Which Processes Are Needed in Five Years? Strategic Process Portfolio Management at the Japan Aerospace Exploration Agency (JAXA)

Ove Armbrust[1], Masafumi Katahira[2], Tatsuya Kaneko[2], Yuko Miyamoto[2], Yumi Koishi[2]

[1]Fraunhofer IESE, Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany
[2]Japan Aerospace Exploration Agency, 2-1-1 Sengen, Tsukuba, Ibaraki, 305-8505 Japan
[1]ove.armbrust@iese.fraunhofer.de
[2]{katahira.masafumi, kaneko.tatsuya, miyamoto.yuko, koishi.yumi}@jaxa.jp

Abstract. During the course of a strategic cooperation, a method was developed that supports an organization in adjusting its process scope. The approach systematically analyzes process needs of products and projects, and then evaluates the organization's processes with respect to the fulfillment of these needs. The result is a recommendation as to which processes to keep and maintain, which ones to discard, and which process to apply in which situation. The method was successfully applied during the development of software development standards for the development of satellite software at the Japan Aerospace Exploration Agency (JAXA). It could be shown that the method can significantly reduce the effort required for creating and maintaining software process standards, while at the same time providing projects with exactly the processes that are needed.

1 Introduction

The aerospace domain is known for its high safety and reliability requirements. This applies to both hardware and software products. While in other domains, most minor software defects will be fixed through updates, and even major defects rarely lead to significant problems, this is not the case for aerospace products. A single software defect may lead to the destruction of equipment worth hundreds of millions of dollars, for example when a launch vehicle and its payload are destroyed. Moreover, in manned space flight, human lives are in danger when software behaves incorrectly. Another problem is the long distance between the command and control platform on earth and the controlled object, for example a satellite or a space probe. In particular, it is much more difficult to fix a defect after delivery (i.e., during a mission) than in other domains: It is not possible to recall a probe, and transmitting a software patch to the probe and rebooting it is very risky. If a probe that has been flying for ten years is lost two weeks before it should start its actual mission, huge amounts of money are lost and years of work have been wasted.

Because of these challenges, the aerospace domain focuses very much on delivering defect-free products in a reliable manner. One generally accepted way of doing this is to define and follow specific processes, in line with the assumption that high-quality processes facilitate high-quality products. Process standards like SPICE [1] or CMMI [2] pick up on that by promoting proven best practices for software development.

Because of its high safety and reliability requirements, the aerospace domain has a very strong process focus. The Japan Aerospace Exploration Agency (JAXA) corresponds to the European Space Agency (ESA) or the National Aeronautics and Space Administration (NASA). As such, it is responsible for the entire Japanese space program, i.e., development and operation of ground equipment such as launch and control facilities, of launch vehicles such as rockets, and of spacecraft such as satellites.

Within JAXA, process responsibility lies with the Software Engineering Team of JEDI, JAXA's Engineering Digital Innovation Center. In very much the same way that product management is responsible for products, JAXA's process management is responsible for the creation, establishment, maintenance, and retirement of all software development processes.

As a federal agency, JAXA acquires most of the technology used from external suppliers. Only a fraction of the equipment is developed by JAXA. All suppliers are obliged to obey specific quality criteria and to provide appropriate proof. For software, this is achieved partially through comprehensive tests and other analytical quality assurance measures, and partially by enforcing the use of specific processes during software development (constructive quality assurance). To support this, JAXA provides a standard development process for all units that must be tailored to the specific needs.

Until recently, every development project prepared and applied its own tailoring of the standard process. Over time, it turned out that many such tailorings were very similar in large areas, but also showed major differences in other areas. In addition, it repeatedly occurred that necessary process adaptations were not discovered and performed until late in the project, which led to (preventable) problems in such projects. A systematic analysis of the available and necessary processes could have prevented this, but had not been performed.

To address this issue, a method was developed and applied that analyzes the process needs of projects and products as well as the process capabilities of existing processes. Based on the analysis results, a recommendation is given as to which processes to maintain in the future, which ones to discard, and which ones to apply in specific projects. The core question was:

> *How can JAXA'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 means, for example, that if a specific design method is anticipated to be used more frequently during the development of (future) satellite software, this method will be integrated into the respective standard right from the beginning – it does not need to be "tailored into" the standard for every project. The strategic process management approach that was developed supports this kind of situation. It consists of five steps that result in a recommendation with respect to the process scope, i.e., which processes to keep, which ones to discard, and when to apply a specific process.

This paper is structured as follows. Section 2 introduces related work from both research and industrial practice. Section 3 introduces the SCOPE approach, which provides a possible answer to the core question posed above. Section 4 describes the application of the approach at the Japan Aerospace Exploration Agency (JAXA). Section 5, finally, shares some of the experience collected during the case study.

2 Related Work

There is a variety of related work connected to managing an organization's process scope. In this section, we analyze research in the field of process scoping as well as approaches taken by industry.

2.1 Research approaches

This section describes a number of research approaches concerned with analyzing, selecting, or optimizing software products and projects. We distinguish product scoping approaches, technique selection approaches, and process-aware approaches.

Product Scoping Approaches. Schmid [3] 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. [4] 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 [5] 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 its 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 [6] 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 [7] 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 [8] 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. [9] developed a simulation model predicting the impact of quality strategies on defect profiles, cost, and risk, using COCOMO II [10] 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 [11], 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 [12] 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 [13], 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 [14] 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 [15] 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. [16] discuss the application of Quality Function Deployment (QFD) [5] 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.

2.2 Industry Approaches

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

Fitzgerald et al. [17] 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 [18]) for its suppliers, called SETG (Tailoring of ECSS Software Engineering Standards for Ground Segments in ESA [19]). 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 safety-critical 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 processes or process 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.

3 The SCOPE Approach for Scoping Software Processes

This section introduces the SCOPE approach for scoping software processes. SCOPE consists of five steps:

1. Product analysis, in order to identify product-imposed process demands;
2. Project analysis, in order to identify project-imposed process demands;
3. Attribute prioritization, in order to distinguish more important from less important demands;
4. Process analysis, using the same attributes as for products and projects in order to identify process capabilities; and
5. Scope determination, based on a mathematical model.

The result of these five steps is an objective analysis of the organization's process demands (steps 1 and 2), a prioritization of these demands (step 3), an analysis of the capabilities of the current processes and possibly potential (external, not yet utilized) processes (step 4), and a recommendation for the scope of the organization's processes (step 5). Note that depending on the kind of organization, product and project analyses may have different weights for assessing process needs. An organization that develops mainly products that follow a fixed-release cycle, for example one release every six months, might put more emphasis on product analysis. An organization that develops customer-individual software in various projects might concentrate more on project analysis. This difference might even exist within a single organization that has several (semi-) independent units.

3.1 Product Analysis

This step identifies existing products that will need maintenance in the future, as well as products that are expected to be developed in the future. These products may be firmly planned or still just a vision; however, they need to be considered in order to manage the organization's processes. To identify existing products, relevant information can be found in typical work products such as requirements, design documentation, test documentation, etc. For planned products (i.e., products that are firmly planned to be developed), market analyses, business strategy documents, product portfolio plans, etc. may be used as information sources. Finally, information about potential products (i.e., products that were already envisioned and / or discussed, but whose realization is yet unclear), mid- to long-term company strategy information or business roadmaps may provide the required information.

Every identified product is analyzed for its process needs using characterization attributes. These attributes describe properties of the product that influence the development process. Because of the multitude of different products, there cannot be a single list of attributes that covers all of them. However, some suggestions can be found in Table 1. While characterization attributes can be determined in various ways, a systematic approach typically leads to better results. For example, the Goal/Question/Metric approach [20] can be used to derive product characterization attributes. In the case of SCOPE's product analysis, a suitable GQM goal would be to *characterize* current and future *products* for their *process demands,* from the viewpoint of a *process engineer,* considering the *organizational context.*

Once the attribute list is complete, the next step is to characterize the products that were identified using the attributes from this list. We propose using a 3-item scale for the characterization activity, with the value "3" standing for a high rating (for example, a "3" in "size" means that the product is large and needs a process that supports large products well) and "1" for a low one (small product), leaving "2" for a medium rating. This scale admittedly neglects some detail knowledge that may be available for individual attributes; however, with an increasing number of values to choose from, rating their products becomes more difficult for most people. Since it is confusing to switch scales from attribute to attribute, one should stick to a single scale throughout the entire attribute characterization activity. A more detailed scale also does not necessarily result in more detailed answers, but possibly just in more variation within similar answers.

Each identified product is characterized along the attributes. The result is a table with the products and their ratings with respect to the attributes, and, additionally, the realization probability of each product, indicating the lower probability of planned and potential products of ever becoming a reality. Depending on the realization probability of the products (future, envisioned products might not be realized at all), the ratings may be discounted, to reflect the lower probability of such future products of becoming a reality at all. This can be done individually for each product, or for classes of products.

Product	**Project**
Size	Degree of distribution
Complexity	Schedule pressure
Criticality	Available personnel
Requirements stability	Cooperation with customer
Safety criticality	Temporal distribution
Developer experience	Developer experience

Table 1: Product and project analysis attributes

3.2 Project Analysis

Analyzing historic and future, planned projects is done similarly to product analysis. The first step is to identify existing, planned, and potential projects. Information about existing projects can be found in project plans, project traces, (tailored) process documentation, etc. Planned projects may also be described in project plans or tailored process documentation. The identification of potential projects depends mostly on expert opinion; however, some information may be extracted from mid- to long-term company strategy information or business roadmaps. Similar to product analysis, attributes are determined to characterize the projects, and projects are rated along these attributes. The corresponding GQM goal would be to *characterize* current and future *projects* for their *process demands,* from the viewpoint of a *process engineer,* considering the *organizational context.*

The result of the project analysis activity is a table with the projects and their ratings with respect to the attributes, and, additionally, the realization probability of each project, indicating the lower probability of planned and potential projects of ever becoming reality. Depending on the realization probability of the projects (future, envisioned projects might not be realized at all), the ratings may be discounted, to reflect the lower probability of such future projects of becoming a reality at all. This can be done individually for each project, or for classes of projects. Just like products, there is no single list of attributes covering all kinds of projects; however, some suggestions can be found in Table 1.

3.3 Attribute Prioritization

The first two steps of the SCOPE approach have elicited the process demands of the organization. The resulting process demand profiles consider the uncertainty of future products and projects. So far, the analysis activities have considered all attributes to be equally important. In reality, this will most likely not be the case. Thus, we propose an additional step in order to prioritize the attributes and to be able to better distinguish processes for different demands. The result is a prioritized list of attributes, i.e., information about how important each attribute is with respect to its peers.

There are a number of ways to prioritize arbitrary entities. The most straightforward way is to just put the attributes in an order ad hoc. A little more sophisticated approaches assign a number to each attribute; the higher the number, the higher the respective attribute's priority. However, both approaches only work with low numbers of attributes. The higher the number of attributes gets, the greater the risk of introducing inaccuracies with respect to priorities.

The reason for this lies in the limits of the human brain to keep things in mind at the same time. While a human being can memorize and compare a low number of objects, this does not work for higher numbers (meaning 10 or more objects). Eventually, some objects will "slip the mind"; effectively leading to some attributes not being compared to all others, but only to some. This situation is aggravated by the fact that these comparison partners change over time, too, leading to invalid priorities.

Therefore, we propose a somewhat more elaborate, but therefore more objective method for prioritizing the characterization attributes: pair-wise comparison [21]. Using this method, every characterization attribute is compared to all other characterization attributes and a value is assigned depending on whether it is considered more important than its counterpart, less important, or equally important. By summing up all comparison values for each attribute, their importance relative to each other is determined. This importance can then be scaled to meet arbitrary demands, e.g., to fit between predefined boundaries for highest and lowest relative values.

3.4 Process Analysis

Now that the needs of products and projects have been identified, the processes are analyzed as to how well each process supports these needs. The first step is to identify the processes that exist within the organization. In order to do so, information sources such as process descriptions, templates for work products, project work products (e.g., project plans, risk plans, design documents, test plans, etc.), or employees' implicit process knowledge may be used.

The identified processes are analyzed using the attributes from the product and project analysis: Each process is analyzed regarding its capability with respect to the attributes. The result of this activity is, for every process, a profile for all attributes. Typically, the analysis is conducted by process experts; however, available empirical knowledge may also be used.

In the case of expert estimation, typically process experts and expert practitioners (e.g., project managers or quality managers) would rate each process with respect to the attributes. This resembles an assessment as it is done, for example, following standards such as CMMI [2] or ISO/IEC 15504 (SPICE) [1]. However, the basis for assessing a process are not a number of specific practices (CMMI) or base practices (SPICE), but the attributes from the product and project attribute lists. Apart from ad hoc judgment, more elaborate methods, such as the standard Delphi method or the wideband Delphi method [22], can be used. Such methods are recommended when a larger number of experts is to be coordinated, or if the persons performing the rating have little experience and thus need better guidance.

If empirical data describing the effects of applying processes is available, for example, in an experience base, this data can be used to evaluate individual attributes. For example, a large number of controlled experiments and industrial case studies provide ample data regarding effectiveness, efficiency, and usability for people with different skills in different inspection approaches [23], [24], [6], [25]. Other studies compare the cost effectiveness of different verification and validation technologies [26], or provide information on the applicability of static analysis methods for safety-critical systems [27]. Information from publications like these can be used to rate individual processes in terms of the attributes defined. If experience is available from within the organization itself, this can obviously be used, too. For attributes for which no studies are available, expert estimation can be used alternatively.

The result of the process analysis is, for each process, a capability profile with respect to the attributes identified during product and project analysis.

3.5 Scope Determination

The process analysis determines how well each process supports the process needs of the organization's products and projects. So far, all attributes are considered to be equally important. Since this is not the case in reality, the results of the attribute prioritization step are used to adjust the attributes' relative importance. For example, if attribute prioritization determined that the least important attribute is only half as important as the most important one, and the other attributes are distributed between these two extremes, the results of the process analysis are adjusted accordingly: The ratings for the least important attribute are halved, the ratings for the most important one are kept as they are, and the other ratings are adjusted according to their attribute's respective importance. This results in a weighted analysis.

In order to determine organization-specific process suitability, we have to consider two aspects beyond the general capability analysis performed during the process analysis:

1. How often is a specific process capability needed for the organization's business?
2. How much does good support for a specific process capability contribute to the organization's success?

If these aspects are considered when evaluating processes, it is possible to determine which process is suitable *for a specific organization.* We do this by introducing an *organizational factor* that combines the aspects above. The organizational factor per attribute is determined by multiplying the process need for the respective attribute (result of the product and project analyses) with the relative importance of this attribute for the organization's success (result of attribute prioritization).

The organizational factor is then applied to the process analysis results. This adjusts the (generic) process capability according to the organization's business. The result indicates the capability of the analyzed processes with respect to the process needs of products and projects and each analysis attribute's contribution to the organization's success. This enables us to bring the processes into an order, with the process on top fulfilling the majority of the most important needs of the organization, and the process on the bottom the fewest.

4 Application at JAXA

The SCOPE approach was applied in the course of a strategic cooperation between JAXA and Fraunhofer IESE. The JAXA application of SCOPE included the performance of product and project analyses, attribute prioritization, and the process analysis. JAXA typically has the role of the customer in its projects, i.e., most parts of the ground and space installations, including software, are created by third-party contractors. JAXA therefore prescribes the general process, while the contractors fill in the details, e.g., specific methods. The most notable constraint of the case study was the language barrier. Many of the JAXA engineers did not speak English, so all activities were performed with few JAXA liaison engineers. However, it is assumed that they accurately reflected the other engineers' views. Overall, the case study was part of an ongoing cooperation project; therefore, the context was fixed and could not be influenced. Attribute prioritization was performed ad hoc by JAXA engineers, which is why it is neglected in the following. For confidentiality reasons, we cannot disclose the detailed analysis results; however; we can describe the experience we collected.

4.1 Product Analysis

Two products were analyzed (Satellite 1 and Satellite 2). The analysis showed that for products, scales with several items were usually required (for instance, for "complexity" or "size"), but that in some instances, binary yes/no scales would have been sufficient (e.g., for "requirements stability"). Table 2 shows an excerpt from the JAXA product analysis results.

		Complexity	Criticality	Size	Req. stability	…
Satellite 1	**Subsystem 1**	3	2	3	3	
	Subsystem 2	2	3	3	3	
	Subsystem 3	1	1	2	3	
Satellite 2	**Subsystem 1**	1	1	2	1	

Table 2: JAXA product analysis (excerpt)

4.2 Project Analysis

In the aerospace domain, products and projects often correlate directly – this was also the case here. Thus, we analyzed the two projects that created the products Satellite 1 and Satellite 2. Table 3 shows an excerpt from the results. When compared to the product analysis, it became apparent that the number of binary attributes is considerably higher, for example for "collaboration type" or "mission type". The results of product and project analysis also proved to be dependent on each other: For instance, the unstable requirements of Satellite 2 required an iterative project approach – so every (potential) supplier had to prove that this could be supported.

	Collaboration type	Mission type	Subsystem type	Supplier
Satellite 1	1	1	1, 2, 3	1, 2
Satellite 2	2	2	3	1

Table 3: JAXA project analysis (excerpt)

4.3 Process analysis and scope determination

Analyzing the JAXA processes using the identified attributes and providing a process scope recommendation proved to be not trivial. In particular, “soft” product characteristics such as “complexity” or “size” could not be used directly to determine new or modified processes. In fact, these factors did not lead to qualitative changes to the process landscape (i.e., new or changed activities or work products), but influenced project planning. For example, large and/or complex products triggered an increase in the number of reviews for certain work products, or an increase in the amount of independent verification and validation (IV&V).

The project analysis results, on the other hand, led to a number of variation points within the JAXA processes. For example, it was determined that for international cooperation projects (e.g., with Europe’s space agency ESA), an additional activity FMECA (Failure Mode, Effects, and Criticality Analysis) had to be performed to create the appropriate (new) work product. The analysis also showed that the process pursued so far was much too heavy for exploratory science projects: In this case, for example, the amount of quality assurance was reduced, and design rationales were waived entirely.

The resulting process was modeled in a graphical process modeling tool and contains 76 activities, 54 work products, and 18 graphical views of the control flow. Figure 1 displays an excerpt of the resulting process model. Depending on the values of the characterization attributes, the generic process (which contains all variants, top part) is instantiated. In Figure 1, this would be a process for a national science-type project (bottom left), or a process for an international engineering-type project (bottom right). The product, project, and process analysis results govern all variability decisions (denoted with “Opt1” and “Opt2” in Figure 1) and provide non-ambiguous advice on which option to choose.

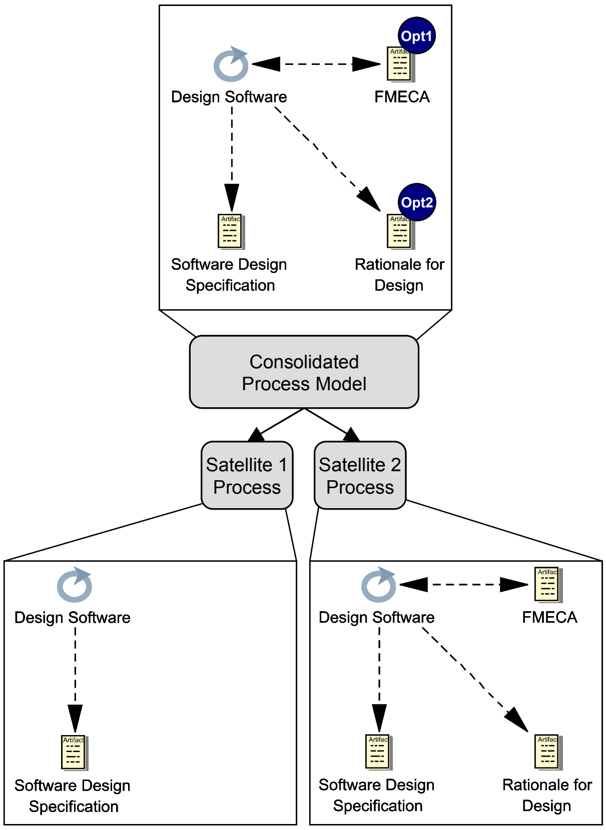

Figure 1: JAXA process scope: consolidated process model (excerpt)

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 work products. So, by using the results of the scoping activity, JAXA could reduce the *potential* variation (i.e., those parts of the process where variation might occur) of their processes by an average of 82%. By using the scoping results, JAXA also needs to maintain the common elements of the two satellite development process standards that were analyzed 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 of 41%.

5 Experience

The SCOPE approach provides very good support for JAXA's process group in their efforts to provide tailored processes for different project types. In particular, the systematic analysis of the process needs of different products and projects and, in addition, the matching selection of processes fulfilling those needs significantly facilitate an optimal process design. For example, process parts that would only be needed in specific situations could be excluded for all other circumstances, which allows JAXA an overall leaner project execution.

At the same time, the systematic analyses ensure that process needs are not overlooked, which prevents problems later on in the project. SCOPE's process scope recommendation, based on the evaluation results, was evaluated by JAXA engineers and accepted with minor modifications. The effort required by the analyses was estimated to be significantly lower than the individual creation of two independent development processes for satellites. In the future, this effort advantage is expected to grow, because large parts of the standard need to be maintained only once instead of twice for the traditional, two-standard approach. The high number of common activities and artifacts surprised some of the involved process engineers; this number was expected to be much lower.

Finally, the JAXA application of SCOPE showed that the product analysis led mostly to quantitative process changes (e.g., more independent verification and validation), whereas the project analysis typically led to qualitative process changes (e.g., new or removed activities and work products).

For the future, JAXA intends to apply this new process engineering approach to other units (ground segment, launch vehicle) in order to harmonize software development processes JAXA-wide.

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] International Organization for Standardization: ISO/IEC 15504:2004, 'Information technology - Process assessment'. ISO/IEC, Geneva, Switzerland (2006).

[2] Capability Maturity Model Integration, http://www.sei.cmu.edu/cmmi/, last visited 2009-01-19.

[3] Schmid, K.: Planning Software Reuse - A Disciplined Scoping Approach for Software Product Lines. PhD Thesis, Department of Computer Science, University of Kaiserslautern, Germany (2003).

[4] 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 Inter-

national Conference on Product-Focused Software Process Improvement (PROFES'2006), Amsterdam, The Netherlands, Lecture Notes in Computer Science 4034, pp. 348-361, (2006).
[5] Cohen, L.: Quality Function Deployment: How to Make QFD Work for You. Addison Wesley Longman (1995).
[6] 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).
[7] Schweikhard, T.: Identification of inspection-variation-factors for a decision-support-tool. Diploma Thesis, Department of Computer Science, University of Kaiserslautern, Germany (2006).
[8] Vegas, S., Basili, V. R.: A Characterization Schema for Software Testing Techniques. Empirical Software Engineering 10 (4), 437-466 (2005).
[9] 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).
[10] Boehm, B. W., Harrowitz, E.: Software Cost Estimation with Cocomo II. Prentice Hall International (2000).
[11] Denger, C., Elberzhager, F.: A Comprehensive Framework for Customizing Quality Assurance Techniques, Kaiserslautern, Germany (2006).
[12] 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).
[13] 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).
[14] Becker, U., Hamann, D., Verlage, M.: Descriptive Modeling of Software Processes, Kaiserslautern, Germany (1997).
[15] 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).
[16] 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).
[17] Fitzgerald, B., Russo, N. L., O'Kane, T.: Software Development Method Tailoring at Motorola. Communications of the ACM 46 (4), 65-70 (2003).
[18] Collaboration website of the European Cooperation for Space Standardization, http://www.ecss.nl/, last visited 2009-4-25.
[19] BSSC Guides and Reports, http://www.esa.int/TEC/Software_engineering_and_standardisation/, last visited 2009-04-25.
[20] Basili, V. R., Caldiera, G., Rombach, H. D.: Goal Question Metric Paradigm. In: Encyclopedia of Software Engineering (2 Volume Set). John Wiley & Sons (1994).
[21] David, H. A.: The Method of Paired Comparisons. Lubrecht & Cramer, Limited (1988).
[22] Äder, M.: Delphi-Befragungen. VS Verlag für Sozialwissenschaften, Wiesbaden, Germany (2002).
[23] Biffl, S.: Analysis of the Impact of Reading Technique and Inspector Capability on Individual Inspection Performance. Proceedings of the 7th Asia Pacific Software Engineering Conference (APSEC), Singapore, pp. 136-145, (2000).
[24] Biffl, S., Gutjahr, W.: Influence of Team Size and Defect Detection Technique on Inspection Effectiveness. Proceedings of the 7th International Software Metric Symposium, London, England, pp. 63-77, (2001).
[25] McCarthy, P., Porter, A. A., Siy, H. P., Votta, L. G.: An Experiment to Assess Cost-benefits of Inspection Meetings and their Alternatives: A Pilot Study. Proceedings of the 3rd International Software Metrics Symposium (METRICS), Berlin, Germany, pp. 100-111, (1996).
[26] Wojcicki, M. A.: Evaluating and combining verification and validation technologies, Brisbane, Queensland, Australia. School of Information Technol and Elec Engineering, University of Queensland (2008).
[27] Ourghanlian, A., Nguyen, T.: Dependability Assessment of Safety-Critical System Software by Static Analysis Methods. Proceedings of the 2003 International Conference on Dependable Systems and Networks (DSN'03), 22-25 June 2003, San Francisco, CA, USA, pp. 75-79, (2003).