C. Wohlin and A. Aurum, "Criteria for Selecting Software Requirements to Create Product Value: An Industrial Empirical Study", In Value-based Software Engineering, edited by S. Biffl, A. Aurum, B. Boehm, H. Erdogan, and P. Grünbacher, Springer Verlag, 2005.

Criteria for Selecting Software Requirements to Create Product Value: An Industrial Empirical Study

Claes Wohlin and Aybüke Aurum

Abstract: Product value is based on which requirements are included in a specific release of a software product. This chapter provides an overview of the value concept and presents an empirical study conducted as an industrial survey. The objective of the survey was to quantify the importance of different decision-making criteria when deciding whether to include a requirement in a project or release. The results reported from the survey are based on responses from two companies. It was discovered that there were similarities in responses at a company level, although major differences existed between individual respondents to the survey. The most important criteria were found to be those related to specific customers or markets and criteria, such as development cost-benefit, delivery date and resources. The least important criteria were those related to development and maintenance. The results also indicate that a better balance between the most important and least important criteria ought to be achieved in the future.

Keywords: Decision support, decision-making, requirements selection, product management, empirical software engineering.

1. Introduction

Organizations operating in a knowledge-based economy are facing new challenges. There is incredible pressure on software companies to achieve and sustain competitive advantage. To remain competitive in an era of increasing uncertainty and market globalization it is important to focus on the value of different customers and markets when developing products. Software companies, like many other organizations, are forced to adapt to the strategic challenges and opportunities presented by the new economy where technological advances cause dramatic changes in business processes, products and services.

According to economics and management science, an organization's ability to create value (in relation to their goals) depends on the utilization of intellectual capital (Drucker, 1998; Prahalad and Hamel, 1990). Intellectual capital is the sum of organizational knowledge which includes ideas, inventions, technologies, software programs, designs, processes, and creativity - which all can be converted to profit, create value and give organizations a competitive edge (Alwis et al., 2003; Sullivan, 1998). Alwis et al. (2003) list several potential approaches that enable organizations to create value from intellectual capital e.g. profit generation from products through sale, strategic positioning through market share, and innovation

technology, customer loyalty, cost reductions, and improved productivity. Effective management of the product development process contributes to sustainable competitive advantage for software companies. This requires that software developers firstly consider customers' requirements, business requirements and technologic opportunities when making decisions. Secondly, they need to have a sound understanding of both technical and business implications of decisions that have been made throughout the development process. Thirdly, it is essential to understand the business dynamics that drive software development in terms of cost, time, and product quality as well as how software processes and products interconnect.

The real challenge for software developers is to understand factors that drive organizational value creation and how to influence them. Value depends on the relationship between customer needs and the benefits of products that satisfy those needs. Value is created when software developers provide products that satisfy customer needs (Alwis et al., 2003). However, focusing on value to a specific customer may lead to the exclusion of considering value to other stakeholders, including other customers, different markets, software developers and project managers. This may jeopardize the long-term viability of the software company. Since customers have different needs and desires that vary with time, software companies are forced to create value along many dimensions, including as economical, physical, emotional, social, cognitive and political dimensions (Nunamaker et al., 2001). There is a vast amount of literature in the management, economics and marketing fields that has recognized the need to make product development decisions in light of their overall effect on value (Browning et al., 2002; Deaton and Muellbauer, 1980; Park, 1998; Urban and Hauser 1993).

Software developers need to know early on what the economic implications of their decisions will be in the development process, particularly when developing new products with attributes that are complex and difficult to characterize during the initial development process (Faulk et al., 2000; Harmon et al., 2003). Analyzing the economic value of a software product is complex. As such, analysis cannot be carried out simply by understanding the functionality and characteristics of software technology alone. An appreciation of the connection of this technology to business as well as to all aspects of the national and international economy is also desirable. Such an analysis must portray the future demand for software product usage accurately. This requires estimation of productivity increases from technical changes as well as estimation of economic growth and cost of software technology. Chillarege (2002) argues that in the last two decades several software product businesses announced gross profit margins of around 80%, however, there is no guarantee that this will continue into the next 20-30 years. During a software product's life cycle, market values change and different characteristics become dominant and drive business. If we can understand how market values vary during the life cycle, it would be easier to identify process models with attributes that highlight market values in a particular stage.

There has been progress made over the years in integrating value-oriented perspectives into the software engineering discipline; a discipline which includes requirements engineering, architecture, design and development, verification and

validation, planning and control, risk management, quality management, people management, and principles and practices (Boehm, 2003, Boehm and Huang, 2003). A detailed discussion of value-based concept in software engineering is also provided in Chapters 11, 14 and 21. A value-based approach aims to align software development with customer requirements and strategic business objectives (Faulk et al., 2000). Understanding the customer-value aspects brings together domain and application engineering within a common framework.

This chapter incorporates the concept of a value-based approach in requirements engineering. It is written based on the understanding that software requirements need to be bundled together such that they are aligned with business and product objectives to create value for the user of the software product. This chapter addresses criteria for how to decide which product requirements will be included in specific software projects. In particular, the chapter presents an empirical survey into two companies where the criteria for including a specific requirement in the next project or release are prioritized. The main research question addressed is: "What defines whether a requirement will be included in a specific release/project?"

The chapter is outlined as follows. Section 2 describes the background and context of the value concept from three different perspectives: management, software engineering and requirements engineering, and also presents some related works. Section 3 describes the design of the empirical study aimed at identifying which criteria are important when deciding whether to include a specific requirement in the next project or release. The results of the study are presented in Section 4. Finally some conclusions and further work is discussed in Section 5.

2. Background

This section presents the value concept for products from three different perspectives: management, software engineering and requirements engineering. Note that a detailed discussion on valuation can be found in Chapters 11, 14 and 21. It also positions the chapter in relation to research conducted in software requirements engineering related areas, in particular release planning and prioritization.

2.1. Value Concept in Management

The Oxford English dictionary defines value as "the ability of a thing to serve a purpose or cause and effect". Value creation is related to achieving desired outcomes. Thus, value can be defined as anything that one might consider useful, convenient or essential (Nunamaker et al., 2001).

In the context of product development, value includes both product and process attributes. Browning et al. (2002) argue that product value is affected not only by the presence of necessary activities in the product development process, but also by the way those activities work together to ensure that they use and produce the

right information. The value of a product to a customer depends on customer preferences and alternatives, as addressed in economics, marketing and value engineering literature (Browning, 2003; Deaton and Muellbauer, 1980; Park, 1998; Urban and Hauser 1993).

Customer value has two aspects (Browning, 2003): (a) Absolute value, which illustrates how well the attributes of a product address customer needs and (b) Relative value, which implies that the change in a product's value depends on alternative solutions to customer needs. There are a vast amount of studies in marketing literature that determine the vector of product values and specify the optimum level of each attribute. For example, Weinstein and Johnson (1999) define absolute value as *Value = perceived benefits/perceived price*, where perceived benefits and price are both measured relative to competing products. Browning (2003) points out that product value is essentially equal to benefit/cost. The author argues that a change in any of these factors can cause a change in the value of product. The question is how to balance these with the preferences of customer or market. Although companies put a great amount of effort into increasing customer value in their product development process, determining *how* and *when* value is added is still a challenge even in marketing and management science. Some of the strategies include focusing effort on eliminating the critical risk in a project. The removal of the critical risk assists in adding value into product development, or using methods such as multi-attribute utility theory, where a change in value/utility typically implies a change in demand for a product (Browning et al., 2002).

2.2. Value Concept in Software Engineering

Keller (1995) argues that value of software is viewed very differently from most other kinds of objects. Thus, it is difficult to define the concept of value in software engineering, as the development process involves many stakeholders that each defines value from their own point of view. The author points out that "*Software is not something that you can hold it, yet it can be duplicated very quickly. …. As more and more software becomes "available" on public networks, the range of value will be extended even more*". An interesting fact is that, the value of a product for a customer is expressed in terms of benefit and cost, whereas to a software company it is expressed in terms of the profit (return) from the product sold. This profit promotes economic value, determined by the net present value of future benefits that ownership of an item brings to its owner (Browning et al., 2002; Alwis et al., 2003).

In the context of software engineering, value creation involves gaining new insights or discovering new patterns of knowledge at the process level, product level or resources level. Information and knowledge transfer to stakeholders facilitates value creation. The ability to assess the impact of changes in a software product, process or resources during the development life cycle is an important aspect in software product management. Alwis et al. (2003) point out that the value of a product increases in proportion to its advantages over competitive products, and decreases in proportion to its disadvantage. Thus the value of any product to a cus-

tomer is a function of its performance and price, relative to other products in the market.

The notion of integrating insights from customer value analysis into the software development process and the difficulties that are associated with the practical application of this have been addressed in software engineering literature. According to Harrison (2001) the software engineering community lacks the ability to quantitatively measure the benefits of reduced uncertainty in a software development project. He proposes evaluating investments in software engineering infrastructure using well-accepted economic and financial models. These models are based on a theory that the inherent value of the organization is defined as the value of all its future profits. The author argues that usage of these techniques can lead to better business cases for investing in software process improvement.

Tanaka et al. (1998) emphasize that there are various existing analysis tools and techniques for quality measurement of software product value throughout the life cycle, however many of them are not fully utilized by software developers. Firstly, it is not easy for managers to understand and utilize analyzed results. Secondly, it is time consuming to evaluate the tools and prepare the environment needed to apply them practically. Thirdly, it is important to acquire the know-how for using tools and measurement data effectively and to incorporate this with the software development process in a timely fashion. Erdogmus et al. (2004) attack the problem from an education point of view. The authors point out that, although the software engineering community has put in an enormous amount of work in the areas of metrics and cost-benefit analysis, they have failed to cover valuation and competitive strategy in a business context. According to the authors, the problem starts from the software engineering education, which ignores to the need to investigate the role of technical projects in the context of overall business requirements

Faulk et al. (2000) emphasize the importance of communication between the business and technical sides of an organization, so that decision makers with different roles can have a better understanding of the software engineering implications of their decisions. The authors point out that current software development models do not support such communication. Furthermore, value creating units such as product management, marketing and development are separated by culture, language and perception of overall goals. Since the software development process does not link business objectives with software design decisions, the outcome is often a mismatch between technical capabilities and business goals. The authors provide a process framework that links strategic business goals, process improvement and the application of domain engineering to software product lines and refer to this approach as a “Value-Based Software Engineering” approach. A value-based approach to software engineering is further discussed in Chapter 21

The fact that software is different than other types of products only serves to complicate the matters. Software is easily changed (in many cases, too easily) and released in several releases. Thus, it is not only a matter of looking at the short-term value of the next release. The long-term evolution of a software product has to be taken into account. There is a constant trade-off between short-term business goals to satisfy customers and different markets, and long-term evolution of the

software to ensure that the software product is competitive in both the short- and long-term.

2.3. Value Concept in Requirements Engineering (RE)

It is critical that software developers integrate insights from customer value analysis into the requirements process. Several researchers emphasize how important it is for managers to understand the implications of their decisions in relation to a cost-benefit analysis, in particular during early life cycle activities (Boehm, 2003, Faulk et al., 2000). Furthermore, it is crucial to ensure that the requirements meet business goals. System engineering and management, and in particular risk management literature, stress the importance of including effort, schedule, cost and risk assessment as part of project planning. Goal modeling techniques in requirements engineering is another approach that serves as a mechanism by which to link requirements to strategic objectives anchored in the context of an overall model of business strategy.

Gordijn and Akkerman (2003) argue that requirements engineering approaches neglect the value proposition of information systems, despite an understanding of this value proposition being key to the development of e-commerce applications. The authors focus on the use of RE and a conceptual modeling approach to articulate, analyze and validate Internet enabled value propositions in an e-business context. They also develop an economic value perspective (called e3-value) by representing an e-commerce idea using principles and techniques which stem from RE.

Favaro (2002) points out that a full cost-benefit analysis of a requirement(s) requires investment in time and resources and is more difficult than design and implementation, as there are more unknown factors in the early stages of the life cycle. Thus it takes a full development cycle before the complete economic impact of a requirement is known. Favaro argues that software developers may add value to requirements in several ways, such as by learning to create reusable requirements that enclose cost-benefit analysis or by studying the new generation of agile development processes to enable them to understand strategic possibilities for adding value to the requirements process over the full product cycle. Furthermore, by learning more about the new tools and financial analysis, they can better understand how strategic flexibility in the requirements process adds value.

The bottom-line is that software development companies are faced with the challenge of deciding which requirements to include in a specific project or release, and which requirements to reject or postpone to later releases. Thus, an empirical study was conducted to increase our understanding of which criteria are in fact the most important to include in the next project or release. It was assumed that each project handles one release. The study was conducted as a survey and results are presented from two companies.

2.4. Related Work in Release Planning and Prioritization

Market driven (as opposed to customized) incremental product development and delivery (release) is becoming increasingly commonplace in the software industry (Ruhe and Greer, 2003, Greer and Ruhe, 2004, Carlshamre 2002). Incremental product development is planned and executed with the goal of delivering an optimal subset of requirements in a certain release (version of a product that is distributed to customers). The idea is to select what a release should contain (requirements), when it should be released (time), and at what cost (effort) this should be achieved. Decisions about which customers get which features, at what level of quality and at what point in time, have to be made, making these activities a major determinant of the success of a product. All of these activities are vitally dependent on product requirements and are elicited/captured, analyzed, and specified before any planning and development activity can commence. Decision-support in a release-planning context is further discussed in Chapter 22.

The contributions in this area include addressing different aspects of requirements management, such as prioritization (Karlsson et al., 1998; Regnell et al., 2001; Ruhe et al., 2003) and dependencies between requirements (Dahlstedt and Persson, 2003; Carlshamre et al., 2001). Moreover, researchers have worked on connecting the requirements engineering process to decision-making (Regnell et al., 2001; Aurum and Wohlin, 2002; Aurum and Wohlin 2003). Some work has also been done on release planning. In (Ruhe and Greer, 2003; Greer and Ruhe 2004), a genetic algorithm approach has been used to plan for different releases, while the work in (Carlshamre, 2002) is focused on understanding release planning.

Thus, work has been conducted on release planning and, as such, there are investigations into prioritization of requirements and dependencies between them. However, to the best of our knowledge no studies have actually looked into the criteria used in decision-making about whether to incorporate a specific requirement into a software project or release. The study presented below is the first step towards filling this gap, and is needed to understand how value is created for software products.

3. Research Approach

This section provides an overview of the design of the survey and, in particular, the questionnaire used. The main objective is to provide insight into the following research question: “What defines whether a requirement will be included in a specific release/project?” This is closely related to understanding the underlying decision process related to requirements. Decision support is further discussed in Chapters 13 and 14. Another related issue is negotiation, which is further discussed in Chapter 36. Situations where it must be decided whether to include a specific requirement in a project or release are often not straightforward and may even involve negotiations. Negotiations are not further discussed here.

3.1. Development of the Survey Questionnaire

A survey was designed to understand and evaluate the importance of different decision-making criteria when determining whether or not to include a specific requirement in a project or release. Industry representatives were asked to prioritize the importance of the different criteria in their decision-making process. The following procedure was chosen to design the survey instrument, i.e. a questionnaire:

- A brainstorming session was held to identify suitable criteria to include in the survey. The session included three researchers involved in requirements engineering research. All three have close industrial contacts.
- Based on the outcome of the brainstorming session, a questionnaire was designed by the main author of this chapter.
- The questionnaire was reviewed by the participants of the brainstorming session, and one additional independent researcher, to further improve the selection of criteria.
- The questionnaire was updated based on feedback from the reviewers, and then sent to a contact person at different companies.

The brainstorming session and the review process included some in-depth discussion about whether it was possible to identify orthogonal criteria. It was concluded that it would only be possible if the criteria were kept at a high level of abstraction. This would mean that very few criteria would be evaluated and prioritized by the subjects in the study. The discussions led to a removal of some all embracing criteria, such as risk, that are related to basically all other criteria; however it also was decided to retain a number of criteria despite dependencies, since it is basically impossible to avoid all dependencies. The intention was for subjects to prioritize without thinking too much about dependencies, and instead focusing on what they viewed as the main criteria. In summary, the objective was that importance should be judged from the individual importance of the criteria and not as consequences of other criteria. The actual outcome points to three different behaviors with respect to this issue. This is further elaborated in Section 4.1.

3.2. Criteria Covered in the Questionnaire

After several iterations the questionnaire was narrowed to include 13 criteria for assessment by subjects. Many of the criteria were general in the sense that they were not solely factors relevant for selecting requirements. They were often referred to in literature discussing software success more generally (Wohlin et al., 2000). Moreover, it was also stated clearly that additional criteria could be added by the subjects. This was done to avoid subjects feeling that missing criteria hindered their completion of the questionnaire. Moreover, the questionnaire was designed this way to capture any additional criteria that were missed in the brainstorming session. It was agreed among the researchers that the 13 criteria covered three important dimensions or stakeholder groups, although this grouping was not communicated to the subjects (respondents). The three groups were: external mar-

ket/customer, company management and development/maintenance personnel. The 13 criteria included in the study are as follows. The text is exactly as communicated to the subjects in the questionnaire, including a short explanation and motivation for each criterion.

External market/customer

1. Competitors
   Explanation: The status of the competitors with respect to the requirement. In other words, it is taken into account whether a competitor has the implied functionality implemented or not.
   Motivation: We may feel forced to include a requirement if our competitors have the functionality, or we may want to implement something that is considered to be leading edge functionality (functionality competitors do not have).
2. Requirement's issuer
   Explanation: The actual issuer of the requirement is taken into account, i.e. which stakeholder (internal or external) generated the requirement.
   Motivation: We may judge some issuers as more important than others, for example, a very important customer or representative for an important market.
3. Stakeholder priority of requirement
   Explanation: The priority of the requirement is taken into account.
   Motivation: We may want to prioritize the requirements that our customers or markets think are of particular importance.
4. Requirement's volatility
   Explanation: This criterion is related to whether the requirement is likely to change or not.
   Motivation: We may want to handle highly volatile requirements differently.

Company management

5. Support for Education/Training
   Explanation: The ability and possibility to provide technical support, education, and training to customers, markets and so forth with respect to the requirement.
   Motivation: We may not want to implement functionality unless we could provide the appropriate technical support, education and training in relation to the requirement.
6. Development cost-benefit
   Explanation: The actual cost-benefit for implementing the requirement.
   Motivation: We may not want to include a requirement if the implementation cost is judged to be high in relation to the expected benefit.
7. Resources/competencies
   Explanation: The availability of resources with the right competencies to implement the requirement.
   Motivation: We may not want to implement a requirement unless we are sure that we have the right people available for the job.

8. Delivery date/Calendar time
   Explanation: The ability to meet the project deadline.
   Motivation: We may not want to introduce a requirement that may affect the deadline of the project negatively.

Development/maintenance personnel

9. System impact
   Explanation: The impact of the requirement on the existing system.
   Motivation: We may not want to implement a requirement if we judge that the actual impact in terms of changes to the existing system is too large.
10. Complexity
    Explanation: The estimated complexity of the requirement and the associated challenges in implementing it.
    Motivation: We may not want to include a requirement that is judged to be very complex to implement and as a consequence the risk of failure as too high.
11. Requirements dependencies
    Explanation: The dependencies between this specific requirement and other requirements, either already implemented or other posed requirements.
    Motivation: The dependency to other requirements (already implemented, scheduled to be implemented, or deferred to later release) may affect our decision regarding the current requirement.
12. Evolution
    Explanation: The impact on the future evolution of the system.
    Motivation: We may not want to implement a requirement if it is believed to make long-term evolution of the system more complicated.
13. Maintenance
    Explanation: The impact on the maintenance of the current system.
    Motivation: We may not want to implement a requirement if it is believed that the requirement may cause many problems in terms of maintenance.

3.3. Conducting the Survey

The above 13 criteria were included in the questionnaire as follows. First, the subjects were given a short introduction. This included positioning the survey within a larger industry-academia collaborative research project, highlighting the value of participating in the survey, the target audience for the survey (important since the communication was done through a contact person at each company), the main research question, estimated time for the questionnaire, and finally the subjects were also guaranteed anonymity. It was clearly stated both in the questionnaire and in an e-mail that the target audience was personnel included in the decision-making process. It was expected to include the following types of management personnel: product management, project management and line management.

The second part contained an introduction to the 13 criteria as listed in Section 3.2. The third part included a characterization of the context in which the subject responded. This included company name, unit within company, type of applica-

tion, whether development was market- or customer-oriented, type of product and the role of the subject within the organization. Contact details were also asked for to ensure that each subject could be contacted for clarification purposes, although no data in the analysis will be connected to specific individuals.

The third and final part was the actual survey. The 13 criteria were listed in a table and the subjects were asked to fill out three columns with respect to the criteria. First, the subjects were asked to answer yes or no regarding whether each criterion was relevant when deciding to include a requirement in a project or release. For the other two columns the subjects were asked to provide relative weights regarding the importance of the criteria. The subjects had 1000 points to spend among the 13 criteria (or more if they chose to add some criterion). A higher number of points meant that a criterion was relatively more important. For example, a criterion obtaining twice as many points as another criterion was viewed to be twice as important. The subjects were allowed to distribute the points as they wished, i.e. there were no requirement that each criterion should be allocated a weighting. In other words, a subject could have given all 1000 points to one criterion.

The second column was concerned with the way different criteria are valued today, and the third column was focused on how the criteria ought to be valued in the future. The objective was to capture both the current state of practice and any changes that industry would like to make in the future. The latter may be viewed as a more idealistic situation.

4. Survey Results and Analysis

The questionnaire was initially sent to two companies, although the intention was to send the survey to more companies. This approach was chosen for two reasons. First, it provided a means of validating that the survey was understandable and that no major problems existed with the questionnaire. Secondly, conducting the study these particular companies became a priority because they had scheduled requirements engineering related workshops, presenting the perfect opportunity to present and discuss the survey. Later, six more companies responded to the survey, however this data has still not been analyzed and hence the results presented here are based on the two first companies only.

Unfortunately, both companies compressed their workshop schedules, which meant that the presentation and survey discussion was removed from the agenda. Feedback has hence only been sent via e-mail.

The two companies are referred to as Company A and Company B respectively. Company A is a major international company, and the responses are provided by one part of the business. This part develops hardware and software solutions for process control systems. Products from the company are sold to a world market. Company B is part of an international enterprise. The company develops hardware and software products for automatic guided vehicles. Their products are also sold on a world market.

In total, 13 subjects responded from these two companies, i.e. seven subjects represent Company A and six subjects work at Company B. The observations, results and analysis presented in this section are based on an analysis of the responses of these 13 subjects. This may be viewed as few respondents, however, it should be remembered that the number of responses was naturally limited by virtue of the fact that the survey targeted key personnel and roles in each organization. Given that company workshops were planned in advance, it was known that only 15 responses could be expected if all relevant people responded. Thus, 13 responses must be viewed as a very positive outcome, given the workload of the people targeted with the survey.

In this section, some general observations from the survey are presented together with the results and analysis of the data collected with respect to the criteria used in relation to the main research question.

4.1. Observations from the Questionnaire

The earlier identifies issue, i.e. that the criteria were not fully orthogonal, was also identified by some of the subjects, and mentioned in their e-mail communications when submitting the questionnaire. As mentioned above, it resulted in three different approaches. One subject took a rather extreme standpoint and only gave points to two criteria, including assigning a high weighting to development cost-benefit, which is arguably related to many of the other criteria. Some subjects divided the criteria into subgroups, either based on judged importance or as a way of handling the inevitable dependencies between some of the criteria. They then assigned the criteria in a subgroup the same number of points. Finally, a third group approached the criteria without really taking the dependencies into account too much. Basically, they filled out the questionnaire from a “main criteria point of view” as was intended by the research design. In other words, they focused on each criterion’s own value rather than considering its connection to other criteria.

It was also observed that one subject allocated more than 1000 points to the criteria (1020 points), thus the points given by the subject were rescaled so that their sum became 1000. Four of the thirteen subjects suggested new criteria for the decision. These four subjects also provided points for the new criteria. The new criteria are further discussed in the following subsection, however it should be noted, given that only 25% of the subjects suggested new criteria, and that there was only a minor overlap between their suggestions, it was hard to include these new criteria when comparing how subjects allocated points to the same criteria. Thus it was decided that, in the case of the subjects who suggested new criteria, the points for the 13 criteria should also be rescaled to ensure comparability. The proposed new criteria are handled separately below.

4.2. Relevant Criteria

The responses provided by the subjects were related to whether the 13 criteria were relevant for this type of decisions or not. Most subjects regarded the criteria as relevant. More precisely, all subjects regarded seven of the criteria as relevant. For the remaining six criteria, the following results were obtained:

- Requirement's issuer: 12 subjects out of 13 viewed this criterion as relevant
- Requirement's volatility: 10
- Support for training/education: 10
- System impact: 12
- Complexity: 12
- Maintenance: 12

We have not further explored the above four situations in which only one person has felt that a particular criterion has no relevance, because we feel that more responses to the survey are needed before any further conclusions can be drawn. It is more interesting to look at the two criteria where three subjects state that the criteria are not relevant. While this is not a definitive answer in general, it should be noted that it probably is easier to say "yes" than "no". The subjects know that the researchers regard these criteria as relevant since the criteria appear in the list, and hence it is easier to agree that they are relevant than to object. Thus, it is interesting when three subjects disagree with the researchers.

The volatility of a requirement is not a relevant criterion according to three of the subjects. This may seem surprising. On the other hand, it may show that requirements are included for other reasons and that volatility has to instead be handled as part of the development project, for example, by postponing the implementation as long as possible until more is known about the requirement.

It is probably not as surprising that the support for education/training is depicted as a criterion that may not be relevant. The inclusion of a requirement is decided based on other criteria and if education/training is needed then this can be provided later.

The other interesting issue is that four subjects proposed new criteria. In total, five new criteria were proposed. One of them obtained two votes. The subjects only provided the names of the criteria and hence comments with respect to the criteria are based on interpretations of the researchers. The comments are not meant to imply that the newly proposed criteria should not be used, for example, in future surveys, although the criteria have relations to the criteria used in the survey as indicated below. The following five new criteria were proposed:

- Strategic importance/alignment (it is assumed that the subjects meant the same criterion, although one used importance and the other alignment)
  Comment: This criterion seems to be related to competitors (criterion 1), i.e. strategic positioning in relation to other competing products on the market.
- Customer value
  Comment: This value is probably partially related to stakeholders' priorities of

a requirement (criterion 3), since a requirement is highly prioritized by a stakeholder, then it ought to have a high value for that stakeholder.

- Product cost
- Comment: This cost is most likely closely related to the development cost-benefit (criterion 6), although there may be differences.
- Market technology trends
- Comment: This is related to competitors (criterion 1), and in particular the text in the motivation above regarding criterion 1 where it is stated that the criterion may be important in relation to leading edge functionality.
- Function is promised/sold
- Comment: This criterion is partially related to the requirement's issuer (criterion 2). It may be viewed as more important to keep a promise to some issuers than others.

The above list of proposed new criteria illustrates that it is very difficult to formulate an exhaustive set of criteria, particularly if criteria should be reasonably independent. For future studies, it must be decided whether any of the 13 criteria included in this study should be removed and if any of the above five new criteria should be incorporated into the list.

4.3. Importance of Criteria Today

The assignment of points for the criteria was divided into two parts: 'today' and 'future'. In this section, the outcome regarding the situation 'today' is reported. The results are presented for the two companies separately. A comparison between the companies is provided in Section 4.6.

The results for each company were aggregated by taking the sum of the points provided by each subject. The sum was then normalized to a percentage figure, which makes it possible to, for example, state which criteria contribute more than X percent to the decision. The results for Company A are shown in Fig. 1. The list of the criteria can be found in Section 3.2.

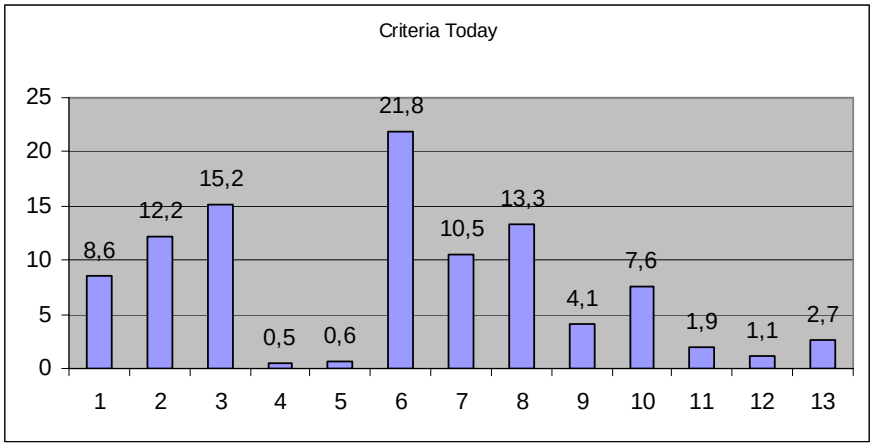

Fig. 1. Percentage values for importance of different criteria at Company A.

It is worth noting that five criteria have percentage values above 10% and six criteria have values below 5%. The results from Company A indicate that some criteria are clearly more important than others. The five most important criteria are (in order): Development cost-benefit, customer/market priority of requirement, delivery date/calendar time, requirement's issuer and resources/competencies. This indicates that issues related to specific customers/markets are important, as are traditional management aspects such as cost-benefit, delivery date and resources. The development/maintenance aspects have low influence on the decision. It is worth noting that for both companies, it has not been possible to identify any relationship between the actual job role of the respondents and their views on which criteria should be taken into consideration.

The outcome for Company B is presented in Fig. 2. The results are similar from an ordering point of view, although the actual percentage figures differ slightly. For Company B, four criteria have a value above 10% and five criteria have a value lower than 5%. The values are slightly more evenly distributed for Company B, which may be explained by the fact that one of the subjects for Company A gave almost all points to "Development cost-benefit". The figures for the two companies become even more similar when this subject was removed from the data set. The four criteria with a value above 10% for Company B are (in order): Development cost-benefit, delivery date/calendar time, customer/market priority of requirement and requirement's issuer. Basically, the only difference between the top five (except for smaller differences in the percentage values) was that criterion 2 and 3 were swapped in order of importance. However, the actual difference in percentage value is small. The patterns are very similar for the two companies when it comes to the least important criteria.

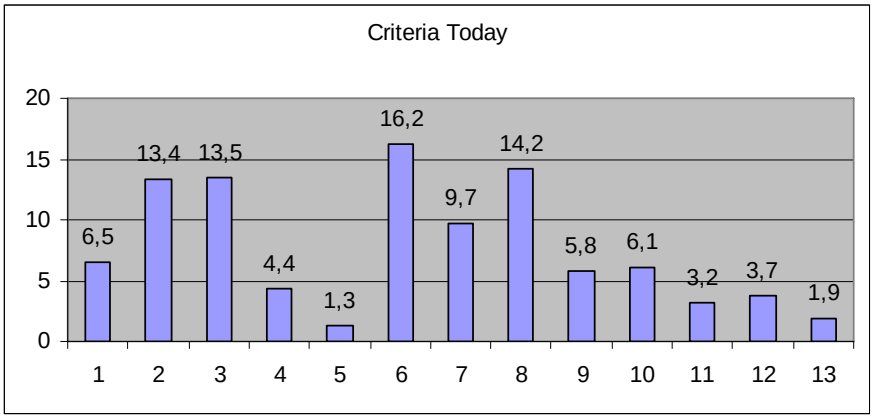

Fig. 2. Percentage values for importance of different criteria at Company B.

In summary, it is quite clear that the two companies have very similar opinions regarding what is important when deciding whether or not to include a specific requirement in the next project or release. This makes the results even more interesting than if the companies had differing opinions, because it points to the possibility of a pattern, or common trend in views, across the software development industry. This could be a first step towards identifying key criteria in the decision-making process with respect to including requirements in software projects or new releases.

As a final note, it is worth stressing that this is the picture that emerges when aggregating the prioritization from the subjects. However, at an individual level the subjects actually have quite different opinions, which are further discussed in Section 4.6.

4.4. Importance of Criteria in the Future

A similar analysis for Company A and Company B was been conducted to examine how the subjects wanted to see the use of the criteria in the future. The objective was to capture what the subjects believed would be a better balance between the criteria than the situation today. The results for Company A are presented in Fig. 3, where it can be seen that only three criteria had a percentage value at or above 10%. The three criteria are among the five ranked the highest in the previous section. The development cost-benefit is still viewed as most important, and the customer/market priority of a requirement is second. However, the gap between the two top criteria is smaller. The development/maintenance criteria (criteria 9-13) still have low values, but they are higher than in the previous section. In general, it seems like the patterns of today will remain in the future, although other criteria will be valued slightly more than today.

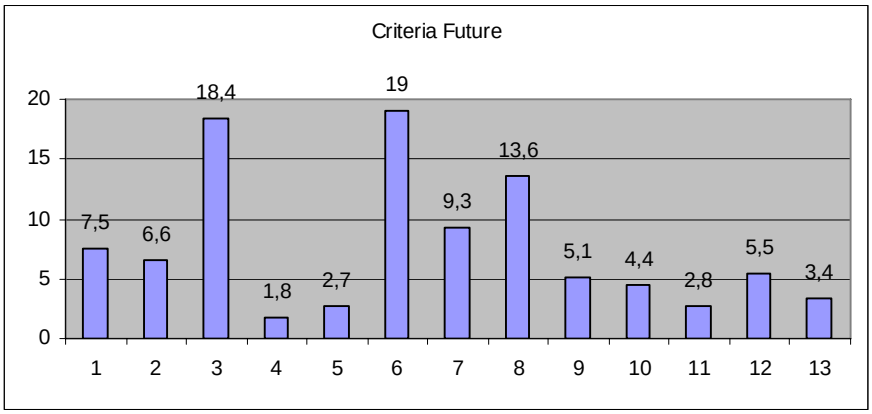

Fig. 3. Percentage values for importance of different criteria at Company A.

The results with respect to the future judgment of the criteria at Company B are shown in Fig. 4.

The trends found for Company A are also visible for Company B, although four criteria have a value of 10% or higher. In addition to the three found for Company A, the first criterion has a high score. The first criterion is related to competitors. Moreover, the order between the two highest ranked criteria has changed. Company B would like to have the main focus to lie on the customer/market priority of a requirement rather than focusing on development cost-benefit, although the latter is still very important. At the lower end, it is also possible to see for Company B that the percentages are closer to each other. In other words, more criteria ought to be used in the future than are used today. There are differences, but the patterns are similar and the differences may very well be the results of having few subjects after all.

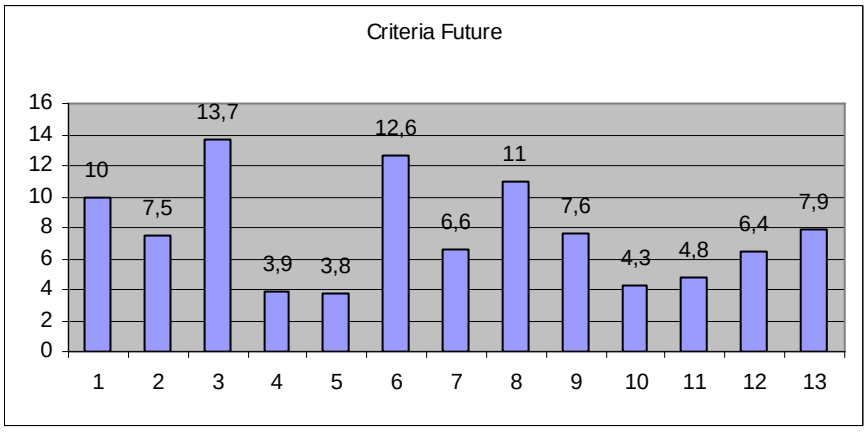

Fig. 4. Percentage values for importance of different criteria at Company B.

4.5. Analysis of Stakeholder Groups

The 13 criteria were divided into three groups in Section 3.2. It is interesting to see how the balance is between these groups and if there are any differences between the situation today and how the subjects say that it ought to be in the future.

In total, 13000 points have been awarded by the 13 subjects (1000 each). The division of these points is shown in Table 1. In this case, the main interest is to study how the importance of the different areas was judged in relation to each other.

Table 1. Division of points between different dimensions of criteria

	Today	**Future**
External market / customer	**4824**	**4503.5**
Company management	**5722**	**5157**
Development / maintenance	**2454**	**3339.5**

The results presented in the table show that the criteria related to company management issues are, and will continue to be, most important. The main difference observable from Table 1 is that there is a general opinion that the development/maintenance oriented criteria should be valued higher than it is today when it comes to decisions regarding which requirements to include in a project or release.

4.6. Individual and Company Comparison

An analysis at an individual level also has been conducted. The analysis points to the fact that there are large differences in opinions between individuals. This is supported both by a visual inspection of the collected data and a statistical analysis. The latter analysis included both a principal component analysis (PCA) and correlation analysis for one company at a time. The PCA showed three groups at each company, which indicates that the subjects represent different views. A correlation analysis yielded similar findings. Correlations between some individuals are rather high (and positive). However, some correlations between individuals are negative, although not high, which shows that there are quite different opinions among the individuals.

Based on the analysis of the opinions and views of the individuals, it is rather surprising to see common patterns at both companies as discussed in relation to Fig. 1 to Fig. 4. A possible explanation is that there are quite different opinions between individuals but, when aggregating the different views, a common pattern becomes visible on a company level. The results on an individual level point to a

need to align the opinions of what is important when deciding what to include in a specific project or release.

4.7. Validity Threats

As for any empirical study, there are some threats to the validity of the findings. The first threat is related to what the two companies represent in terms of population. The two companies have several things in common, such as development of real time systems for control purposes on an international market. This means that the companies may not be representative of all types of companies, and hence the results must be interpreted with some caution when moving away from the characteristics of the two studied companies.

On an individual level, there is a risk that it is easier to agree to relevance of the criteria than to disagree. However, this is partially taken care of by allowing the subjects to assign zero points to some criteria if they so wish. Moreover, it is easier to stick to the stated criteria than proposing new criteria. This means that important criteria may be missing, for example, the criteria mentioned by two subjects related to strategic importance/alignment.

Another potential threat is related to the questionnaire. It is always difficult to know whether the respondents have understood the questions as intended and in a similar fashion to one another. This threat is somewhat addressed by providing the outcome of the survey to the respondents so that the results can be discussed both at the respective companies and with the researchers.

The threats point to the need to analyze the other companies included in the study, although the number of subjects for the other companies vary considerably. Moreover, the threats also highlight the need for replication of this type of study.

5 Conclusions and Further Work

In this chapter, the value-based concept has been discussed and studied from the viewpoint of decision-making in requirements engineering. We have analyzed the determinants of whether or not a specific requirement should be included in a specific project/release. The inclusion or exclusion of specific requirements affects the value of the final product and hence the actual criteria for making these decisions are important to understand when discussing value-based software engineering.

The results from a survey conducted at two companies with 13 subjects representing roles such as product managers, project managers and line managers are reported. It is demonstrated that the patterns from both companies were quite similar in terms of the judged importance of different criteria, although individuals had quite different opinions of what is most important. Overall, it is agreed that *who* states a requirement is important and their priority of that requirement. Moreover, issues such as development cost-benefit, delivery date and resources available are

also important. Criteria related to development and maintenance aspects, such as complexity and system impact, have lower importance. When comparing the situation today with a judgment of how it ought to be in the future, subjects expressed the desire to weight the criteria slightly differently in the future, although the general pattern remains the same. Subjects felt that criteria related to development and maintenance ought to be more important in the future than they are today.

Future work should include improving the set of criteria based on the feedback from this study or similar studies. Replications are also needed to uncover whether the findings provide a general picture of how decisions are made in the software industry with respect to which requirements to include in a project or release.

Acknowledgment

We would like to extend our thanks to Patrik Berander, Tony Gorschek and Per Jönsson for their contribution to the brainstorming session and for reviewing the survey material. We are also grateful to the companies that have shared their experiences with us. We are particularly grateful to the contact champions and the respondents at the companies. Finally, we would like to express our gratitude to Irem Sevinc for helping us improve the English.

References

(Alwis, Hlupic and Fitzgerald, 2003) Alwis, D., Hlupic, V., Fitzgerald G.: Intellectual capital factors that impact of value creation. 25th Int. Conf Information Technology Interfaces, ITI, Cavtat, Croatia, pp 411-416, June 16-19, (2003)

(Aurum and Wohlin, 2002): Aurum, A., Wohlin, C.: Applying decision-making models in requirements engineering. Proceedings of Requirements Engineering for Software Quality, Essen Germany, 9-10 December, (2002)

(Aurum and Wohlin, 2003): Aurum, A., Wohlin, C.: The fundamental nature of requirements engineering activities as a decision-making process. Information and Software Technology, **45**(14), 945-954, (2003)

(Boehm, 2003): Boehm, B.W.: Value-based software engineering. ACM SIGSOFT, Software Eng. Notes, **28**(2), 1-12, March, (2003)

(Boehm and Huang, 2003): Boehm, B.W., Huang L.G.: Value-based software engineering: A case study. IEEE Computer Society, Computer, pp 33-41, March, (2003)

(Browning, 2003): Browning T.R.: On customer value and improvement in product development processes. Systems Engineering, **6**(1), 49-61 (2003)

(Browning, Deyst, Eppinger, and Whitney, 2002): Browning, T.R., Deyst, J.J., Eppinger S.D., Whitney, D.E.: Adding value in product development by creating information and reducing risk. IEEE Transactions on Engineering Management, **49**(4), 428-442, (2002)

(Carlshamre, 2002): Carlshamre, P.: Release planning in market-driven software product development: Provoking an understanding. Requirements Engineering **7**, 139-151, (2002)

(Carlshamre, Sandahl, Lindvall, Regnell and Natt och Dag, 2001): Carlshamre, P., Sandahl, K., Lindvall, M., Regnell, B., Natt och Dag, J.: An industrial survey of requirements interdependencies in software product release planning. Proceedings Fifth IEEE International Symposium on Requirements Engineering, IEEE, Los Alamitos CA, pp 84-92, (2001)

(Chillarege, 2002): Chillarege, R.: The marriage of business dynamics and software engineering. IEEE, Software, November, 43-49, (2002)

(Dahlstedt and Persson 2003): Dahlstedt, Å., Persson, A.: Requirements interdependencies - Moulding the state of research into a research agenda. Proceedings Ninth International Workshop on Requirements Engineering (REFSQ'03), Klagenfurt/Velden, Austria, 71-80, (2003)

(Deaton and Muellbauer, 1980): Deaton, A., Muellbauer J.: Economics and consumer behavior. New York, Cambridge University Press, (1980)

(Drucker, 1998): Drucker, P.F.: Management's new paradigm. Forbes, **162**(7), 152-177, (1998)

(Erdogmus, Favaro, and Strigel, 2004): Erdogmus, H., Favaro, J., Strigel, W.: Return on investment. IEEE Software, 18-22, (2004)

(Faulk, Harmon, Raffo, 2000): Faulk, S.R., Harmon, R.R., Raffo D.M.: Value-base software engineering: A value-driven approach to product-line engineering. 1st Int. Conf on Software Product-Line Engineering, Colorado, August, (2000)

(Favaro, 2002): Favaro, J.: Managing requirements for business value. IEEE Software, **19**(2), 15-17, March/April, (2002)

(Gordijn and Akkerman 2003): Gordijn, J., Akkerman H.: Value-based requirements engineering: Exploring innovative e-Commerce ideas. Requirements Engineering, **8**(2) 114-134, July (2003)

(Greer and Ruhe, 2004): Greer, D., Ruhe, G.: Software release planning: An evolutionary and iterative approach. Information and Software Technology **46**, 243-253, (2004)

(Harmon, Raffo, and Faulk, 2003): Harmon, R., Raffo, D., Faulk, S.: Incorporating price sensitivity measurement into the software engineering process. IEEE, Portland International Conference on Technology Management for Reshaping the World, PICMET'03, 316–323, 20-24 July, (2003)

(Harrison, 2001): Harrison, W.: Using economic value of the firm as a basis for assessing the value of process improvements. Proceedings of 26th Annual NASA Goddard Software Engineering Workshop, 123-127, 27-29 November, (2001)

(Karlsson, Wohlin, and Regnell, 1997-98): Karlsson, J. Wohlin, C., Regnell, B.: An evaluation of methods for prioritizing software requirements. Information and Software Technology, **39**(14-15), 939-947, (1997-98)

(Keller, 1995): Keller, E.: The value of software. Manufacturing Systems, ABI/INFORM Global, **13**(1), 16, January, (1995)

(Nunamaker, Briggs, De Vreede, and Sprague, 2001): Nunamaker J.F, Briggs R.O., De Vreede G.J., Sprague R.H.: Enhancing organization's intellectual bandwidth: The quest for fast and effective value creation. Special issue: Journal of Management Information Systems, **17**(3), 3-8, (2001)

(Park, 1998): Park, R.J.: A plan for invention. New York, St Lucie Press, USA, (1998)

(Prahalad and Hamel, 1990): Prahalad C.K., Hamel G.: The core competence of the corporation. Harvard Business Review **68**(3), 79-81, (1990)

(Regnell, Höst, Natt och Dag, and Hjelm, 2001): Regnell, B. Höst, M. Natt och Dag, J., Hjelm, T.: Case study on distributed prioritisation in market-driven requirements engineering for packaged software. Requirements Engineering **6**, 51-62, (2001)

(Regnell, Paech, Aurum, Wohlin, Dutoit, and Natt och Dag, 2001): Regnell, B. Paech, B., Aurum, A. Wohlin, C., Dutoit, A., Natt och Dag, J.: Requirements mean decisions! - Research issues for understanding and supporting decision-making in requirements engineering. Proceedings of First Swedish Conference on Software Engineering Research and Practice, Ronneby, Sweden, 49-52, (2001)

(Ruhe, and Greer 2003): Ruhe G., Greer D.: Quantitative studies in software release planning under risk and resource constraints. Proceedings of International Symposium on Empirical Software Engineering (ISESE), IEEE, Los Alamitos CA, 262-271, (2003)

(Ruhe, Eberlein, and Pfahl, 2003): Ruhe, G., Eberlein, A., Pfahl D.: Trade-off analysis for requirements selection. International Journal of Software Engineering and Knowledge Engineering, **13**(4), 345-366, (2003)

(Sullivan, 1998): Sullivan P.H.: Profiting from intellectual capital: Extracting value from innovation. New York: John Wiley and Sons, (1998)

(Tanaka, Aizawa, Ogasaware and Yamada, 1998): Tanaka, T., Aizawa, M., Ogasaware H., Yamada, A.: Software quality analysis and measurement service activity in the company. IEEE Proceedings of the 20th International Conference on Software Engineering, pp 426-429, 19-25 April, (1998)

(Urban, and Hauser, 1993): Urban, GL., Hauser, J.: Design and marketing of new products. Englewood Cliffs NH: Prentice Hall, (1993)

(Weinstein and Johnson, 1999): Weinstein A., Johnson, WC.: Designing and delivering superior customer value: concepts, cases, and applications. St. Lucie Press, Boca Raton, FL, USA, (1999)

(Wohlin, von Mayrhauser, Höst and Regnell, 2000): Wohlin, C., von Mayrhauser, A., Höst, M. and Regnell, B.: Subjective evaluation as a tool for learning from software project success. Information and Software Technology, **42**(14), 983-992, (2000)

Author Biography

Prof. Claes Wohlin is a professor in Software Engineering at the School of Engineering at Blekinge Institute of Technology in Sweden. Prior to this, he has held professor chairs in software engineering at Lund University and Linköping University. He has an MSc in Electrical Engineering and a PhD in Communication Systems both from Lund University, and he has five years of industrial experience. Dr. Wohlin is co-editor-in-chief of the journal of Information and Software Technology published by Elsevier. He is on the editorial boards of Empirical Software Engineering: An International Journal, and Software Quality Journal. Dr. Wohlin is the recipient of the Telenor Nordic Research Prize in 2004 for his achievements in software engineering and improvement of software reliability for telecommunication systems.

Dr. Aybüke Aurum is a senior lecturer at the School of Information Systems, Technology and Management, University of New South Wales. She received her BSc and MSc in geological engineering, and MEngSc and PhD in computer science. She is the founder and group leader of the Requirements Engineering Research Group (ReqEng) at the University of New South Wales. She also works as a visiting researcher in National ICT, Australia (NICTA). Her research interests include Management of Software Development Process, Software Inspection, Requirements Engineering, Decision Making and Knowledge Management.