11-24-2008

A Methodology to Support Software Release Decisions

Hans Sassenburg
Software Engineering Institute, hanss@sei.cmu.edu

Follow this and additional works at: http://aisel.aisnet.org/sprouts_all

A Methodology to Support Software Release Decisions

Hans Sassenburg
Software Engineering Institute, Germany

Abstract
A relatively unexplored area in the field of software management is the implementation or release decision, deciding whether or not a software product can be transferred from its development phase to operational use. Many software manufacturers have difficulty in determining the "right" moment to release their software products. It is a trade-off between an early release, to capture the benefits of an earlier market introduction, and the deferral of product release, to enhance functionality, or improve quality. In this research project software release decisions are researched from three perspectives: economics, decision-making and software management. All perspectives are reviewed, explored indepth, both from a theoretical and from an empirical point of view, by studying practical examples. The results are used in a proposed methodology to improve strategic software release decisions, characterized by the existence of large prospective financial loss outcomes, including the presence of high costs for reversing a decision. Based on validation results in a practicalsetting, it is concluded that this methodology has a descriptive and a judgmental character, and can therefore support understanding, analysing, assessing and improving the capability of software manufacturers in this problematic area.

Keywords: Software Releasing, Economics, Decison making,

citer

A methodology to support software release decisions

Hans Sassenburg

斗|Sprouts

University of Groningen
Centre of IT Economic
Research
P.O. Box 800
9700 AV Groningen
The Netherlands

CITER WP/008/HS

21 May 2006

CITER Mission

CITER is an independent research group within the Department of Economics, University of Groningen. Our research is focused on the economics of information technologies. Our research aims at understanding and analyzing the dynamics and the processes of development, distribution and implementation of information and communications technologies and improving their efficiency and effectiveness.

We investigate particular economic issues in the economics of information technologies. For instance, the differences between 'Open' and 'proprietary' technologies, the characteristics of hardware and software commercial demand and supply and the diffusion of new technologies. We also study the efficient and effective use of those technologies, how we can improve IT management and increase the benefits of investment in information technologies.

The objectives of our research are especially useful for organizations using information technologies and to firms competing in this arena, as well as to policy makers and to society as a whole.

Our research is conducted in close coopera-tion with industry, non-profit organizations and governmental partners, as our field of research is subject to frequent technological and political changes.

Contact information

University of Groningen
CITER-WSN827
P.O. Box 800
9700 AV Groningen
The Netherlands
Tel. +31-50-363-3721
info@CITER.nl

Projects and main venues of research

- Cost/benefit management of IT.
- Decision-support methods for implementation decisions within organizations.
- Evaluation of legacy systems.
- Innovation and technical change in ICT.
- 'Open' vs. 'proprietary' software modes of development.
- Software patenting and appropriation strategies.
- Tools and strategies for the IT Control Officer.

Sponsors

UWV

Getronics PinkRoccade

Researchers

Prof. dr. E.W. Berghout	e.w.berghout@rug.nl
Drs. Ing. A.L. Commandeur	a.l.commandeur@ rug.nl
C.E. Elsenga, MSc	c.e.elsenga@rug.nl
Dr. E. Harison	e.harison@rug.nl
Dr. ir. M.H. Nijland	m.h.nijland@lse.ac.uk
Prof. dr. P. Powell	mnspp@management.bath.ac.uk
Dr. T.J.W. Renkema	t.j.w.renkema@rug.nl
Dr. Ir. H. Sassenburg	hanss@sei.cmu.edu
P.M. Schuurman, MSc.	p.m.schuurman@rug.nl
E.J. Stokking, MSc.	e.j.stokking@rug.nl
S. Orie, MSc.	sieraadj@asset-control.com

citer

A Methodology to Support Software Release Decisions

Hans Sassenburg
Software Engineering Institute
An der Welle 4
D-60322 Frankfurt, Germany
+41 (033) 7334682

hanss@sei.cmu.edu

ABSTRACT
A relatively unexplored area in the field of software management is the implementation or release decision, deciding whether or not a software product can be transferred from its development phase to operational use. Many software manufacturers have difficulty in determining the 'right' moment to release their software products. It is a trade-off between an early release, to capture the benefits of an earlier market introduction, and the deferral of product release, to enhance functionality, or improve quality. In this research project software release decisions are researched from three perspectives: economics, decision-making and software management. All perspectives are reviewed, explored in-depth, both from a theoretical and from an empirical point of view, by studying practical examples. The results are used in a proposed methodology to improve strategic software release decisions, characterized by the existence of large prospective financial loss outcomes, including the presence of high costs for reversing a decision. Based on validation results in a practical setting, it is concluded that this methodology has a descriptive and a judgmental character, and can therefore support understanding, analysing, assessing and improving the capability of software manufacturers in this problematic area.

Categories and Subject Descriptors
D.2.8 [**Software Engineering**]: Metrics – *process metrics, product metrics.*
K.6.3 [**Management of Computing and Information Systems**]: Software Management – *software development, software maintenance.*

General Terms
Management, Measurement, Economics, Reliability.

Keywords
Software releasing, economics, decision-making.

1. INTRODUCTION
There are many (indefinite) points of evaluation along the life-cycle of a software product. The various milestones in between the life-cycle stages in particular, draw the attention of researchers and practitioners in the software engineering disciplines. Important milestones are the upfront investment appraisal, the implementation or release decision, and disinvestment in an operational software product [6]. A relatively unexplored area in the field of software management is the implementation or release decision, deciding whether or not a software product can be transferred from its development phase to operational use. A release decision is a trade-off where, in theory, the objective is to maximize the economic value. Inputs into the release decision are expected cash inflows and outflows if the product is released. In a practical setting, the decision to release a software product can be a problem, best illustrated with examples:

☐   In practice, cost and time constraints will normally be present in retrieving complete and reliable information. This search for information should be taken into account as an economic activity with associated costs and time. This leaves the software manufacturer with the problem of finding the optimal level of information, where marginal value equals marginal costs and thus marginal yield is zero. Gigerenzer holds this optimal level is difficult, if not impossible, to find [1].

☐   Decision-making in the real world is often unstructured [5], and normally involves various stakeholders, and there might, for example, be reasons to release a system or software product, due to political or business pressures, even though knowing it still contains defects. A study of spacecraft accidents, for example, reveals that, although inadequate system and software engineering occurred, management and organizational factors played a significant role, including the diffusion of responsibility and authority, limited communication channels and poor information flows [4].

☐   Research has revealed there are many obstacles to the successful implementation of almost any decision [5], including:

   -   The reduced importance of a decision once it is made and implemented.

   -   The control of the outcome of a decision by stakeholders not involved in its making.

   -   The development of new situations and problems to command the attention of the decision-makers once the choice has been implemented.

In this research project these different perspectives were reviewed, explored in-depth, both from a theoretical and from an

empirical point of view, by studying practical examples. The results are used in a proposed methodology to improve strategic software release decisions, characterized by the existence of large prospective financial loss outcomes, including the presence of high costs for reversing a decision. Based on validation results in a practical setting, it is concluded that this methodology has a descriptive and a judgmental character, and can therefore support understanding, analysing, assessing and improving the capability of software manufacturers in this problematic area.

2. EXPLORATORY CASE STUDIES

2.1 Introduction

Seven exploratory case studies were conducted. The selected environments varied with respect to the software manufacturer types (custom system written in-house versus commercial software), geographical locations (The Netherlands and Switzerland), the product version developed (new product versus new version of existing product), and the process maturity level (ranging from CMMI level 1 to 3). The aggregated results are discussed in the next subsection (see [7] for a broader and more detailed overview and discussion).

2.2 Aggregated Case Study Results

Aggregating the results of the exploratory case studies leads to four main identified problem areas:

1. *Definition of the release criteria.* Documented and commonly-accepted product development strategies were not common in the cases studied. Not having consensus among stakeholders about priority setting in a product development strategy could imply that stakeholders do not work towards a common goal. It leaves room for self-imposed controls and restrictions, and performing activities (costs) that add no value.

2. *Information about the implemented values of the release criteria.* In all cases, information as input to the decision-making process was incomplete. Two examples are:

   - In most cases non-functional requirements were not broken down during product development to subsystems and/or lower level components. It was only during testing that reliability again received attention, which may be too late to guarantee a high reliability level. The level of maintainability obtained was not addressed.

   - Information on the availability of relevant documentation and the quality of this documentation was limited in a number of cases.

   As a result, organizations faced difficulty in making firm statements about expected post-release maintenance costs.

3. *Decision-making process.* The process descriptions found did not explicitly focus on software release decisions. Through the questionnaires, and during interviews, informants confirmed that no formal collective decision-making process for release decisions was available, but that their organisation probably would benefit from such a process by creating transparency on responsibilities (who), activities (what), timing (when), and support methods (how).

4. *Implementation of the release decision.* The process descriptions found paid no or limited attention to the implementation of the release decision, once it was made. Although, in all cases, corrective actions were implemented for defects found after the release decision implementation, most cases revealed the absence of an institutionalized process to analyse the defects found and evaluate the business case, or project, afterwards to supplement organizational knowledge. This makes it difficult to plan expected post-release maintenance costs for future projects based on prior experience, and prevents the identification of areas for improvement.

The problem areas identified in these exploratory case studies corroborate the need for a formal process to support software release decisions.

3. STRATEGIC DECISION SUCCESS

A formal process offers a structured mechanism to provide visibility of threats to release decision success. The net result of a formal approach is to help avoid preventable surprises late in the project, and improve the chance of meeting initial project commitments, and reducing the level of uncertainty. Reducing uncertainty has a cost, which should be balanced against the potential cost a software manufacturer could incur if the uncertainty is not reduced. It may not be cost-effective to try and reduce uncertainty too much. Formal approaches are of special concern when common interests increase, and when strategic value is present. In this study, a decision is considered as being of strategic value when large prospective financial loss outcomes to a software manufacturer and its customers/end-users of the software are present [3]. This is often true for software release decisions due to high costs for reversing the software release decision once made. Strategic value also has a long-term character as prospective loss outcomes may arise long after the decision has been made (for example, in cases where liability issues lead to lawsuits). Decisions with strategic value should be made at a high level of the organization, require a formal decision-making process, and should be of concern to top management [2]. Routine software release decisions, without strategic value, can be handled with a higher degree of certainty, and should be left to management at tactical, or even operational, level. Strategic software release decisions require a formal, collective decision-making process. Decision-making is defined as the combined activity of comparing alternatives and the act of choice. However, Harrison divides a decision-making process into six functions; broadening the scope with preceding and proceeding activities, as illustrated in Figure 1 [2].
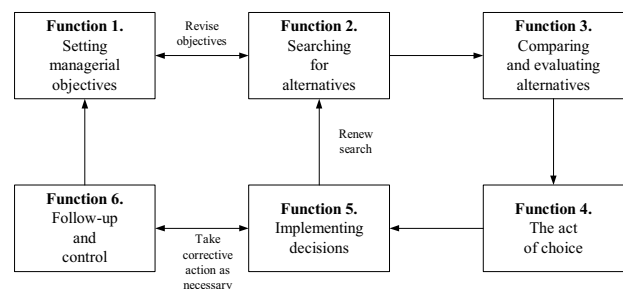

Figure 1. Components of a Decision-making Process [2].

In this framework, decision-making is illustrated as a dynamic process. Decision-making is considered to be a non-linear, recursive process. That is, most decisions are made by moving back and forth between the choice of criteria or objectives (the characteristics the choice should meet) and the identification of alternatives (the possibilities one can choose from). The alternatives available influence the objectives applied, and similarly the objectives defined influence the alternatives to be considered. Other conditions increasing the likelihood of strategic decision success are [2]:

1. *Decision-making process*. The primary factors here are the availability of well-defined, *attainable objectives* (Condition 1) as opposed to unattainable objectives and a mindset toward an *open decision model* (Condition 2), giving weight to the environment (dynamic objectives, imperfect information, time and cost constraints, cognitive limitations), opposed to a closed decision model.

2. *Decision*. The primary factors here are a *judgmental decision strategy* (Condition 3): choosing an alternative based on judgment applied to information that is imperfect, instead of a computational strategy and the search for a *satisficing outcome* (Condition 4): strong preference for a desirable result; complemented by an acceptance of less-than-perfect knowledge about the outcome, meeting the defined objectives instead of a maximizing outcome.

4. METHODOLOGY

4.1 Overview

In this framework, four process areas in the software release decision-making process are distinguished, each addressing the process from different perspectives. These process areas match problem areas identified for software release decisions, as discussed in section 2.
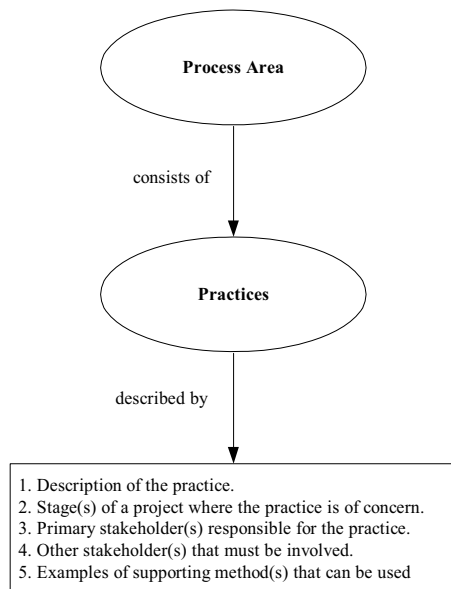

Figure 2. Structure of the Methodology [7].

A process area is defined as a cluster of related practices which, when performed collectively, achieve a set of goals considered important for establishing process capability in that area. See Figure 2. The next step in designing the methodology is the identification of relevant practices for each process area, which should describe 'what' is to be accomplished (general guidelines) but not 'how'. Taking this approach, the descriptions of practices still offer the possibility for interpretation and customization to the external market environment, and to internal strategic and functional characteristics of a software manufacturer organization.

The identified process areas are:

1. *Release Definition*. Decision-making is mainly viewed from a quantitative perspective, assuming that information is near to perfect: complete and reliable. It emphasizes the *maximizing behaviour* approach with emphasis on the mathematic, economic and statistic disciplines. In software release decisions, decision-making from a quantitative perspective is concerned with the definition and control of a product development strategy: setting the managerial objectives with their priorities (Function 1), and ensuring they are attainable (Condition 1). The availability of a product development strategy will enable the comparison/evaluation of different release alternatives (Function 3), thus answering the question: which alternative maximizes economic value?

2. *Release Information*. This process area is concerned with the search for alternatives (Function 2) during product development, for example, the identification and collection of information that is needed to compare and evaluate different release alternatives. This search is derived from the formulated product development strategy. Decision-making is also viewed from a quantitative perspective, but with the recognition that information is imperfect in the sense that not everything can be expressed in numbers, and that information has its price, in time and money. For this process the mathematic, economic and statistic disciplines still play an important role, but the maximizing behaviour approach is extended with an *optimizing behaviour* approach: what is the optimal volume of information? Insufficient information increases uncertainty and hampers the decision-making process, whereas too much information is a waste of scarce resources; there is an optimum above which the cost for searching for more information exceeds the benefits.

3. *Release Decision*. Decision-making is viewed from a psychological, sociological and socio-psychological perspective, addressing factors that influence individual and group behaviour. It recognizes the imperfections of information, and stakeholders, involved in the act of choice (Function 4), will possibly have different preferences with respect to the decision outcome; an open decision-making process (Condition 2). The challenge is to use a judgmental strategy (Condition 3) to reach a decision outcome that meets the objectives formulated, and is agreeable to all stakeholders involved. The concept of optimizing behaviour is extended with a *satisficing behaviour* approach (Condition 4): which outcome satisfies the needs of all stakeholders involved?

4. *Release Implementation*. Decision-making is viewed from an implementation perspective once a decision has been made and is implemented (Function 5), assuming a successful decision requires follow-up and control (Function

6) of the implemented decision. For software release decisions, it is necessary to identify the factors that ensure congruence between the expected and the actual outcome. To increase organizational learning, the decision-making process and its outcome should be evaluated.
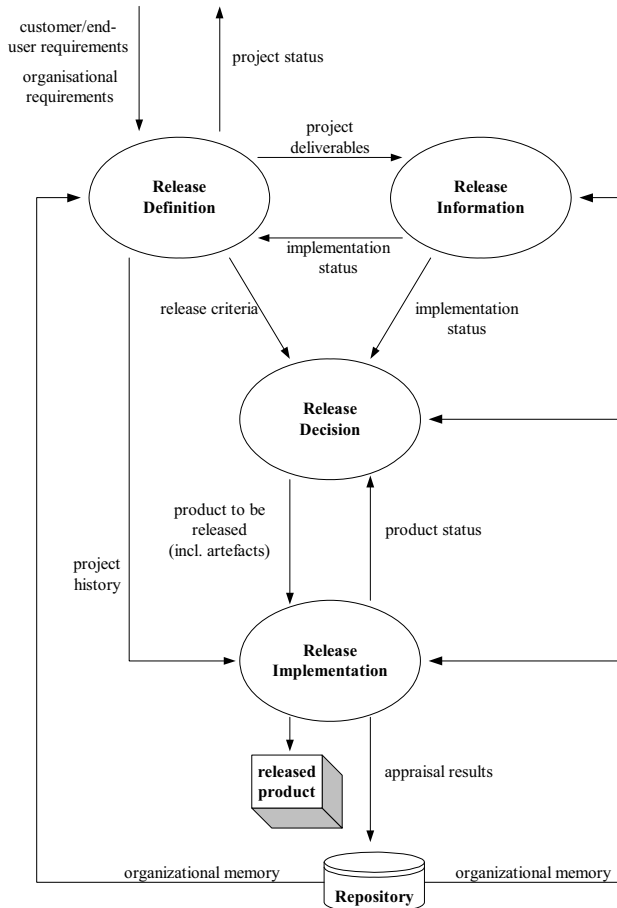

Figure 3. Overview of the Methodology [7].

In Figure 3, the data-flow-diagram of the methodology is illustrated, combining the four identified process areas. In Figure 3, the underlying practices of each process area are summarized.

4.2 Properties

The designed methodology implements all inter-related functions of managerial decision-making and meets the conditions for strategic decision success. Implementation of all practices of the methodology ensures that all relevant stakeholders are actively involved before a project is started (proposal phase) and stay involved until the released product functions well in its operational environment. This is an important property of the methodology, as it ensures product development is continuously discussed among stakeholders representing different perspectives. This multi-perspective approach enables the sharing of knowledge among stakeholders and, where problems arise, all perspectives are represented in evaluating an alternative course of action. Specific advantages are illustrated by examples:

Table 1. Methodology – Process Areas and Practices [7].

Process Area: Release Definition	
Project Objectives	Define product development strategy
Project Control	Control the project's progress with respect to the product development strategy
Uncertainty Management	Identify sources of uncertainty and implement effective measures to reduce or eliminate them
Selection of Alternatives	Select alternatives that most closely meets the product development strategy
Process Area: Release Information	
Verification Definition	Define in which way the correct implementation of the functional requirements and non-functional requirements is verified
Verification Implementation	Deploy activities to verify the correct implementation of the functional requirements and non-functional requirements using the available definitions
Artefact Definition	Identify which artefacts related to the product are to be developed to support future maintenance and exploitation activities
Artefact Implementation	Deploy activities to implement the identified artefacts
Process Area: Release Decision	
Information Perfection	Assure that the completeness and reliability of the information is high enough to reduce uncertainty to an acceptable level without overspending resources
Aspiration Levels	Reduce differences in opinions through the sharing of convincing information
Stakeholder Involvement	Involve all stakeholders throughout the project, especially in the release decision
Decision Choice	Apply a negotiated decision-making strategy, and reach a state of mutual agreement among the stakeholders using consensus as the decision rule (interacting group type)
Process Area: Release Implementation	
Maintenance Budget	Reserve a maintenance budget for corrective maintenance actions in case problems are encountered during the product rollout
Product Rollout	Carefully monitor the implementation of the released product and take appropriate corrective actions in case of encountered problems
Project Discharge	Officially discharge the Project Steering Committee responsible for the development of the product and the implementation of the released product from these responsibilities when all obligations have been met
Project Appraisal	Appraise the important aspects of the project (for instance the identification of the reasons for discrepancies between initial project objectives and actual results, the identification of strengths and weaknesses to augment the software manufacturer organization's memory (repository) as a source for increasing its capabilities

☐ Involving the Maintenance department during the project proposal phase helps define a product development strategy that includes important post-release requirements of the product. The involvement of maintenance, an important stakeholder in the release decision-making process, is considered crucial, as they are responsible for the decision implementation.

□ All stakeholders remain involved once the release decision has been made. This is especially important for the Development project, which is only discharged from its responsibilities when the product is proven stable. This prevents the organization from assigning development resources to other projects before the actual outcome meets the expected outcome.

□ Senior management is assigned both responsibility and involvement during the various stages. This is important as the release decision and its successful implementation are of strategic value to the organization and requires the involvement of higher management.

5. VALIDATION RESULTS

5.1 Validated Properties

The process areas of the methodology cover the important aspects of strategic software release decisions: defining and controlling the product development strategy (*'Release Definition'* process area), defining and acquiring the information needed as input for the release decision (*'Release Information'* process area), establishing a broad basis for the release decision outcome (*'Release Decision'* process area), and establishing congruence between the expected and actual release decision outcomes and determining lessons learned (*'Release Implementation'* process area). Both the descriptive and judgmental character of the methodology were validated in the cases studied.

□ On the descriptive character of the methodology, the following conclusions are drawn. When the information level is too low, uncertainty is high and this is likely to have a negative impact on post-release cash outflows (corrective maintenance due to limited verification). This may lead to differences in aspiration levels and is likely to reveal challenges amongst stakeholders instead of sharing convincing information. This was confirmed in case study A, as in Figure 4a (outer circle equals a 'High' score, middle circle 'Medium' and the inner circle 'Low'). When information increases, uncertainty is reduced and this is likely to have a positive impact on post-release cash outflows (increased verification and artefacts). Differences in aspiration levels are reduced or even eliminated and the decision-making process is likely to reveal the sharing of convincing information to reach consensus about the decision outcome. This was confirmed in the other case studies B and C, as in Figure 4b and 4c respectively.

□ On the judgmental character of the methodology, the following conclusions are drawn: Decision success requires a high quality for the decision-making process and for decision implementation. In this way, it is likely there will be congruence between the expected and actual outcome, in meeting the objectives that gave rise to the decision. This was confirmed in all cases. Case study A revealed a low quality for the decision-making process, and a relatively high quality for release implementation, however the original objectives are not met. The two other case studies revealed a high quality for the decision-making process and release implementation, both meeting the original project objectives. It is concluded that the judgmental character as an assumed property of the methodology is validated in a practical context. Using the proposed methodology, the quality of the decision-making process and quality of decision implementation can be determined, offering the possibility of assessing the strengths and weaknesses of strategic software release decisions. This judgmental character of the methodology offers the possibility of identifying areas of improvement, and meets the primary research objective of this study.
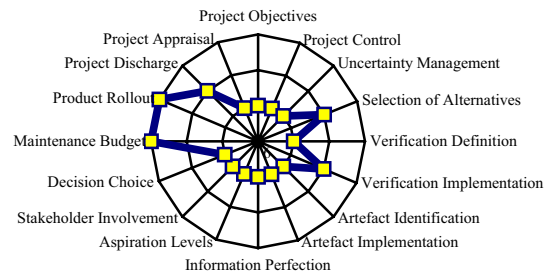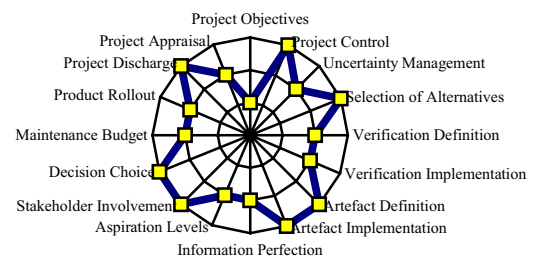

Figure 4a. Practice-scores case study A [7].

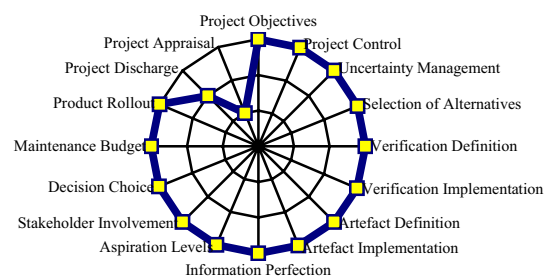**Figure 4b. Practice-scores case study B [7].**

Figure 4c. Practice-scores case study C [7].

5.2 Added Value

When comparing the methodology with project management methodologies, development methodologies, standards and models, some overlap can be observed: defining the project objectives and controlling the project's progress during its

execution. However, the methodology offers added value by explicitly recognizing that:

- there needs to be a clear rationale for a project throughout its existence;
- information has its price in time and money;
- there is a need to reduce the aspiration levels of all stakeholders involved early during product development, and find consensus amongst all stakeholders when making the release decision, and
- product development only ends when the product has been successfully rolled out and lessons learned have been collected.

6. VALIDITY OF STUDY RESULTS

For the external validity of the results to a wider context beyond the cases studied, the following conclusions are drawn:

☐ *Generalization of results to similar and other software manufacturer types*. The first question to be answered is the extent to which the descriptive and judgmental character of the methodology can be generalized, to similar and other software manufacturer types. The case studies selected are software manufacturer types developing software products for internal use. A review of the methodology indicates no practices specific to a software manufacturer type. The cases studied revealed environments with high pressure on both time and quality, in relatively turbulent environments (especially one case), similar to environments in which, for example, customer software or mass-market software is developed. It is therefore considered that no major obstacles exist in successfully applying the methodology to other similar or different software manufacturer environments.

☐ *Generalization of results to more routine software release decisions*. The second question that arises is whether the conclusions are restricted to strategic software release decisions (non-routine decisions). The methodology has been designed for strategic software release decisions. As discussed in section 2, routine decisions should not be the concern of higher-level management, and can probably be made at operational level. As such, the need for the establishment of a Project Steering Committee at tactical level to control the project, with involvement of Senior Management at strategic level, is limited for more routine release decisions, as controversial issues between different stakeholders, requiring a negotiated decision-making strategy addressing the perspective of satisficing behaviour, is less likely. This methodology can be considered for more routine software release decisions, however for each practice it must be carefully considered if its implementation gives sufficient added value and whether the involvement of higher management levels is required.

☐ *Generalization of results to other product development decisions*. A third question that arises is whether the conclusions are restricted to [strategic] software release decisions. Could, for example, the methodology also be used for investment decisions or product design decisions; important milestones during product development? Although the methodology has been designed for strategic software release decisions, its general nature makes this worth considering. The methodology focuses on the decision-making process (*'Release Decision'* process area), extending it with defining and controlling the decision objectives (*'Release Definition'* process area), the definition and collection process of information as input to the decision-making process (*'Release Information'* process area), and the implementation and evaluation of the release decision (*'Release Implementation'* process area). These are common aspects of decision-making and usage for other product development decisions can, therefore, be considered. The underlying practices should, for such cases be revised to focus more specifically on the decision type considered.

Ongoing research is planned to investigate the completeness of the methodology. Organizations interested in participation are invited to contact the author.

7. REFERENCES

[1] Gigerenzer, G., Striking a Blow for Sanity in Theories of Rationality. In *Essays in honor of Herbert Simon,* Augier, B., March, J.D., (Eds.), Cambridge, MA: MIT Press, 2004.

[2] Harrison, E.F., *The Managerial Decision-Making Process.* Houghton Mifflin Company, 1987.

[3] Kunreuther, R.M., et al., High Stakes Decision Making: Normative, Descriptive and Prescriptive Considerations. *Marketing Letters, 13, 3,* 2002, 259-268.

[4] Leveson, N.G., The Role of Software in Spacecraft Accidents. *AIAA Journal of Spacecraft and Rockets, 41, 4,* 2004.

[5] March, J.G., Olsen, J.P., *Ambiguity and Choice in Organizations.* 2nd Edition, Universitetsforlaget (Norway), 1979.

[6] Sassenburg, H., Berghout, E., A managerial release-decision model for IT-applications. *Proceedings of the 11th European Conference on the Evaluation of IT,* November 11-12, Amsterdam (The Netherlands), 2004.

[7] Sassenburg, H., *Design of a Methodology to Support Software Release Decisions: Do the Numbers really Matter?* PhD thesis, University of Groningen (The Netherlands), 2006.