, a controlled experiment was conducted to evaluate hypotheses H1 through H4. The test subjects were students from the master course “Process Modeling” at the University of Kaiserslautern, held during the summer term 2009. The 13 participating students originated from China, Columbia, Germany, India, Indonesia, the Netherlands, Spain, and Thailand. They were randomly divided into two groups A and B, to account for possibly inhomogeneity. The experiment was divided into five sessions (see Table 1). Three sets of experiment materials were distributed throughout the experiment, each describing a total of 9 projects and 16 processes, 6 of which were designed to be suitable for an organization and 10 were not.

The experiment took three hours and 15 minutes and resulted in eleven usable sets of works results. The work results were analyzed in terms of identified processes and the students’ assessment of their suitability (true and false positives/negatives) (H1); the students’ correct identification of process gaps (i.e., no suitable process exists) (H2); and with respect to the reduction of process management effort, derived from the number of removed processes (H3). A Shapiro-Wilk test of normality and an analysis of skewness, kurtosis, and variances of the values denied normality; hence, a one-tailed Wilcoxon signed-ranks test [20] was performed for these hypotheses. A significance criterion of $\alpha = 0.1$ was chosen as a compromise between α error (erroneously accept hypothesis) and β error (erroneously reject hypothesis). H4 was tested using an abridged version of Venkatesh's UTAUT questionnaire [21], resulting in six variables. For two of the variables, normality testing allowed for t-tests, whereas for the other four, only a one-tail binomial sign test could be applied. All tests were significant for $\alpha = 0.05$.

Table 1. Experiment design: groups and experiment material

	Group A	Group B
Session 0	Familarization	
Session 1	Ad-hoc Data set 1	Ad-hoc Data set 2
Session 2	SCOPE Training Data Set 3	
Session 3	SCOPE Data Set 2	SCOPE Data Set 1
Session 4	Questionnaire	

Table 2 displays a summary of the statistical test results. Concerning the hypotheses, the experiment indicates that, compared to ad hoc approaches, SCOPE users:

- (a) make a decision (at all, H1.1) significantly more often, and they can spot suitable processes significantly better (H1.2, H1.4); but no significant difference could be proven for unsuitable processes (H1.3, H1.5)
- (b) detect significantly more process gaps (150% improvement, H2)
- (c) remove significantly more unsuitable processes (on average 83% more), promising a proportional reduction in process management effort (H3)
- (d) perceived the approach very positive (averaging 4.09 on a five-point Likert scale, H4).

Table 2. Experiment results

Hypothesis		Significant	Average effect
H1	Process variations		
H1.1	Decisions	yes	32%
H1.2	True positives	yes	28%
H1.3	True negatives	no	31%
H1.4	False negatives	yes	-79%
H1.5	False positives	n/a*	101%
H2	Process gaps	yes	150%
H3	Process management effort	yes	83%
H4	Acceptance	yes	4.09**

*ranks indicate non-applicability of test
**on a scale of 1 ("completely disagree") to 5 ("completely agree")

JAXA Case Study

The case study evaluated hypotheses H1, H3, and H4 in JAXA's satellite development segment and actually went beyond pure scope determination, also taking the first steps towards a comprehensive software process line [18]. This resulted in sets of common and variable entities, which were used to evaluate the hypotheses. During the course of the case study, JAXA engineers performed product and project analyses, attribute prioritization, and a process analysis, and followed the output-constrained scenario, where the goal was to fully support all projects. For confidentiality reasons, we cannot disclose the detailed process suitability results here, but we can sketch the general conclusions that were drawn.

The results of the scoping activities showed that all identified project types (national/international and scientific/engineering) share 86% of their activities and 77% of their artifacts. This means that only 14% of the activities and 23% of the artifacts *must* vary between the analyzed project types – the rest is unnecessary, yet real variation. So, by using the results of the scoping activity, JAXA could reduce the *potential* variation of their processes across activities and artifacts by an average of 82%: Instead of creating, establishing, and maintaining two completely independent satellite development process standards, they could share all common entities – effectively reducing variation for these entities to zero, which confirms hypothesis H1. By using the scoping results, JAXA also needs to maintain the common elements of the process standards only once, thus reducing management effort for each by half. Assuming that maintenance effort for all activities and artifacts is identical, SCOPE thus enabled a reduction in process management effort by 41%, confirming hypothesis H3. Finally, the feedback collected from JAXA engineers during the application of the SCOPE approach was positive. While the language barrier turned out to be something of an obstacle, product and project as well as process analyses could be performed in the course of the daily work of the JAXA process engineers. The case study results therefore support our assumption that the results from the controlled experiment with respect to hypothesis H4 can be transferred to industrial practice. Table 3 displays an overview of the results of the performed studies.

Table 3. Study results overview

Hypothesis		Controlled experiment	JAXA case study
H1	Process variations	(✓)[1] 46%	✓ 82%
H2	Process gaps	✓ 150%	
H3	Process management effort	✓ 83%	✓ 41%
H4	Acceptance	✓ 4.09 [2]	(✓) [3]

[1] Accepted with respect to making decisions and identifying suitable process variants, but not unsuitable variants
[2] On average, on a scale of 1 ("completely disagree") to 5 ("completely agree")
[3] No comparison value available: based on qualitative data

To summarize, the validation showed that (a) the application of the approach in a controlled experiment led to a 46% reduction in unnecessary process variability compared to ad-hoc approaches and allowed for an 82% reduction in an industrial case study; (b) SCOPE users identified 150% more misalignments between processes and demands in a controlled experiment than when working ad-hoc; (c) the application of the SCOPE approach allowed for a reduction in process management effort of 83% in a controlled experiment and of 41% in an industrial case study; and (d) the SCOPE approach and results were accepted by the controlled experiment participants as well as by the engineers in the industrial case study as a means of providing adequate support for process management.

6 Discussion and Future Work

In our opinion, the greatest advantage of the SCOPE approach is that it makes explicit a number of facts and decisions that are implicit at best otherwise. This way, they can be discussed and evaluated, something that is not possible for implicit knowledge. Another advantage is that the approach makes an organization very flexible within its scope. Setting up a new (or modified) process based on the process landscape can be completed very quickly, as opposed to fully tailoring a standard. For products or projects outside the scope, this is obviously not the case. However, from our experience, this kind of flexibility on a global scale ("we're great at everything") is an illusion anyway. Therefore, SCOPE assists organizations in determining their scope and then achieving process excellence for this scope.

Both product and project analyses encourage an organization's process engineers to think about the product and project future of the organization. This likely leads to identifying information that would otherwise have been neglected. The same applies to process analysis: A thorough analysis of the currently used processes' capabilities with respect to the actual needs of the organization is hardly ever done. The results of this analysis can help to rationalize otherwise sometimes rather emotional discussions regarding advantages and disadvantages of individual processes.

The two scenarios for using the suitability index can help an organization decide about its process future. They reflect two typical industry scenarios, where either an existing budget should be used optimally, or past experience is used as benchmark for process evaluation. From our experience, assistance with these types of questions is often sought, but typical model-based SPI approaches such as CMMI or SPICE do not provide this.

So far, the scenarios support determining what could be called the "global" suitability of an organization's processes. While this helps to determine the "value" of individual processes for the respective organization, there may be scenarios where a process-individual evaluation might not yield the best possible result for an organization. For example, within some organizations, one process may score high for one half of the analysis attributes, while yielding only low scores for the other half. Another process may behave vice versa. In total, these two processes would reach a mediocre suitability index, "losing" against a third process that is slightly better for all attributes – but not as good as any of the two is for some. Fig. 2 displays this situation. It shows the analysis scores for three processes for four attributes and the resulting suitability index S_o (red box to the right, assuming equal attribute importance). It becomes apparent that process 3 achieves the highest value for S_o, qualifying it for selection. However, a combination of process 1 and process 2 might prove to be more beneficial for the organization if their advantages with respect to the four attributes can be combined. We plan to investigate this interesting possibility in the future.

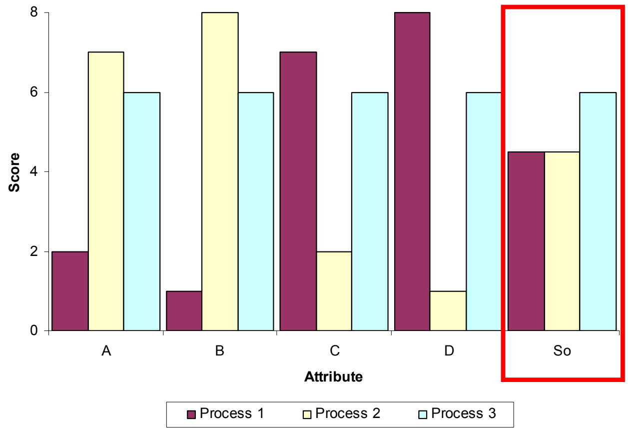

Fig. 2. Combining very different processes

Acknowledgments

We would like to thank Ms. Sonnhild Namingha from Fraunhofer IESE for reviewing the first version of this article. We also thank the anonymous reviewers for their valuable comments on the first version of this article.

References

1. Schmid, K.: Planning Software Reuse - A Disciplined Scoping Approach for Software Product Lines. PhD Thesis, Department of Computer Science, University of Kaiserslautern, Germany (2003).
2. Bayer, J., Kose, M., Ocampo, A.: Improving the Development of e-Business Systems by Introducing Process-Based Software Product Lines. In: Münch, J., Vierimaa, M. (eds.). Proceedings of the 7th International Conference on Product-Focused Software Process Improvement (PROFES'2006), Amsterdam, The Netherlands, Lecture Notes in Computer Science 4034, pp. 348-361, (2006).
3. Cohen, L.: Quality Function Deployment: How to Make QFD Work for You. Addison Wesley Longman (1995).
4. Biffl, S., Halling, M.: Managing Software Inspection Knowledge for Decision Support of Inspection Planning. In: Aurum, A., Jeffery, R., Wohlin, C., Handzic, M. (eds.) Managing Software Engineering Knowledge. Springer Verlag, Berlin (2003).
5. Schweikhard, T.: Identification of inspection-variation-factors for a decision-support-tool. Diploma Thesis, Department of Computer Science, University of Kaiserslautern, Germany (2006).

6. Vegas, S., Basili, V. R.: A Characterization Schema for Software Testing Techniques. Empirical Software Engineering 10 (4), 437-466 (2005).
7. Madachy, R., Boehm, B.: Assessing Quality Processes with ODC COQUALMO. In: Wang, Q., Pfahl, D. R. D. M. (eds.). Proceedings of the International Conference on Software Process (ICSP 2008), May 10-11, 2008, Leipzig, Germany, Lecture Notes in Computer Science, pp. 198-209, (2008).
8. Boehm, B. W., Harrowitz, E.: Software Cost Estimation with Cocomo II. Prentice Hall International (2000).
9. Denger, C., Elberzhager, F.: A Comprehensive Framework for Customizing Quality Assurance Techniques, Kaiserslautern, Germany (2006).
10. Rus, I., Collofello, J. S.: A Decision Support System for Software Reliability Engineering Strategy Selection. Proceedings of the 23rd Annual International Computer Software and Applications Conference, Phoenix, AZ, USA, pp. 376-381, (1999).
11. Jedlitschka, A., Pfahl, D.: Towards Comprehensive Experience-based Decision Support. Proceedings of the 11th European Software Process Improvement Conference (EuroSPI 2004), November 10-12, 2004, Trondheim, Norway, Lecture Notes in Computer Science 3281, pp. 34-45, (2004).
12. Becker, U., Hamann, D., Verlage, M.: Descriptive Modeling of Software Processes, Kaiserslautern, Germany (1997).
13. Avison, D. E., Wood-Harper, A. T.: Information Systems Development Research: An Exploration of Ideas in Practice. The Computer Journal 34 (2), 98-112 (1991).
14. Becker, A. L., Prikladnicki, R., Audy, J. L. N.: Strategic Alignment of Software Process Improvement Programs Using QFD. Business Impact of Process Improvement (BIPI 2008), May 13, 2008, Leipzig, Germany (2008).
15. Fitzgerald, B., Russo, N. L., O'Kane, T.: Software Development Method Tailoring at Motorola. Communications of the ACM 46 (4), 65-70 (2003).
16. Collaboration website of the European Cooperation for Space Standardization, http://www.ecss.nl/, last visited 2009-4-25.
17. BSSC Guides and Reports, http://www.esa.int/TEC/Software_engineering_and_ standardisation/, last visited 2009-04-25.
18. Armbrust, O., Katahira, M., Miyamoto, Y., Münch, J., Nakao, H., Ocampo, A.: Scoping Software Process Lines. Software Process: Improvement and Practice 14 (3), 181-197 (2008).
19. David, H. A.: The Method of Paired Comparisons. Lubrecht & Cramer, Limited (1988).
20. Sheskin, D. J.: Handbook of Parametric and Nonparametric Statistical Procedures. CRC Press, Boca Raton Boston London New York Washington, D.C. (1997).
21. Venkatesh, V., Morris, M. G., Davis, G. B., Davis, F. D.: User Acceptance of Information Technology: Toward a Unified View. MIS Quarterly 27 (3), 425-478 (2003).