Improving Business Value Assurance in Large-Scale IT Projects—A Quantitative Method Based on Founded Requirements Assessment

GILBERT FRIDGEN, University of Bayreuth
JULIA KLIER, University of Regensburg
MARTINA BEER and THOMAS WOLF, FIM Research Center, University of Augsburg

The probability of IT project failures can be mitigated more successfully when discovered early. To support a more insightful management of IT projects, which may also facilitate an early detection of IT project failures, transparency regarding a project's cash flows shall be increased. Therefore, an appropriate analysis of a project's benefits, costs, requirements, their respective risks and interdependencies is inevitable. However, to date, in requirements engineering only few methods exist that appropriately consider these factors when estimating the ex ante project business case. Furthermore, empirical studies reveal that a lot of risk factors emerge during the runtime of projects why the ex ante valuation of IT projects even with respect to requirements seems insufficient. Therefore, using the Action Design Research approach, we design, apply, and evaluate a practicable method for value-based continuous IT project steering especially for large-scale IT projects.

Categories and Subject Descriptors: K.6.1 [**Management of Computing and Information Systems**]: Management techniques

General Terms: Economics, Management, Measurement

Additional Key Words and Phrases: Requirements engineering, business value of IT, IT project controlling, value assurance, Action Design Research

ACM Reference Format:
Gilbert Fridgen, Julia Klier, Martina Beer, and Thomas Wolf. 2014. Improving business value assurance in large-scale IT projects—A quantitative method based on founded requirements assessment. ACM Trans. Manag. Inform. Syst. 5, 3, Article 12 (August 2014), 17 pages.
DOI: http://dx.doi.org/10.1145/2638544

1. INTRODUCTION

Companies continuously increased their IT investments over the last decades. Especially the number and complexity of large IT projects is growing. The complexity itself is intensified by dependencies within one or between different projects and processes and is boosted even further by the growing number of large IT projects. Another important influence is the rising uncertainty in an increasingly dynamic project management environment. These developments have implications for IT projects success. To cope with these challenges, requirements engineering (RE) concentrates on design decisions and interventions by capturing, sharing, representing, analyzing, negotiating, and prioritizing requirements in recent years (cf. Zave [1997], van Lamsweerde [2000],

---

Authors' addresses: G. Fridgen, University of Bayreuth, Friedrich-von-Schiller-Str. 2a, 95444 Bayreuth, Germany; J. Klier, Department of Management Information Systems, University of Regensburg, Universitätsstr. 31, 93053 Regensburg, Germany; M. Beer and T. Wolf, FIM Research Center, University of Augsburg, Universitätsstr. 12, 86159 Augsburg, Germany.

DOI: http://dx.doi.org/10.1145/2638544

Cheng and Atlee [2009], and Jarke et al. [2010]). Based on the evolution of IT, new opportunities and challenges in the field of RE emerge. Jarke et al. [2010], for example, state that “the environment in which RE is practiced has changed dramatically” and, therefore, reveal demand for new ways to manage requirements. In that context, modern software development processes and especially methods of agile software developments allow for the ongoing verification and update of these requirements.

However, despite the scientific achievements in the context of RE, there are still a significant portion of IT projects that fail in the way that they run out of time, budget, or do not generate the planned value. According to a recent study by the IT Governance Institute [ITGI 2011] about one out of five investments into IT is terminated before implementation. A study by the Project Management Institute [PMI 2012] found that despite the fact that organizations increasingly applied a variety of means to manage their projects, still 36% of projects did not successfully meet their initially objectives or business intent in 2011. Flyvbjerg and Budzier [2011] found that, on average, IT projects overrun their budgets by 27%. The question is why companies still fail to achieve the successes initially expected from these IT projects.

Among other reasons, unexpected economic risk factors that emerge during the runtime of projects cause budget and time overruns and consequently those high termination rates. Those risk factors lead to the late conclusion that—in contrast to prior expectations—anticipated results cannot be achieved [ITGI 2011]. In that context, Flyvbjerg and Budzier [2011] found that the continuous measurement and controlling of expected projects benefits (beyond costs) seems to be positively related to IT project success. However, this insight is often not considered in practice to date: If requirements are reconsidered during the runtime of a project, then typically because of technical or cost reasons (e.g., “Which features are feasible with the limited budget?”). Financial dependencies between different project parts as well as the measurement of expected projects benefits are mostly neglected so far. Moreover, there is a lack of methods to compare the current financial project status with the ex ante valuation of the IT project (e.g., regarding the realized benefits). In many situations, if companies have decided to make a project once, they continue the project even if financial environments have changed.

While scientific literature on RE and project management methods primarily focuses on technical aspects [Jarke et al. 2010] or on the financial ex ante valuation of IT projects [Walter and Spitta 2000; Wehrmann and Zimmermann 2005] the continuous value-based management of IT projects (also with respect to requirements) is mostly neglected so far. To be able to identify emerging risks during the runtime of projects early and to counteract reasonably, processes and methods for a continuous value-based IT project steering are necessary, which as of today, to the best of our knowledge, are missing within scientific literature. Thus, based on the first idea presented by Fridgen and Heidemann [2013], the aim of this article is to develop a method for a continuous value-based IT project steering especially for large-scale systematically assessable IT projects. Our approach helps companies in their strive to measure the current success of an IT project during its lifecycle, allows them to provide a control mechanism, and to make future-oriented decisions.

2. METHOD

2.1. Action Design Research

For the development of a method for continuous value-based IT project steering, we decided to draw on Action Design Research (ADR), a design research method that has been developed by Sein et al. [2011]. The ADR method is based on different stages as well as corresponding principles that guide the research process [Sein et al. 2011]. In contrast to other design research methods [cf., e.g., March and Smith [1995] and Peffers et al. [2008]), ADR does not separate and sequence the design of an artifact and its evaluation in a "build and *then* evaluate" cycle [Sein et al. 2011]. ADR rather supports ensemble artifacts that "emerge from the contexts of both their initial design and continual redesign via organizational use" [Sein et al. 2011]. Thus, the simultaneous development and evaluation of an artifact, which is done in mutual cooperation between practitioners and researchers, is a specific characteristic of this research method. Since the actual perception of a method for continuous value-based IT project steering by decision makers and its acceptance in business practice cannot be investigated solely driven by theories without actively engaging organizations (cf. Beer et al. [2013]), we believe that ADR is especially well suited to our problem because of three reasons. First, ADR supports research driven by design theories and inspired by problems from practice (stage 1, "problem formulation") that allows for an organization dominant building, intervention, and evaluation of artifacts (stage 2, "building, intervention, and evaluation") [Sein et al. 2011]. Therefore, ADR helps us to structure and guide the initial development of a novel method for continuous value-based IT project steering driven by the need of our business partners and the lack of suitable approaches in theory, its improvement by "reciprocal shaping" and "mutually influential roles" using the expertise of researchers and practitioners, and its concurrent evaluation of the artifact by the promptly use of the new method by practitioners (alpha version) and end users (beta version) within an organizational context. Second, as we create a completely novel method ADR helps to reflect on the design of the artifact ("guided emergence") and to "generate and evolve design principles" that partly might have been already derived in stage 1 (stage 3, "reflection and learning") [Sein et al. 2011]. Third, ADR ask for a generalization of outcomes from the "specific-and-unique to generic-and-abstract" (stage 4, "formalization of learning") [Sein et al. 2011]. Thus, we believe that ADR allows us to derive general recommendations that help to further improve project-steering methods in general.

2.2. Research Setting

In order to avoid IT failures and due to the lack of scientific rigor methods being available and especially easy applicable in practice, there is a need of companies for IT project steering. Therefore, we design, apply, and evaluate a practicable method for continuous value-based IT project steering in collaboration with one of the world's leading strategy consulting companies (in the following referred to as CC in the following).We were gathering feedback from practice regarding efficacy and applicability of the method on a regular basis. Besides the feedback from practice, we also continuously took scientific literature into account when designing the method to uphold the scientific rigor. In addition, we tested the developed method at an industrial client, namely a multinational manufacturing company (referred to as MC in the following), who used the method to IT project steering of multiple mobile app development projects. Although mobile app projects are rarely large-scale IT projects, we were able to apply and evaluate our artifact in this IT project context too. Furthermore, with respect to the evaluation, we were able to gather additional qualitative feedback from the CC that applied our method (at least in parts) in three more IT projects. It may be argued, that our case study gives back just qualitative feedback and insights on our method in a first step. However, according to common literature (e.g., Dubé and Paré [2003]) qualitative feedback in case studies (e.g., gathered by interviews) is a validate approach, which also brings rigor and flexibility to case studies concerning the complex field of IS [Dubé and Paré 2003, p. 598]. We, therefore, draw on this approach for a first but indispensable step toward the evaluation of our method.

In sum, the valuable feedback in different evaluation cycles of both business partners—CC (alpha cycle and beta cycle) as well as MC (beta cycle)—gave us the opportunity to satisfy the criteria of ADR and to develop an artifact, which fulfills the requirements of all stakeholders from business practice and science. Figure 1 shows the ADR approach based on the depiction in Sein et al. [2011], adjusted to our specific research setting.

The remainder of this paper is organized in accordance with the previously mentioned stages. We first outline the theoretical foundations and the specific practical need of our research (stage 1). Subsequently, we describe the building, intervention, and evaluation that finally led to our method for continuous value-based IT project steering (stage 2). Afterward, we reflect on our findings (stage 3) and generalize by deriving design principles for a continuous value-based IT project steering for decision makers in the context of project management (stage 4). In the last section, we summarize our results, point out limitations, and suggest areas for further research.

3. PROBLEM FORMULATION

The management of large scale IT projects in an increasingly dynamic and complex project environment is a challenging task for decision makers in companies [Denne and Huang 2003]. Although IT management processes, methods, and techniques have improved significantly over the last couple of years—in the context of RE methods for agile software development allow for example for easily changeable requirements associated with the evaluation of potential changes [Ernst et al. 2012]—there are still a high number of "out-of-control tech projects" that fail [Flyvbjerg and Budzier 2011] in the way that they run out of time, budget, or do not generate the planned value. Flyvbjerg and Budzier [2011], for example, analyzed 1,471 projects and found that on average they overrun their budget by 27%, on average, and one out of six projects even by 200%. Recognizing this risk, there is a specific practical need of companies for techniques in order to avoid these IT failures. During different interviews with CC, they specified this need for a methodically sound as well as easy to use and practical applicable method of a continuous value-based IT project steering for especially large IT projects. Our method may be more influencing on large-scale IT projects, as in this context complexity and risks are usually higher. Nevertheless, it can be applied to different kinds and sizes of IT projects because we draw on a generic approach, but the type and extent of application is subject to further research.

In theory, RE is an acknowledged phase within every IT project's lifecycle [Pohl 1993] and an important factor for the success of an IT project [Cheng and Atlee 2009]. Thereby, RE can be seen as a process to identify the purpose, a certain IT project has to fulfill [Nuseibeh and Easterbrook 2000]. It is realized by the analysis, documentation, communication, and implementation of the IT projects' stakeholders needs, also known as requirements [Nuseibeh and Easterbrook 2000]. One of the demises of RE is the fact that some of a projects' requirements may change during the projects' lifecycle and, therefore, are hard to manage and may lead to increased payoffs [Nuseibeh and Easterbrook 2000; Cheng and Atlee 2009]. To address changing requirements in the context of software development, Denne and Huang [2003], for instance, develop an incremental funding methodology that values timely and incremental subfunctionalities. Another challenge of RE is to create a strong alignment between science and practice, which has become more and more important due to the changing economics of RE [Jarke and Lyytinen 2010]. These changing economics of RE can be seen in the increasing number of large business and technical systems, which need a more rigorous analysis of Return-on-Investment (ROI) [Jarke and Lyytinen 2010].

4. BUILDING, INTERVENTION AND EVALUATION (BIE)

4.1. Alpha Cycle

Based on the results of the problem formulation stage, the ADR team aimed to develop a method for a continuous value-based IT project steering. In this context, the objective of

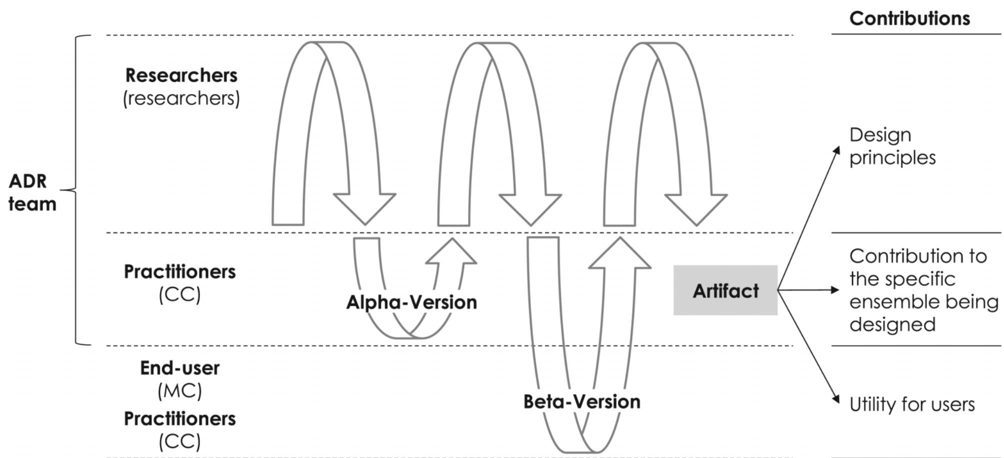

Fig. 1. Building, intervention and evaluation scheme in ADR (cf. Sein et al. [2011]).

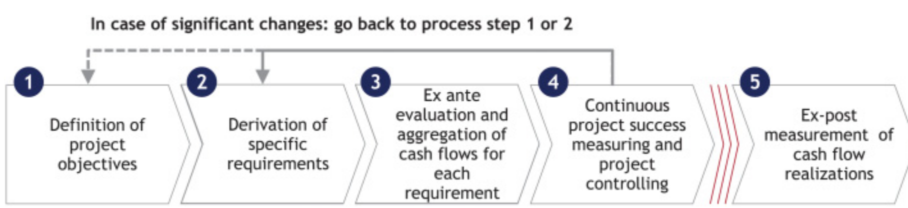

Fig. 2. Value assurance in requirements engineering process.

this research is to derive insights for the quantification and management of a specific project, which possibly can be generalized and transferred to other project settings afterwards. The BIE stage was initiated by the design of a process for value assurance in IT projects over their lifecycle by considering different steps. The initial process is depicted in Figure 2.

We assume the general project objectives to be defined in step 1. In step 2, requirements r_i with $i = 1 \ldots n$ are derived from these objectives using established methods of RE. Requirements can be defined on different levels of granularity. For instance, there may be projects in which one requirements cash flow and risk, which should be estimated, might be on a very fine, detailed and technical level (e.g., two variables need to have a technical connection for exchanging integer-type data) or in extreme contrast, there may also be other projects, in which the level of requirement might be very functional, abstract, and coarse (e.g., a new CRM system is needed). When developing our method, we primarily had requirements on this coarse level in mind that presumably can be measured by monetary values. However, through input by our business partners and literature [Feather and Cornford 2003], we made sure that there are ways to handle different levels of granularity. Depending on the difficulties to estimate cash flows and risks of requirements of a specific level, it is common in practice to subsume few fine requirements to one coarser requirement and estimate its cash flow and risk. There may be some requirements for which the estimation of cash flows and requirements may not be possible, but the application of our method and estimation of as many project-relevant cash flows and risks of requirements is still better than making decisions concerning project steering just based on gut feelings.

Yet, this article focuses on steps 3 to 5 that basically apply the same quantitative method set in different phases of the project. This allows for intertemporal comparability and thus for a quantitative analysis of the project course.

Ex ante Evaluation and Aggregation of Cash Flows (Step 3)

On the one hand, there are quantitative aspects of each requirement that can be directly transformed into cash flows. On the other hand, there are qualitative aspects, which are difficult to transform into monetary units [Walter and Spitta 2004]. According to the feedback from CC, many approaches applied in business today refrain from quantifying these qualitative aspects as no decision maker dares to name exact numbers for parameters difficult to estimate.

However, our method requires that in step 3 cash flows cf_{it} are initially evaluated for each one of the requirements i in each period t with $t = 0 \ldots T$. Note, that we assume that cash flows can also be determined for requirements that must be implemented (e.g., due to legal requirements). This can be achieved by comparing the IT project to its alternatives (e.g., doing a task manually). As stated earlier, the granularity of requirements can vary. To facilitate the quantification it therefore may be easier in some situations to subsume some requirements to estimate the respective cash flow on a coarser level even if a finer level may be more accurate. The evaluation of cash flows is then repeated multiple times in step 4 and finally in step 5. In the following, we describe how to accomplish this initial evaluation for benefits of requirements and describe the adaptations for steps 4 and 5.

Assumption 1. The cash flows are normally distributed random variables $\widetilde{cf}_{it} \sim N(\mu_{it}, \sigma_{it})$. The cash flows are stochastically independent between different periods.

Normally distributed project cash flows are a common assumption in IT portfolio management (cf. Wehrmann and Zimmermann [2005], Wehrmann et al. [2006], Zimmermann et al. [2008], and Fridgen and Müller [2011]). Although our assumption might not picture reality in every case, especially projects' benefits cash flows are often market driven and thus a normal distribution seems applicable. Furthermore, the more cash flows or requirements are aggregated, the better the central limit theorem and variations thereof will apply. Assuming the cash flows to be independent is obviously simplifying means, too. However, as the model could easily be adapted to picture intertemporal dependencies between cash flows, this is subject to further research.

Treating cash flows as random variables clearly eases their estimation, as no decision maker has to commit to exact values. Their deviation then contributes to the project's risk (i.e. cash flows that are hard to estimate increase the project risk more than cash flows that are easy to estimate). In a first attempt, we designed our method so that the distributional parameters μ_{it}, σ_{it} and the correlation ρ_{ij} would be directly obtained from the decision makers. Within the loops of the alpha cycle, we got the feedback from CC that this approach is hardly feasible in practice as (a) decision makers may not have the relevant statistical knowledge available and (b) absolute values of many these parameters (e.g., correlations) are hard to interpret even for trained people. To simplify the estimation of these parameters, we hence draw back on a basic but acknowledged procedures, which we adapted for our problem setting and which we strive to in further ADR cycles. For instance, in the case of cash flows, we draw on behavioral economics by using an interval-based scheme for the evaluation of each cash flow (please refer to Tversky and Kahneman [1974] for a critical discussion on these estimation methods, or for some kind of similar approach refer to Feather and Cornford [2003], who estimate for each requirement different criteria in ranges). For a more detailed and elaborate description of this approach, please refer to Beer et al. [2013]. Assuming normally distributed cash flows, we are able to derive expected values μ_{it} and standard deviations σ_{it} for each requirement r_i in each period t from this interval.

Having identified all cash flows $\widetilde{cf}_{it}$ and their distribution parameters, and assuming stochastical independence between periods (assumption 1), we can then calculate the distribution parameters of the net present value $\widetilde{npv}_i \sim N(\mu_i, \sigma_i)$ for each individual requirement based on the interest rate p:

$$\mu_i = \sum_{t=0}^{T} \frac{\mu_{it}}{(1+p)^t} \tag{1}$$

$$\sigma_i = \sqrt{\sum_{t=0}^{T} \left(\frac{\sigma_{it}}{(1+p)^t}\right)^2} \tag{2}$$

Aggregation of a Project Value Considering Risk and Dependencies

To determine the overall value (business case) of an IT project, we need to aggregate the $\widetilde{npv}_i$ of each requirement r_i to the project's $\widetilde{NPV} \sim N(\mu, \sigma)$. The project's overall expected value then is depicted by μ.

$$\mu = \sum_{i=1}^{n} \mu_i \tag{3}$$

To calculate the overall standard deviation σ of the IT project, we have to account for dependencies between requirements, which sometimes react similar for instance to external influences. For example, in case of technological innovation, multiple requirements and, therefore, cash flows might be affected simultaneously.

Assumption 2. The net present values $\widetilde{npv}_i$ of the requirements r_i are linearly dependent. Their Bravais-Pearson correlation coefficient of ρ_{ij} describes the dependencies between requirements i, j.

The identification of the correlation coefficients between every pair of requirements is a complex task becausee a high number of elements are involved and the context is hard to understand by project staff. As the CC suggested, we developed an easier approach for a gradually and guided determination of interdependencies (cf. Beer et al. [2013]). We can calculate the overall standard deviation σ of an IT project by aggregating the standard deviation of the single requirements and their respective correlation coefficients:

$$\sigma = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_i \sigma_j \rho_{ij}}. \tag{4}$$

Using these parameters, firms can apply various methods of an integrated risk/return management (e.g. (Conditional) Value-at-Risk). Synchronized with the CC, we decided to use a risk-adjusted project value as our means for project evaluation, which is in line with the Bernoulli principle and developed according to established methods of decision theory (e.g., Bernoulli [1738, 1954], Markowitz [1959], and von Neumann and Morgenstern [1947]). Similar formal approaches and assumptions for risk adjusted economic value analysis have been derived by Longley-Cook [1998] and have been applied in the context of IT numerous times, for example, in Hanink [1985], Bardhan et al. [2004], Zimmermann et al. [2008], Fogelström [2010], and Fridgen and Müller [2011].

Assumption 3. We define α as the parameter of risk aversion and assume that the decision-maker is risk-averse ($\alpha > 0$).

The risk-adjusted project value then is depicted by ϕ.

$$\phi_{EA} = \mu - \alpha\sigma^2 \tag{5}$$

The risk-adjusted project value can be interpreted as the certainty equivalent for normally distributed random variables and an exponential utility function and thus as an amount of money. The parameter α is a linear transformation of the Arrow-Pratt characterization of absolute risk aversion [Arrow 1971]. The higher the value of α, the more risk averse the decision maker. For practitioners, the concept of risk aversion is fairly abstract. Therefore, a precise determination thereof is very difficult. Again, we considered the input of the CC and designed a survey to determine a company's parameter of risk aversion at the executive level. Such an approach can also be found in behavioral finance [Sautner et al. 2007]. Thereby, the relevant decision makers are asked multiple questions about their maximum willingness to pay for different fictive project settings to determine the risk class, which is afterward assigned to a corresponding value of risk aversion.

Continuous Business Case (Step 4)

So far we described the first three steps of the process for value assurance in IT projects depicted in Figure 2. Because projects usually endure over a period of time T, a continuous project and business case management is essential for a lasting value assurance. Therefore, the main contribution of this research is the design, application, and evaluation of a continuous IT project steering indicated by step 4 of the described process in Figure 2.

In step 4, we are in the point of time $0 < \check{t} < T$. In $\check{t}$, some requirements $0 \ldots \check{\imath} - 1$ might already have been fully implemented and generate certain and noninfluenceable returns. Therefore, their associated cash flows cf_{it} are no more random variables for all t. For all other requirements $\check{\imath} \ldots n$, the past cash flows cf_{it} are realized and thus no more random variables for $t < \check{t}$. However, for $t \geq \check{t}$, the $\widetilde{cf}_{it}$ are still prone to risk and thus random variables. All $\widetilde{cf}_{it}$ need to be re-evaluated, as their values or distribution parameters, respectively, might have changed.

We identify two possible means to enable and support the continuous IT project steering: the *project success measuring* measure and the *project controlling* measure. The objective of the *project success measuring* is a comparison of the ex ante business case target value and the corresponding actually realized project value. In contrast, the *project controlling* enables to validate the ex ante estimated future cash flows from today's point of view, considering current information. In the following, we will examine these two measures in detail.

Project Success Measuring

Project success measuring (PSM) can be used in the course of the project lifecycle or as an ex-post means to investigate value deviations from the ex ante business case. To ensure comparability with the ex ante business case, all cash flows need to be discounted to $t = 0$. The project's expected value is then:

$$\mu = \sum_{i=0}^{\check{\imath}-1} \sum_{t=0}^{T} \frac{cf_{it}}{(1+p)^t} + \sum_{i=\check{\imath}}^{n} \left(\sum_{t=0}^{\check{t}-1} \frac{cf_{it}}{(1+p)^t} + \sum_{t=\check{t}}^{T} \frac{\mu_{it}}{(1+p)^t} \right). \tag{6}$$

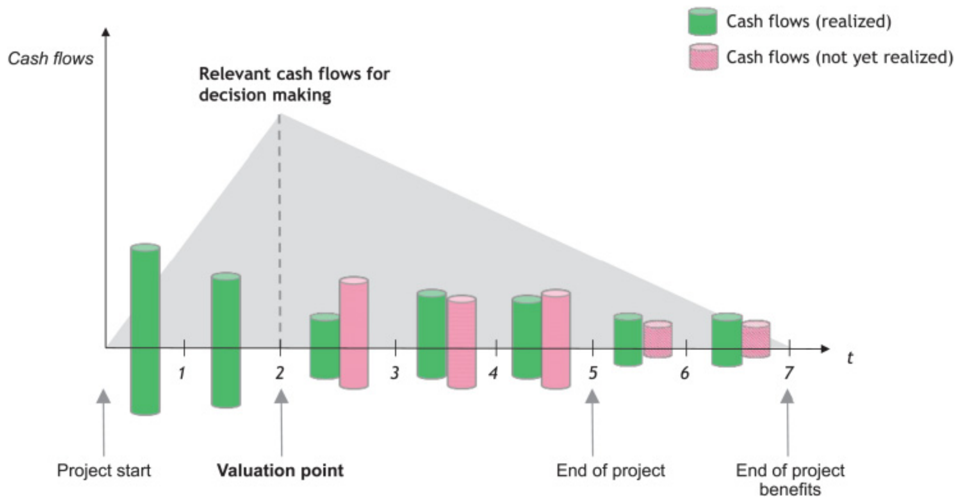

Fig. 3. Project success measuring.

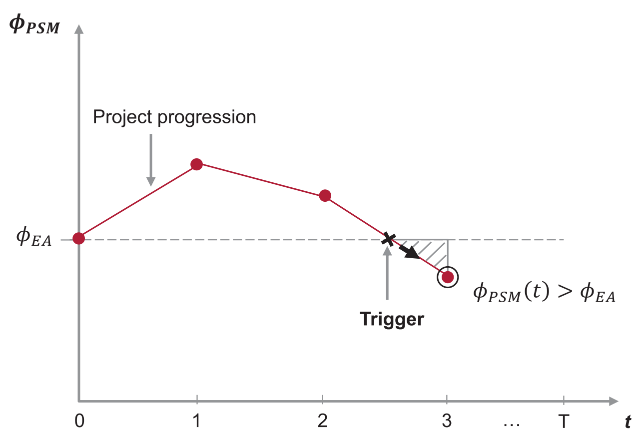

Fig. 4. Project success measuring: Trigger 1.

As only future cash flows of unfinished requirements are risky, the project's standard deviation is then:

$$\sigma = \sum_{i=\check{\imath}}^{n} \sum_{j=\check{\imath}}^{n} \sigma_i \sigma_j \rho_{ij} \; with \; \sigma_i = \sqrt{\sum_{t=\check{t}}^{T} \left(\frac{\sigma_{it}}{(1+p)^t} \right)^2} \; for \; i \geq \check{\imath}. \tag{7}$$

We can then calculate $\phi_{PSM}(\check{t})$ using Equation (5). Figure 3 illustrates the formally described coherences for a better understanding.

Using PSM within the projects lifespan may enable the identification of deviations between planed and actual progression in an early stage of the project. To ease the recognition of critical deviations, specific kinds of triggers can be defined in reality. Such triggers include, for example, planed project or benchmark values at specific points of time or any combination of these. However, for reasons of simplicity, we just examine some of them in the following. Additional triggers can be easily defined, though. A natural lower bound for the trigger is the ex ante business case ϕ_{EA}, as one

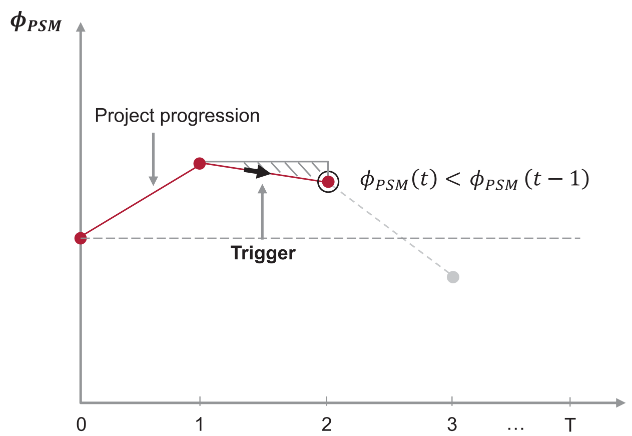

Fig. 5. Project success measuring: Trigger 2.

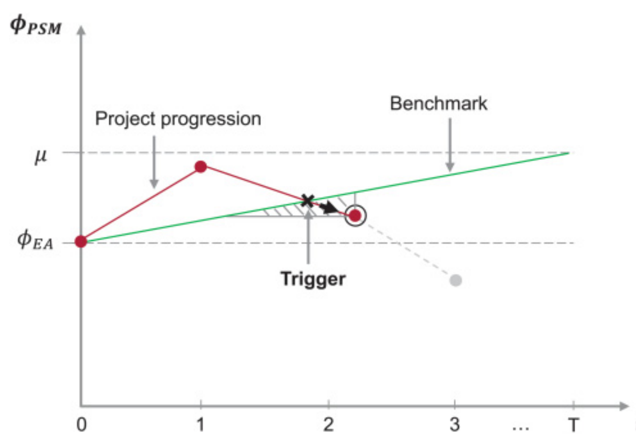

Fig. 6. Project success measuring: Trigger 3.

would assume that not all anticipated risks actually occur during the project and thus $\phi_{PSM}(t) > \phi_{EA}$. This coherence is described in Figure 4.

As this trigger might give a warning too late, an earlier warning would be triggered when $\phi_{PSM}(t) < \phi_{PSM}(t-1)$. This trigger, illustrated in Figure 5, consequently indicates a slowdown of the project progression and may sensitize the decision maker for the current project situation.

In accordance with the CC, we furthermore identified the necessity of a triggering system that monitors if a project makes steady progress in terms of realizing value and avoiding risk. Similar to the lower part of the cone of uncertainty (cf. Boehm [1984] and Armour [2008]), one would expect the project to reach its ex ante expected value at the end of the project. We use a linear benchmark that runs between ϕ_{EA} and the ex ante expected value μ. The question if differently shaped (e.g., convex) benchmarks, shown in Figure 6, are more suitable in a project setting is subject to further research.

The information about the current project value, the linear approximation and the respective triggers can be used for a continuous management of IT projects. It enables responsible decision makers to initiate adequate actions like a reallocation of resources in time and therefore mitigates the risk of project failure.

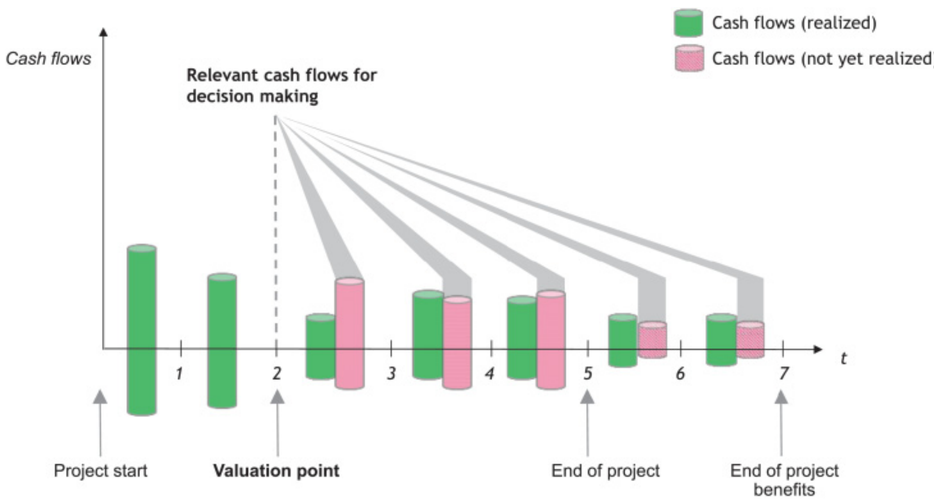

Fig. 7. Project controlling.

Project Controlling

For a rational project steering, PSM is not enough. In fact, it can still make sense to continue a project that has a negative project success $\phi_{PSM}(t)$. That is because the past cash flows and even the noninfluenceable future cash flows that are considered in $\phi_{PSM}(t)$ need to be treated as "sunk" to make the decision on continuing a project. If influenceable future cash flows show a positive risk-adjusted net present value, then it is rational to continue the project. As in IT projects benefits usually occur late [Buhl 2012] while big parts of the costs are already sunk, oftentimes finishing an unsuccessful project is favorable.

The project controlling (PC) supports the decision maker by deciding whether to continue a project. It is a future-riented project management measure and can be calculated during the projects lifecycle at different points of time. In t, it includes current information about already accomplished requirements (seen as sunk costs and requirements) and considers only the cash flows that can still be influenced. The net present value (for reasons of simplicity still discounted to 0) used for PC then has the following expected value:

$$\mu = \sum_{i=\check{i}}^{n} \sum_{t=\check{t}}^{T} \frac{\mu_{it}}{(1+p)^t} \tag{8}$$

Its standard deviation equals the one used in PSM. We can then calculate the risk-adjusted residual project value $\phi_{PCM}(\check{t})$ using Equation (5). To ease the understandability of the formally described coherences, they are illustrated in Figure 7.

As already realized, cash flows are not considered in the calculation; it is hardly possible to compare the risk-adjusted residual project value of different evaluation points. Hence, the objective of the PC is rather to indicate the necessity for project termination or at least safeguarding measures than to compare the overall cash flow situation of the project at different points of time. This can avoid expensive project failure at the end of the implementation phase. Analogous to the PSM, specific triggers in the PC context can ease the recognition of critical project situations. However, the following triggers are also in this case just examples and additional triggers can easily

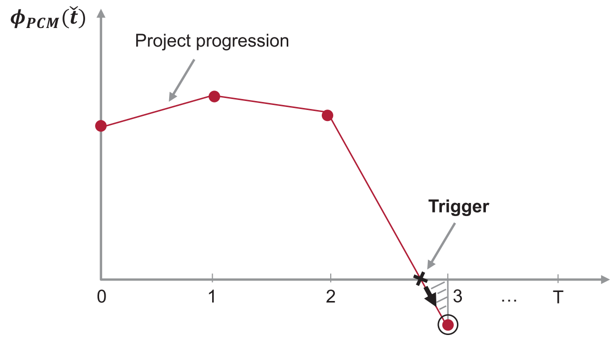

Fig. 8. Project controlling: Trigger 1.

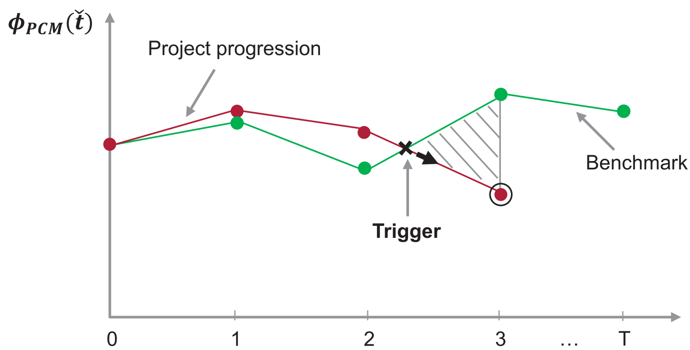

Fig. 9. Project controlling: Trigger 2.

defined. A natural lower bound for the trigger in this case is a negative risk-adjusted residual project value $\phi_{PCM}(\check{t}) < 0$, illustrated in Figure 8.

In terms of value realization and risk avoidance, we again developed a triggering system together with the CC that monitors the actual project progress compared to the initial estimation. This enables earlier warnings to be given than in the case of $\phi_{PCM}(\check{t}) < 0$. Therefore, we calculate the risk-adjusted residual project value based on the initial assessments of the ex ante business case and use it as a benchmark, like depicted in Figure 9.

Similar to the PSM, it enables responsible decision makers to initiate adequate actions in time and, therefore, mitigates the risk of project failure. Furthermore, it indicates whether it is more reasonable to continue or to terminate the project at a specific point in time. However, it is important to understand that a negative PC not necessarily indicates a project failure: If all value-adding requirements have already been implemented and the residual requirements provide negative business value, it makes sense to not finish the project.

Ex-post Measurement of Cash Flow Realizations (Step 5)

The ex post measurement is necessary to compare the ex ante estimated project values with the actual realized ones after the projects lifecycle and to gain valuable insights for upcoming projects. To achieve this, the results of the PSM are calculated at the end of the project and compared to the ex ante anticipated project value. Furthermore, it allows to associate critical environmental incidents occurred during the projects lifecycle to deviations between actual and estimated project cash flows. Analyzing this information enables to initiate a process of learning to improve the quality of ex ante business case estimations. Furthermore, it enables to build up a knowledge base that can support the prediction of a projects progression in the context of specific environmental influences.

4.2. Beta Cycle

During the beta cycle, we identified possible improvements for our method for a continuous value-based project steering on the basis of experiences in implementing it at the MC and on the basis of the CC who were implementing it at several of their clients.

The MC applied the method in several small projects with a project volume between 0.3 million € and 2.0 million €. Thereby, we received two major insights. First, while our presented risk-adjusted project value incorporates all relevant information including the decision maker's risk attitude and can be interpreted as a security equivalent, there might still be settings where other statistical measures might be more suitable. In the concrete case, the decision makers preferred a Value-at-Risk approach measuring which project value will be exceeded with 80% probability as this measure could be more easily interpreted and was more compatible with existing decision procedures. However, as risk attitude is not part of this measure, the comparability of projects and between different points of time is not given in general. While this was not conceived to be problematic by MC as their projects were comparatively small, it poses opportunities for further research on how interpretability can be improved while ensuring rationality using decision theory. Second, applying the method to several small projects revealed a problem of incentives when doing the interval-based estimation of values: While costs were estimated in MCs IT department that was also held responsible for a realistic estimation, the estimation of the projects' benefits required the involvement of several business units. However, in the given project setting, those units were not held responsible for the results and therefore did only put little effort in the estimation resulting in very vague answers like "the benefit of this feature will be between 0 and 20,000 € with 80% probability," merely ensuring that they cannot be blamed for project failures afterward. This reinforced existing and opens up areas for further research in incentivizing realistic project value and risk estimation.

In addition, to enhance our beta cycle, we draw back on additional qualitative feedback from the CC. They adopted different elements of the developed method in three IT projects with a bank, an insurance company, and an industrial client with an IT project volume sizing from 5 million to 150 million €. One of the IT projects used parts of the method to assure and steer an entire IT portfolio. Although the approach has not yet been implemented as a whole at one client, we were able to gather valuable feedback from the practitioners with regard to the benefits as well as the obstacles of our method. In three independent feedback cycles, our business partners validated the used principles and proceedings. In that context, we conducted three in-depth interviews with the project leaders of the three IT projects in September 2013. As already stated by Sein et al. [2011], "ADR is useful for open-ended IS research problems that require repeated intervention in organizations to establish the in-depth understanding of the artifact–context relationship." To date, we got the following feedback regarding our method (see Table I):

5. FOMALIZATION OF LEARNING

Based on our research results and in meaning of ADR, we are able to derive generalized insights that can be assigned to different kinds of problems in the context of value assurance in IT projects. The first three steps of the process for value assurance in RE (cf. Figure 2) focus on identifying and considering all project requirements and their transformation in a practicable method for an integrated quantification of IT projects. The challenge of maintaining applicability while upholding scientific rigor turned out

Table I. Further Feedback from CC Projects

Observed benefits of the method	• *General proceeding:* All interviewed business partners appreciate the general proceeding (especially with respect to the ex ante evaluation of the IT project requirements and the continuous IT project success measuring and controlling) and value the ex ante monetary assessment of cash flows and risks with respect to the project requirements. • *Estimation of the parameters:* In most cases, they were able to monetize the costs and with respect to the benefits the expected savings (one procedure is to ask different experts and to average the estimations in order to reduce the mistakes); we learned that they (bank and insurance) consider the interval-based estimation as practicable procedure. • *Continuous update:* They stated that the continuous update of PSM and PC supported the management of their projects.
Observed obstacles and improvement ideas of the method	• *Mathematical approach:* They consider the method in parts still too mathematically challenging and too hard to interpret for average top management purposes. To simplify the interpretation, they propose an initial estimation of cash flows on a higher granularity (e.g., estimating net present values of whole project parts instead of cash flows of individual requirements), which is then refined during the project while still staying within the same theoretical framework. Another solution would be to have specialized employees who are trained in applying the method. • *Visualization:* They suggest an easier visualization of the PSM and PC in form of a simple management cockpit. • *Context of the IT project:* According to the CC, the willingness to implement a monetary project steering like the proposed one depends on the context of the IT project. To give one example, the risk department of a large financial institution may be more open to apply it than the manufacturing department of a small industrial corporation.

to be a recurring topic throughout the action design research project. The estimation of accurate values for cash flows, risk aversion, and dependency parameters, which are necessary for a holistic calculation of the overall project value, is according to both our business partners a difficult task for project staff in practice. In this context, we are able to state following generalizable findings: First, the interval-based scheme, a method from behavioral economics being discussed by Tversky and Kahneman [1974], is a practicable and rigor means to assess the value of a project's requirements and cash flows. Second, to assess a value for the risk aversion of decision makers, an approach of behavioral finance (cf. Sautner et al. [2007]) doing a survey containing questions about the decision makers' willingness to pay in different project settings is advisable. Beyond that, the acceptance of monetary IT project management methods seems to depend on various parameters that need to be further examined (e.g., in empirical studies). Within our project, we identified the following parameters: the complexity of the method itself, the company's size and industry, the projects' size, the responsible division, the involved divisions and top-management support, and the decision makers' analytical education and skills.

6. CONCLUSION AND LIMITATIONS

We introduce a novel integrated approach for a continuous value-based IT project steering for large IT projects, which—unlike existing methods—considers costs, cash flows of requirements, risks, and interdependencies between requirements comprehensively. Therefore, this approach complements existing scientific literature in the context of RE and project management methods for the financial ex ante valuation of IT projects. The approach was designed, applied, and evaluated according to the ADR cycle in collaboration with two business partners from practice. In addition, we were able to gather additional qualitative feedback from the CC that applied our method (at least in parts) in three more IT projects. In the context of our collaborative project, methods were identified that can measure different project parameters and meet academic standards while at the same time preserve practical applicability. Furthermore, these methods have to be embedded in a project management process to enable the value assurance over the lifecycle of IT projects. As stated in Section 2, the case study gives back qualitative feedback and insights on our method. Even though this is a validate approach in the complex field of IS [Dubé and Paré 2003, p. 598], according to common literature (e.g., Dubé and Paré [2003]), we hope to draw on quantitative feedback in further evaluations in real-life IT projects. In this context, we develop two means to ensure a continuous value-based IT project steering: First, the PSM ratio, enabling a comparison of the overall cash flow situation of the project at different points of time and, therefore, an early detection of deviations from the ex ante business case. Second, in the context of PC, we develop the risk-adjusted residual project value, indicating the necessity for project termination or at least safeguarding measures. Therefore, our approach supports practitioners to measure the success of an IT project during its lifecycle enables a control mechanism for the project progression and eases future-oriented decisions regarding the projects continuation, which may also reduce the overall risk of IT project failure.

Nevertheless, because our model is based on several assumptions, it is not without limitations that are described in the following. First, although normally distributed cash flows are a common assumption in IT portfolio management (cf. Wehrmann and Zimmermann [2005], Wehrmann et al. [2006], Zimmermann et al. [2008], and Fridgen and Müller [2011]) and can also be braced by practical observations, they nevertheless are a restriction to the applicability of the model. Second, for the calculation of the risk-adjusted project value, we consider the standard deviation as measure of risk. This two-sided risk measure scales risk as symmetric deviation of the expected value. Likewise, it is conceivable that the model might be adapted to include different risk measures like Lower Partial Moments or Value at Risk (VaR). In cooperation with our business partners, we noticed that especially the VaR might be easier to interpret for decision makers. Third, we consider linear dependencies between requirements only, as we picture them by a Bravais-Pearson correlation coefficient. Yet, realistically, dependencies between requirements in some cases may also be nonlinear or there may be n-ary dependencies between requirements. But because this is a complex subject and not satisfactorily solved by academia or practice, it is justifiable to work with this simplifying assumption of linear dependencies in order to derive first results. Beyond these assumptions, our model has further limitations as follows. Because the developed approach has only been applied in depth to mobile app development projects and only in parts to three other large-scale IT projects, we are yet not able to draw general conclusions about miscellaneous IT projects, varying in context, scope, and size. However, the first results derived from application in practice indicate that for specific kinds of projects, the approach might have to be adjusted in order to reduce complexity of mathematical expense (cf. beta cycle). Because detailed calculations can be substituted by more vague estimations, this adjustment can be easily performed, though. In this case, the resulting lack of mathematical rigor might be overcompensated by the increase in practicability. Consequently, the results are worse compared to the mathematical rigor procedure but still better compared to isolated, nonmathematical procedures. As the applicability of this approach in different, varying IT projects obviously is an important issue to practitioners, it is topic to further research and evaluation. This may also be a helpful input for the desired knowledge base.

We presented an integrated approach, combining RE and IT project value quantification in a continuous value-based IT project steering process. This approach enables to derive generalized insights for interval-based estimation, the inquiry of the correlations between requirements, and the determination of the risk-aversion parameter. These insights provide a first basis for further development as it now shall be analyzed, for which kind and size of IT projects the approach is especially applicable. As already described by Sein et al. [2011], such consecutive and sustainable analyzes are parts of the ADR. In this case, they might be of great significance to practitioners as well as to researchers.

ACKNOWLEDGMENTS

This research was (in part) carried out in the context of the Project Group Business and Information Systems Engineering of the Fraunhofer Institute for Applied Information Technology FIT.

REFERENCES

Phillip G. Armour. 2008. The inaccurate conception. *Communications of the ACM* 51, 3 (2008), 13–16.

Kenneth J. Arrow. 1971. The theory of risk aversion. In: K. J. Arrow, K. J. (ed.) *Essays in the Theory of Risk-Bearing*. Markham, Chicago, 90–120.

Indranil Bardhan, Sugato Bagchi, and Ryan Sougstad. 2004. Prioritizing a portfolio of information technology investment projects. *Journal of Management Information Systems* 21 (2004), 33–60.

Martina Beer, Gilbert Fridgen, Hanna-Vera Müller, and Thomas Wolf. 2013. Benefits quantification in IT projects. In *Proceedings of the 11th International Conference on Wirtschaftsinformatik (WI'13)*. Leipzig.

Daniel Bernoulli. 1954. Exposition of a new theory on the measurement of risk. *Econometrica* 22 (1954), 23–36.

Daniel Bernoulli. 1738. Specimen theoriae novae de mensura sortis. *Commentarii Academiae Scientarum Imperialis Petropolitanae* 5 (1738), 175–192.

Barry W. Boehm. 1984. Software engineering economics. *IEEE Transactions on Software Engineering* SE-10, 1 (1984), 200–217.

Hans-Ulrich Buhl. 2012. The contribution of business and information systems engineering to the early recognition and avoidance of "Black Swans" in IT projects. *Business & Information Systems Engineering* 4, 2 (2012), 55–59.

Betty HC. Cheng and Joanne M. Atlee. 2009. Current and future research directions in requirements engineering. In K. Lyytinen, P. Loucopoulos, J. Mylopoulos, W. Robinson, W. (eds.) *Design Requirement Engineering—A Ten-Year Perspective*. LNBIP, Vol. 14. Springer, Heidelberg.

Mark Denne and Jane Huang. 2003. Software by numbers: Low-risk, high-return development. Prentice Hall PTR.

Line Dubé and Guy Paré. 2003. Rigor in information systems positivist case research: Current practices, trends, and recommendations. *MIS Quarterly* 27 (2003), 597–636.

Neil A. Ernst, Alexander Borgida, John Mylopoulos, and Ivan Jureta. 2012. Agile requirements evolution via paraconsistent reasoning. In *Proceedings of the 24th International Conference on Advanced Information Systems Engineering (CAiSE'12)*. Gdansk.

Martin S. Feather and Steven L. Cornford. 2003. Quantitative risk-based requirements reasoning. *Requirements Engineering* 8 (2003), 248–265.

Bent Flyvbjerg and Alexander Budzier. 2011. Why your IT project may be riskier than you think. *Harvard Business Review* 89 (2011), 23–25.

Gilbert Fridgen and Julia Heidemann. 2013. The importance of continuous value based project management in the context of requirements engineering. In J. Cleland-Huang, M. Jarke, L. Liu, K. Lyytinen (eds.), *Requirements Management—Novel Perspectives and Challenges, Dagstuhl Reports* 2, 10, 129.

Gilbert Fridgen and Hanna-Vera Müller. 2011. An approach for portfolio selection in multi-vendor IT outsourcing. In *Proceedings of the 32nd International Conference on Information Systems (ICIS'11)*.

Nina D. Fogelström, Emil Numminen, and Sebastian Barney. 2010. Using portfolio theory to support requirements selection decisions. In *Proceedings of the 4th International Workshop on Software Product Management (IWSPM'10)*. IEEE, 49–52.

Dean M. Hanink. 1985. A mean-variance model of MNF location strategy. *Journal of International Business Studies* 16 (1985), 165–170.

IT Governance Institute (ITGI). 2011. Global status report on the governance of enterprise IT (GEIT)—2011. Retrieved from http://www.isaca.org/Knowledge-Center/Research/Documents/Global-Status-Report-GEIT-10Jan2011-Research.pdf.

Matthias Jarke and Kalle Lyytinen. 2010. High impact requirements engineering. *Business & Information Systems Engineering* 2, 3 (2010), 123–124.

Matthias Jarke, Pericles Loucopoulos, Kalle Lyytinen, John Mylopoulos, and William Robinson. 2010. The brave new world of design requirements: Four key principles. In *Proceedings of the 22nd International Conference (CAiSE'10)*. 470–482.

Alastair G. Longley-Cook. 1998. Risk-adjusted economic value analysis. *North American Actuarial Journal* 2 (1998), 87–100.

Salvatore T. March and Gerald F. Smith. 1995. Design and natural science research on information technology. *Decision Support Systems* 15 (1995), 251–266.

Harry M. Markowitz. 1959. Portfolio Selection: Efficient Diversification of Investments. John Wiley & Sons. New York.

Bashar Nuseibeh and Steve Easterbrook. 2000. Requirements engineering—A roadmap. In *Proceedings of the Conference on The Future of Software Engineering*. ACM, New York, 35–46.

Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. 2008. A design science research methodology for information systems research. *Journal of Management Information Systems* 24 (2008), 45–77.

PMI – Project Management Institute. 2012. *PMI's Pulse of the Profession. Driving Success in Challenging Times.* Available at http://www.pmi.org/~/media/PDF/Research/2012_Pulse_of_the_profession.ashx.

Klaus Pohl. 1993. The three dimensions of requirements engineering. In *Proceedings of the 5th International Conference (CAiSE'93).*

Zacharias Sautner, Marin Weber, and Markus Glaser. 2007. What determines how top managers value their stock options? *SSRN* 1155084. (2007).

Maung Sein, Ola Henfridsson, Sandeep Purao, Matti Rossi, and Rikard Lindgren. 2011. Action design research. *MIS Quarterly* 35 (2011), 37–56.

Amos Tversky and Daniel Kahneman. 1974. Judgment under uncertainty: Heuristics and biases. *Science* 185 (1997), 1124–1131.

Axel van Lamsweerde. 2000. Requirements engineering in the year 00: A research perspective. In *Proceedings of the 22nd International Conference on Software Engineering*. 5–19.

John von Neumann and Oskar Morgenstern. 1947. *The Theory of Games and Economic Behavior*. Princeton University Press.

Sascha G. Walter and Thorsten Spitta. 2004. Approaches to the ex-ante evaluation of investments into information systems. *Wirtschaftsinformatik* 46 (2004), 171–180.

Alexander Wehrmann, Bernd Heinrich, and Frank Seifert. 2006. Quantitatives IT-Portfoliomanagement: Risiken von IT - Investitionen wertorientiert steuern. *Wirtschaftsinformatik* 48, 4 (2006), 234–245.

Alexander Wehrmann and Steffen Zimmermann. 2005. Integrierte Ex-ante Rendite-/Risikobewertung von IT-Investitionen. W*irtschaftsinformatik* 47, 4 (2005), 247–257.

Pamela Zave. 1997. Classification of research efforts in requirements engineering. *ACM Computing Surv*eys 29, 4 (1997), 315–321.

Steffen Zimmermann, Arne Katzmarzik, and Dennis Kundisch. 2008. IT sourcing portfolio management for IT service providers a risk/cost perspective. In *Proceedings of 29th International Conference on Information Systems (ICIS'08).*

Received April 2013; revised September 2013; accepted February 2014