K. Henningsson and C. Wohlin, "Risk-based Trade-off between Verification and Validation - An Industry-motivated Study", Proceedings of International Conference on Product Focused Software Process Improvement (PROFES05), LNCS-series, Springer Verlag, Oulu, Finland, 2005.

Risk-based Trade-off between Verification and Validation – An Industry-motivated Study

Kennet Henningsson, Claes Wohlin

[1]Blekinge Institute of Technology, PO BOX 520, 37225 Ronneby, Sweden
Kennet.Henningsson@bth.se, Claes.Wohlin@bth.se

Abstract. Within industry the demand for short lead-time and reduced effort consumption is in focus. For an associated industry partner the lead-time and effort focus has meant turning the interest towards the Verification and Validation (V&V) process. The industry cooperation motivating this study aims at providing a tailored and applicable V&V process, where the order of verification and validation may be changed as well as the amount of V&V activities conducted. Through the industry cooperation as well as industrial and academic experience, a method has been formulated that address how to select a suitable V&V process depending on the functionality being developed. The method describes how a suitable process is created and selected, where the appropriate process is identified based on functionality and coupling between the system entities being developed. It is concluded that the method provides support, structure and clarification to address the possibilities to a trade-off between verification and validation.

1. Introduction

Software process models have evolved over the years starting with the waterfall model [1] into agile processes [2] In-between several process models have been proposed and used, including the spiral model [3], incremental and evolutionary development [4, 5] among others. The introduction of new models is a response to the constant need of improvement in software development. Companies are constantly making trade-offs between cost, delivery time, quality, and delivered functionality. The software community also recognizes that not all types of software need the same treatment. Different processes, methods, techniques, and tools are used depending on the type of software [6] [7, 8]. Safety-critical software has different requirements than computer games, and hence the development process has to be adapted accordingly.

It is clear that different types of software need different processes. If taking this one-step further, different functionality needs different processes. However, this requires processes that are possible to tailor [9, 10]. In other words, the software process needs to be flexible. Process tailoring is discussed in the context of the Capability Maturity Model (CMM) in [9]. In a joint project with an industrial partner, in this case UIQ Technology (a subsidiary of Symbian), it was decided that an attempt should be made to tailor their verification and validation (V&V) activities [11] based on the types of functionality being developed. Verification is concerned with whether

the software is developed correctly and validation is focused on whether the correct software is developed. The company works in an environment where some functionality is fairly standard or similar to other functionality that has been developed previously and other functionality is leading edge services that have not been developed before. The company wants to work with flexible processes to ensure as short lead times as possible for their software.

These different types of functionality require different approaches to V&V, or at least open up for an opportunity to run the V&V activities differently. For example, standard functionality is straightforward to develop from a user point of view; the company is confident of what functionality the user desires. On the other hand, the user has high expectations when it comes to the quality of service for this type of functionality. Thus, in this case a possible scenario is that there is rather little need for validation, but a high need for verification. At the other end of the spectrum, there are completely new functions that are not currently available on the market. To ensure that these services meet the needs on the market, it is important to validate their usefulness with users or representatives for the markets in some sense. Normally, the expectations on correctness are less for new functionality and hence it may be discussed how much verification is needed. In this case, the validation is probably more critical than the verification, the situation can occur when it is necessary to negotiate quality, or functionality, of the delivered product [6]. This reasoning implies that different ways of handling V&V can be applied for different types of functionality within the same project.

The functionality characteristic indicates the uncertainty whether the correct functionality is implemented or not, i.e. likelihood for changes and, to some extent, faults. However, this does not provide the true picture of the risk (risk meaning exceeded budgets, too low quality, and increased project lead-time). There is a need for determining the risk impact as well. A system entity's coupling, i.e. connections to other system entities indicate the magnitude of the risk. The magnitude shows the number of dependencies each entity has, and how large the ripple effect would be by a late change or fault. It is likely that for a system entity with a high number of connections, the impact would be larger than for a low-coupled entity. By determining the functionality characteristic for the system entity as well as the coupling, sufficient information is available for risk assessment.

Bottom-line is that the company wanted a risk-based approach to their V&V activities, providing the possibility to determine which risks to take in terms of how much verification, and how much validation that should be performed for different types of functionality.

This paper presents a method for selecting a suitable V&V process based on being flexible regarding both the amount (in relative terms) of V&V needed and the order of the activities. The method is motivated by industry needs, and it is developed from an industrial case using an exploratory approach. The method is presented in a number of steps that can be adapted to different situations depending on the functionality being developed. This method is partly illustrated using input from the industrial partner where the need for this type of approach was identified.
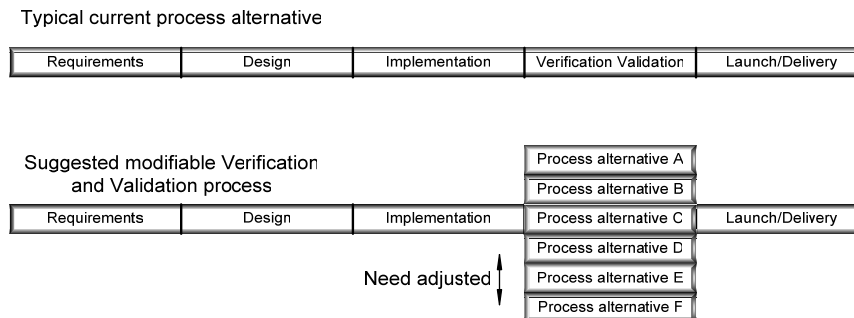
The contribution of this paper is the formulation of a method for tailoring the V&V process, involving changing order and extent of the V&V process. Moreover, the method is illustrated using input from an industrial interview study. The suggested

method is presented in Section 2, where the method is presented in generic terms. Section 3 presents a partial application of the method in the company context. In Section 4, the paper is concluded.

2. Verification and Validation Process Tailoring Method

The method presented supports the verification and validation (V&V) trade-off by addressing the traditional order and scope (or amount) of V&V activities. Adaptation of the V&V process adheres to a system entity's functional characteristics. The term system entity is used as the part of the product that provides specified functionality to the overall product or user of the product. The system entity relates to a coherent collection of requirements, defining the functionality. The V&V trade-off within the proposed method is implemented by process selection among a set of pre-developed and described process alternatives. A prerequisite for this method is a flexible and adjustable process for V&V being present in the company applying this method. The method does not suggest or exclude any specific V&V activities. The objective is to provide decision support for choosing how to apply the existing V&V activities.

Fig. 1 presents the addition to the standard process typically present for handling verification and validation.

Typical current process alternative

| Requirements | Design | Implementation | Verification Validation | Launch/Delivery |

Suggested modifiable Verification and Validation process

			Process alternative A	
			Process alternative B	
Requirements	Design	Implementation	Process alternative C	Launch/Delivery
			Process alternative D	
		Need adjusted	Process alternative E	
			Process alternative F	

Fig. 1. The replacement of the traditional V&V process with a selection of alternatives corresponding to the varying functionality demands and risk impact assessment

The core of the method is the adaptation of the V&V processes. The adaptation is guided by the product needs, the essential attributes, and risk reduction. Risk is assessed by determining the coupling for a system entity. The list below outlines the steps of the method.

1. Functionality classification – Classifying the functionality according to the validation complexity and scope required to assure a valid product.
2. Coupling classification – A subjective evaluation in early stages regarding the level of coupling for the system entity at hand.
3. Process alternatives – Developing a set of process alternatives defining order and scope for the V&V processes.

4. Attributes affected by the process alternatives – Identifying the core attributes involved in determining the success of the project.
5. Process recommendation – Based on the system entity at hand, i.e. the type of functionality and coupling of the entity and the alternative processes, recommendations of a suitable process or processes are provided.
6. Establish impact on the attributes per process alternative – Based on the alternative processes, what is the likely impact on the monitored core attributes.
7. Analyzing attribute impact – From the process recommendation it is possible to see whether that process fulfils the project goals with sufficient probability. It may be decided whether to take a specific alternative process or not, depending on the risk associated with the choice.
8. Feedback and re-evaluation of the process selection – Improving the process selection and monitoring the process impact on a continuous basis improves the method and builds trust, and hence it is an important part of the method.

Sections 2.1 through 2.8 present the steps of the method in more detail, and Section 2.9 discusses possible future extensions of the method.

2.1. Functionality Classification

For adapting the processes according to the system entity, it is required to examine each system entity. Each system entity has a set of properties; one property is the functionality. A system entity's characteristics, i.e. functionality, govern, along with the coupling classification (Section 0), the recommendation of what process alternative or alternatives to use. Given that the functionality drives the choice of V&V activities, it is suitable for classification. The apparent alternative is to classify requirements, but the vast amount of requirements would make it impossible to base the recommendation on single requirements. Functionality classification focuses on a group of requirements associated with a specific system entity.

The functionality classification scheme depends on the company's domain and prior history. The classification scheme shown in Table 1 is an illustration and exemplification. The number of classification alternatives also needs to be decided in each specific application of the method. The functionality classification accounts for if the validation is internal (performed within the company) or external (performed or involving external parties). Further, the classification considers if the verification and validation is particularly challenging for any reason. Each classification is assigned a classification on a nominal scale. The classification represents the difficulty and potential risk of changes and faults connected to each entity.

It is important to differentiate between internal and external validation. Internal validation takes place within the organization and is associated with less lead-time consumption, based on easier transition between development and testing and that the developing organization still has the control over the V&V activities.

External validation involves external resources, being customer, users, user reference groups, legislators, assessors, or other. This process consumes additional lead-time and is not under the developing organization's control. However, the accuracy of the V&V might improve in comparison to internal V&V.

Table 1. Suggested generic functionality classification

Value	Classification Name	Validation opportunities	Exemplification
1	Well-known functionality	Internal – based on experience	Functionality typically developed by the organization prior to this specific project, historically the organization is experienced and capable with the domain and the required functionality.
2	Standard functionality	Internal – with aid of standards External – if standard interpretation is required	Functionality related to complying with a standard, for example communication protocol, language specification, storage facility, encryption, or other standard.
3	Familiar functionality	External, possible internal – based on the level of experience	Functionality not previously developed by the organization, but partial experience within the general functionality and domain exists.
4	New experience functionality	External – based on the knowledge possessed concerning the requirements	Functionality where the organization lacks previous experience and domain knowledge. This is typically related to breaking new ground for the organization in question, but experience is present in the community.
5	Exploratory functionality	None – not possible to determine what is the correct functionality	Functionality that is typically breaking new ground also within the community, implying that the domain knowledge and experience does not exist.

The classification needs to reflect the nuances of the functionality affecting the V&V process. Concerns are directed towards that a too detailed classification might render it impossible to separate classifications without extensive education effort. The recommendation is to use common sense when creating the classifications. The classification is essential for the process recommendations later in the method, and for that reason, needs special attention. The functionality classification needs to represent the functionality categories present in the product and related to the verification and validation in full, at the same time, it must be feasible to separate the classes clearly.

The next section discusses the coupling classification, illustrating dependencies and connections for a system entity.

2.2. Coupling Classification

By classifying the functionality, it is possible to assess the probability for a change occurring in the system entity dealt with. However, it is also necessary to discuss the impact of the potential change for providing a complete risk assessment.

The impact of interest in this paper is to handle the time-consuming changes and faults to stay in control of, and try to minimize, lead-time and effort. The lead-time and effort consumption of a change or fault is influenced by the number of dependencies an entity has. It seems reasonable to believe that an alteration made to a central, highly coupled entity consumes more lead-time and effort, than a similar alteration to an entity with low coupling.

The coupling is subjectively assessed on an ordinal scale from 1-5, 1 lowest, and 5 highest coupling. The experience of the person assigning the coupling determines the value. Though this method is blunt, subjective, and not always repeatable, it is possible to perform early, and provides sufficient support for further evaluation of the risk.

The next section, Section 2.3, addresses the creation of, and describes the process alternatives.

2.3. Process Alternatives

The recommendation is to create the processes alternatives in advance, for two reasons: 1) Not limiting the process alternatives to the problems faced and 2) The upcoming project can facilitate the process alternatives directly, i.e. the process alternatives are in place from the start.

When handling verification and validation, there are two tailoring alternatives. One is the order, and the other is the scope of the V&V activity. Order and scope are labeled modification variables in this paper. When creating the process alternative all combinations of the modification variables is documented and investigated.

Dealing with the order of V&V, there are two natural alternatives, verification first or validation first. The characteristics of the system entity's functionality are typically more or less suitable for one or the other of these alternatives. Section 2.5 describes this further.

In addition to rearrange the order of V&V, the scope (or amount) is also subject for modification. In our model the scope of V&V activities are separated into two groups, high and low, shown in Table 3. The implication of high and low is not determined by the method, and it is up to the organization applying the method to make that definition. The characteristics of the system entity's functionality as well as change impact (moderated by coupling) motivate the variance in scope. An example is that well-known functionality might not require extensive external validation; instead could be directly submitted for verification to more rapidly reach a final state for the system entity. The process alternatives and the scope generate eight possible alternatives; these alternatives are listed along with the affected attributes in Table 3. Section 2.4 discusses the affected attributes further.

2.4. Attributes Affected by the Process Alternatives

Depending on the requirements on the V&V process, a generic set of important attributes is presented, namely number of changes, number of defects, project effort, and project lead-time. These attributes are often mentioned or indicated in literature, for example in [12, 13], and are judged as important for many companies. Though these attributes are generic, an adaptation is possible to suit the needs of an organization for selecting an appropriate process.

Conflicts between attributes are probable, meaning that maximization of two attributes may be contradictory, and in the example, there is, for example, a contradiction between low numbers of faults and minimum project effort [1, 14, 15]. The rationale being that in many cases a low number of faults typically requires extensive testing, which consumes effort. Table 3 lists the attributes along with the process alternatives. Section 2.5 continues by describing the process recommendation task.

2.5. Process Recommendation

The goal of decreasing effort and shorten project lead-time is vital. However, to shorten project lead-time it is necessary to reduce the scope of for example verification and validation. But simply decreasing the scope of these activities increases the risk for overrun of both time and budget, due to a high number of changes and faults. To handle this risk and focus the resources on the crucial parts of the system, the functionality and coupling classification is used to recommend a suitable process.

The process recommendation combines the classification of functionality and coupling by building a process selection key for each system entity. The characteristics of the functionality and coupling reflect in this two-parted key. The structure of the key is: [funct.class.value : coupl.rank]. The functional value is obtained from Table 1 (nominal scale) and the coupling is a rank (1-5) on an ordinal scale. By producing a key for each component, it is possible to use the key to recommend a process.

The mapping between a system entity's characteristics and process alternatives is the basis for process recommendation. The suitability of the recommendations is essential, and it is a task for the organization to establish. One or more processes, as described in Table 2, are matching the keys produced by the classification process.

At this stage, the focus is on recommending an appropriate process based on the key generated by the classifications, without reflecting on the impact on the attributes.

The founding idea behind the recommended processes is that if the functional characteristic of an entity represents uncertainty, it is wise to start the validation process early, to avoid a high number of changes due to un-validated requirements late in the process. On the other hand, if the functional characteristic represents familiarity and stability, it is possible to decrease the effort of validation in favor of early and thorough verification. This is based on the assumption of few changes in the requirements and minimal uncertainty and misunderstanding. Late changes are unwanted for many reasons, promoting sufficient validation activities preventing the

risk for additional and unnecessary changes, possibly avoided by the selection of an appropriate process.

Additionally, the process selection reflects the potential impact of late changes and faults, from the aspect of system entity coupling. A change to high coupling component is likely to consume more effort and project lead-time, and hence posing a greater risk if not properly handled.

Table 2. Functionality classification and process recommendation

Key range	Processes	Motivation
[1-2 : 1-3]	P4, P3, P2, P8	Key representing easy internal validation, with low coupling, possibility to go for low V&V, or focusing on finalizing the product through high verification.
[1-2 : 3-5]	P1, P2	Key representing easy internal validation, with high coupling, high verification is recommended.
[3-4 : 1-2]	P6, P3	Key representing challenging validation, with low coupling, high validation early is recommended, possible lower verification
[3-4 : 2-4]	P6, P3, P5	Key representing challenging validation, with intermediate coupling, early and high validation recommended, but also high verification.
[4-5 : 4-5]	P5	Key indicating tough validation and high coupling, early and high validation is recommended.

The processes alternatives are assigned an identification number, as in Table 3. The identification number is used to connect suitable processes to each key or key range. The keys produce links to one or more process alternatives. Table 2 states the generated keys, and recommended processes, as well as a motivation for the process recommendations. In Table 2 a key range is used, comprising a range of classification also some values occur more than once, i.e. there is an overlap. The main motivation being that there are borderline cases in the classification of functionality due to the subjective judgment of the classification.

The next section analyzes the impact on the attributes based on the selected processes.

2.6. Establish Impact on the Attributes per Process Alternative

The selection of appropriate process alternatives requires that the likely impact on the attributes presented in Table 3 is established.

Based on the keys created, a set of alternative and suitable processes are determined. For each of the keys and the suitable processes the predicted impact on the attributes is established. The impacts presented in Table 4 are used as an illustration. Actual impacts have to be determined in each specific case. The effects of the impact on the attributes are limited to two, i.e. increase, or decrease. A horizontal

arrow indicates no impact. In Table 3, arrows indicate the expected impact. It is important to stress that the impacts need to be established within each organization and representing the best knowledge of the organization. Section 3.1 discusses the company cooperation. However, the recommendation is to use surveys to capture the knowledge within the organization. This includes using existing documentation from past projects, available literature, and the tacit experience of the personnel.

The assumption is that the selection is done in a sensible way. In other words, the selection is done so that the best possible outcome is expected. For example, a process with extensive validation is not chosen for well-known functionality.

As described, the order of verification and validation change when creating the process alternatives. By starting with verification and ending with validation, it is assumed that the number of faults found late in the process decreases, but postpones the validation. This means that there is a risk that the number of changes late in the process increases. This alternative is used when a small number of changes are expected.

Starting with validation is expected to decrease the number or change requests, but might increase the number of faults at late stages in the process. The reason is that early validation will prevent misunderstandings and varying interpretations, but may delay the fault identification. This alternative is used when a high number of changes are expected if no proper action is taken.

When discussing the amount of V&V, two alternative levels are used, i.e. high and low. The actual meaning of these levels is not defined here, and is up to each company applying the method. High verification is likely to decrease the number of faults, but will consume more effort and also increase project lead-time. Low verification is likely to increase the number of faults. However, a beneficial influence on effort and probably also project lead-time, with certain reservations for that high number of faults is likely to prolong the project. This alternative should be used when a low number of changes are expected, thus fully using the verification effort. It is also important to acknowledge the system entity's coupling, and the impact of potential changes and defects. Changes to highly coupled entities might cause ripple effects, generating faults or changes in other entities, or require extensive re-verification.

High validation decreases the number of change requests, but is likely to increase project effort and lead-time. Low validation is likely to increase the number of change requests, and possible the number of faults, but project effort and lead-time is likely to decrease, given that the cost for correcting the changes and faults surpasses the gain of decreased validation. This alternative should be used when a high number of changes are expected and not continuing with major verification efforts based on invalid functionality specifications.

When the impact on all attributes is established for each key and process alternative, the table is completed and a suggestion is presented as in Table 3. Table 3 is an exemplification based on the authors' experience though company cooperation.

The next section, Section 2.7, addresses the analysis of the attribute impact.

Table 3. Table presenting process alternatives, attributes, and expected impact

Process ID	Order	Verification	Validation	#Change requests	#Fault	Project Effort	Project Lead-time
P1	Effort	High	High	↓	↓	↑	↑
P2		High	Low	↑	↓	↑	↑
P3		Low	High	↓	↑	↓	↓
P4		Low	Low	↑	↑	↓	↓
	Order	Validation	Verification				
P5	Effort	High	High	↓	↓	↑	↑
P6		High	Low	↓	↑	↓	↓
P7		Low	High	↑	↓	↑	↑
P8		Low	Low	↑	↑	↓	↓

2.7. Analyzing Attribute Impact

This step compares the recommended processes and the expected outcome for the monitored variables to the prioritized attributes for the project at hand. The goal is to find the most appropriate process based on the functionalities' characteristics for the specific system entity, providing the desired impact on the monitored attributes.

The recommended process(es) presents an expected impact on the attributes, as presented in Table 3. This needs to be compared to the outcome of the project. If the project goal is to decrease project lead-time, process alternatives with the likely impact of decreased project lead-time is a suitable choice. However, based on the functionality characteristics, the recommended processes might not provide the desired impact on the important attributes from an overall project perspective.

The analysis is influenced by risk assessment, asking if the recommended process is likely to cause overrun of the limited resources, typically effort and project lead-time. Further, the quality is also an important factor to evaluate, if an increase in faults and changes is likely, then the risk for lower quality can be too high.

The main usage of Table 3 is to see how the attributes are affected based on the selected process. However, it is possible to exclude or include processes alternatives based on the attribute viewpoint as well. If there are specific demands for a system entity, for example, the number of fault must be low, when addressing the number of faults column in Table 3, it is possible to exclude process alternatives that is likely to increase the number of faults.

The exclusion or inclusion of processes from the column viewpoint is also a desirable alternative when dealing with prioritized or fixed attributes.

Besides excluding processes, the alternative is to modify the input, i.e. the functionality characteristics or coupling for the system entity developed. In this case, an adaptation would alter the product to suit the desired process. For example, this can be done to meet highly prioritized lead-time demands.

A final alternative is to accept the potential mismatch between the recommended process and the predicted outcome, and be informed of the potential risks, and address these risks as they appear. This is still an improvement over being surprised when the risks turn into problems. The final handling is still up to the organization implementing this method for verification and validation trade-off.

2.8. Feedback and Re-evaluation of the Process Selection

Despite testing and evaluation, no process is without openings for improvement. The approach suggested in this paper is by itself a process improvement approach targeting lead-time and effort optimization. However, the method described, including this section, does not stand above improvements.

To assure that the best possible method is used for tailoring the company's process, evaluation and improvement are required. Within this method, there are three central areas for improving the accuracy and correctness of the method: 1) Functionality and coupling classification, 2) Process recommendation, and 3) Variable outcome.

The feedback and re-evaluation of the process selection aims at assuring the correctness of these three areas. If they are not performing according to expectations, i.e. supplying correct recommendations, the outcome of the process selection is likely to fail.

To address the evaluation of the three central areas, a three-phased approach is suggested. The approach is described briefly, and should be implemented according to suitability within the organization implementing this tailoring process.

The reason for presenting a three alternative approach is to cover the majority of situations, from when little historical data is available to when the historical database is sufficient for deducting statistics. The three alternatives are: 1) Value-based evaluation, 2) General statistic evaluation, and 3) Specific statistic evaluation.

Value-based evaluation: This is carried out by surveying project personnel, typically project managers and stakeholders interested in the core attributes. The focus is on the compliance and correctness related to the functionality classification and interconnections. Also, the prediction reliability of the core attributes is subject for evaluation, it should be determined if the right processes were recommended.

General statistic evaluation: This is done by comparing the outcome of the project by any previously completed project regarding the level on the core attributes. This would indicate if the process tailoring was beneficial or not.

Specific statistic evaluation: This is done by comparing the outcome with a prior project representing the same categories in terms of classification.

2.9. Method Extension

There are possible extensions of the process described in Sections 2.1 through 2.7. The described process is though sufficient for supplying value for the organization adopting the approach to verification and validation trade-off. Four possible extensions are identified:
- Additional events of V&V.

- Finer grained scope for V&V, i.e. not only low and high.

- Increased set of attributes, i.e. more than four attributes.

- Magnitude of impact on the variables, i.e. more different levels.

Additional events of V&V are to add a third or fourth, and so on, activity of V&V, creating a process of verification then validation, followed by another event of verification activities.

Increasing the granularity of the scope of V&V is also a possible extension. The scope could be divided into more levels for example, high, medium, low or any other division.

The proposed method handles a set of basic attributes these attributes are not final. Thus, it is possible to extend the model with more attributes.

The magnitude of impact on the attributes is also a possible extension. In the current proposal the impact is indicated as an increase, a decrease or not affected. It would be desirable to have a magnitude on these increases or decreases as well. This would, for example, make it possible to prioritize a small increase in project lead-time in favor of a large decrease in the number of changes.

3.   Investigation at the Company

The method presented in Section 2 is a result of a close cooperation between researchers at the university and UIQ Technology. The focus is on improving their V&V processes by tailoring the process based on characteristics of the different system entities in the software products. This section provides some first reactions and experiences when introducing the concept of adaptable V&V processes.

3.1. Company Cooperation

The cooperation between the company and the researcher is a long-term activity. The researcher has been present in the company's facilities and also through related studies, introduced and familiarized with challenges and goals of the company, becoming familiar with the company, its personnel and the products.

The development of method for V&V trade-off originates from the environment and knowledge of challenges and desires from the company.

Through close interaction and frequent discussion with a company representative having long experience and a good understanding of the goals of the company the method has emerged. The main contribution by the researcher is to formulate and structure the model based on industry input. In addition, the objective has been to make the method generally applicable and focusing on the few but prominent and important issues. The key contact person and main point for cooperation is operating as a line manager with extensive experience and insight within the company. The line manager is head of the development department, being responsible for several parallel projects. Additional responsibilities include providing sufficient support and knowledge in the form of process support to the developers. The line manager is also

in involved in customer negotiations addressing contracting issues such as delivery dates and product content and quality.

Thus, the line manager is interested in process improvements that could help the company to deliver higher quality products in a shorter time with less effort.

3.2. Company Challenge

The main challenge for the company is to deliver high quality in as short time as possible. The demands typically force the company to deliver their products with increased functionality, extended features, higher quality and in less time. This challenge requires a new perspective on the commonly used processes within the company. An adapted way of handling the processes has been seen as one way of meeting the challenges ahead. The alternative process (including selection of a suitable V&V process) requires handling the system or system parts differently based on the characteristics of them. To gain effort and lead-time, adaptable processes are considered, but changing processes imply risk. To handle the risk it is necessary to trade-off verification and validation in such a way that the overall project lead-time and effort does not suffer, typically by finding a large number of faults or changes late in the process. On the other hand, too much lead-time may be consumed if both verification and validation is operated to a full extent, when not called for.

The proposed method addresses the verification and validation trade-off by dealing with the order and scope of these two activities. Both verification and validation is quality assuring activities, focusing typically on providing answers to whether the right functionality is being developed and if it is operating as intended.

3.3. Company Result

The company study, at this stage, is performed through discussions. In addition, a structured interview took place, which discussed the handling of the creation of alternative processes and the process selection at the company.

The interview focused on three parts: important project success attributes for the company, the feasibility and applicability of the suggested method, and finally the creation and difference between the alternative processes.

This section describes the result from the interview. The most important attributes for the company is the ones described in Section 2.4: lead-time, effort, number of defects, and number of changes. The success of the project relates to the timeliness of the delivery as well as the quality of the product. During the interview, the applicability of the method was discussed. The method is apprehended as applicable and suitable in the requirements management, and project planning phase of a typical project within the company.

The issues of creating the process alternatives and determining the likely outcome resulted in that four classes of functionality were found suitable and hence to be used. However, this adaptation is natural since the method has been formulated from a balance between a generally applicable method and the wishes from UIQ Technology. As the method suggests that each key is accompanied with a set of suggested

processes and likely impacts. Thus, the suitable processes were decided upon for each key and the outcome variables were determined.

The result shows that it is clear that different keys generate different selections of processes. However, it is also clear that some process alternatives are most desired and highly prioritized, and some process alternatives are not selected at all. It is clear that the preferred processes are initiated with validation, which to some extent contradicts the typical process starting with verification and ending with validation.

4. Conclusions

The main contribution of this paper is how to deal with adaptability and in particular how classification and the creation of tangible keys can be used to select a suitable process alternative. The proposed method deals with the creation of keys focused on tangible characteristics for the functionality and component under development, providing guidance on selecting a suitable process alternative.

The verification and validation trade-off, discussed in this paper, deals with the necessity of finding the right balance between these two important activities without extending the resource consumption or lower quality in any case. In the case of the studied company there are two challenges. The first is to avoid a too safe route, and hence not fulfilling the timeliness of the product. The second is to avoid a too risky route, and risking a vast amount of faults and changes in the later stages of development, and as a result pushing the delivery date and increasing the cost.

There is also a clear connection between the success in reaching the timeliness and fulfilling market demand, and the success of the company. This strengthens the necessity of tailored processes that delivers as good results as possible. To tailor processes, it is required to monitor the circumstances under which the process shall operate, motivating the attribute and characteristics monitoring described in this paper. However, it is easy to understand that these characteristics and attributes are not the only ones of importance. Further work could be directed towards investigating other characteristics and attributes and their influence on the verification and validation process, and hence the process selection. However, it is impossible to cover all possible characteristics, and hence simplifications are required. It is argued in this paper that the method described in Section 2 is a sufficient and working starting point for the discussion of an adaptable approach to verification and validation trade-off, based on functional and coupling characteristics.

In summary, it is concluded that this approach should be feasible to obtain processes that are better tailored for the needs of different types of functionality.

5. Acknowledgement

References

[1]   W. Scacchi, "Process Models in Software Engineering", in J. J. Marciniak (ed.), Encyclopedia of Software Engineering, Wiley, New York, 2003.

[2]   K. Beck, Extreme Programming Explained: Embrace Change, Addison-Wesley, Publishing Company, Reading, 2000.

[3]   B. Boehm, "A Spiral Model of Software Development and Enhancement", Computer, vol. 21, pp. 61-72, 1988.

[4]   W. Royce, "Trw's Ada Process Model for Incremental Development of Large Software Systems," presented at 12th International Conference on Software Engineering, 1990.

[5]   T. Gilb and S. Finzi, Principles of Software Engineering Management, Addison-Wesley Publishing Company, Wokingham 1988.

[6]   B. Ramesh, J. Pries-Heje, and R. Baskerville, "Internet Software Engineering: A Different Class of Processes", Annals of Software Engineering, vol. 14, pp. 169-195, 2002.

[7]   F. Keenan, "Agile Process Tailoring and Problem Analysis (APTLY)", presented at 26th International Conference on Software Engineering, Edinburgh, Scotland, UK, 2004.

[8]   P. Xu, "Knowledge Support in Software Process Tailoring", presented at 38th Hawaii International Conference on System Sciences, Hawaii, 2005.

[9]   M. P. Ginsberg and L. H. Quinn, "Process Tailoring and the Software Capability Maturity Model", Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Technical report, ESC-TR-94-024, November 1995.

[10]   P. Donzelli, "Tailoring the Software Maintenance Process to Better Support Complex Systems Evolution Projects", Journal of Software Maintenance and Evolution: Research and Practice, vol. 15, pp. 27-40, 2003.

[11]   J. Radatz, "IEEE Standard Glossary of Software Engineering Terminology," IEEE Standards Board, New York, Standard IEEE std. 610.12-1990, 1990.

[12]   R. T. Futrell, D. F. Shafer, and L. I. Shafer, Quality Software Project Management, Prentice Hall, Upper Saddle River, 2002.

[13]   G. Succi and M. Marchesi, Extreme Programming Examined, Addison-Wesley Publishing Company, Boston, 2001.

[14]   L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos, Non-Functional Requirements in Software Engineering, Kluwer Academic, Boston, 2000.

[15]   B. Boehm and H. In, "Identifying Quality-Requirement Conflicts", IEEE Software, vol. 13, pp. 25-35, 1996.