Universität
Augsburg
University

Information Technology: Instrument and Object of Risk/Return Management

Dissertation
der Wirtschaftswissenschaftlichen Fakultät der
Universität Augsburg
zur Erlangung des Grades eines
Doktors der Wirtschaftswissenschaften
(Dr. rer. pol.)

vorgelegt von

Gilbert Stefan Fridgen

Diplom-Kaufmann Univ.,
Bachelor of Science in Informatik und Electronic Commerce

Augsburg, April 2010

Erstgutachter:	Prof. Dr. Hans Ulrich Buhl
Zweitgutachter:	Prof. Dr. Manfred Steiner
Vorsitzender der mündlichen Prüfung:	Prof. Dr. Marco C. Meier
Datum der mündlichen Prüfung:	08.04.2010

We should willingly take risks in supporting new projects. The tendency is to play it safe when funding is low, but we need to remember that the greatest risks have the greatest payoffs. […] *It is clear to me that under the right conditions, future technologies will be created that we cannot even imagine.*

Jerome Isaac Friedman,
1990 Physics Nobel Prize Laureate

Table of Contents

Please note: Page numbering is separate for each *chapter* or for each *paper,* respectively. Numbering of figures, tables, assumptions and footnotes is separate for each included *paper*. References are listed separately at the end of each *chapter*.

Table of Included Papers

The following published and submitted papers are included in this dissertation and will be referenced by the respective numbering (P1-P4):

P1. Buhl HU, Fridgen G, Hackenbroch W (2009) An Economic Analysis of Service-Oriented Infrastructures for Risk/Return Management. In: Newell S, Whitley E, Pouloudi N, Wareham J, Mathiassen L (eds) Proceedings of the 17th European Conference on Information Systems, ECIS, Verona: 2061-2072
Nominated for the Claudio Ciborra award for the most innovative paper.
(VHB-JOURQUAL2: 7,37 points, category B)

P2. Buhl HU, Fridgen G (2010) IT-Enabled Risk/Return Management: Service-Oriented Infrastructures vs. Dedicated Systems.

P3. Fridgen G (2009) Using a Grid for Risk Management: Communication Complexity of Covariance Calculations. In: Proceedings of the 15th Americas Conference on Information Systems, AMCIS, San Francisco, California: Paper 476
http://aisel.aisnet.org/amcis2009/476. Access 2010-01-06
(VHB-JOURQUAL2: 5,92 points, category D)

P4. Fridgen G, Müller H (2009) Risk/Cost Valuation of Fixed Price IT Outsourcing in a Portfolio Context. In: Proceedings of the 30th International Conference on Information Systems, ICIS, Phoenix, Arizona: Paper 135
http://aisel.aisnet.org/icis2009/135. Access 2010-01-06
(VHB-JOURQUAL2: 8,48 points, category A)

I Introduction

The ongoing financial and economic crisis (FEC) that started in 2007 already caused and still causes unprecedented damage for the economy worldwide. Its reasons and consequences illustrate the complex and dynamic environment in which companies act nowadays: Increasingly harmonized markets and the growing use of information technology (IT) enable companies to be part of globally connected value networks and therefore to realize quick return opportunities all over the world.

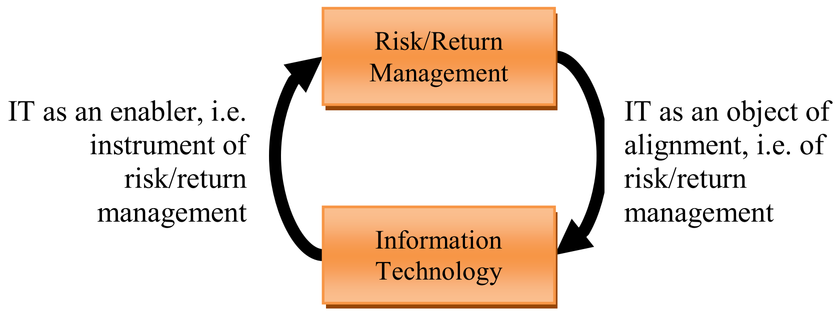
The FEC definitely showed that these value networks are also risk (spreading) networks: Companies can only create sustainable value if they are able to control the associated risk. Heretofore, they took too much risk on assets they believed to understand but whose risk contribution was in fact completely unknown. Similar to a domino effect, these risks spread rapidly all over the world, not only in the financial sector. Due to a lack of equity, banks did for example not renew credit contracts of companies in production or service sectors. Their respective bankruptcy brought suppliers and customers who relied on their business partners into liquidity issues, especially, when they had problems with their own credit renewals. Although the financial services industry pulled the trigger, the problems and solutions presented in this dissertation apply to all industries: Producing and service industries highly depend on the reliability of customers, suppliers, and many of them also of resource prices.

Companies better cope with such a crisis, if they are able to consistently manage those dependencies and their effects on projects as well as business processes. Crisis such as the FEC either will not happen again, or at least cause much less damage, first, if companies know their risk/return position regarding all relevant assets, second, if they know the respective interdependencies, and third, if they use their knowledge correctly.

Considering the FEC, IT can be regarded as both, a blessing and a curse. On the one hand, IT may enable ad hoc collection and aggregation of information on the current risk/return position, on the other hand IT is not only a relevant risk factor itself, it even made the FEC possible in the first place. Unfortunately, there are neither IT systems available that are able to determine a company's risk position within its value network, nor sophisticated methods to measure the risk of IT itself. These are the challenges of current research.

This dissertation therefore regards IT not only as an instrument but also as an object of risk/return management – IT is not only an enabler, but also has to be aligned (see figure 1).

Figure 1: Two Perspectives on IT: Enable and Align



In order to **enable** decision support (particularly on capital market based hedging) the risk/return-position of the company has to be determined shortly before market actions, after market actions, and – due to dynamic markets – on a regular basis, too (P1 (Buhl et al. 2009) describes how to determine the optimal interval of risk/return calculations). As described in P3 (Fridgen 2009, p. 2), the computing capacity necessary for risk calculation can be substantial, especially in globally acting enterprises. For a complete examination, the correlation between every possible pair of (relevant) assets has to be determined. This requires IT to be able to make extensive calculations in very short time, even when examining only parts of a company's assets. As this functionality is often required, it needs advanced algorithms and/or parallel computing. Currently "service-oriented architectures" (SOA) and especially "grid computing" (newly also labeled "cloud computing") that are in vogue in academia and practice seem to be promising concepts to solve this need for flexible and distributed utilization of IT resources.

Therefore, relevant references on the "enable" perspective can mainly be found in literature on SOA and grid computing (see P1 (Buhl et al. 2009), P2 and especially Hackenbroch (2007) for overviews on the relevant literature in this context). Here, resources and services are the most important components. According to Neumann et al. (2006) an IT resource is a representation of a logical or physical entity (e.g. computing or data capacity, software licenses, hardware, or network infrastructure). An IT service delivers a specific functionality to the user and requires different IT resources. IT services can thus be considered as software components that are used to support business processes. In contrast, grid computing is considered to be an infrastructure technology that allows for the virtualization of physical resources (see e.g. Foster et al. 2001; Foster and Kesselman 2003). "Grid services" are a way to realize SOA using grid technologies. They are based upon certain web service standards (e.g. "Open Grid Services Architecture" and "Web Services Resource Framework") and extend web services by dynamically allocating distributed resources using a grid middleware. Therefore, grid services are well suited for calculations required by risk/return management (P2 compares SOA and dedicated systems regarding their suitability for this task).

IT usually constitutes a big part of a company's cost, at the same time it bears high opportunities and risk. On the one hand, this is due to the fact that IT is widely used to support not only numerous business processes, but especially business-critical core processes of a company. On the other hand, this is due to special characteristics like irreversibility and dependencies between the different IT assets. Hence, it is necessary to design IT considering risk and return and to **align** it thoroughly to the company's strategy.

IT assets can be distinguished in already implemented IT assets (e.g. running systems) and IT project opportunities. Especially IT project opportunities have to be evaluated regarding their support for the business processes and the company's strategy. Furthermore, related innovations and their potential to enable new and improved business models are relevant parameters. When setting up IT projects, one not only has to take dependencies to other assets into account, but also the relatively high freedom of decisions, which can have an influence on risk and return of the project and thus of the whole company.

When considering IT assets as parts of a portfolio, especially stochastic dependencies are of relevance (Wehrmann et al. 2006). According to Zimmermann (2008), those can either be intratemporal (negative/positive effects on other currently running assets), or intertemporal (negative/positive effects on future projects). To exemplify the aforementioned freedom of decisions, one can choose an IT asset's design, schedule and sourcing strategy. Software can be developed as a monolithic system or according to the concepts of service-oriented architectures. The schedule of an IT project is flexible: It can e.g. be aborted, stripped-down or accelerated to a certain extent by additional resources. Finally, and most important for this dissertation, sourcing strategies might have similar effects: Lower cost of an outsourced asset is oftentimes associated with higher risk regarding the quality of service.

Phrases like "death of distance" (Cairncross 1997) show that technology rendered globally distributed value networks possible. One central motive for many IT sourcing decisions is the high cost pressure on IT departments. On the one hand, many argue that outsourcing to IT service providers or offshoring to low-wage countries can lead to high savings on the IT budget without losses in quality. On the other hand, these decisions may also imply high risk – especially in the IT sector. Nevertheless, well managed outsourcing at a fixed price can help optimize the risk/return position of the whole IT portfolio as described in P4 (Fridgen and Müller 2009). Thus, decisions on sourcing must be made in due consideration of the associated risk. IT project development or operation at different locations or by different service providers can have fundamental impact on the success of outsourcing projects (Aubert et al. 1999). Regarding the number of unsuccessful or even aborted outsourcing deals, one can guess that these failures are due to the nonexistence of adequate methods of evaluating outsourcing alternatives. Today, there are only few and very basic approaches for decision support on IT sourcing in science and practice (see e.g. Lacity and Willcocks 2000; Aron et al. 2005; von Campenhausen

2005; Dutta and Roy 2005; Wehrmann and Gull 2006; Zimmermann et al. 2008; Henneberger et al. 2009). At the same time, not only studies confirm the importance of research and need for action in this area (Lacity and Willcocks 2003; Weill and Ross 2005), but also CIOs name the development of IT sourcing strategies as one of the most important tasks of IT governance (Quack 2004).

Altogether, both perspectives on IT and risk/return management – align and enable – bear great potential for research. Before motivating the addressed research questions of this dissertation in section I.2, I will outline the objectives and structure in section I.1.

I.1 Objectives and Structure

The objective of the papers included in this dissertation is the decision support on selected topics at the interface of information technology and risk/return management. As illustrated before, two perspectives are important here: On the one hand, information technology can be an instrument of risk/return management; on the other, it has to be an object of risk/return management. As described in the following table, the first three papers included in this dissertation focus the use of information technology as an instrument of risk/return management. The fourth and last included paper focuses the governance of IT projects, especially of outsourcing, using methods of risk/return management.

I. Introduction – IT: Instrument and Object of Risk/Return Management

- Objective I.1: Outlining the objectives and structure of this dissertation
- Objective I.2: Motivating the addressed research questions of the papers

II. Information Technology as an Instrument of Risk/Return Management (P1, P2, P3)

- Objective II.1: Illustrating the basic characteristics of calculations in risk/return management, especially of distributed calculations of covariance matrices (P1, P2, P3)
- Objective II.2: Developing a quantitative model for the optimization of computing capacity dedicated to risk/return management (P1)
- Objective II.3: Developing a quantitative model comparing service-oriented infrastructures and dedicated systems regarding their suitability for risk/return management calculations (P2)
- Objective II.4: Developing and analyzing algorithms for distributed covariance calculations (P3)

III. Information Technology as an Object of Risk/Return Management (P4)

- Objective III.1: Developing a quantitative model to determine optimal outsourcing degrees of an IT portfolio
- Objective III.2: Illustrating the relevance of these results by simulating outsourcing decisions based upon real world data
- Objective III.3: Describing the deviation of the optimal solution caused by the common IT decision process on projects and on outsourcing

IV. Conclusion and Outlook

- Objective IV.1: Summarizing the findings
- Objective IV.2: Identifying areas of further research

I.2 Motivation of the Addressed Research Questions

The research questions of chapters II&III and papers P1-P4, respectively, are outlined and motivated in the following sections, structured by the aforementioned two perspectives.

I.2.1 IT as an Instrument of Risk/Return Management (Chapter II)

Today's market environment can be characterized by tight competition and global integration of markets. As already described above, risk/return management is crucial for enterprises in order to survive in these surroundings. The problems arising from the FEC emphasized this more than ever: As generally decided by the G20 meeting in April 2009 and announced by US President Barack Obama in January 2010, already strict rules and regulations (that nevertheless did not prevent the crisis) are going to be much more tightened, especially in the financial services industry. Risk/return management will require new and consistent implementation. Very sophisticated and resource intensive methods for risk/return quantification and aggregation will have to come in place.

Innovative approaches of distributed computing, like grid computing, cloud computing, cluster computing or service-oriented architectures, are principal topics of IT related science and business. They are offering potentially suitable infrastructures for the described complex calculations. In this context, we will speak of service-oriented infrastructures (SOI) as an instance of (the abstract principle of) service-oriented architectures where (mostly resource-intensive) distributed services are made available transparently over a grid network.

Up to now, the scientific focus on SOI is in general technically oriented, only. Although approaches like "grid economics" or "utility computing" address economic aspects in general, the business perspective is oftentimes completely neglected. For specific application domains, the consideration of benefit and cost still constitutes a widely unresolved issue. This dissertation is therefore narrowing the research gap between technical issues of service-oriented computing and its application in risk/return management.

While there is an extensive amount of literature on risk/return management, this dissertation is primarily based on prior research addressing risk forecasting and in particular the estimation of covariance matrices. Covariances can be used for a comprehensive enterprise-wide risk/return management as described by Huther (2003, pp. 111) and Faisst and Buhl (2005, pp. 408) and as later picked up by Fill et al. (2007), Kundisch et al. (2007) and Gericke et al. (2009). Other publications are dealing with the empirical estimation and forecasting of covariances using historical data. For example, there is plenty of literature already covering the question of how many and which historic quotations should be used to determine a suitable risk forecast. According to Elton and Gruber (1972, p. B-409) forecasting is in fact "one of the important problems in finance". For applicable techniques refer to Engle (1982), Kupiec (2007) and Hull and White (1998). An overview of the corresponding methods used in volatility and correlation forecasting can be found in Alexander (2008, pp. 89). Taylor (2005) presents recent approaches to volatility forecasting like conditional autoregressive Value-at-Risk (CAVaR) models.

Not only to meet regulatory requirements, but also to conform to the business strategy, it is necessary to frequently estimate the risk exposure associated with a portfolio of investment objects (e.g. securities). It is common practice in today's financial services industry to calculate the risk exposure within fixed time intervals (e.g. several days). SOI based on grid technologies can embrace resources of the whole enterprise and even of external resource providers. Therefore, they offer huge amounts of computing capacity that can be used to accelerate calculations dramatically (Middlemiss 2004).

In this context, Paper P1, entitled *"An Economic Analysis of Service-Oriented Infrastructures for Risk/Return Management",* seeks an answer to the following research question:

> *What is the optimal amount of computing capacity that should be allocated to risk quantification, considering benefits as well as cost?*

A dedicated system where a fixed set of resources is dedicated to the risk/return calculations could be a sensible alternative to a SOI. The second part of paper P2, entitled *"IT-Enabled Risk/Return Management: Service-Oriented Infrastructures vs. Dedicated Systems",* contains new results on this topic intended for journal publication. The first part is to some extend identical to P1, which was presented and published in conference proceedings. So, following the approach of P1, paper P2 aims at answering the research question:

> *Under which conditions is a SOI superior to a dedicated system regarding calculations in risk/return management?*

In both papers P1 and P2, we formulated analytical optimization models delivering solutions to these decision problems. Yet, even when using a SOI, resources are not unlimited. Thus, it is necessary to design sophisticated algorithms for the corresponding calculations and to consider their respective complexity. Paper P3, entitled *"Using a Grid for Risk Management: Communication Complexity of Covariance Calculations",* therefore analyzes several different standard network topologies regarding the following research questions:

> *Which grid resources shall be allocated to the calculation of which parts of the covariance matrix? How shall the input data necessary to perform these calculations be distributed?*

What communication complexity do algorithms feature on different network topologies?

I.2.2 IT as an Object of Risk/Return Management (Chapter III)

The already mentioned tight competition on globalized markets forces companies to deal with cost cutting that is necessary to stay in business. Within the big area of research on managing IT according to principles of risk/return management, paper P4, entitled *"Risk/Cost Valuation of Fixed Price IT Outsourcing in a Portfolio Context",* focuses on the (partial) outsourcing of IT projects. Firms mainly pursue outsourcing strategies to reduce costs and mitigate risks associated with their business processes (Lacity and Hirschheim 1993). The market for outsourcing services therefore not only increased significantly over the past years but is also about to outgrow previous prospects (Aspray et al. 2006). According to Currie (1997) IT service providers become more specialized and competitive as they benefit from this development. Especially software development skills are often called "global commodities" (Dutta and Roy 2005; Lacity and Willcocks 2003). Therefore, companies face the opportunity to close more profitable outsourcing deals particularly on software development projects.

Today, the evaluation of and decision process on IT projects is in many companies neither specified nor documented. Therefore, it is difficult to develop a viable outsourcing strategy. The Standish Group (2006) reports that two thirds of IT projects fail or miss their targets. Difficulties concerning the development of integrative outsourcing strategies are certainly one reason for problems with successfully conducting IT projects.

On the contrary, projects of a manageable size, following reasonable plans and tactics, and risk management by a capable team, can lead to far better outcomes (Sauer et al. 2007). To make a project manageable it can be effective to outsource parts of a project to a specialized service provider. Slaughter and Ang (1996) state, that through outsourcing, projects can be managed more successfully. Therefore, to enable a company to implement a profitable outsourcing strategy, P4 examines the effects of fixed price outsourcing on costs and risks of an IT project portfolio by posing the following research questions:

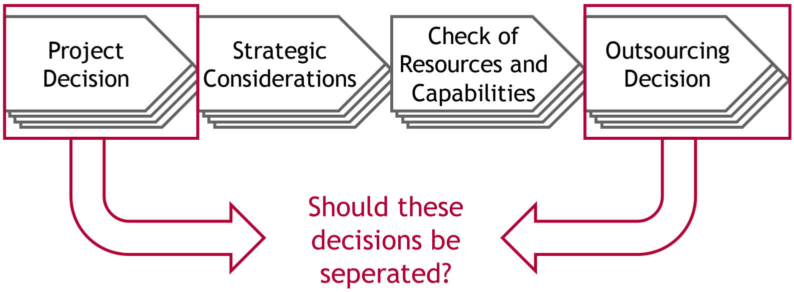
Which degree of outsourcing should a client choose for a single project to minimize the risk adjusted costs of a software development project?

Which degrees of outsourcing should a client choose for a <u>given</u> multiple projects portfolio to minimize the risk adjusted total portfolio costs?

In today's IT departments it is common practice to decide on the implementation of individual projects first and then to decide case by case if and to what extent a project shall be outsourced, as depicted in figure 2.

> *What is the gain of simultaneously selecting both, projects and their respective outsourcing degrees, against determining efficient outsourcing degrees for a previously selected (therefore given) optimal project portfolio?*

Figure 2: The Common IT Project and Outsourcing Decision Process



P4 thus provides a formal-deductive model that enables companies to determine an optimal outsourcing strategy by considering the project portfolio selection and the decision on outsourcing degrees simultaneously. The validity of the results is furthermore documented by a simulation based on data gathered in a business context.

The chapters II and III include the aforementioned papers. Chapter IV contains a summary of the main results and an outlook on further areas of research.

I.3 References (Chapter I)

Alexander C (2008) Market Risk Analysis Volume II: Practical Financial Econometrics, 1 ed. Wiley, Chichester

Aron R, Clemons EK, Reddi S (2005) Just Right Outsourcing: Understanding and Managing Risk. JMIS 22(2):37-55

Aspray W, Mayadas F, Vardi MY (2006) Globalization and Offshoring of Software. A Report of the ACM Job Migration Task Force. http://www.acm.org/globalizationreport. Access 2010-02/17

Aubert BA, Dussault S, Patry M, Rivard S (1999) Managing the Risk of IT Outsourcing. In: Sprague RH (ed) Proceedings of the 32nd Annual Hawaii International Conference on System Science, HICSS-32, IEEE Computer Society, Maui, Hawaii

Buhl HU, Fridgen G, Hackenbroch W (2009) An Economic Analysis of Service-Oriented Infrastructures for Risk/Return Management. In: Newell S, Whitley E, Pouloudi N, Wareham J, Mathiassen L (eds) Proceedings of the 17th European Conference on Information Systems, ECIS, Verona

Cairncross F (1997) The Death of Distance: How the Communications Revolution Will Change Our Lives. Mcgraw-Hill Professional

Currie WL (1997) Expanding IS Outsourcing Services through Application Service Providers. Growth 8(1):1

Dutta A, Roy R (2005) Offshore Outsourcing: A Dynamic Causal Model of Counteracting Forces. JMIS 22(2):15-35

Elton EJ, Gruber MJ (1972) Earnings Estimates and the Accuracy of Expectational Data. Management Science 18(8):B-409-B-424

Engle RF (1982) Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. Econometrica 50(4):987-1007

Faisst U, Buhl HU (2005) Integrated Enterprise Balancing mit integrierten Ertrags- und Risikodatenbanken. Wirtschaftsinformatik 47(6):403-412

Fill H, Gericke A, Karagiannis D, Winter R (2007) Modellierung für Integrated Enterprise Balancing. Wirtschaftsinformatik 49(6):419-429

Foster I, Kesselman C (2003) The Grid 2: Blueprint for a new Computing Infrastructure, 2 ed. Morgan Kaufmann, San Francisco, California

Foster I, Kesselman C, Tuecke S (2001) The Anatomy of the Grid - Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications 15(3):220-222

Fridgen G (2009) Using a Grid for Risk Management: Communication Complexity of Covariance Calculations. In: Proceedings of the 15th Americas Conference on Information Systems, AMCIS, San Francisco, California: Paper 476. http://aisel.aisnet.org/amcis2009/476. Access 2010-01-06

Fridgen G, Müller H (2009) Risk/Cost Valuation of Fixed Price IT Outsourcing in a Portfolio Context. In: Proceedings of the 30th International Conference on Information Systems, ICIS, Phoenix, Arizona: Paper 135. http://aisel.aisnet.org/icis2009/135. Access 2010-01-06

Gericke A, Fill H, Karagiannis D, Winter R (2009) Situational method engineering for governance, risk and compliance information systems. In: Vaishanvi V, Purao S (eds) Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology, DESRIST '09, Philadelphia, paper 24

Hackenbroch W (2007) Integrated Risk/Return Management on Service-Oriented Infrastructures: Financial Applications and their Economic Value: the risk-at-risk approach, 1 ed. Sierke, Göttingen

Henneberger M, Katzmarzik A, Müller S, Pleie FM (2009) Sourcing and Automation Decisions in Financial Value Chains. In: Newell S, Whitley E, Pouloudi N, Wareham J, Mathiassen L (eds) Proceedings of the 17th European Conference on Information Systems, ECIS, Verona

Hull J, White A (1998) Incorporating Volatility Updating into the Historical Simulation Method for Value-at-Risk. Journal of Risk 1(1):5-19

Huther A (2003) Integriertes Chancen- und Risikomanagement - Zur ertrags- und risikoorientierten Steuerung von Real- und Finanzinvestitionen in der Industrieunternehmung. Deutscher Universitäts-Verlag, Wiesbaden

Kundisch D, Löhner FM, Rudolph D, Steudner M, Weiss C (2007) Bank Management Using Basel II-Data: Is the Collection, Storage and Evaluation of Data Calculated with Internal Approaches Dispensable? In: Enterprise Risk Management Symposium Monograph, Chicago, Illinois

Kupiec PH (2007) Financial Stability and Basel II. Annals of Finance 3(1):107-130

Lacity MC, Hirschheim RA (1993) Information Systems Outsourcing: Myths, Metaphors, and Realities. Wiley, New York

Lacity MC, Willcocks LP (2003) IT Sourcing Reflections: Lessons for Customers and Suppliers. Wirtschaftsinformatik 45(2):115-125

Lacity MC, Willcocks LP (2000) Global Information Technology Outsourcing: In Search of Business Advantage. Wiley, New York

Middlemiss J (2004) Gearing up for Grid. http://www.wallstreetandtech.com/showArticle.jhtml?articleID=17603501. Access 2009-12/20

Neumann D, Veit D, Weinhardt C (2006) Grid Economics: Market Mechanisms for Grid Markets. In: Barth T, Schüll A (eds) Grid Computing: Konzepte, Technologien, Anwendungen, 1 ed. Vieweg, 64-83

Quack K (2004) Zwölf Themen, die den CIO bewegen. Computerwoche Spezial (December 2004):16-17

Sauer C, Gemino A, Reich BH (2007) The Impact of Size and Volatility on IT Project Performance. Communications of the ACM 50(11):79-84

Slaughter S, Ang S (1996) Employment Outsourcing in Information Systems. Communications of the ACM 39(7):47-54

Standish Group (2006) Chaos Report 2006.

Taylor JW (2005) Generating Volatility Forecasts from Value at Risk Estimates. Management Science 51(5):712-726

von Campenhausen C (2005) Offshoring Rules – Auslagern von unterstützenden Funktionen. ZfB 75(1):5-13

Wehrmann A, Gull D (2006) Offshoring von Softwareentwicklungsprojekten - Ein COCOMO-basierter Ansatz zur Entscheidungsunterstützung. Wirtschaftsinformatik 48(6):407-417

Wehrmann A, Heinrich B, Seifert F (2006) Quantitatives IT-Portfoliomanagement: Risiken von IT-Investitionen wertorientiert steuern. Wirtschaftsinformatik 48(4):234-245

Weill P, Ross J (2005) A Matrixed Approach to Designing IT Governance. MIT Sloan Management Review 46(2):26

Zimmermann S, Katzmarzik A, Kundisch D (2008) IT Sourcing Portfolio Management for IT Service Providers - A Risk/Cost Perspective. In: Proceedings of the 29th International Conference on Information Systems, ICIS, Paris: Paper 133. http://aisel.aisnet.org/icis2008/133. Access 2010-01-06

Zimmermann S (2008) IT-Portfoliomanagement - Ein Konzept zur Bewertung und Gestaltung von IT. Informatik-Spektrum 31(5):460-468

II IT as an Instrument of Risk/Return Management

This chapter includes the papers *"An Economic Analysis of Service-Oriented Infrastructures for Risk/Return Management"*, *"IT-Enabled Risk/Return Management: Service-Oriented Infrastructures vs. Dedicated Systems"*, and *"Using a Grid for Risk Management: Communication Complexity of Covariance Calculations"*. Some prior versions of these papers have also been published in Hackenbroch (2007).

II.1 An Economic Analysis of Service-Oriented Infrastructures for Risk/Return Management

Authors:	Buhl, Hans Ulrich, Fridgen, Gilbert, FIM Research Center Finance & Information Management, Department of Information Systems Engineering & Financial Management, University of Augsburg, Universitätsstraße 12, 86159 Augsburg, Germany, hans-ulrich.buhl@wiwi.uni-augsburg.de gilbert.fridgen@wiwi.uni-augsburg.de Hackenbroch, Wolfgang, Hypo Real Estate Holding AG, Unsöldstraße 2, 80538 Munich, Germany, wolfgang.hackenbroch@hyporealestate.com
In:	Newell S, Whitley E, Pouloudi N, Wareham J, Mathiassen L (eds) Proceedings of the 17th European Conference on Information Systems, ECIS, Verona: 2061-2072. *Nominated for the Claudio Ciborra award for the most innovative paper.*

Risk/return management has not only evolved as one of the key success factors for enterprises especially in the financial services industry, but is in the times of the financial crisis crucial for the survival of a company. It demands powerful and at the same time flexible computational resources making it an almost ideal application for service-oriented computing concepts. An essential characteristic of service-oriented infrastructures is that computational resources can be accessed on demand and paid per use. Taking the estimation of covariances for a portfolio of risky investment objects as an example, we propose quantification for the economic value of fast risk/return management calculations. Our model analyzes the influence factors on the optimal computing capacity dedicated to these calculations and reveals interesting insights in how far the optimal computing capacity depends on market parameters. Our main result is that more volatile markets require a lower computing capacity as the optimal computing capacity depends positively on changes of the market risk but negatively on the risk itself.

II.1.1 Introduction

Risk/return management is crucial for today's enterprises in order to strive and even to survive in a market environment that can be characterized by tight competition and global integration of markets. This is more than ever emphasized by the problems arising from the current crisis on the financial markets. Especially in the financial services industry, already strict rules and regulations – that nevertheless have not prevented the crisis – are expected to be much more tightened as generally decided by the G20 meeting in April 2009. Consequently, very sophisticated and resource intensive methods for risk/return quantification and aggregation have to be in place. Innovative approaches of distributed computing like grid computing, cluster computing or service-oriented architectures (SOA) are en vogue in academia as well as in practice, offering potentially suitable infrastructures for the corresponding complex calculations. We will speak of service-oriented infrastructures (SOI) in this context as an instance of (the abstract principle of) SOA where (mostly resource-intensive) distributed services are made available transparently over a grid network. Up to now the intellectual treatment of SOI is usually technically oriented and most often neglecting the necessary economic aspects. Even though these aspects are addressed in general by approaches like for instance "grid economics" or "utility computing", the reflection on benefits and cost still constitutes a widely unresolved issue for specific application domains. We are therefore striving to narrow the gap between the technical capabilities of service-oriented computing and its economical application in risk/return management.

One basic task in risk/return management is the frequent estimation of the risk exposure associated with a portfolio of investment objects (e.g. securities). Today, enterprises mostly calculate their risk exposure during fixed time intervals like e.g. several days. With the possibly huge amount of computing capacity a SOI based on grid technologies offers (embracing resources of the whole enterprise or even of external resource providers), calculations can be accelerated dramatically (Middlemiss 2004). Nevertheless, economic models quantifying the business value of a more frequent recalculation of the risk exposure are not available yet. Thus, the question arises, what is the optimal amount of computing capacity that should be allocated to risk quantification, considering benefits as well as cost? We will deliver a solution to this problem in the form of an optimization model.

Concerning risk/return management[1], we restrict our considerations to publications addressing risk forecasting and in particular the estimation of covariance matrices. Huther (2003, pp. 111), or Faisst and Buhl (2005, pp. 408) for example describe the use of covariances for a comprehensive enterprise-wide risk/return management. Other publications are dealing with the question of how covariances can be empirically estimated or forecasted by analyzing historical data. In fact forecasting is "one of the important problems in finance" (Elton and Gruber 1972, p. B-409) and consequently there are a lot of publications already covering the question of how many and which historic quotations should be used to determine a suitable risk forecast. To give an impression about applicable techniques we refer to Engle Engle (1982), Kupiec (2007) and Hull and White (1998). Alexander (1996, pp. 233) provides an overview of the corresponding methods used in volatility and correlation forecasting. Recent approaches to volatility forecasting like conditional autoregressive Value-at-Risk (CAVaR) models are presented in Taylor (2005). Although it is widely known that the corresponding calculations are very resource and time intensive, there is to the best of our knowledge so far no publication dealing with the specific problem of quantifying the economic value of a frequent (re)calculation of risk.

Grid computing can be regarded as an infrastructure technology enabling the virtualization of physical resources. Available definitions for the term grid computing are mostly of descriptive nature and provide little more than certain essential characteristics (see e.g. Foster 2002; Foster et al. 2001; 2002; Foster and Kesselman 1998). However, various proponents have agreed with Foster and Kesselman (1998) that "a computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive and inexpensive access to high-end computational capabilities". Recently, an evolution towards SOA can be observed Longworth (2004). Grid technologies are one possibility to realize a SOA consisting of so called "grid services". Grid services are based on specific web service standards, like the specifications (Open Grid Services Architecture) and (Web Services Resource Framework). They extend web services insofar as they imply the dynamic, yet for the user transparent, allocation of (physical) resources to services by a grid middleware and therefore are especially suited to fulfil resource intensive tasks. There is an extensive literature on service-oriented computing or grid computing in general (Foster and Kesselman 1998; see for instance Berman 2005; Silva 2006; Singh and Huhns 2004). Some publications even consider the application of these technologies for portfolio management, derivatives pricing or other areas of financial risk management (Brow-

---

[1]Whereas literature most often focuses on either side separately, we will rather speak of risk/return management emphasizing an integrated view because risk management can only unfold its potential in combination with the management of the corresponding return.

nlees et al. 2006; Crespo et al. 2006; Schumacher et al. 2006). However, these approaches describe how grid technologies can be applied, but do not quantify the resulting business value. In the context of grid computing also resource allocation mechanisms have been widely discussed—most often under the term "grid economics". Regev and Nisan (1998), Buyya et al. (2000), Nabrzyski et al. (2003) and Wolski et al. (2004) which provide an overview of this area. They most often dwell either on the question which principles are appropriate to resource management or on technical and architectural issues connected with the development of resource management systems. Accordingly, in most of the existing approaches demand for computational resources is merely an external factor whereas in our approach it is subject to optimization.

In this context an important characteristic of SOI based on grid technologies is the on-demand access to distributed resources. When resources stem from an external provider this concept is often labelled utility computing, meaning that resources can be consumed und priced as easy as for instance electricity or water. Utility computing has been subject to research as well. Bhargava and Sundaresan (2004) analyze pay-as-you-go pricing scenarios where providers guarantee computing capacity, but users do not make a commitment towards actual use. Our paper to some extent continues the ideas of Bhargava and Sundaresan (2004). However, we take the perspective of a service user and present a rationale for decisions on computing capacity in the context of risk/return management.

II.1.2 Risk/Return Management

The term "risk" is used heterogeneously in general speaking as well as in academic circles. Therefore we feel that it is appropriate to begin with a definition of risk before we describe the various requirements and objectives of risk/return management applications. While in the economic literature risk is often generically explained as the "possibility of missing a planned outcome" we will follow a more finance-related approach. From this point of view we define with Schröck (2001, p. 24) risk as "the deviation of a financial value from the expected value". A positive deviation is often in general speaking referred to as "chance" while a negative deviation is characterized as "danger". Because of this two-sided perception of risk, variance or standard deviation of a risky value are suitable and well accepted measures of risk. We will use the standard deviation of historical portfolio returns as the risk measure later in this text. Synonymously we will speak of the volatility of a portfolio and define it as the "annualized standard deviation of percentage change in daily price" (Spremann 2003, pp. 154).

Enterprises are investing capital into investment objects in order to generate cash inflows and subsequently to increase the return of the invested capital. Typically risk-averse management is making risky investments hoping to achieve an excess return over the risk-free rate. There is a general connection between risk and return of an investment object: higher return is systematically associated with higher risk. This connection is theoretically explained by economic models like the CAPM (the "Capital Asset Pricing Model" was originally developed by Sharpe (1964), Lintner (1965) and Mossin (1966)) and empirically evaluated later on (an overview of relevant empirical studies can be found e.g. in Copeland and Weston (1988, pp. 212)). Following the argumentation of Wilson (1996, pp. 194) it is therefore crucial for the survival and success of an enterprise to be able to allocate the available capital to the right combination of investment objects, taking into account their specific contributions to the overall risk and return. Investment objects in this context are not restricted to securities. Almost all business transactions are associated with uncertainty and thus contribute to an enterprises overall risk exposure. Thus in the spirit of an enterprise-wide risk/return management all investments an enterprise is engaged in, like credit decisions or even customers or projects can be seen as components of the enterprise's overall investment portfolio, having a return and a variance (as a measure of risk).

One major goal of risk/return management in this context is the prevention of bankruptcy by restricting potential losses resulting from risky investment objects. The increasing importance of this goal is emphasized by the current crisis on the financial markets. A growing number of rules and regulations require enterprises to hold a part of their available capital to back their risky investments (Jackson et al. 1998, pp. 8). This share of the available capital then makes less or no contribution to the overall earnings. By management decisions these restrictions are broken down along the organizational hierarchies into guidelines on business unit or departmental level. We are assuming in the following text that those guidelines are essentially representing limits for the maximum risk a department, business unit and consequently an enterprise is willing (or able) to take.

In order to evaluate whether an enterprise or department complies with a given risk limit, it is necessary to calculate the current risk exposure frequently. For simplifying means, we concentrate on one fundamental instrument in this paper: The covariance approach. This constitutes a basic principle in finance and forms the foundation for many risk/return management applications ranging from Markowitz portfolio optimization to Value-at-Risk (VaR) calculations. In our context covariances are used for determining the overall risk position of an enterprise taking into account diversification effects that exist between the investment objects. Nevertheless, the proposed methods are also applicable to other risk measures as long as they take dependencies between single investment objects into account.

Following the covariance approach, we can represent risky investment objects by random variables. Typically historical data are used in order to derive a distribution for a random variable[2] and calculate the distribution parameters. It is worth mentioning that considering the portfolio risk and return merely on an aggregated level is not satisfactory because all information is lost about the risk attributable to a single investment object. It is crucial to separate the portfolio and decompose it into data per investment object, i.e. to calculate the covariances. Only then the enterprise can perform economically rational investment decisions on different aggregation levels. With σ_i^2 and Cov_{ij} denoting variance and covariance of investment objects respectively we can determine the overall risk of a portfolio σ_P^2, consisting of $n \gg 0$ investment objects (numbered from 1 to n), as[3]

$$\sigma_P^2 = \sum_{i=1}^{n} \sigma_i^2 + \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} Cov_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{n} Cov_{ij}$$

The so defined matrix of all covariances is called *covariance matrix*. An important characteristic of covariances is that $Cov_{ij} = Cov_{ji}$. This makes the matrix symmetric and thus not all of its values have to be calculated. The total number of covariance calculations necessary is given by $n(n+1)/2$.

II.1.3 A Valuation Model for fast Risk Quantification

We consider an enterprise which has access to (and is possibly engaged in) a set of risky investment objects as well as to a risk-free investment alternative. It is frequently (re)calculating its overall risk position by estimating the covariance matrix of its portfolio. Our main hypothesis for the valuation of benefits is: the faster risk/return management calculations can be executed the higher will be the return of the enterprise because given risk limits can better be exploited.

Since the enterprise is acting in an uncertain and dynamic environment its risk position is changing willingly (by investment decisions) or unwillingly (by “movement” of the markets). Because the estimation of risk cannot be accomplished in real-time the covariances at hand are always significantly outdated. We are in the following recurring to the fact that the enterprise is adjusting its risk position to a value somewhere below a certain threshold thus constituting a

---

[2] A thorough overview and discussion of the statistical analysis of financial data can be found in Shiryaev (1999, pp. 314).

[3] As we analyze the complete investment objects and not parts or quantities of investment objects (e.g. stocks), this formula contains no weights.

"safety margin". In the regulatory context this is often called "haircut", like in Basel Committee on Banking Supervision (2004). The financial crisis made clear, that – depending on the assets invested in – a high safety margin is necessary to account for changes of parameters, especially in liquidity. It is doing so by using the capital allocation between the risky investment objects and the risk-free alternative for balancing their overall risk position. It is important to understand that our model is not addressing the evaluation of the efficient set of investment objects or portfolio optimization (both require covariances), but the aggregation and management of the risk position of an enterprise. Whenever covariances are available the safety margin can be adjusted immediately in a way that the resulting (and over time changing) overall risk position of the enterprise does not exceed the given risk limit at any time. Hence, the faster covariances are available, the smaller the safety margin can be. We will use this effect to quantify the benefits of fast covariance estimations depending on the time needed for the completion of one covariance matrix.

II.1.3.1 Model Setting and Basic Assumptions

The time interval under consideration consists of equidistant periods periods such that $t = 0$ denotes the beginning of the current period. We shall write for example σ_t to indicate the value of a model parameter at the end of period t. Correspondingly (dis)investment decisions take effect only at the end of each period. If not mentioned otherwise all variables assume real values, i.e. values $\in \mathbb{R}$.

The enterprise is equipped with a total capital of $K > 0$ which is always completely allocated to the risky portfolio and/or the risk-free alternative. We denote the risky portion of K with $x \geq 0$ and furthermore use x for risk adjustment. T indicates the length of the calculation time frame, $T \in \mathbb{N} \backslash 0$ with $\mathbb{N}$ as the set of natural numbers. At the end of each time frame we choose x in a way that the risk limit is "probably" not exceeded during the next time frame. We will formulate more precisely what is meant by "probably" later on. Future returns of the portfolio are modelled as independent random variables. Their probability distribution for each period can be characterized by mean and standard deviation. This implies that the investment objects can be marked to market, i.e. there is a price attached to them. We additionally need a set of assumptions for the deductions following thereafter.

Assumption 1 *The enterprise is generally risk-averse and striving for efficient combinations of investment objects. Investment objects are perfectly divisible and traded on a no-frictions market.*[4]

Assumption 2 *The risky part of the enterprise's capital yields the expected return* μ*, the risk-free investment pays the time-invariant risk-free interest rate i, which is equal to the borrowing rate.*[5] *We always have* $\mu > i > 0$.

Assumption 2 is made in the spirit of the model of Modigliani and Miller (1958) where enterprises and investors can borrow or place money at will for a risk-free rate, but expect a premium for taking the risk associated with the investment in risky assets. With the so defined parameters and x as decision variable we can determine the overall expected return $\mu^U(x)$ and risk $\sigma_t^U(x)$ of the enterprise according to common rules of statistics as

$$\mu^U(x) = x\mu + (1-x)i = i + x(\mu - i) \text{ and } \sigma_t^U(x) = x\sigma_t \quad (1)$$

The overall risk of the enterprise is expressed by the portfolio risk (the enterprise is the weighted "sum" of its investment objects) and thus changes over time driven by the varying σ_t. Note that due to our focus on changing risk we do not regard a changing $\mu^U(x)$ over time (which would result in an index t as in $\sigma_t^U(x)$). As we see, with ascending x the overall returns as well as the overall risk of the enterprise are both increasing. On the one side the enterprise certainly strives for the highest possible return, on the other side a limitation exists for x from the given risk limit.

It is common practice to use some variation of a random walk for the price movement on security and commodity markets. This approach goes ultimately back to Louis Bachelier (1900) who compared the stock market with a "drunkards walk". Although controversially discussed, it was picked up more than half a century later by Mandelbrot (1963; 1972) and Fama (1965) among others. Following this theory of random walks historical (e.g. daily or weekly) portfolio returns can be used for estimating mean μ and standard deviation σ_t of future portfolio returns.

---

[4]In the sense of the "Portfolio Selection" theory (Markowitz 1971) investors are trying to achieve the highest possible return on their investment for a given risk. They are acting under perfect trading conditions, i.e. no arbitrage, no transaction cost, strong information efficiency etc.

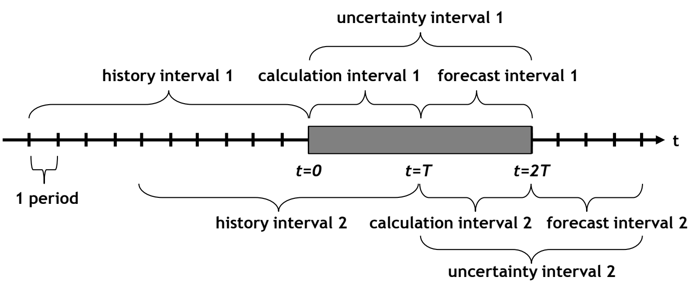
[5]The equality of lending and borrowing rate is assumed for sake of simplicity and justifies the case $x > 1$, where the enterprise is actually borrowing money for making risky investments.

It is important to understand that in our model the standard deviation is possibly changing in each period (indicated by its index t).

Assumption 3 *The initial calculation of covariances starts in $t = 0$ and is finished after T periods. Each new covariance calculation begins in the finishing period of the previous covariance calculation.*

According to assumption 3, whenever a covariance matrix is completed, the input data used for its calculation are T periods old. We can immediately determine the portfolio risk by summing up the covariances in the matrix. This can then be used for a risk adjustment decision as well as for portfolio optimization. The moment before the next matrix is finished the input data used for risk calculations are already $2T$ periods outdated. Therefore the uncertainty interval that has to be taken into account spans $2T$ periods: in the worst case the risk has been going up over $2T$ periods before the enterprise realizes that it is exceeding the maximum risk it is willing (or able) to take (see figure 1). Without loss of generality we will concentrate our analysis on the first covariance matrix calculation and the corresponding adjustment decision, therefore focussing on the time interval $[0; 2T[$. During this time the portfolio risk is fluctuating in a non-predictable way.

Figure 1: Period Model and Relevant Time Intervals



II.1.3.2 The Risk-at-Risk Approach

We will now dwell on the portfolio risk at time t, denoted as σ_t, modelling it as a random variable. This relates to a phenomenon known from the behavior of stock market prices called *heteroscedasticity* (see e.g. Spremann 2003, pp. 152). In analogy to the periodical returns we write

Assumption 4 *The σ_t are normally distributed.*

This distribution assumption can (and would in practice) be relaxed by approximating the distribution of the σ_t delivered by the calculated sequence of standard deviations. Arranged in increasing order one can easily deduce the quantiles needed in our model. Nevertheless, for reasons of simplicity and without significantly changing the general result we assume σ_t to be normally distributed here.

We will again focus on two distribution parameters: Our notation for the (strictly positive) mean will be μ_σ, for the standard deviation σ_σ (both tagged with an σ indicating the fact that the distribution applies to the portfolio *risk*), thus $\sigma_t \sim N(\mu_\sigma; \sigma_\sigma)$ with N short for the normal distribution.

The distribution parameters in our model can again be estimated using historical data. For example, the standard deviation of the portfolio risk can be taken as an estimate for the expected portfolio risk μ_σ and thus as the starting point for the random walk of σ_t. In order to maximize x in equation (1) under the given constraints we have to consider the uncertainty interval $[0; 2T[$. Because of assumption 4 the standard deviation of the expected portfolio risk after $2T$ periods is (as a sum of normally distributed random variables) again normally distributed according to $N(\mu_\sigma; \sigma_\sigma\sqrt{2T})$. As a consequence for the overall risk of the enterprise we have $\sigma_t^U(x) \sim N(x\mu_\sigma; x\sigma_\sigma\sqrt{2T})$.

In order to rephrase the fuzzy formulation "the risk limit is probably not exceeded" we will follow an approach comparable to the VaR for quantifying portfolio risk. We speak of a Risk-at-Risk over a holding period and a confidence level α, $0 \ll \alpha < 1$ and think of it as the standard deviation $\bar{\sigma}$ which is exceeded within the holding period only with the (small) probability of $(1 - \alpha)$. With $\Phi(x)$ denoting the standardized normal distribution function, we know for the distribution of $\sigma_t^U(x)$ over $2T$ periods that

$$P(\sigma_t^U(x) \leq \bar{\sigma}) = \Phi(\frac{\bar{\sigma} - x\mu_\sigma}{x\sigma_\sigma\sqrt{2T}})$$

At the same time we require in the spirit of the Risk-at-Risk approach the probability given above to be greater than or equal to the confidence level α, i.e.

$$P(\sigma_t^U(x) \leq \bar{\sigma}) \geq \alpha \text{ for } t = 2T \tag{2}$$

In the marginal case both sides of the equation are equal and we can therefore state—with q_α as the (onesided) α-quantile of the standardized normal distribution—that

$$\frac{\bar{\sigma} - x\mu_\sigma}{x\sigma_\sigma\sqrt{2T}} = q_\alpha \Leftrightarrow x = \frac{\bar{\sigma}}{q_\alpha\sigma_\sigma\sqrt{2T} + \mu_\sigma} \tag{3}$$

x gives us the portion of risk-free and risky investment objects in a way that equation (2) holds. We can calculate the overall expected earnings of the enterprise, given this capital allocation, as

$$B(x) = \mu^U(x) \cdot K = (i + x(\mu - i)) \cdot K$$

Obviously $B(x)$ represents only expected, calculatory (and not real) earnings because the returns are not fully cash-flow effective and the earnings themselves subject to interpretation. We will neglect the adjectives "expected" and "calculatory" and speak of earnings or, more generally, of benefits.

By inserting x from equation (3) into $\mu^U(x)$ from equation (1) we find for the benefits (with the number of periods needed for the completion of one covariance matrix as the independent variable)

$$B(T) = \left(i + \frac{\bar{\sigma}(\mu - i)}{q_\alpha \sigma_\sigma \sqrt{2T} - \mu_\sigma}\right) \cdot K \tag{4}$$

Considering equation (4) an enterprise could maximize its benefit by minimizing the time T that is needed to calculate a covariance matrix. Yet there is a trade-off between the benefits and the cost, i.e. cost caused by the infrastructure that is necessary to compute the calculations.

II.1.4 Service-Oriented Infrastructures for Risk Quantification

From a SOI point of view the risk calculation can be regarded as a service providing its user transparently with up-to-date risk information for the relevant investment universe. In this section we derive the relationship between the computing capacity (i.e. cost) allocated to risk quantification and the time needed for the computation.

II.1.4.1 Computing Capacity for Risk Quantification

We denote with z the computing capacity necessary for estimating one covariance matrix in exactly T periods. We use CPUs as a measure for computing capacity and are aware of the fact that this means a one-dimensional view on matters as other determinants of system performance are ignored. We denote with w the workload—measured in CPU hours—for estimating one covariance. Applying a simple moving average technique with a rolling sample of historical data (Elton and Gruber 1972, pp. 409) we get unbiased estimators of the expected value and (co)variances for every point in time.

Assumption 5 *The same workload w is necessary for estimating variances and covariances.*[6] *Furthermore every covariance is estimated from scratch, i.e. no intermediate results are used.*[7]

Assumption 6 *The length of the calculation time frame T depends solely on the time needed for the computation, neglecting e.g. latency or transmission times. Correspondingly the only cost relevant is cost for computation which occurs in the form of a (internal or external) factor price per CPU hour over a given time.*

Note that for the calculation of covariance matrices on a SOI it is convenient that the computations can be distributed on several nodes and executed in parallel, as all pairwise covariances can be calculated independently from each other. Thus efficiency losses are considerably low.

We can now deduce the computing capacity $z(T)$ (workload per time) that is required in every period over T periods. We already know that for n investment objects $n(n+1)/2$ covariances have to be calculated. Multiplied with the workload per covariance this determines the total number of CPU hours needed. This in turn—divided by the calculation time frame—leads to the functional relationship

$$z(T) = \frac{n(n+1)w}{2T} \Leftrightarrow T(z) = \frac{n(n+1)w}{2z} \tag{5}$$

Given n investment objects and a computing capacity of z CPUs, the covariance matrix will be completed after $T(z)$ periods.[8] This constitutes an important parameter of the covariance estimation service since it describes the economic value that should be attributed to the consump-

---

[6]Variance estimation requires only one historical time series, so its intrinsic workload is smaller than the workload for covariance estimation. Nevertheless, this effect can be neglected since for n variances there are $n(n-1)/2$ covariances in a given covariance matrix. With n sufficiently large the variances have merely no effect on the number of calculations (e.g. with $n = 201$, the number of variances is only 1% of the number of covariances to be calculated).

[7]This could be considered awkward for the simple moving average procedure but is a realistic approach for more sophisticated methods.

[8]Here as well as for the optimal $T(z^*)$ later in this text the outcome is assumed real-valued. In reality and in order to fit it to the discrete-time period model one has to check the neighboring integer values to obtain the discrete optimum.

tion of capacity for covariance estimation. By inserting (5) into equation (4) and neglecting the mean risk compared to the standard deviation of the risk[9] we quantify the benefits as

$$B(z) = a + 2b\sqrt{z} \text{ with } a = iK > 0, b = \frac{\overline{\sigma}(\mu - i)K}{2q_\alpha \sigma_\sigma \sqrt{n(n+1)w}} > 0$$

$$B'(z) = \frac{b}{\sqrt{z}} > 0,\ B''(z) = -\frac{b}{2\sqrt{z^3}} < 0 \tag{6}$$

$B(z)$ is a strictly increasing and concave function.

II.1.4.2 Optimizing Computing Capacity on a Service-Oriented Infrastructure

In this paper we abstract from a specific SOI framework or technological implementation, but only consider the essential characteristics of such an infrastructure: A service user can consume exactly the amount of resources needed and is charged by the service provider on a pay-per-use basis. Thereby, it makes no difference whether resources stem from internal sources like enterprise-owned desktop computers and servers that are connected via a grid network or from an external provider. In the case of allocating the (limited) resources of an internal SOI, the enterprise faces opportunity cost that may be taken as a usage price. For external resources a price per unit of computing capacity is set by the provider. The price can be changing depending on the contract and service level agreement used. For example resources could be more expensive when delivered fail-safe during peak times while covering basic load on a lower service level might be cheaper (Bhargava and Sundaresan 2004, p. 203).

In an external provisioning scenario we assume a straightforward cost function using a factor price p (measured in e.g. \$ per workload over a period). Using equation (6), our cost function $C(z)$ and our objective function $Z(z)$ then are defined as

$$C(z) = pz,$$

$$Z(z) = B(z) - C(z) = a + 2b\sqrt{z} - pz.$$

[9] The exact quantification would lead to a strictly increasing and concave benefit function B, as well. It would nevertheless be tedious to continue in our analysis with the exact expression. In order to avoid writing overhead we deliberately simplify our objective function by neglecting the mean risk, which is a numerically justifiable approximation in our context.

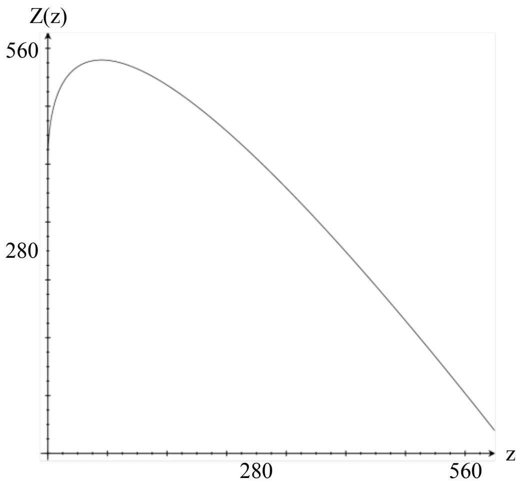
Note that - as it is generally the case for the parameters in our model - $C(z)$ describes the cost per period and $B(z)$ the benefits per period depending on the capacity used per period for the completion of the covariance matrix over T periods. As the difference of a strictly concave benefit function and a linear cost function, $Z(z)$ is again a strictly concave function. The first derivative $Z'(z) = \frac{b}{\sqrt{z}} - p$ features its only null at $\frac{b^2}{p^2} > 0$. Due to the strict concavity of $Z(z)$ and with respect to equation (6) the only maximum of $Z(z)$ is at

$$z^* = \frac{b^2}{p^2} = \frac{\bar{\sigma}^2(\mu - i)^2 K^2}{4q_\alpha^2 \sigma_\sigma^2 n(n+1)wp^2} \tag{7}$$

On a short-term, iterative basis this result can be used to allocate resources of an external provider to risk/return management services. It is possible to examine in detail how input parameters affect z^*. For example the more capital the enterprise has to its disposal the more (in absolute terms) it will invest into risky investment objects. Higher risk exposure in turn increases the importance of risk/return management which is correctly reflected by the positive sign of $\frac{\partial z^*}{\partial K}$. The same argumentation holds when the enterprise faces a higher risk limit $\bar{\sigma}$. In this case it should allocate more capacity to risk/return management applications, which is consistently leading to an increasing z^*. Eventually when the risk premium $(\mu - i)$ rises (due to higher μ and/or lower i) investing into risky objects becomes more attractive, resulting in a larger share of risky capital. In order to manage the consequently more voluminous portfolio our model suggests that additional capacity should be allocated. In the denominator of equation (7) we have the parameter w determining the CPU hours needed for one covariance estimation. Increasing w generates higher cost. This in turn leads to less capacity allocation in the optimal case, reflecting the known tradeoff between accuracy and speed of risk/return management calculations. The behavior of z^* depending on the confidence level α is quite surprising. Theoretically q_α could grow infinitely (with increasing confidence level α) leading to an infinitesimal small z^*. This is the case because in our model the only way to account for a higher confidence level is a larger safety margin. This causes diminishing benefits and therefore a decreasing z^* in the optimum. Another interesting and to some extent counter-intuitive result is produced in combination with the parameter σ_σ. One would possibly expect that with increasing volatility of the portfolio risk the optimal capacity allocation increases which is actually not true. Basically, for a given risk limit $\bar{\sigma}$ a higher volatility of the portfolio risk can be leveraged by faster risk calculations (so that the enterprise can still get close to the risk limit without hazardously exceeding it during the uncertainty interval). Going back to Bachelier's proposition the benefits of fast covariance calculations are of order $\frac{1}{\sqrt{T}}$. On the other side, the cost are depending on T in a reciprocal $(\frac{1}{T})$ fashion. As a consequence for higher volatility of the portfolio risk the cost are increasing more quickly than the benefits, leading ultimately to an increasing T^* and decreasing z^*, respectively.[10]

Figure 2 shows a numerical example of the described optimization. All values are per hour and were in parts estimated from intraday data of the German stock index DAX. The example pictures a company with capital of $K = 50 million\$$ and $n = 200{,}000$ investment objects. The company tries to hold a risk limit of $\bar{\sigma} = 0.1\%$ with a confidence level $\alpha = 99\%$. It can realize a risk free return $i = 0.0008\%$ and a risky expected return $\mu = 0.001\%$. The workload per covariance is assumed $w = 2{,}5 \cdot 10^{-10}$ CPU hours, the price per CPU $p = 2\$$. It features an optimum at 72 CPUs.

Figure 2: Numerical Example for Z(z)



Besides this straight forward linear cost function, especially in an internal provisioning scenario, more complex cost structures may apply. For instance, in a scenario where a high number of services are competing for a limited number of resources, opportunity cost may not increase proportionally, because ever more services with increasing opportunity cost are suppressed. After all, it becomes clear, that a detailed analysis of cost structures along with the knowledge of the concrete utility function of a service not only delivers the opportunity to allocate resources on-demand. The analysis also supports design decisions concerning for instance the

[10] Such a situation could be observed e.g. during a “regime switch” between two volatility clusters where a time interval with relatively low volatility switches into an interval with higher volatility or vice versa.

questions whether external or internal provision or a mixture of both is preferable, whether or not the service should be deployed doing risk calculations in the background and whether an existing SOI is sufficient or should be enlarged due to an already high utilization rate resulting in high opportunity cost.

II.1.5 Limitations of the Model and Conclusion

In this paper we demonstrated how the economic value that can be derived from risk/return management calculations can be measured considering an enterprise that has to decide on the amount of capital it wants to reserve to cover potential losses resulting from a risky investment portfolio. Several assumptions (e.g. regarding the distribution of the expected portfolio risk) were necessary to achieve an analytical solution.

Using the covariance approach as an example we moreover developed an optimization model that delivers the optimal amount of computing capacity that should be allocated to risk calculations at a time. In doing so we restricted our analysis to one well-defined risk/return management problem. Although covariances are fundamental and widely used in financial applications we thereby covered only one element of numerous risk/return management methods and algorithms. Other approaches and applications for SOI concepts (like for instance Monte-Carlo simulations which also have a very high parallelization potential) have to be examined as well. In fact, most of the basic principles introduced in this paper can be adapted to other scenarios in more sophisticated and complex surroundings.

Putting it all together a SOI is especially advantageous when market parameters determining the benefits of risk calculations are highly volatile as could be observed during the crisis since July 2007 resulting in varying demand for computing capacity. With a SOI, resources can be reallocated at any time to reach an economic optimum. As discussed, not only benefits but also (opportunity) cost may vary depending on the total demand for capacity. For example, during "quiet times" risk calculations may be computed more frequently generating added value out of readily available excess capacity even if benefits are comparably small. One caveat not mentioned in this paper is information security. As information on investment objects may be sensitive business data, spreading the calculations over the company or even over service providers may not be desired. Implementing a system as described would therefore require additional security mechanisms and persuasion of the management.

After all, this paper is a contribution to understand the application of service-oriented infrastructures in the specific domain of risk/return management. Although a validation of our findings based on real-world data is still subject to further research, in our point of view, such a systematic and economic analysis is a requirement as a first step for the further development of the new concepts like service-oriented computing or utility computing.

II.2 IT-Enabled Risk/Return Management: Service-Oriented Infrastructures vs. Dedicated Systems

Authors:	Buhl, Hans Ulrich, Fridgen, Gilbert, FIM Research Center Finance & Information Management, Department of Information Systems Engineering & Financial Management, University of Augsburg, Universitätsstraße 12, 86159 Augsburg, Germany, hans-ulrich.buhl@wiwi.uni-augsburg.de gilbert.fridgen@wiwi.uni-augsburg.de

Risk/return management has not only evolved as one of the key success factors for enterprises especially in the financial services industry, but is in the times of the economic crisis initiated by the financial markets crucial for the survival of a company. It demands powerful and at the same time flexible computational resources making it an almost ideal application for service-oriented computing concepts. An essential characteristic of service-oriented infrastructures is that computational resources can be accessed on demand and paid per use. Taking the estimation of covariances for a portfolio of risky investment objects as an example, we propose quantification for the economic value of fast risk/return management calculations. Our model then compares the cost structures of service-oriented infrastructures and dedicated systems in this domain. On this basis, we can determine under which circumstances the one or the other architectural strategy is superior.

II.2.1 Introduction

Risk/return management is crucial for today's enterprises in order to strive and even to survive in a market environment that can be characterized by tight competition and global integration of markets. This is more than ever emphasized by the problems arising from the current economic and financial crisis. Especially in the financial services industry, already strict rules and regulations – that nevertheless have not prevented the crisis – are expected to be much more tightened as generally decided by the G20 meeting in April 2009, but are still lacking consistent implementation. Consequently, very sophisticated and resource intensive methods for risk/return quantification and aggregation have to be in place. Innovative approaches of distributed computing like grid computing, cloud computing, cluster computing or service-oriented architectures (SOA) are en vogue in academia as well as in practice, offering potentially suitable infrastructures for the corresponding complex calculations. We will speak of service-oriented infrastructures (SOI) in this context as an instance of (the abstract principle of) SOA where (mostly resource-intensive) distributed services are made available transparently over a grid network. Up to now, the intellectual treatment of SOI is usually technically oriented and most often neglecting the necessary economic aspects. Even though these aspects are addressed in general by approaches like for instance "grid economics" or "utility computing", the reflection on benefits and cost still constitutes a widely unresolved issue for specific application domains. We are therefore striving to narrow the gap between the technical capabilities of service-oriented computing and its economical application in risk/return management.

One basic task in risk/return management is the frequent estimation of the risk exposure associated with a portfolio of investment objects (e.g. securities). Today, enterprises mostly calculate their risk exposure during fixed time intervals like e.g. several days. With the possibly huge amount of computing capacity a SOI based on grid technologies offers (embracing resources of the whole enterprise or even of external resource providers), calculations can be accelerated dramatically (Middlemiss 2004).

Concerning risk/return management[11], we restrict our considerations to publications addressing risk forecasting and in particular the estimation of covariance matrices. Huther (2003, pp. 111), or Faisst and Buhl (2005, pp. 408) for example describe the use of covariances for a comprehensive enterprise-wide risk/return management (see Fill et al. 2007; Kundisch et al. 2007; Gericke et al. 2009). Other publications are dealing with the question of how covariances can

[11]Whereas literature most often focuses on either side separately, we will rather speak of risk/return management emphasizing an integrated view because risk management can only unfold its potential in combination with the management of the corresponding return.

be empirically estimated or forecasted by analyzing historical data. In fact forecasting is "one of the important problems in finance" (Elton and Gruber 1972, p. B-409) and consequently there are a lot of publications already covering the question of how many and which historic quotations should be used to determine a suitable risk forecast. To give an impression about applicable techniques we refer to Engle (1982), Kupiec (2007) and Hull and White (1998). Alexander (1996, pp. 233) provides an overview of the corresponding methods used in volatility and correlation forecasting. Recent approaches to volatility forecasting like conditional autoregressive Value-at-Risk (CAVaR) models are presented in Taylor (2005). It is widely known that the corresponding calculations are very resource and time intensive. Therefore, Buhl et al. (2009) already proposed a model identifying the optimal computing capacity to be invested into risk/return management on a SOI. Nevertheless, a Dedicated System (DS) where a fixed set of resources is dedicated to the risk/return calculations could be a sensible alternative to a SOI. By following the approach of Buhl et al. (2009), this paper aims at answering the following research question: *Under which conditions is a SOI superior to a DS regarding calculations in risk/return management?* We formulate an analytical optimization model delivering a solution to this decision problem.

II.2.2 Grid Computing

Grid computing can be regarded as an infrastructure technology enabling the virtualization of physical resources. Available definitions for the term grid computing are mostly of descriptive nature and provide little more than certain essential characteristics (see e.g. Foster 2002; Foster et al. 2001; 2002; Foster and Kesselman 1998). However, various proponents have agreed with Foster and Kesselman (1998) that "A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities". Today, an evolution towards SOA can be observed Longworth (2004). Grid technologies are one possibility to realize a SOA consisting of so called "grid services". Grid services are based on specific web service standards, like the specifications (Open Grid Services Architecture) and (Web Services Resource Framework). They extend web services insofar as they imply the dynamic, yet for the user transparent, allocation of (physical) resources to services by a grid middleware and therefore are especially suited to fulfil resource intensive tasks. There is an extensive literature on service-oriented computing or grid computing in general (see e.g. Foster and Kesselman 1998; Berman 2005; Silva 2006; Singh and Huhns 2004). Some publications even consider the application of these technologies for portfolio management, derivatives pricing or other areas of financial risk management (Brownlees et al. 2006; Crespo et al. 2006; Schumacher et al. 2006). However, these approaches describe how grid technologies can be applied, but do not quantify the resulting business value. In the context of grid computing also resource allocation mechanisms have been widely discussed—most often under the term "grid economics". We refer to Regev and Nisan (1998), Buyya et al.

(2000), Nabrzyski et al. (2003) and Wolski et al. (2004) which provide an overview of this area. They most often dwell either on the question which principles are appropriate to resource management or on technical and architectural issues connected with the development of resource management systems. Accordingly, in most of the existing approaches demand for computational resources is merely an external factor whereas in our approach it is subject to optimization.

In this context an important characteristic of SOI based on grid technologies is the on-demand access to distributed resources. When resources stem from an external provider this concept is often labelled utility computing, meaning that resources can be consumed und priced as easy as for instance electricity or water. Utility computing has been subject to research as well. Bhargava and Sundaresan (2004) analyze pay-as-you-go pricing scenarios where providers guarantee computing capacity, but users do not make a commitment towards actual use. Our paper to some extent continues the ideas of Bhargava and Sundaresan(2004). However, we take the perspective of a service user and present a rationale for decisions on computing capacity in the context of risk/return management.

Enterprises are still facing the question when and where to adopt the new service-oriented computing concepts. This may be accomplished by investing into an own internal SOI or by making use of utility computing offers by major infrastructure providers like HP, SUN or IBM (Bhargava and Sundaresan 2004, p. 202) but also online service providers like e.g. Google Amazon and Microsoft, whose services are today oftentimes marketed under the term “cloud computing”. The fundamental question underlying this paper is under which circumstances a SOI for risk quantification is favorable over (traditional) DS.

II.2.3 Risk/Return Management

The term “risk” is used heterogeneously in general speaking as well as in academic circles. Therefore we feel that it is appropriate to begin with a definition of risk before we describe the various requirements and objectives of risk/return management applications. While in the economic literature risk is often generically explained as the “possibility of missing a planned outcome” we will follow a more finance-related approach. From this point of view we define with Schröck (2001, p. 24) risk as “the deviation of a financial value from the expected value”. A positive deviation is often in general speaking referred to as “chance” while a negative deviation is characterized as “danger”. Because of this two-sided perception of risk, variance or standard deviation of a risky value are suitable and well accepted measures of risk. We will use the standard deviation of historical portfolio returns as the risk measure later in this text. Synonymously we will speak of the volatility of a portfolio and define it as the “annualized standard deviation of percentage change in daily price” (Spremann 2003, pp. 154).

Enterprises are investing capital into investment objects in order to generate cash inflows and subsequently to increase the return of the invested capital. Typically risk-averse management is making risky investments hoping to achieve an excess return over the risk-free rate. There is a general connection between risk and return of an investment object: higher return is systematically associated with higher risk. This connection is theoretically explained by economic models like the CAPM (the "Capital Asset Pricing Model" was originally developed by Sharpe (1964), Lintner (1965) and Mossin (1966)) and empirically evaluated later on (an overview of relevant empirical studies can be found e.g. in Copeland and Weston (1988, pp. 212)). Following the argumentation of Wilson (1996, pp. 194) it is therefore crucial for the survival and success of an enterprise to be able to allocate the available capital to the right combination of investment objects, taking into account their specific contributions to the overall risk and return. Investment objects in this context are not restricted to securities. Almost all business transactions are associated with uncertainty and thus contribute to an enterprises overall risk exposure. Thus in the spirit of an enterprise-wide risk/return management all investments an enterprise is engaged in, like credit decisions or even customers or projects can be seen as components of the enterprise's overall investment portfolio, having a return and a variance (as a measure of risk).

One major goal of risk/return management in this context is the prevention of bankruptcy by restricting potential losses resulting from risky investment objects. The increasing importance of this goal is emphasized by the current economic crisis. A growing number of rules and regulations requires enterprises to hold a part of their available capital to back their risky investments (Jackson et al. 1998, pp. 8). This share of the available capital then makes less or no contribution to the overall earnings. By management decisions these restrictions are broken down along the organizational hierarchies into guidelines on business unit or departmental level. We are assuming in the following text that those guidelines are essentially representing limits for the maximum risk a department, business unit and consequently an enterprise is willing (or able) to take.

In order to evaluate whether an enterprise or department complies with a given risk limit, it is necessary to calculate the current risk exposure frequently. For simplifying means, we concentrate on one fundamental instrument in this paper: The covariance approach. This constitutes a basic principle in finance and forms the foundation for many risk/return management applications ranging from Markowitz portfolio optimization to Value-at-Risk (VaR) calculations. In our context, covariances are used for determining the overall risk position of an enterprise taking into account diversification effects that exist between the investment objects. Nevertheless, the proposed methods are also applicable to other risk measures as long as they take dependencies between single investment objects into account.

Following the covariance approach, we can represent risky investment objects by random variables. Typically historical data are used in order to derive a distribution for a random variable[12] and calculate the distribution parameters. It is worth mentioning that considering the portfolio risk and return merely on an aggregated level is not satisfactory because all information is lost about the risk attributable to a single investment object. It is crucial to separate the portfolio and decompose it into data per investment object, i.e. to calculate the covariances. Only then the enterprise can perform economically rational investment decisions on different aggregation levels. With σ_i^2 and Cov_{ij} denoting variance and covariance of investment objects respectively we can determine the overall risk of a portfolio σ_P^2, consisting of $n \gg 0$ investment objects (numbered from 1 to n), as[13]

$$\sigma_P^2 = \sum_{i=1}^{n} \sigma_i^2 + \sum_{i=1}^{n} \sum_{j=1;j\neq i}^{n} Cov_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{n} Cov_{ij}$$

The so defined matrix of all covariances is called *covariance matrix*. An important characteristic of covariances is that $Cov_{ij} = Cov_{ji}$. This makes the matrix symmetric and thus not all of its values have to be calculated. The total number of covariance calculations necessary is given by $n(n+1)/2$.

II.2.4 A Valuation Model for fast Risk Quantification

We consider an enterprise which has access to (and is possibly engaged in) a set of risky investment objects as well as to a risk-free investment alternative. It is frequently (re)calculating its overall risk position by estimating the covariance matrix of its portfolio. Our main hypothesis for the valuation of benefits is: the faster risk/return management calculations can be executed the higher will be the return of the enterprise because given risk limits can better be exploited.

Since the enterprise is acting in an uncertain and dynamic environment its risk position is changing willingly (by investment decisions) or unwillingly (by "movement" of the markets). Because the estimation of risk cannot be accomplished in real-time the covariances at hand are

---

[12] A thorough overview and discussion of the statistical analysis of financial data can be found in Shiryaev (1999, pp. 314).

[13] As we analyze the complete investment objects and not parts or quantities of investment objects (e.g. stocks), this formula contains no weights.

always significantly outdated. We are in the following recurring to the fact that the enterprise is adjusting its risk position to a value somewhere below a certain threshold thus constituting a "safety margin". In the regulatory context this is often called "haircut", like in Basel Committee on Banking Supervision (2004). The financial and economic crisis made clear, that – depending on the assets invested in – a high safety margin is necessary to account for changes of parameters, especially in liquidity. It is doing so by using the capital allocation between the risky investment objects and the risk-free alternative for balancing their overall risk position. It is important to understand that our model is not addressing the evaluation of the efficient set of investment objects or portfolio optimization (both require covariances), but the aggregation and management of the risk position of an enterprise. Whenever covariances are available the safety margin can be adjusted immediately in a way that the resulting (and over time changing) overall risk position of the enterprise does not exceed the given risk limit at any time. Hence, the faster covariances are available, the smaller the safety margin can be. We will use this effect to quantify the benefits of fast covariance estimations depending on the time needed for the completion of one covariance matrix.

II.2.4.1 Model Setting and Basic Assumptions

The time interval under consideration consists of equidistant periods such that $t = 0$ denotes the beginning of the current period. We shall write for example σ_t to indicate the value of a model parameter at the end of period t. Correspondingly (dis)investment decisions take effect only at the end of each period. If not mentioned otherwise all variables assume real values, i.e. values $\in \mathbb{R}$.

The enterprise is equipped with a total capital of $K > 0$ which is always completely allocated to the risky portfolio and/or the risk-free alternative. We denote the risky portion of K with $x \geq 0$ and furthermore use x for risk adjustment. T indicates the length of the calculation time frame, $T \in \mathbb{N}\backslash 0$ with $\mathbb{N}$ as the set of natural numbers. At the end of each time frame we choose x in a way that the risk limit is "probably" not exceeded during the next time frame. We will formulate more precisely what is meant by "probably" later on. Future returns of the portfolio are modelled as independent random variables. Their probability distribution for each period can be characterized by mean and standard deviation. This implies that the investment objects can be marked to market, i.e. there is a price attached to them. We additionally need a set of assumptions for the deductions following thereafter.

Assumption 1 *The enterprise is generally risk-averse and striving for efficient combinations of investment objects. As sufficient*[14] *number of investment objects are perfectly divisible, liquid, and traded on a no-frictions market.*

Assumption 2 *The risky part of the enterprise's capital yields the expected return* μ*, the risk-free investment pays the time-invariant risk-free interest rate i, which is equal to the borrowing rate.*[15] *We always have* $\mu > i > 0$.

Assumption 2 is made in the spirit of the model of Modigliani and Miller (1958) where enterprises and investors can borrow or place money at will for a risk-free rate, but expect a premium for taking the risk associated with the investment in risky assets. With the so defined parameters and x as decision variable we can determine the overall expected return $\mu^u(x)$ and risk $\sigma_t^U(x)$ of the enterprise according to common rules of statistics as

$$\mu^U(x) = x\mu + (1-x)i = i + x(\mu - i) \text{ and } \sigma_t^U(x) = x\sigma_t \quad (1)$$

The overall risk of the enterprise is expressed by the portfolio risk (the enterprise is the weighted "sum" of its investment objects) and thus changes over time driven by the varying σ_t. Note that due to our focus on changing risk we do not regard a changing $\mu^U(x)$ over time (which would result in an index t as in $\sigma_t^U(x)$). As we see, with ascending x the overall returns as well as the overall risk of the enterprise are both increasing. On the one side the enterprise certainly strives for the highest possible return, on the other side a limitation exists for x from the given risk limit.

It is common practice to use some variation of a random walk for the price movement on security and commodity markets. This approach goes ultimately back to Louis Bachelier (1900) who compared the stock market with a "drunkards walk"). Although controversially discussed, it was picked up more than half a century later by Mandelbrot (1963; 1972) and Fama (1965) among others. Following this theory of random walks historical (e.g. daily or weekly) portfolio returns can be used for estimating mean μ and standard deviation σ_t of future portfolio returns.

---

[14]In the sense of the "Portfolio Selection" theory (Markowitz 1971) investors are trying to achieve the highest possible return on their investment for a given risk. They are acting under perfect trading conditions, i.e. no arbitrage, no transaction cost, strong information efficiency etc. For our model this is only required within the safety margin.

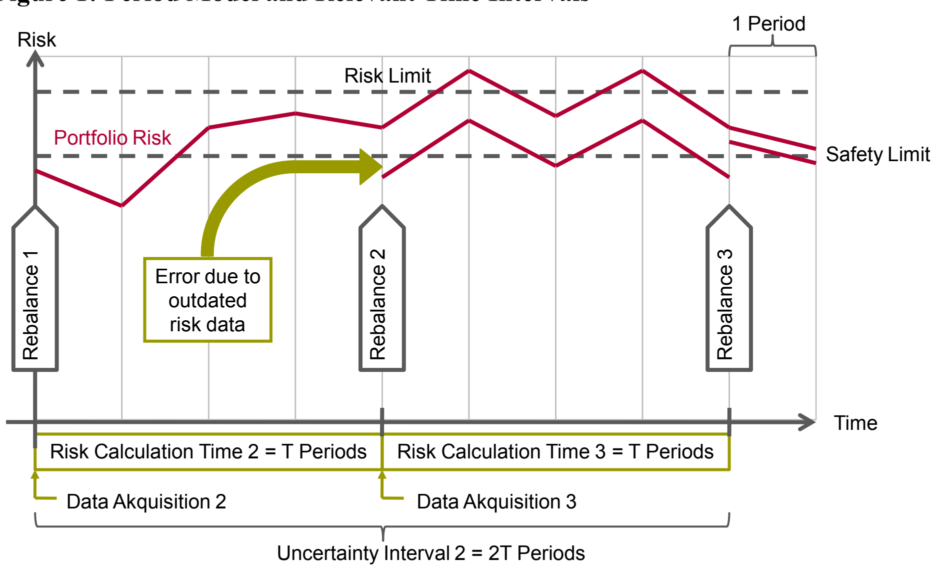
[15]The equality of lending and borrowing rate is assumed for sake of simplicity and justifies the case $x > 1$, where the enterprise is actually borrowing money for making risky investments.

It is important to understand that in our model the standard deviation is possibly changing in each period (indicated by its index t).

Assumption 3 *The initial calculation of covariances starts in $t = 0$ and is finished after T periods. Each new covariance calculation begins in the finishing period of the previous covariance calculation.*

According to assumption 3, whenever a covariance matrix is completed, the input data used for its calculation are T periods old. We can immediately determine the portfolio risk by summing up the covariances in the matrix. This can then be used for rebalancing the portfolio. The moment before the next matrix is finished the input data used for risk calculations are already $2T$ periods outdated. Therefore the uncertainty interval that has to be taken into account spans $2T$ periods: in the worst case the risk has been going up over $2T$ periods before the enterprise realizes that it is exceeding the maximum risk it is willing (or able) to take (see figure 1). Without loss of generality we will concentrate our analysis on the first covariance matrix calculation and the corresponding adjustment decision, therefore focussing on the time interval $[0; 2T[$. During this time the portfolio risk is fluctuating in a non-predictable way.

Figure 1: Period Model and Relevant Time Intervals



II.2.4.2 The Risk-at-Risk Approach

We will now dwell on the portfolio risk at time t, denoted as σ_t, modelling it as a random variable. This relates to a phenomenon known from the behavior of stock market prices called *heteroscedasticity* (see e.g. Spremann 2003, pp. 152). In analogy to the periodical returns we write

Assumption 4 *The σ_t are normally distributed.*

This distribution assumption can (and would in practice) be relaxed by approximating the distribution of the σ_t delivered by the calculated sequence of standard deviations. Arranged in increasing order one can easily deduce the quantiles needed in our model. Nevertheless, for reasons of simplicity and without significantly changing the general result we assume σ_t to be normally distributed here.

We will again focus on two distribution parameters: Our notation for the (strictly positive) mean will be μ_σ, for the standard deviation σ_σ (both tagged with an σ indicating that the distribution applies to the portfolio *risk*), thus $\sigma_t \sim N(\mu_\sigma; \sigma_\sigma)$ with N short for the normal distribution.

The distribution parameters in our model can again be estimated using historical data. For example, the standard deviation of the portfolio risk can be taken as an estimate for the expected portfolio risk μ_σ and thus as the starting point for the random walk of σ_t In order to maximize x in equation (1) under the given constraints we have to consider the uncertainty interval $[0; 2T[$. Because of assumption 4 the standard deviation of the expected portfolio risk after $2T$ periods is (as a sum of normally distributed random variables) again normally distributed according to $N(\mu_\sigma; \sigma_\sigma\sqrt{2T})$. As a consequence for the overall risk of the enterprise we have $\sigma_t^U(x) \sim N(x\mu_\sigma; x\sigma_\sigma\sqrt{2T})$.

In order to rephrase the fuzzy formulation "the risk limit is probably not exceeded" we will follow an approach comparable to the VaR for quantifying portfolio risk. We speak of a Risk-at-Risk over a holding period and a confidence level α, $0 \ll \alpha < 1$ and think of it as the standard deviation $\bar{\sigma}$ which is exceeded within the holding period only with the (small) probability of $(1-\alpha)$. With $\Phi(x)$ denoting the standardized normal distribution function, we know for the distribution of $\sigma_t^U(x)$ over $2T$ periods that

$$P(\sigma_t^U(x) \leq \bar{\sigma}) = \Phi(\frac{\bar{\sigma} - x\mu_\sigma}{x\sigma_\sigma\sqrt{2T}})$$

At the same time we require in the spirit of the Risk-at-Risk approach the probability given above to be greater than or equal to the confidence level α, i.e.

$$P(\sigma_t^U(x) \leq \bar{\sigma}) \geq \alpha \text{ for } t = 2T \tag{2}$$

In the marginal case both sides of the equation are equal and we can therefore state—with q_α as the (onesided) α-quantile of the standardized normal distribution—that

$$\frac{\bar{\sigma} - x\mu_\sigma}{x\sigma_\sigma\sqrt{2T}} = q_\alpha \Leftrightarrow x = \frac{\bar{\sigma}}{q_\alpha\sigma_\sigma\sqrt{2T} + \mu_\sigma} \tag{3}$$

x gives us the portion of risk-free and risky investment objects in a way that equation (2) holds. We can calculate the overall expected earnings of the enterprise, given this capital allocation, as

$$B(x) = \mu^U(x) \cdot K = (i + x(\mu - i)) \cdot K$$

Obviously $B(x)$ represents only expected, calculatory (and not real) earnings because the returns are not fully cash-flow effective and the earnings themselves subject to interpretation. We will neglect the adjectives "expected" and "calculatory" and speak of earnings or, more generally, of benefits.

By inserting x from equation (3) into $\mu^U(x)$ from equation (1) we find for the benefits (with the number of periods needed for the completion of one covariance matrix as the independent variable)

$$B(T) = \left(i + \frac{\bar{\sigma}(\mu - i)}{q_\alpha\sigma_\sigma\sqrt{2T} - \mu_\sigma}\right) \cdot K \tag{4}$$

Considering equation (4) an enterprise could maximize its benefit by minimizing the time T that is needed to calculate a covariance matrix. Yet there is a trade-off between the benefits and the cost, i.e. cost caused by the infrastructure that is necessary to compute the calculations.

II.2.5 Required Computing Capacity on Service-Oriented Infrastructures

From a SOI point of view the risk calculation can be regarded as a service providing its user transparently with up-to-date risk information for the relevant investment universe. In this section we derive the relationship between the computing capacity (i.e. cost) allocated to risk quantification and the time needed for the computation.

We denote with z the computing capacity necessary for estimating one covariance matrix in exactly T periods. We use CPUs as a measure for computing capacity and are aware of the fact that this means a one-dimensional view on matters as other determinants of system performance are ignored. We denote with w the workload—measured in CPU hours—for estimating one covariance. Applying a simple moving average technique with a rolling sample of historical data (Elton and Gruber 1972, pp. 409) we get unbiased estimators of the expected value and (co)variances for every point in time.

Assumption 5 *The same workload w is necessary for estimating variances and covariances.*[16] *Furthermore every covariance is estimated from scratch, i.e. no intermediate results are used.*[17]

Assumption 6 *The length of the calculation time frame T depends solely on the time needed for the computation, neglecting e.g. latency or transmission times. Correspondingly the only cost relevant is cost for computation which occurs in the form of a (internal or external) factor price per CPU hour over a given time.*

Note that for the calculation of covariance matrices on a SOI it is convenient that the computations can be distributed on several nodes and executed in parallel, as all pairwise covariances can be calculated independently from each other. Thus efficiency losses are considerably low.

We can now deduce the computing capacity $z(T)$ (workload per time) that is required in every period over T periods. We already know that for n investment objects $n(n+1)/2$ covariances have to be calculated. Multiplied with the workload per covariance this determines the total number of CPU hours needed. This in turn—divided by the calculation time frame—leads to the functional relationship

$$z(T) = \frac{n(n+1)w}{2T} \Leftrightarrow T(z) = \frac{n(n+1)w}{2z} \tag{5}$$

Given n investment objects and a computing capacity of z CPUs, the covariance matrix will be completed after $T(z)$ periods.[18] This constitutes an important parameter of the covariance esti-

---

[16]Variance estimation requires only one historical time series, so its intrinsic workload is smaller than the workload for covariance estimation. Nevertheless, this effect can be neglected since for n variances there are $n(n-1)/2$ covariances in a given covariance matrix. With n sufficiently large the variances have merely no effect on the number of calculations (e.g. with $n = 201$, the number of variances is only 1% of the number of covariances to be calculated).

[17]This could be considered awkward for the simple moving average procedure but is a realistic approach for more sophisticated methods.

mation service since it describes the economic value that should be attributed to the consumption of capacity for covariance estimation. By inserting (5) into equation (4) and neglecting the mean risk compared to the standard deviation of the risk[19] we quantify the benefits as

$$B(z) = a + 2b\sqrt{z} \text{ with } a = iK > 0, b = \frac{\overline{\sigma}(\mu - i)K}{2q_\alpha \sigma_\sigma \sqrt{n(n+1)w}} > 0$$

$$B'(z) = \frac{b}{\sqrt{z}} > 0,\ B''(z) = -\frac{b}{2\sqrt{z^3}} < 0 \tag{6}$$

$B(z)$ is a strictly increasing and concave function.

In the following we will consider two alternatives for balancing the benefits described by equation (6) with the cost of computation: On the one hand an enterprise may employ a SOI. On the other hand it may invest into a DS like a server or a cluster of servers dedicated only to risk quantification.

II.2.6 SOI vs. Dedicated Systems

As a base of the decision on SOI vs. DS, the enterprise faces the problem of determining the required computing capacity for its risk-/return management in advance. Especially in a DS setting, it is obviously necessary to acquire or reserve resources for a longer planning horizon (e.g. one year). Even in a SOI scenario, where one might expect capacity planning to be superfluous, under realistic conditions it is still necessary: In the case of internal provision the enterprise needs to decide on the overall capacity of its SOI thus affecting the availability and prices of the resources.[20] When resources are provided externally it may be required to forecast and reserve capacity in advance. This is in accordance with the findings of Bhargava and Sundare-

[18]Here as well as for the optimal $T(z^*)$ later in this text the outcome is assumed real-valued. In reality and in order to fit it to the discrete-time period model one has to check the neighboring integer values to obtain the discrete optimum.

[19]The exact quantification would lead to a strictly increasing and concave benefit function B, as well. It would nevertheless be tedious to continue in our analysis with the exact expression. In order to avoid writing overhead we deliberately simplify our objective function by neglecting the mean risk, which is a numerically justifiable approximation in our context.

[20]It is worth emphasizing that a DS is specifically dedicated to risk calculations whereas a SOI is universally applicable for different services. Thus in a SOI cost and benefits need to be interpreted as contributions to overall benefits and cost of the infrastructure.

san (2004) who find that a pure pay-as-you-go model without reservation is not reasonable in most cases.

We will therefore determine suitable cost functions depending on the computing capacity used for risk quantification both on the DS and the SOI. In our notation of the model parameters, we will use the lower indices D for the DS and S for the SOI, respectively. For the DS, we apply a cost function $C_D(z)$ proportional to the computing capacity z provided per period which in fact assumes that there exist suppliers for DS with virtually any capacity. This leads us to a continuous cost function with a price of p_D per computing capacity used over one period. For the SOI we also specify a proportional cost function with slope p_S, but with a fixed minimum cost of $p_S z_m$, $z_m > 0$. Thus the SOI cost function has a slope of 0 to the point z_m and afterwards a constant slope of p_S, i.e. for all $z \geq z_m$ every unit of capacity over one period cost p_S. The break at point z_m accounts for the following situations that need to be considered in practice for an internal and external provisioning scenario respectively:

- The SOI capacity is obtained by an external service provider for the usage price p_S. The service consumer on the other side is obligated to a minimum purchase of z_m. Consuming less capacity nevertheless induces cost of $p_S z_m$.
- The SOI is instituted by an investment project of the enterprise. After initial investment cash-flows free capacity up to the amount of z_m is available for covariance estimations at variable cost of 0 because of pooling existing internal resources, until the free capacity generated by the SOI project is fully exploited. A capacity demand exceeding z_m induces a usage price of p_S – because of competing service consumers in the case of an internal SOI or because of additional external capacity priced with p_S.

We assume that always $0 < p_S < p_D$[21] reflecting the typical advantages of a SOI. Firstly, an intrinsic characteristic of a SOI is that a potentially high number of services share a common infrastructure. Capacity must not be determined by summing up the requirements of each single service independently. Instead multiplexing gains resulting from less than perfectly correlated demand structures need to be considered (Chandra et al. 2003). Therefore, total capacity may be significantly smaller compared to independent dedicated systems. Additionally, a SOI may comprise low level standardized components which can be assumed to be substantially cheaper. Finally, on a SOI capacity can be easily varied by adding or removing resources. Thereby it is possible to react to varying environment conditions which is a crucial part of the results of our optimization model presented in the previous section.

[21]For $z_m = 0$ we would therefore obviously have the trivial solution that the SOI is always preferable to the DS thus narrowing the case to a mere SOI capacity planning problem.

Consequently, we define the cost functions, C_D and C_S, as well as the objective functions, Z_D and Z_S, for the DS and the SOI with recourse to equations (5) and (6) for $z > 0$ as

$$C_D(z) = p_D z, \qquad Z_D(z) = B(z) - C_D(z)$$

$$C_S(z) = \max(p_S z_m; p_S z), \qquad Z_S(z) = B(z) - C_S(z)$$

$Z_D(z)$ and $Z_S(z)$ are again strictly concave functions. $C_D(z)$, $C_S(z)$ and $Z_D(z)$, $Z_S(z)$ respectively are intersecting at $z_1 = \frac{p_S z_m}{p_D} < z_m$. We set $c = p_S z_m = p_D z_1$ and obtain a cost setting as depicted in figure 2.

Figure 2: Cost Functions for DS and SOI

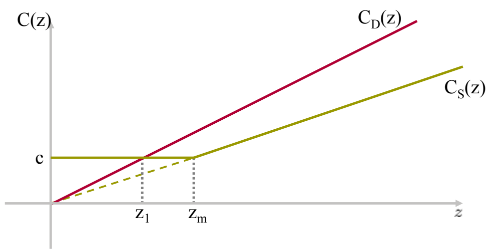
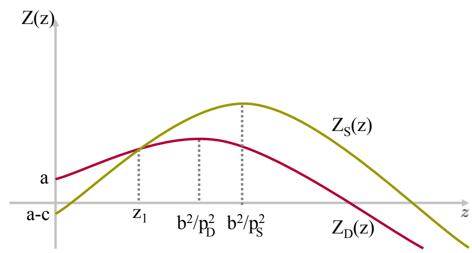


Figure 3: Objective Functions for DS and SOI



In practice one would insert realistic numbers (e.g. expected values for the planning interval) into the model parameters. By comparing the values of the objective functions it would be easy to determine the overall optimum *numerically*. In the following we will nevertheless deduce *analytically* the conditions under which a long-term investment in a DS and a SOI respectively is optimal. We will achieve this in three steps.

Step 1: Obviously we have, as illustrated in figure 3, $Z_D(z) \geq Z_S(z) \Leftrightarrow 0 < z \leq z_1$, with equality exactly for $z = z_1$. The first derivative $Z'_D(z) = \frac{b}{\sqrt{z}} - p_D$ features its only null at $\frac{b^2}{p_D^2} > 0$. Due to the strict concavity of $Z_D(z)$ the only maximum of $Z_D(z)$ on $]0; z_1]$ is at $z_D^* = min\left(\frac{b^2}{p_D^2};\ z_1\right)$.

The associated value of the objective function is

$$Z_D(z_D^*) = \begin{cases} Z_D\left(\frac{b^2}{p_D^2}\right) = a + \frac{b^2}{p_D} \text{ for } \frac{b^2}{p_D^2} < z_1 \text{ and thus } z_D^* = \frac{b^2}{p_D^2} \\ Z_D(z_1) = a - c + 2b\sqrt{z_1} \text{ for } \frac{b^2}{p_D^2} \geq z_1 \text{ and thus } z_D^* = z_1 \end{cases} \tag{7}$$

Step 2: Furthermore we have $Z_S(z) \geq Z_D(z) \Leftrightarrow z \geq z_1$, with equality exactly for $z = z_1$. Since $Z_S(z)$ is strictly increasing on $]0; z_m[$ the maximum can only occur on $[z_m; \infty[$. The first derivative $Z'_S(z) = \frac{b}{\sqrt{z}} - p_S$ for $z > z_m$ features its only null at $\frac{b^2}{p_S^2} > 0$.

- Case 1: $\frac{b^2}{p_S^2} \leq z_m \Rightarrow Z'_S(z) < 0$ for $z > z_m \Rightarrow Z_S(z) = Z_S(z_m) + \int_{z_m}^{z} Z'_S(t)dt < Z_S(z_m)$ for $z > z_m$. Therefore in this case $z_S^* = z_m$ is the only maximum.
- Case 2: $\frac{b^2}{p_S^2} > z_m \Rightarrow \exists z_0 > z_m$ with $Z'_S(z_0) > 0$ and hence $Z'_S(z) > 0$ for $z_m < z \leq z_0$
  $\Rightarrow Z_S$ has its only maximum on the right side of z_m at the point $\frac{b^2}{p_S^2}$, i.e. $z_S^* = \frac{b^2}{p_S^2}$ in this case.

Due to the strict concavity of $Z_S(z)$ for $z > z_m$ the only maximum of $Z_S(z)$ on $[z_1; \infty[$ is at $z_S^* = max\left(\frac{b^2}{p_S^2}; z_m\right)$

The associated value of the objective function is

$$Z_S(z_S^*) = \begin{cases} Z_S\left(\frac{b^2}{p_S^2}\right) = a + \frac{b^2}{p_S} \text{ for } \frac{b^2}{p_S^2} > z_m \text{ and thus } z_S^* = \frac{b^2}{p_S^2} \\ Z_S(z_m) = a - c + 2b\sqrt{z_m} \text{ for } \frac{b^2}{p_S^2} \leq z_m \text{ and thus } z_S^* = z_m \end{cases} \tag{8}$$

Step 3: Finally the optimal values of the objective functions have to be compared for the DS and the SOI respectively. We have

$$Z_D(z_D^*) >^{[=]} Z_S(z_S^*) \Leftrightarrow a + \frac{b^2}{p_D} >^{[=]} Z_S(z_S^*),$$

since $Z_D(z_1) = Z_S(z_1) < Z_S(z_S^*)$. This is equivalent to

$$a + \frac{b^2}{p_D} >^{[=]} Z_S(z_m) = a - c + 2b\sqrt{z_m},$$

since $Z_S\left(\frac{b^2}{p_S^2}\right) = a + \frac{b^2}{p_S} > a + \frac{b^2}{p_D}$, which again can only be satisfied (with $c = p_S z_m$) for

$$2b\sqrt{z_m} - \frac{b^2}{p_D} <^{[=]} p_S z_m \Leftrightarrow p_S >^{[=]} \frac{2b\sqrt{z_m} - \frac{b^2}{p_D}}{z_m}.$$

The overall optimum z^* and the associated type of system can then be expressed without loss of generality depending on the value of the parameter p_S relative to the other relevant model parameters p_D, b and z_m. Substituting for simplicity $\frac{b}{\sqrt{z_m}} = p_1$ and $\frac{2b\sqrt{z_m} - \frac{b^2}{p_D}}{z_m} = p_2$ we can state the overall optimum[22]

$$z^* = \begin{cases} \frac{b^2}{p_D^2} \Leftrightarrow p_2 \le p_S \text{ and } p_1 \le p_S \Rightarrow \text{ DS optimal} \\ z_m \Leftrightarrow p_1 \le p_S \le p_2 \Rightarrow \text{ SOI optimal} \\ \frac{b^2}{p_S^2} \Leftrightarrow p_S < p_1 \Rightarrow \text{ SOI optimal} \end{cases} \tag{9}$$

This result can be interpreted as a decision rule determining whether or not the investment into a SOI or into a DS is ex ante economically rational; at the same time the question regarding the optimal capacity is resolved: Clearly, the investment into a DS is optimal when the variable cost of the SOI exceeds a certain threshold relative to the other relevant model parameters. For p_S at this threshold the enterprise would be indifferent on the DS and the SOI (with different capacities each). Below the threshold the SOI is optimal with a capacity – depending on the value of z_m – of either z_m or $\frac{b^2}{p_S^2}$.

From a managerial perspective it is an important aspect that IT investment decisions in this specific context can be based on cost as well as benefits. In general cost for IT infrastructures can be quantified quite well, whereas benefits most often are only roughly estimated. With our

---

[22]In the case $p_S = p_2$ the DS as well as the SOI are optimal.

model we provide an overall quantification (under certain assumptions). We can moreover examine how input parameters like K, μ or i affect investment decisions. For example the larger the capital K, the higher is b and p_1 respectively. Accordingly, the price p_S is likely to be lower than p_1 and thus investing into SOI becomes more attractive. The same argumentation holds for the risk premium $(\mu - i)$.

II.2.7 Limitations of the Model and Conclusion

In this paper we demonstrated how the economic value that can be derived from risk/return management calculations can be measured considering an enterprise that has to decide on the amount of capital it wants to reserve to cover potential losses resulting from a risky investment portfolio. Several assumptions (e.g. regarding the distribution of the expected portfolio risk) were necessary to achieve an analytical solution.

Using the covariance approach as an example we moreover developed an optimization model that delivers the optimal amount of computing capacity that should be allocated to risk calculations at a time. In doing so we restricted our analysis to one well-defined risk/return management problem. Although covariances are fundamental and widely used in financial applications we thereby covered only one element of numerous risk/return management methods and algorithms. Other approaches and applications for SOI concepts (like for instance Monte-Carlo simulations which also have a very high parallelization potential) have to be examined as well. In fact, most of the basic principles introduced in this paper can be adapted to other scenarios in more sophisticated and complex surroundings.

Putting it all together a SOI is especially advantageous when market parameters determining the benefits of risk calculations are highly volatile as could be observed during the crisis since July 2007 resulting in varying demand for computing capacity. With a SOI, resources can be reallocated at any time to reach an economic optimum. As discussed, not only benefits but also (opportunity) cost may vary depending on the total demand for capacity. For example, during "quiet times" risk calculations may be computed more frequently generating added value out of readily available excess capacity even if benefits are comparably small. On a DS on the other hand an economical allocation cannot be guaranteed. When expected values are applied for capacity planning of a DS the economic optimum is systematically missed when parameters are deviating from expectations. We analyzed the different cost structures of DS and SOI resulting from these fundamental differences and provided a decision rule for investing into SOI for risk/return management.

One caveat not mentioned in this paper is information security. As information on investment objects may be sensitive business data, spreading the calculations over the company or even over service providers may not be desired. Implementing a system as described would therefore require additional security mechanisms and persuasion of the management.

After all, this paper is a contribution to understand the application of service-oriented infrastructures in the specific domain of risk/return management. Although a validation of our findings based on real-world data is still subject to further research, in our point of view, such a systematic and economic analysis is a requirement as a first step for the further development of the new concepts like service-oriented computing or utility computing.

II.3 Using a Grid for Risk Management: Communication Complexity of Covariance Calculations

Authors:	Fridgen, Gilbert, FIM Research Center Finance & Information Management, Department of Information Systems Engineering & Financial Management, University of Augsburg, Universitätsstraße 12, 86159 Augsburg, Germany, gilbert.fridgen@wiwi.uni-augsburg.de

In:	Proceedings of the 15th Americas Conference on Information Systems, AMCIS, San Francisco, California: Paper 476. http://aisel.aisnet.org/amcis2009/476. Access 2010-01-06

Risk management has evolved as one of the key success factors for enterprises especially in the financial services industry. It is highly demanding in terms of business requirements and technical resources, making it an almost ideal application for distributed computing concepts like e.g. grid computing. In this paper we focus on a specific problem—the estimation of covariance matrices that provide a powerful tool for decisions on investments. In this context we analyze different network topologies that the corresponding calculations can be performed on. We derive complexity classes for a distributed calculation scenario on these topologies. As a general result we find an upper and lower bound for the complexity of a distributed calculation in an arbitrary network structure. These results not only provide a different view on grid resource allocation but also make a contribution towards better understanding the business value of grid computing.

II.3.1 Introduction

Whereas in the beginning grid computing was restricted to large-scale scientific applications, it has over the past years evolved to an increasingly relevant technology for the commercial sector. It is to some extent difficult to differentiate grid computing from related concepts of e.g. distributed computing, cluster computing, utility computing, or Service-Oriented Architectures. The meaning we intend to convey by our use of the term grid computing in this article is best captured by a definition of Buyya (2002): ‘A Grid is a type of parallel and distributed system that enables the sharing, selection, and aggregation of geographically distributed, autonomous resources dynamically at runtime depending on their availability, capability, performance, cost, and users’ quality-of-service requirements’. An overview of the status quo and current applications of grid computing provide e.g. Berman (2005), Foster and Kesselman (2003), and Abbas (2004).

The availability of grid enabled business applications is a critical success factor for the wider adoption and further development of this powerful technology. One of the most promising domains for grid computing is the financial services industry with its information-driven business models and high needs for computing power. In fact this sector is often mentioned among the key industries for grid computing applications (Ricadela 2002; Smith and Konsynski 2004; Friedman 2003). In this context resource-intensive risk management applications seem to be especially suitable. With the potentially huge amount of computing capacity a grid infrastructure offers (embracing resources of the whole enterprise or even of external providers) they can possibly be accelerated dramatically. Yet, even on a grid infrastructure resources are not unlimited. Thus it is necessary to consider the complexity associated with the corresponding calculations. For our analysis we focus on a specific problem in risk management—the calculation of covariance matrices that, on the one hand, provide a powerful tool for decisions on investments but, on the other hand, are a very complex and time-consuming assignment.

II.3.1.1 Principles of Risk Management

Enterprises are investing capital into risky investment objects in order to generate cash inflows and subsequently to increase the return of the invested capital. Especially with the background of the financial crisis and following the argumentation of Wilson (1996, pp 194) it is therefore crucial for the survival and success of an enterprise to be able to allocate the available capital to the right combination of investment objects taking into account their specific contributions to the overall risk. In order to abide by given risk limits the knowledge of the current overall risk position is an essential prerequisite. Consider for example a trading unit that needs exact and timely information about its risk exposure when deciding on security or option trades.

Finance offers a set of instruments to calculate risk measures, some of them seeming to be promising with regard to grid computing concepts. In this paper we focus on the 'variance-covariance approach'. When measuring portfolio risk it is not sufficient to take into account the variances of the investment objects' periodical returns alone. Instead, also correlation effects that exist when returns are not perfectly positively correlated have to be considered which can be achieved using covariance matrices.

II.3.1.2 The Variance-Covariance Approach

We can represent risky investment objects by stochastic variables. Typically historical data are used to derive a distribution for a stochastic variable. It is then possible to determine the actual risk position of the investment objects or to extrapolate from history to the future for the support of investment decisions. For two stochastic variables, X_i and X_j, we define (with $E[X_i]$ and $E[X_j]$ denoting the expected values of the variables) the covariance as

$$Cov\left[X_i, X_j\right] = E\left[(X_i - E[X_i])\left(X_j - E[X_j]\right)\right]$$

Writing Cov_{ij} instead of $Cov\left[X_i, X_j\right]$ and σ_i^2 short for the variance of X_i we can determine the overall risk of a portfolio, σ_P^2, consisting out of k investment objects (numbered from 1 to k) as

$$\sigma_P^2 = \sum_{i=1}^{k} \sigma_i^2 + \sum_{i=1}^{k} \sum_{j=1;j\neq i}^{k} Cov_{ij} = \sum_{i=1}^{k} \sum_{j=1}^{k} Cov_{ij}$$

The instrument of covariance matrices is not restricted to the area of security portfolios. Instead all investments of an enterprise, like credit decisions in banks or even customers or projects can be seen as components of the enterprise's overall investment portfolio, having a return and a variance.

II.3.1.3 Characteristics of the Calculation of a Covariance Matrix

The covariance of two investment objects is not previously known and additionally changing over time (heteroscedasticity). Therefore covariances have to be empirically estimated by analyzing historical data. Using the basic approach for estimating covariances, the covariance matrix can also be calculated by

$$Cov_{ij} = E\left[X_i \cdot X_j\right] - E[X_i] \cdot E\left[X_j\right].$$

While the expected values $E[X_i]$ only have to be calculated once for each investment object, the expected value $E[X_i \cdot X_j]$ has to be calculated for each pairwise combination of investment objects. This shows that while the calculation of a single covariance is not very resource intensive, the calculation of a whole covariance matrix is are more complex problem. Although the equality $Cov_{ij} = Cov_{ji}$ makes the matrix symmetric, for k investment objects the total number of covariance calculations is still given by $k(k+1)/2$.

II.3.2 Modelling Approach

An algorithm for computing covariances in parallel must tackle questions like which grid resources to allocate to the calculation of certain parts of the covariance matrix and how to distribute the input data necessary to perform these calculations. Both questions are inseparably intertwined since calculating part of the matrix requires only part of the input data and unnecessary transportation of input data should be avoided. These tasks are usually done by a grid middleware. However, the proposed problem requires too much domain knowledge for a standard grid middleware to distribute it efficiently.

The grid infrastructure considered for covariance calculation consists of n computing nodes and is depicted by an undirected and connected graph. Its vertexes model the nodes while the edges represent physical connections between the nodes. Loops and multiple edges connecting two vertexes are not allowed in our model.

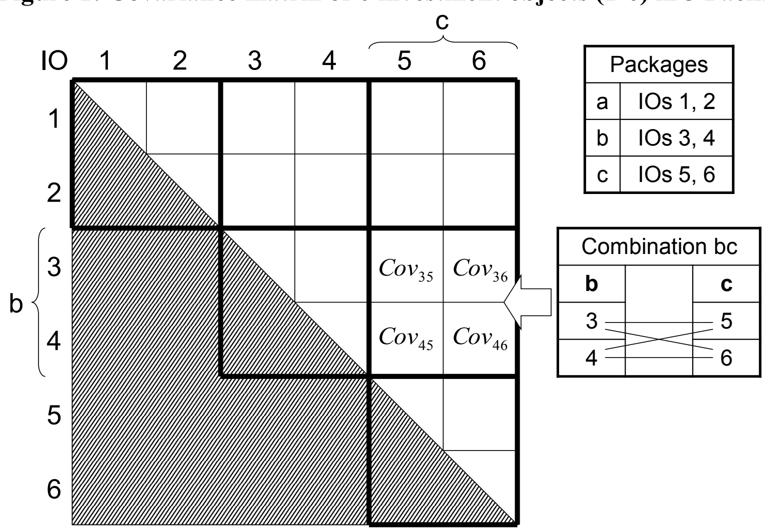
Advancing towards the complexity of grid-based risk management calculations we restrict our considerations on *communication complexity* i.e. the amount of data (we will concretize the unit for measuring the 'amount of data' in the following section) that has to be exchanged between the nodes of the grid. For a *complexity class* we use the common asymptotic notation with Ω, O or Θ denoting the lower, upper or tight bound for the complexity class respectively. It is not surprising that complexity varies for different grid network *topologies*. Therefore, we propose calculation algorithms for a set of standard topologies to find their complexity.

II.3.2.1 Preliminaries

From here on we will use the term *investment object* synonymously with the investment objects' historic quotations. We assume the number of investment objects, k, and the number of nodes in the grid network, n, to be 'sufficiently' large. When necessary to ease our calculations we consider all variables $\epsilon \mathbb{R}$ and omit small constant addends as it is common practice in complexity theory. For simplicity and to avoid case differentiations without relevance we implicitly assume an odd or even number of nodes n as needed.

We refer to the investment objects for which one node holds the historic quotations as to the node's *package*. The number of packages then equals the number of nodes, n. We define a *message* as the transmission of one package over one edge and the *distance* between two nodes as the number of edges/messages to get from one node to the other. We can then measure complexity by the number of messages necessary to calculate one covariance matrix (this connection obviously only holds if all messages are of the same size or at least order of size which is assured for our model by assumption 3). $C^{[topology]}$ denotes the complexity for a specific topology while C denotes the complexity for an arbitrary topology. We use the terms 'complexity', 'amount of communication' and 'number of messages' synonymously.

Figure 1: Covariance matrix of 6 investment objects (1-6) in 3 Packages (a-c)



The nodes holding a certain package are denoted by uppercase Latin letters $A, B, C, ...$, the packages with the corresponding lowercase Latin letters $a, b, c, ...$. We identify investment objects in our examples with numbers $1,2,3, ...$. Figure 1 shows an example of a covariance matrix.

On the package level we constitute a *package combination* out of two packages by calculating the pairwise covariances of each investment object in the one package with each in the other. For our complexity considerations it is then sufficient to analyze the package combinations rather than the covariances themselves. By building all package combinations we get one complete covariance matrix.

Assumption 1 *Covariances are calculated for a given and fixed number* $k \gg 1$ *of investment objects. The necessary input data is available in the form of historic quotations.*

Assumption 2 *The grid infrastructure consists out of* n *nodes with* $1 \ll n \leq k$*, each with the same constant computing capacity available for covariance calculations.*

Assumption 3 *The investment objects are distributed equally over the network, i.e. each node holds a package of the same size* k/n*.*

Assumption 4 *All nodes are used equally if every node calculates the same number of package combinations.*

This is obviously simplifying matters but without significant influence on the results. By concentrating on package combinations, we measure complexity depending on the number of nodes. Intuitively, it might be more interesting to consider the complexity depending on the number of investment objects k. We implicitly assume proportional growth of the number of nodes (e.g. workstations) and the number of investment objects, as both values somehow represent a company's size. Therefore, there exists a constant package size k/n which can be used to transform the complexities. The complexity class is not affected by this transformation.

II.3.2.2 Similar Problems and Other Publications

There are basically two areas that cover parts of the given problem: There is research on algorithms and routing on parallel infrastructures and there is research on grid computing covering especially questions of resource allocation.

In the relevant literature for grid networks one can find many articles addressing allocation problems for certain grid applications. For example Buyya et al. (2002) give an overview about existing systems and their underlying economic models. In particular auctions as described e.g. in Wolski et al. (2001) or agent technologies as proposed in Foster et al. (2004) are used to automatically allocate the available resources efficiently. However these publications try to find general ways to prioritize applications without analyzing a specific application domain whereas this paper analyzes a specific problem's complexity. Yu and Buyya (2005) present a taxonomy of workflow management systems for grid computing. While neither of these systems specifically addresses our problem, they can be used for implementing our resulting algorithms.

Publications in the area of parallel algorithms cover parts of our problem in very specific ways. Algorithms for calculating convolutions could be adapted to calculate covariances as it is necessary for both problems to build pairwise combinations of the input data. Such algorithms have been intensively discussed, most times with the application of integer multiplication, e.g. in Atrubin (1965), and Cappello and Steiglitz (1983). The question of routing packets through a network of processors has already been analyzed e.g. by Valiant and Brebner (1981) in a general way, by Krizanc et al. (1988) for mesh-connected arrays and by Leighton (1990) for greedy algorithms on arrays. Leighton (1992) gives an overview on more algorithms for parallel architectures. All of these algorithms were created for specifically designed or at least prior known topologies. A grid network is by nature inhomogenous, so these algorithms are not directly applicable. In contrast, our model identifies the complexity for communication in an arbitrary and possibly heterogeneous network infrastructure.

There are also several articles dealing with questions of data distribution in distributed systems in general and independent from a given problem. Basically two problems seem to be similar to a distributed calculation of a covariance matrix: The 'all-pairs shortest path' problem (APSP) and the 'optimal broadcast' problem. An APSP algorithm finds the shortest paths between every distinct pair of nodes in a graph (for recent research on APSP algorithms refer to Demetrescu and Italiano (2004)). This might come useful for our problem as all pairs of packages have to be brought together. Nevertheless, this perception disregards the fact that nodes can store and forward information. To distribute a package to several nodes it actually might be more efficient to accept a longer path but at the same time to reach more nodes on the way. According to assumption all nodes have to receive sufficient data to calculate the same number of combinations but an APSP algorithm does not give us any information where to calculate what combinations.

A broadcast algorithm distributes data available at a single node to all the other nodes efficiently. According to Awerbuch et al., 1990; Awerbuch and Schulman, 1997; Awerbuch et al., 1998 and Tiwari, 1987 the complexity for a single node distributing its data to all other nodes in an arbitrary network structure with n nodes is $\Theta(n)$. For the distributed calculation of a covariance matrix n nodes have to distribute their data to several others so a tight bound of $n \cdot \Theta(n) = \Theta(n^2)$ could be expected for our problem.

Buhl et al. (2009) quantify the utility of grid-based covariance calculations for risk management. As there are, to the best of our knowledge, no further articles covering the complexity of grid-based covariance calculation, we provide the first paper in this research area.

II.3.3 Complexity for Different Topologies

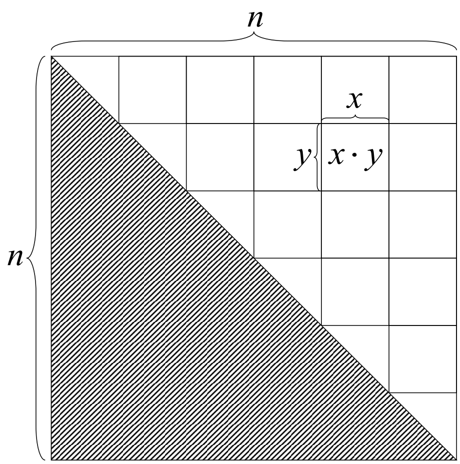
We will examine several idealized standard grid network topologies and propose different algorithms for distributed covariance calculations in order to analyze the complexity. It is already pointed out that we would expect upper and lower bounds of $O(n^2)$ and $\Omega(n^2)$, respectively, adapting the models of Awerbuch et al. (1990), Awerbuch and Schulman (1997), Awerbuch et al. (1998), and Tiwari (1987).

Conjecture 1 *The lower bound for the complexity class in the distributed computation scenario of a covariance matrix in an arbitrary network structure is* $\Omega(n^2)$*, the corresponding upper bound is* $O(n^2)$.

II.3.3.1 Fully Connected Topology

In the fully connected topology every node has direct connections to all other nodes. Every single package can be transmitted from one node to an arbitrary other by one message. Thus minimal communication is accomplished when sending every node exactly the packages for it to perform its calculations. We use this general idea to establish the theoretically minimal communication necessary.

Figure 2: Schematic covariance matrix



We consider figure 2 that shows a schematic covariance matrix for n packages. As the matrix is symmetric only $n^2/2$ package combinations have to be calculated. Thus, according to assumption 4 each of the n nodes has to accomplish $n/2$ package combinations—in effect one half of the matrix is divided into n areas of equal size $n/2$. We will disregard the fact that every node already holds one package. Receiving $x + y$ distinct packages, a node can then calculate $x \cdot y$ package combinations. Thus, $x + y$ represents the number of packages necessary to accomplish $x \cdot y = n/2$ package combinations. It is immediately obvious that for a given and fixed $x \cdot y$ the term $x + y$ is minimized for $x = y$. Therefore, with $x \cdot y = n/2$ and $x = y$, we can state that the minimal number of messages c_{sq} for one node to calculate its required package combinations (a square out of the matrix) is determined by

$$x^2 = \frac{n}{2} \Leftrightarrow x = \sqrt{\frac{n}{2}} \Rightarrow c_{sq} = x + x = 2\sqrt{\frac{n}{2}} = \sqrt{2n} \tag{1}$$

Nodes computing package combinations at the triangles in the diagonal of the covariance matrix will need comparatively more messages to accomplish their (non-quadratic) share. We will neglect this minor inaccuracy—the additional communication for this purpose will only increase linearly with n—and determine the complexity for n nodes as

$$C^{full} \approx n \cdot c_{sq} = n \cdot \sqrt{2n} = \sqrt{2} \cdot n\sqrt{n} \tag{2}$$

As we were minimizing communication when designing this algorithm, we found the lower bound for a calculation on a fully connected topology to be $C^{full} \in \Omega(n\sqrt{n})$. As the algorithms are only sensitive to n, the upper bound equals the lower bound: $C^{full} \in O(n\sqrt{n})$. As all other topologies can be built from the fully connected topology by omitting edges, the fully connected topology must allow the minimal amount of communication possible. So, at the same time, we found the lower bounds for C in the general case: $C \in \Omega(n\sqrt{n})$.

This implicitly disproves conjecture 3.1, since for our problem an even lower complexity than $\Omega(n^2)$ can be reached. We will now analyze if (and if so how) this minimal complexity can be reached in other topologies.

II.3.3.2 Star Topology

The star topology is characterized by one 'hub' with direct connection to all other nodes. Hence, it has a direct neighborhood of $n-1$ nodes. The neighbors themselves are not directly adjacent to each other. The minimal communication necessary in the star topology for calculating the covariance matrix is very similar to the communication considered in the fully connected topology. The hub can receive all packages with one message each. This results in a complexity of $c_1 = n - 1$. As soon as the packages are collected at the hub each package can be transmitted to any node with one message just as in the fully connected topology ($c_2 = C^{full}$). The hub itself already has all the data it needs: thus according to equation 1 we have $c_3 = -c_{sq}$.

$$\begin{aligned} C^{star} &= c_1 + c_2 + c_3 = n - 1 + n \cdot \sqrt{2n} - \sqrt{2n} \\ &= \sqrt{2} \cdot n\sqrt{n} + n - \sqrt{2n} - 1 \end{aligned} \tag{3}$$

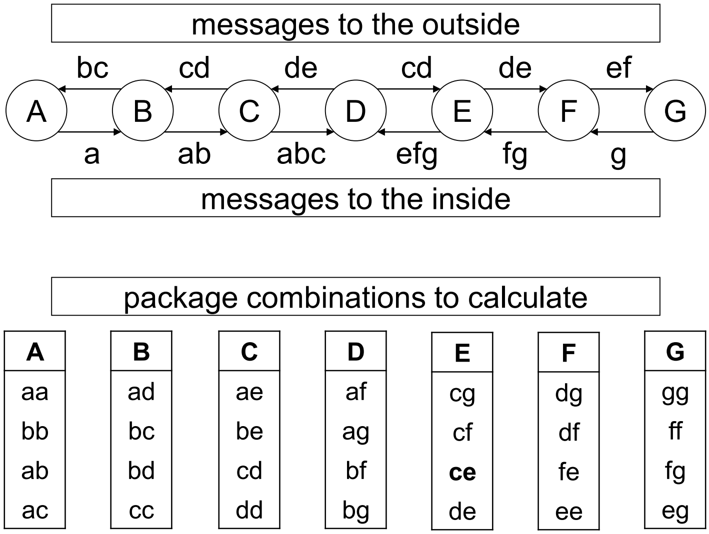
The proposed algorithm for a star topology therefore realizes $O(n\sqrt{n})$ and thus also the theoretical lower bound.

II.3.3.3 Line Topology

In the line topology with n nodes $n-2$ nodes have a degree of 2 and two nodes have a degree of 1. The algorithm for distributed calculation in a line topology is more complicated than in the two previous topologies. Basically there are two reasons for sending a package to a node: The node needs a package to calculate package combinations or the node has to relay the package to a neighbor. We will call the two nodes at the end of the line (each with a degree of 1) the 'outmost' nodes (in the example of figure 3: nodes A and G). For them it makes no sense to forward any packages as the only nodes they could send packages to are the nodes they received the same packages from. Thus the only reason for sending data to the outmost nodes is their calculations to be done. On the other hand, the outmost nodes are not able to calculate all package combinations involving their own packages so they have to pass their packages on to their neighbors. As soon as this is finished one can treat the outmost nodes and their direct neighbors as a 'black box' (here: $A+B$, $F+G$). Again it makes no sense to send these black boxes more packages than they need to perform their calculations. And again they will pass their own data to their neighbors (C and E). As soon as this process is finished, one can again treat the black boxes and their direct neighbors like a larger black box (here: $A+B+C$, $E+F+G$). This can be iterated until the central node is reached from both directions, and then exchanges the data between the two halves of the line.

In order to determine the complexity of this algorithm we will disregard smaller inaccuracies, like there are more computations possible with the available data than with the available computing power or the fact that some combinations may already have been computed at on the other half. Quantifying the communication to the inside, denoted by c_1 is like sending all packages to a central node which has the position $(n+1)/2$ (from both ends of the line).

Figure 3: Communication in a line topology



In the line topology, the communication can be expressed as a sum over the distances between all nodes and the central node. On each side there are $(n-1)/2$ nodes. As the nodes are ordered in a line, when going from the calculating node to the outside of the line, each node's distance to the calculating node increases by one. The overall complexity can thus be described by twice the sum over an increasing control variable i:

$$c_1 = 2\sum_{i=1}^{\frac{n-1}{2}} i = \frac{n^2-1}{4} \tag{4}$$

Quantifying the communication to the outside, denoted by c_2, is more complex. The first question arising is the number of packages to send into a black box. We define ψ as the iteration of our black box, with the first iteration ($\psi = 1$) describing a black box containing the outmost node, the second one describing a black box containing the outmost node and its direct neighbor and so on. We also define $n_b(\psi) = \psi$ as the number of nodes in a black box, $p_b(\psi)$ as the number of combinations to be calculated in the black box, $q_b(\psi)$ as the minimally necessary number of packages to do these calculations (compare to equation 1), and finally $c_b(\psi)$ as the necessary number of messages that have to be sent to the black box.

$$p_b(\psi) = \frac{n+1}{2} \cdot n_b(\psi) = \frac{n+1}{2} \cdot \psi$$

$$q_b(\psi) \approx \sqrt{2p_b(\psi)} = \sqrt{(n+1) \cdot \psi}$$

$$c_b(\psi) = q_b(\psi) - n_b(\psi) = \sqrt{(n+1)\psi} - \psi \approx \sqrt{n \cdot \psi} - \psi$$

The parameter c_2 can be quantified by summing up the $c_b(\psi)$ for all the black boxes. As mentioned above there are black boxes for $\psi = 1, \ldots, (n-1)/2$ and as the algorithm is applied from both sides of the line we have to count every black box twice.[23]

$$c_2 \approx 2 \cdot \sum_{\psi=1}^{\frac{n-1}{2}} \left(\sqrt{n \cdot \psi} - \psi\right) = 2 \cdot \left(\sum_{\psi=1}^{\frac{n-1}{2}} \sqrt{n \cdot \psi} - \sum_{\psi=1}^{\frac{n-1}{2}} \psi \right)$$

$$\approx \frac{1}{3}\sqrt{2}n^2 - \frac{4}{3}\sqrt{n} - \frac{n^2-1}{4} \qquad (5)$$

With c_1 and c_2 we can calculate the total number of messages for a line topology.

$$C^{line} = c_1 + c_2 = \frac{n^2-1}{4} + \frac{1}{3}\sqrt{2}n^2 - \frac{4}{3}\sqrt{n} - \frac{n^2-1}{4} = \frac{1}{3}\sqrt{2}n^2 - \frac{4}{3}\sqrt{n} \qquad (6)$$

According to $C^{line} \in O(n^2)$ for our proposed algorithm we have the first topology not allowing a distributed calculation in $O(n\sqrt{n})$.

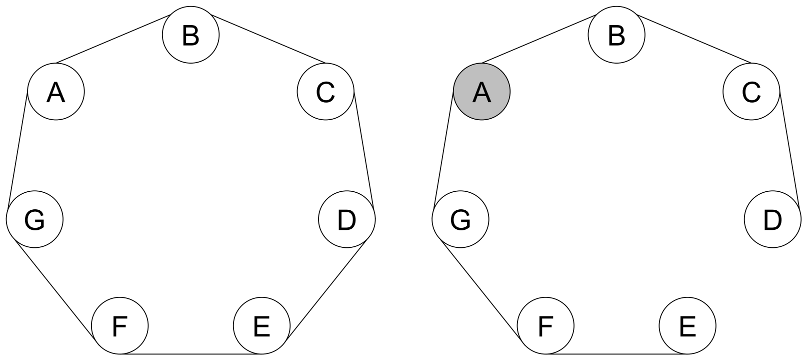
II.3.3.4 Ring Topology

The ring topology is a ring of n nodes with a degree of 2 each. The ring topology is quite similar to the line topology as it only consists out of one more edge, connecting the two ends of the

[23] We approximate the sum by an integral with the same bounds.

line. As we can easily transform a ring into a line, the complexity for a ring topology is surely not higher than the complexity for a line topology.

Figure 4: Example of a ring topology vs. line topology



Having a closer look on the ring topology we can state that the maximal distance between two nodes is $(n-1)/2$. This means that either A's package has to be transported to D, or vice versa or A's and D's packages have both to be transported to B or to C. Either way this causes $(n-1)/2$ messages and for n nodes $n(n-1)/2$ messages. One can easily see, if every node behaves equally, every package combination can be calculated and all nodes and edges are equally utilized. The total number of messages for a ring topology can thus be described by $C^{ring} = n(n-1)/2$ which is $O(n^2)$, too.

II.3.3.5 Tree Topology

We define the tree topology as a network structure whose nodes are arranged in a perfect λ-ary tree with a depth δ. The proposed algorithm is very similar to the line topology's. Its calculation first requires to determine the dependencies of the number of nodes n, the depth of the tree δ and the order of the tree $\lambda > 1$.

$$n = \sum_{i=0}^{\delta} \lambda^i = \frac{\lambda^{\delta+1}}{\lambda - 1} \tag{7}$$

By solving equation 7 to δ we can determine the depth of a tree depending on its number of nodes and its order.

$$\delta = log_\lambda(n \cdot (\lambda - 1) + 1) - 1 \approx log_\lambda(n\lambda) - 1 = log_\lambda(n) \cdot log_\lambda(\lambda) - 1$$
$$\approx log_\lambda(n) \tag{8}$$

The number of messages to the root, c_1, is then given by the tree's depth sum (the sum of all nodes' depths) and thus can be expressed as

$$c_1 = \sum_{i=1}^{\delta} i \cdot \lambda^i = \sum_{j=1}^{\delta}\sum_{i=1}^{\delta} \lambda^i = \cdots = \frac{\delta \cdot \lambda^\delta \cdot \lambda}{\lambda - 1} - \frac{\lambda^\delta \cdot \lambda - \lambda}{(\lambda - 1)^2}$$
$$\approx n \cdot log_\lambda(n) - \frac{n-1}{\lambda - 1} \tag{9}$$

The amount of communication to the other nodes is established by the least number of packages necessary to do all computations there. In order to determine the number of packages, we first need to quantify the number of nodes n_s, in a subtree with its root in depth φ. We have $n_s(\varphi) = \frac{1}{\lambda^\varphi}\sum_{i=\varphi}^{\delta} \lambda^i = \sum_{i=0}^{\delta-\varphi} \lambda^i$. We define, in analogy to the line topology, $p_s(\varphi)$ as the number of calculations to be done in the subtree, $q_s(\varphi)$ as the necessary number of packages for these calculations and $c_s(\varphi)$ as the necessary packages to transfer into the subtree.

$$p_s(\varphi) = \frac{n+1}{2}\sum_{i=0}^{\delta-\varphi} \lambda^i$$

$$q_s(\varphi) = \sqrt{2p_s(\varphi)} = \sqrt{(n+1)\sum_{i=0}^{\delta-\varphi} \lambda^i}$$

$$c_s(\varphi) = q_s(\varphi) - n_s(\varphi) = \sqrt{(n+1)\sum_{i=0}^{\delta-\varphi} \lambda^i - \sum_{i=0}^{\delta-\varphi} \lambda^i} \tag{10}$$

The amount of communication to the outside c_2, is the sum over all c_s for all depths and all subtrees at a specific depth.

$$c_2 = \sum_{\varphi=1}^{\delta-1}\left(\lambda^{\varphi} \cdot \left(\sqrt{(n+1)\sum_{i=0}^{\delta-\varphi}\lambda^{i} - \sum_{i=0}^{\delta-\varphi}\lambda^{i}}\right)\right)$$

$$\approx n^{\frac{3}{2}} \cdot \frac{\sqrt{\lambda}}{\sqrt{\lambda}-1} - n \cdot log_{\lambda}(n) + \frac{n-1}{\lambda-1} - n \cdot \frac{\sqrt{\lambda}}{\sqrt{\lambda}-1} \qquad (11)$$

The overall number of messages necessary to perform a distributed calculation in a perfect tree is constituted as

$$C^{tree} = c_1 + c_2 \approx n^{\frac{3}{2}} \cdot \frac{\sqrt{\lambda}}{\sqrt{\lambda}-1} - n \cdot \frac{\sqrt{\lambda}}{\sqrt{\lambda}-1} \qquad (12)$$

Surprisingly, the perfect tree structure can also realize $O(n\sqrt{n})$ although there is no direct similarity to the algorithms for the fully connected or the star topology.

II.3.4 Generic Complexity Considerations

Table 1: Communication Complexities of the Analyzed Topologies

Topology	Fully Connected	Star Topology	Line Topology	Ring Topology	Tree Topology
Complexity	$O(n\sqrt{n})$	$O(n\sqrt{n})$	$O(n^2)$	$O(n^2)$	$O(n\sqrt{n})$

Assuming optimality for the algorithms proposed in section 3, there is some interesting results worth to be pointed out. There seems to be no clear coherence between complexity and the number of edges relative to the number of nodes. In this section we will try to find a general way to determine the upper and lower bounds of complexity for an arbitrary topology. For enabling our proof we need an assumption regarding the availability of data.

Assumption 5 *Each node receives the packages it needs for its calculations from the nodes with the smallest distances first. Each of these nodes holds at least one required package.*

Obtaining packages from the nodes within the smallest distances first is certainly a good idea for minimizing the complexity. All required packages are available for the first node receiving packages as any package is usable for calculation (no combinations have been calculated yet) and every node holds exactly one package. Taking this finding as a starting point we can make

Proposition 1 *The complexity for the distributed calculation of a covariance matrix in an arbitrary network structure is* $\Omega(n\sqrt{n})$ *and* $\mathrm{O}(n^2)$.

We will prove these upper and lower bounds using the following

Lemma 2 *The marginal number of messages necessary for a single node in an arbitrary topology that receives* i *packages is at least constant and grows at most linearly with each additional* i.

Proof 3 *We define the distance* $d(A,Z)$ *between two (not necessarily distinct) nodes* A *and* Z. *The number of messages necessary for* A *to receive a package from* Z *equals* A*'s distance to* Z. *To minimize communication* A *will receive the packages of the nodes with the smallest distances first. As* A *is part of a connected graph it has at least one direct neighbor* B *with* $d(A,B) = 1$. *In the best case each package is available within a distance of* 1 *and can thus be received with a constant number of messages (exactly* 1 *message). In the worst case every node holds only one usable package and the next reachable node* C *is not a neighbor of* A *but of* B, *so* $d(A,C) = d(A,B) + d(B,C) = d(A,B) + 1 = 2$. *Again, in the worst case, the next reachable node* D *is neither neighbor of* A *or* B *but a neighbor of* C *a.s.o. Generally: In the worst case with* N_i *denoting the* i*th reachable node whose package is to be received at* A: $d(A,N_i) = d(A,N_{i-1}) + 1 = d(A,N_{i-1}) + 2 = \cdots = d(A,N_1) + i - 1 = i$. *As* $d(A,N_1)$ *is constant, the necessary number of messages to receive* i *packages increases in the worst case linearly with each additional* i.

With lemma 4.2 and assumption 5 we can prove proposition 4.1.

Proof 4 *The lower bound of a distributed calculation* $\Omega(C)$ *has already been deduced before as* $C = \sqrt{2} \cdot n\sqrt{n} \Rightarrow C \in \Omega(n\sqrt{n})$. *We can derive the upper bound of a distributed calculation* $\mathrm{O}(C)$ *using almost the same model as for the lower bound. Equation 1 delivers the necessary number of packages for calculating one node's workload (with* $n/2$ *package combinations) as* $\sqrt{2n}$. *According to lemma 4.2 the number of messages required to receive these packages grows at most linearly. Accounting for linear dependency in the worst case we derive an upper bound for the resulting complexity as*

$$C = n \cdot \sum_{i=1}^{\sqrt{2n}} i = n \cdot \frac{\sqrt{2n} \cdot (\sqrt{2n} + 1)}{2} \approx n \cdot \frac{\sqrt{2n} \cdot \sqrt{2n}}{2} = n^2 \tag{13}$$

As this is a worst case scenario we can state that $C \in \mathrm{O}(n^2)$.

II.3.5 Conclusion

In this paper we restricted our analysis to one well-defined problem: the grid-based calculation of covariance matrices. Although covariances are widely used in financial applications, we thereby covered only a small subset of risk management methods and algorithms. Other approaches and applications for grid computing (like e.g. Monte-Carlo simulations which have a high parallelization potential as well) still have to be evaluated.

We considered different grid network topologies and scrutinized suitable algorithms for covariance calculation on these network structures. From there we derived the corresponding complexity classes for a distributed calculation on each topology. The basic algorithms proposed in this paper assume a very regular topology and an omniscient coordinator. Therefore none of them will be directly applicable to a real company's complex infrastructure. Nevertheless, when perceiving the nodes of the topology not as single workstations but as whole subsidiaries of a large enterprise an e.g. line topology is absolutely realistic. Furthermore the resulting complexity classes could in combination with a quantification for the benefits of the calculation be utilized for the determination of an optimal investment in covariance calculations.

For an arbitrary topology these results are as well meaningful: For example when designing a specialized algorithm for a company's network infrastructure one can apply the insights generated by the algorithms proposed (e.g. that it makes sense to send as few packages to the outmost nodes as possible). Furthermore, even if in a company's network each node's effort to receive data grows in a logarithmic fashion (e.g. if each node's degree is greater than 2) one can use $\mathrm{O}(n\sqrt{n})$ as a benchmark for the design of a specific covariance calculation method.

II.3.6 Acknowledgements

A prior version of this research has been presented at the 7th International Conference on Optimization: Techniques and Applications (ICOTA7), Kobe, (Japan), December 2007. Thank you to the audience and especially to Dr. Hackenbroch for very useful feedback.

II.4 References (Chapter II)

Abbas A (2004) Grid Computing: A Practical Guide to Technology and Applications. Charles River Media, Hingham, Massachusetts

Alexander C (1996) The Handbook of Risk Management and Analysis, 1 ed. Wiley, Chichester

Atrubin A (1965) A One-Dimensional Real-Time Iterative Multiplier. IEEE Transactions on Electronic Computers EC-14(3):394-399

Awerbuch B, Cidon I, Kutten S (1990) Optimal Maintenance of Replicated Information. In: Proceedings of the 31st IEEE Symposium on Foundations of Computer Science, St. Louis, Missouri

Awerbuch B, Cidon I, Kutten S, Mansour Y, Peleg D (1998) Optimal Broadcast with Partial Knowledge. SIAM Journal on Computing 28(2):511-524

Awerbuch B, Schulman LJ (1997) The Maintenance of Common Data in a Distributed System. JACM 44(1):86-103

Bachelier L (1900) Thèorie de la spèculation. Annales Scientifiques de l'Ècole Normale Supèrieure 17:21-86

Basel Committee on Banking Supervision (2004) International Convergence of Capital Measurement and Capital Standards. http://www.bis.org/publ/bcbs107.pdf. Access 2009-12/20

Berman F (2005) Grid Computing: Making the Global Infrastructure a Reality. Wiley, Chichester

Bhargava HK, Sundaresan S (2004) Computing as Utility: Managing Availability, Commitment, and Pricing Through Contingent Bid Auctions. JMIS 21(2):201-227

Brownlees CT, Contini S, Di Meo R, Sullo V (2006) Financial Risk Management Via Multi Mode Inference GRID Applications. In: Proceedings of Science, 1st International Workshop on Grid Technology for Financial Modeling and Simulation, Palermo

Buhl HU, Fridgen G, Hackenbroch W (2009) An Economic Analysis of Service-Oriented Infrastructures for Risk/Return Management. In: Newell S, Whitley E, Pouloudi N, Wareham J, Mathiassen L (eds) Proceedings of the 17th European Conference on Information Systems, ECIS, Verona

Buyya R (2002) Grid Computing Info Centre (GRID Infoware). http://www.gridcomputing.com/gridfaq.html. Access 2009-12/27

Buyya R, Abramson D, Giddy J, Stockinger H (2002) Economic Models for Resource Management and Scheduling in Grid Computing. The Journal of Concurrency and Computation: Practice and Experience 14(Special Issue on Grid Computing Environments):1507-1542

Buyya R, Chapin S, DiNucci D (2000) Architectural Models for Resource Management in the Grid. In: Lecture Notes in Computer Science, Proceedings of the First IEEE/ACM International Workshop: Grid Computing — GRID 2000, Bangalore

Cappello PR, Steiglitz K (1983) A VLSI layout for a pipelined Dadda multiplier. ACM Transactions on Computer Systems 1(2):157-174

Chandra A, Goyal P, Shenoy P (2003) Quantifying the Benefits of Resource Multiplexing in On-demand Data Centers. In: Proceedings of the First Workshop on Algorithms and Architectures for Self-Managing Systems, San Diego, California

Copeland TE, Weston JF (1988) Financial Theory and Corporate Policy, 3 ed. Addison-Wesley, West Sussex

Crespo D, Gil J, Gómez A, Mota E (2006) Grid Solution for Market and Counterparty Risk Calculation. Real Life Problems from the point of view of the System Developer. In: Proceedings of Science, 1st International Workshop on Grid Technology for Financial Modeling and Simulation, Palermo

Demetrescu C, Italiano GF (2004) A new Approach to Dynamic All Pairs Shortest Paths. JACM 51(6):968-992

Elton EJ, Gruber MJ (1972) Earnings Estimates and the Accuracy of Expectational Data. Management Science 18(8):B-409-B-424

Engle RF (1982) Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. Econometrica 50(4):987-1007

Faisst U, Buhl HU (2005) Integrated Enterprise Balancing mit integrierten Ertrags- und Risikodatenbanken. Wirtschaftsinformatik 47(6):403-412

Fama EF (1965) The Behavior of Stock-Market Prices. The Journal of Business 38(1):34-105

Fill H, Gericke A, Karagiannis D, Winter R (2007) Modellierung für Integrated Enterprise Balancing. Wirtschaftsinformatik 49(6):419-429

Foster I (2002) What is the Grid? Three Point Checklist. GRID today 1(6)

Foster I, Jennings NR, Kesselman C (2004) Brain Meets Brawn: Why Grid and Agents Need Each Other. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS '04, IEEE Computer Society, New York

Foster I, Kesselman C (2003) The Grid 2: Blueprint for a new Computing Infrastructure, 2 ed. Morgan Kaufmann, San Francisco

Foster I, Kesselman C (1998) The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco

Foster I, Kesselman C, Nick JM, Tuecke S (2002) The Physiology of the Grid - An Open Grid Services Architecture for Distributed Systems Integration. http://www.globus.org/alliance/publications/papers/ogsa.pdf. Access 2009-12/20

Foster I, Kesselman C, Tuecke S (2001) The Anatomy of the Grid - Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications 15(3):220-222

Friedman M (2003) Grid Computing: Accelerating the Search for Revenue and Profit for Financial Markets. Building an Edge (1):143-145

Gericke A, Fill H, Karagiannis D, Winter R (2009) Situational method engineering for governance, risk and compliance information systems. In: Vaishanvi V, Purao S (eds) Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology, DESRIST '09, Philadelphia, paper 24

Hackenbroch W (2007) Integrated Risk/Return Management on Service-Oriented Infrastructures: Financial Applications and their Economic Value: the risk-at-risk approach, 1 ed. Sierke, Göttingen

Hull J, White A (1998) Incorporating Volatility Updating into the historical Simulation Method for Value-at-Risk. Journal of Risk 1(1):5-19

Huther A (2003) Integriertes Chancen- und Risikomanagement - Zur ertrags- und risikoorientierten Steuerung von Real- und Finanzinvestitionen in der Industrieunternehmung. Deutscher Universitäts-Verlag, Wiesbaden

Jackson P, Maude DJ, Perraudin W (1998) Back Capital and Value at Risk. Bank of England working papers 79

Krizanc D, Rajasekaran S, Tsantilas T (1988) Optimal Routing Algorithms for Mesh-Connected Processor Arrays. In: Reif JH (ed) Proceedings of the 3rd Aegean Workshop on Computing: VLSI Algorithms and Architectures, Corfu

Kundisch D, Löhner FM, Rudolph D, Steudner M, Weiss C (2007) Bank Management Using Basel II-Data: Is the Collection, Storage and Evaluation of Data Calculated with Internal Approaches Dispensable? In: Enterprise Risk Management Symposium Monograph, Chicago, Illinois

Kupiec PH (2007) Financial Stability and Basel II. Annals of Finance 3(1):107-130

Leighton FT (1990) Average Case Analysis of Greedy Routing Algorithms on Arrays. In: Proceedings of the Second Annual ACM Symposium on Parallel Algorithms and Architectures, Island of Crete

Leighton FT (1992) Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes. Morgan Kaufmann, San Mateo, California

Lintner J (1965) Security Prices, Risk and Maximal Gains from Diversification. Journal of Finance 20(4):587-615

Longworth D (2004) Grid Lock-in on Route to SOA. http://www.looselycoupled.com/stories/2004/grid-lock-infr0831.html. Access 2009-12/20

Mandelbrot B (1972) Correction of an Error in "The Variation of Certain Speculative Prices". Journal of Business 45(4):542-543

Mandelbrot B (1963) The Variation of Certain Speculative Prices. Journal of Business 36(4):394-419

Markowitz HM (1971) Portfolio Selection: Efficient Diversification of Investments. Yale University Press, New York et al.

Middlemiss J (2004) Gearing up for Grid. http://www.wallstreetandtech.com/showArticle.jhtml?articleID=17603501. Access 2009-12/20

Modigliani F, Miller MH (1958) The Cost of Capital, Corporation Finance, and the Theory of Investment. American Economic Review 48:261-297

Mossin J (1966) Equilibrium in a Capital Asset Market. Econometrica 34:768-783

Nabrzyski J, Schopf JM, Weglarz J (2003) Grid Resource Management: State of the Art and Future Trends. Kluwer Academic, Dordrecht

Regev O, Nisan N (1998) The POPCORN Market - an Online Market for Computational Resources. In: Proceedings on the First International Conference on Information and Computation Economies, Charleston, South Carolina

Ricadela A (2002) Living on the Grid. InformationWeek (893):30-35

Schröck G (2001) Risk Management and Value Creation in Banks. Wiley, Hoboken

Schumacher J, Jaeckel U, Zimmermann F (2006) Grid Services for Derivatives Pricing. In: Proceedings of Science, 1st International Workshop on Grid Technology for Financial Modeling and Simulation, Palermo

Sharpe WF (1964) Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk. Journal of Finance 19(3):425-442

Shiryaev AN (1999) Essentials of Stochastic Finance - Facts, Models, Theory, 1 ed. World Scientific, Singapore

Silva V (2006) Grid Computing for Developers, 1 ed. Charles River Media, Hingham, Massachusetts

Singh M, Huhns MN (2004) Service-Oriented Computing. Semantics, Processes, Agents. Wiley, West Sussex

Smith H, Konsynski B (2004) Grid Computing. MIT Sloan Management Review 46(1):7-9

Spremann K (2003) Portfoliomanagement, 2 ed. Oldenbourg Wissenschaftsverlag GmbH, München

Taylor JW (2005) Generating Volatility Forecasts from Value at Risk Estimates. Management Science 51(5):712-726

Tiwari P (1987) Lower bounds on communication complexity in distributed computer networks. JACM 34(4):921-938

Valiant LG, Brebner GJ (1981) Universal schemes for parallel communication. In: Proceedings of the thirteenth annual ACM symposium on Theory of computing, ACM, Milwaukee, Wisconsin

Wilson TC (1996) Calculating Risk Capital. In: Alexander C (ed) The Handbook of Risk Management and Analysis. Wiley, Chichester, 193-232

Wolski R, Brevik J, Plank JS, Bryan T (2004) Grid Resource Allocation and Control Using Computational Economies. In: Berman F, Fox G, Hey T (eds) Grid Computing - Making the Global Infrastructure a Reality. Wiley, West Sussex, 747-771

Wolski R, Plank JS, Brevik J, Bryan T (2001) Analyzing Market-Based Resource Allocation Strategies for the Computational Grid. International Journal of High Performance Computing Applications 15(3):258-281

Yu J, Buyya R (2005) A Taxonomy of Workflow Management Systems for Grid Computing. Journal of Grid Computing 3(3):171-200

III IT as an Object of Risk/Return Management

This chapter includes only one paper: *"Risk/Cost Valuation of Fixed Price IT Outsourcing in a Portfolio Context"*, as described below.

III.1 Risk/Cost Valuation of Fixed Price IT Outsourcing in a Portfolio Context

Authors:	Fridgen, Gilbert, Müller, Hanna-Vera FIM Research Center Finance & Information Management, Department of Information Systems Engineering & Financial Management, University of Augsburg, Universitätsstraße 12, 86159 Augsburg, Germany, gilbert.fridgen@wiwi.uni-augsburg.de hanna.mueller@wiwi.uni-augsburg.de
In:	Proceedings of the 30th International Conference on Information Systems, ICIS, Phoenix, Arizona: Paper 135. http://aisel.aisnet.org/icis2009/135. Access 2010-01-06.

By optimizing its outsourcing strategy, a company faces the opportunity to lower the overall costs of its IT project portfolio. Without considering risk and diversification effects appropriately, companies make wrong decisions about how much of a project is reasonable to outsource. In this paper, we elaborate a model to identify a project's optimal degree of outsourcing at a fixed price, considering both, costs and risks of software development, as well as diversification effects. We also examine optimal outsourcing degrees in an IT portfolio context. To date, it is common practice to decide on the implementation of projects first and then decide on outsourcing. We provide a model that enables companies to determine an optimal outsourcing strategy which minimizes the total risk adjusted costs of an IT project portfolio by considering the portfolio decision and the selection of outsourcing degrees simultaneously. This model is then evaluated by simulation using real-world data.

III.1.1 Introduction

According to Lacity and Hirschheim (1993) firms pursue outsourcing strategies to reduce costs and mitigate risks associated with their business processes. Increased competition forces companies to deal with the cost cutting that is necessary to stay in business. Therefore, the market for outsourcing services increased significantly over time and is about to outgrow previous prospects (Aspray et al. 2006). IT service providers benefit from this development and become more specialized and competitive (Currie 1997). This provides the opportunity for companies to close more profitable outsourcing deals. Especially software development projects are affected, in consideration of the fact that today software development skills are global commodities (Dutta and Roy 2005; Lacity and Willcocks 2003). It is of particular importance for companies to identify a profitable software development outsourcing strategy, which encompasses not only strategic, but also economic and social perspectives (Lee et al. 2003). For the time being, in the majority of companies, a viable outsourcing strategy is either unknown or difficult to determine, because project evaluation processes are neither specified nor documented. Therefore, many companies struggle with the implementation of an integrative outsourcing strategy and still have difficulties to succeed in the implementation of IT projects. The Standish Group reports that two thirds of the IT projects fail or miss their targets (Standish Group 2006). On the contrary, Sauer et al. (2007) illustrate, when project risks are managed by a capable team, follow reasonable plans and tactics, and are of a manageable size, the outcomes are far better. To meet the desired requirement of making a project manageable, a project partitioning between a company and a service provider can be effective. Through outsourcing, projects can be managed more successfully (Slaughter and Ang 1996). Therefore, to enable a company to implement a profitable outsourcing strategy, we examine the effects of fixed price outsourcing on costs and risks of an IT project portfolio.

In today's IT departments it is common practice to decide on the implementation of individual projects first and then to decide case by case if and to what extent a project shall be outsourced. We illustrate that this causes inferior results compared to a simultaneous selection of projects and respective outsourcing degrees. For this purpose we demonstrate how a company can identify the optimal outsourcing degree of a single project as well as an optimal set of outsourcing degrees for a project portfolio. Moreover, we examine the selection of outsourcing degrees for a previously determined project portfolio and compare the results to an integrated portfolio selection and outsourcing decision. We thus provide a formal-deductive model that enables companies to determine an optimal outsourcing strategy by considering the project portfolio selection and the decision on outsourcing degrees simultaneously. The validity of our results is documented by a simulation based on data gathered in a business context. We point out that there are up to now no scientific papers addressing this special characteristic of outsourcing.

Subsequent to a brief survey of the essential literature, we describe the basic setting and assumptions of our approach. We first analyze a price negotiation between an outsourcing client and a service provider for a given degree of outsourcing. The risk-adjusted costs of a project constitute our objective function. From the objective function of a single project we deduce the one for multiple projects. We identify an optimal degree of outsourcing analytically – both, for a single project, as well as for an optimal vector of outsourcing degrees of a project portfolio. Then, we demonstrate our findings in a two projects example. We evaluate our model through simulations with real-world data. First, after describing the simulation framework, we simulate a fixed multiple projects portfolio and identify the best outsourcing solution. Second, we determine an optimal project portfolio and subsequently identify its best combination of outsourcing degrees. Third, we compare these results with a simultaneously identified best portfolio and its respective outsourcing degrees. Finally, we address practical implications, limitations and prospects of our model.

III.1.2 Literature Overview

IT outsourcing is defined as the decision on relocating an IT department's tasks to a third party vendor, who conducts them and charges a certain fee for the service (Lacity and Hirschheim 1993; Loh and Venkatraman 1992; Apte et al. 1997). The reasons for IT outsourcing are manifold, e.g. excess human and technological resources, focusing on core competencies, and exploitation of global strategic advantages, just to name a few. But the main motive is the cost advantage outsourcing bears, if implemented appropriately (Standish Group 2006; Dibbern et al. 2004; Lacity and Willcocks 1998). To succeed in the implementation, firms need a strategy to manage the costs and risks of outsourcing decisions (Willcocks et al. 1999). In recent years, instead of closing "outsourcing megadeals" selective outsourcing evolves, where companies decide deliberately on their outsourcing activities (Lacity et al. 1996). An integrated view of outsourcing, containing strategic, economic and social aspects, helps firms to realize the anticipated gains (Lee et al. 2003). Aron et al. (2005) coin the term "rightsourcing", which means that a conscious risk and relationship management with multiple outsourcing vendors enables companies to reap benefits. Besides the cost and efficiency benefits, drawbacks have to be taken into account, when deciding on outsourcing. Outsourcing can entail disadvantages like unauthorized knowledge transfer, inflexibility though long term contracts, poor relationship management and accompanying poor loyalty and quality (Bryce and Useem 1998). These drawbacks must be included into the evaluation of outsourcing decisions. The costs and risks of outsourcing need to be assessed carefully. Different methods of estimating development costs are discussed in Boehm et al. (2000). The estimation of the associated risk is equally important. Many articles focus on the qualitative assessment of risk, for example Willcocks et al. (1999) or Aron et al. (2005), whereas few focus on the quantification and computation of risk, like Aubert et al. (1999).

Another research stream relevant for our contribution is the theory on transaction costs of outsourcing. Besides the risky costs of development, transaction costs occur, if a project is outsourced to an IT service provider (Aubert et al. 2004; Lammers 2004). These costs can be split into fixed and variable transaction costs. Fixed transaction costs occur as soon as certain projects or fractions of a project are outsourced, for example costs of negotiation and project initiation (Patel and Subrahmanyam 1982). Variable transaction costs are dependent on the magnitude of the fraction or project outsourced, e.g. costs of communication and control (Dibbern et al. 2006).

Investments in IT increased significantly over time, but the gains of successfully implemented IT projects are required to be managed alongside with the accompanying costs and risks, in order to reap worthwhile benefits. Therefore, firms are trying to establish a comprehensive IT portfolio management, in order to get the most advantageous rate of return (Weill and Aral 2005; Oh et al. 2007). But still, shortfalls cause the failure of numerous IT projects (Standish Group 2006). Therefore, many papers address the issue of how to govern an IT project portfolio. Quantitative approaches on IT portfolio management, e.g. Verhoef (2005), work with economic theory such as the discounted cash flow but mostly omit interdependencies between projects. Some approaches model interdependencies by using Modern Portfolio Theory (MPT) (Santhanam and Kyparisis 1996; Butler et al. 1999). Zimmermann et al. (2008) for example adapt the MPT to propose a decision model for global IT sourcing decisions. They consider the costs of site/project combinations as risky and build a portfolio optimization model.

Like most of the aforementioned articles, our model does not consider the risk of outsourcing in its entirety (e.g. qualitative vs. quantitative risk, risk of costs vs. returns). Moreover, we only consider projects which fit into strategic considerations and passed the analysis of available resources and capabilities. In this model, we focus on one specific aspect of outsourcing. We provide an economic model that delivers relevant insights supporting the design of outsourcing decision processes in today's business.

III.1.3 Model

Our focus is the analysis of a situation where an outsourcing client tries to optimize the software development outsourcing strategy by minimizing the risk adjusted total costs generated by a certain project portfolio. For reasons of simplicity we focus on costs of outsourcing only, as we consider the outsourcing client's cash inflows from a certain project to be independent from whether fractions of the projects are outsourced or not. For this paper, we model outsourcing as a fixed price and thus risk-free alternative for project development that can be used to control IT portfolio risk. Thereby, we define risk as a negative or positive deviation from an expected value (as common in finance). This corresponds to a business setting, where a contract between the outsourcing client and the vendor assures characteristics and price of the service. By outsourcing a fraction of a software development project at a fixed price, the associated risk (according to our definition) can be conveyed to the vendor. By combining internal and external development of all projects in an efficient way, the risk adjusted costs of the IT project portfolio can be lowered to a minimum. To the best of our knowledge there are no further publications regarding this effect, so this is the first contribution to this area.

In the following, each portfolio consists of a limited number of projects, each of which can be only conducted once. We consider two parties, a client as initiator of an IT project, and an IT service provider as possible contractor for partial or entire project development. For each single project, the client has to decide on the fraction that is outsourced to the IT service provider. The size of an outsourced fraction, in the following referred to as outsourcing degree, is our decision variable. We analyze if the appropriate selection of an outsourcing degree, which means an optimal combination of internal and external project development, has effects on the risk adjusted costs of a single project or a project portfolio.

We only consider development activities, which can be outsourced. Essential project phases, which have to be accomplished internally, are not taken into account. For example, we exclude tasks concerning core competencies of the client, which cannot be outsourced, as well as crucial project phases, e.g. requirements analysis. Especially the department which initiated the software request is essentially involved in the development process, at least by participating in the specification of the desired outcomes, like software characteristics concerning functionality and quality (Lacity et al. 1996).

For a better understanding, we provide a rough overview over the influencing parameters below, before we start specifying our assumptions. Since we focus on the costs of outsourcing only, we consider the outsourcing client's cash inflows on a certain project to be constant without considering the modality of development. The service provider's cash inflows are given by a certain reward he obtains for his work performed, in the following referred to as price for the externally developed fraction. In addition to the price, outsourcing a fraction of a project causes transaction costs at the client's side, which we consider risk-free. Table 1 provides a rough overview of the values relevant to the decisions of the respective party.

Table 1: Overview of the Setting

	Outsourcing Client	Service Provider
Risky costs	• Costs of the internally developed fraction of a project	• Costs of the fraction of a project developed on behalf of the client

Risk-free costs	• Price for the externally developed fraction of a project • Fixed and variable transaction costs	• --
Sum	• A project's risk-adjusted costs	• Costs of the fraction of a project developed on behalf of the client
Cash inflow	• Cash inflow of a project	• Price for externally developed fraction of a project

To distinguish the parameters of the two parties, we introduce n as a subscript representing internal, client-related variables and x as a subscript representing external, service provider-related variables. The variable $g = (1, \dots, m)$ is a subscript referencing an arbitrary but definite project, with for example $g = 7$ for project #7. As stated above, the internal costs caused by a certain project are risky. The outsourcing client wants to outsource a fraction of a project to minimize the risk adjusted costs of development. To model this situation, we make the following simplifying assumption 1:

Assumption 1

The costs of an entire project g are $C_{n,g}$ for internal development at the client's responsibility and $C_{x,g}$ for external development at the service provider's responsibility. Both are normally distributed, i.e. $C_{n,g} \sim N(\mu_{n,g}, \sigma_{n,g})$ and $C_{x,g} \sim N(\mu_{x,g}, \sigma_{x,g})$.

To decide under which conditions an outsourcing agreement is advantageous for the parties involved, we have to model the pricing of an outsourced project that, in reality, would be subject to negotiation. The outcome of this price assessment for each single project is determined by the client's and the provider's decision rules, which are specified by their respective risk adjusted costs as described in assumption 2:

Assumption 2

The risk adjusted costs are measured by both parties and follow the general structure $\Phi = \mu - \alpha\sigma^2$ with μ denoting the expected value of the costs, σ denoting its standard deviation. We define $\alpha < 0$ as the parameter of risk aversion. The outsourcing client and the service provider are risk-averse regarding costs. The risk adjusted costs of the outsourcing client shall be minimized.

The risk adjusted costs correspond to a preference function which is developed according to established methods of decision theory and integrates an expected value, its deviation, and the decision maker's risk aversion. A related model has been developed by Freund (1956). It was applied in similar contexts over the last decades, for example by Hanink (1985) and Zimmermann et al. (2008). Since normally distributed random variables and risk-averse decision makers are considered, this preference function and its corresponding utility function are compatible to the Bernoulli principle (Franke and Hax 2004; Bernoulli 1954). The parameter α, $\alpha = -2\hat{\alpha}$, conforms to $\hat{\alpha}$, the Arrow-Pratt characterization of risk aversion (Arrow 1971), but since we focus on costs not on returns, the algebraic sign changes. Here, $\alpha < 0$ indicates risk aversion. The lower the value of α, the more risk-averse is the decision maker.

According to assumption 2, the risk adjusted costs of an entire single project g follow the structure $\Phi = \mu_{n,g} - \alpha_n \sigma_{n,g}^2$ for the outsourcing client and $\Phi = \mu_{x,g} - \alpha_{x,g} \sigma_{x,g}^2$ for the service provider, respectively. For reasons of simplicity and to be able to identify an efficient outsourcing degree, we state the following assumption 3:

Assumption 3

A project is infinitely divisible between internal and external development. Every fraction of a project is perfectly correlated to every other fraction. Equal sized fractions of a project carry the same risk.

In the past, due to interdependencies in development tasks, a project could not be cut into arbitrary pieces, several cohesive parts existed. Due to recent developments in computing concepts, like service oriented architectures, software development becomes more rapid, competitive, transparent and flexible. Formerly, complex and complicated amounts of source code where produced, nowadays distinct modules of software can be developed independently from each other. Therefore, the assumption of divisibility, or at least a convergence to infinite divisibility, is justifiable. For example Zimmermann et al. (2008) make an analogous assumption.

As a consequence of assumption 3 there is a proportional relationship between the volume of a project's fraction and the costs and associated risks, respectively. This implies that the larger a considered fraction of a project, the higher the costs of development and the higher the associated standard deviation. This is obviously simplifying matters, as different phases of software projects naturally carry different risk and costs (Conrow and Shishido 1997). Nevertheless, a differentiation between project phases goes beyond the scope of this paper and is subject to further work in this area.

To identify the optimal degree of outsourcing, we define the decision variable λ_g, $0 \leqq \lambda_g \leqq 1$, as the percentage of a project's costs that refers to external development (at the service provider's responsibility). Therefore, $(1 - \lambda_g)$ is the percentage of a project's costs that refers to internal development (at the outsourcing client responsibility). The outsourcing degree $\lambda_g = 1$

stands for a project that is developed completely externally, $\lambda_g = 0$ for a project that is developed completely internally.

If a fraction of a project is outsourced to an IT service provider, transaction costs occur. These are for example costs of communication and coordination (Aubert et al. 2004). Transaction costs are either dependent on the fractions' size, or become due independently of the magnitude of the outsourced fraction. Therefore, we state the following assumption 4:

Assumption 4

When a project g is outsourced to a service provider with an outsourcing degree $\lambda_g > 0$, risk-free transaction costs $K(\lambda_g)$ occur, consisting of fixed transaction costs F_g and variable transaction costs $\lambda_g f_g$.

The fixed transaction costs are considered through a signum function[24]. The variable transaction costs are composed of the cost factor f_g, multiplied with the volume of the outsourced fraction λ_g. Therefore, the term for the transaction costs follows the structure stated below.

$$K(\lambda_g) = sgn(\lambda_g)F_g + \lambda_g f_g \tag{1}$$

Transaction costs are risk-free and become due as soon as a fraction of a project is outsourced. Besides the transaction costs, the externally developed fraction causes costs to the outsourcing client in terms of a price $P(\lambda_g)$ that the service provider demands from the client for the service offered. The service provider and the client agree on this price, as well as on all specifications of the service, by contract.

Assumption 5

The service's characteristics and quality, as well as a certain price, are contractually assured and carry no risk for the client.

As a consequence of assumptions 1, 3 and 5, the client's expected costs of a project g with an external developed fraction λ_g, have the distribution parameters $\mu_{n,g}(1-\lambda_g)$ and $\sigma_{n,g}(1-$

[24]The signum function implies, that for $\lambda_g = 0$, the term for the fixed transaction costs turns 0. Then, the entire project is developed internally, thus no transaction costs occur. For $\lambda_g > 0$, the term turns 1, i.e. if fractions of the project are outsourced. Then, the full amount of fixed transaction costs becomes due (Courant and John 1965).

λ_g). The service provider's expected costs of a project g have the distribution parameters $\mu_{x,g}\lambda_g$ and $\sigma_{x,g}\lambda_g$, respectively.

The negotiation of the price for the externally developed fraction is, in reality, a process of several bargaining rounds, which are difficult to picture. However, the bargaining positions of the two parties can be modeled by inserting the aforementioned distribution parameters into the valuation equations. The pricing function for an outsourced project fraction is derived in the following section.

III.1.3.1 Price Assessment

We use the individual preferences of the two parties to serve as a valuation criterion. Therefore, the price is assessed on the basis of the risk adjusted costs of the client, on the one hand, and the risk adjusted costs of the service provider, on the other. As can be seen in Table 1, the risk adjusted costs of the client are made up of the internal risk adjusted development costs, the assessed price of the external fraction, and the transaction costs. In contrast, the risk adjusted costs of the service provider are made up of the external risk adjusted development costs, only. Consequently, for each project a price assessment according to the following scheme takes place.

The price $P(\lambda_g)$ for a certain externally developed fraction of a project g ranges between an upper bound $U(\lambda_g)$, determined by the client's willingness to pay, and a lower bound $L(\lambda_g)$, determined by the service provider's minimum asking price. Between these limits, the two parties agree on an assessment outcome.

The client's willingness to pay for the external developed fraction is determined by the risk adjusted costs the development of the external fraction would cause internally. The client determines his maximum price by evaluating the risk adjusted costs which would occur if he develops the entire project by himself. Therefore, the upper bound consists of the costs and risk of the supposed additionally internally developed fraction. The covariance between the costs of the already internally developed fraction $(1-\lambda_g)$ and the supposed additionally internally developed fraction λ_g adjusts the aforementioned risk. Then, the sum of the transaction costs is subtracted. This concludes in the following formula 1:

$$\begin{aligned} U(\lambda_g) &= \mu_{n,g}\,\lambda_g - \alpha_n\left(\left(\sigma_{n,g}\lambda_g\right)^2 + 2Cov\left((1-\lambda_g)C_{n,g}, \lambda_g C_{n,g}\right)\right) - sgn(\lambda_g)F_g \\ &\qquad - \lambda_g f_g \\ \Leftrightarrow U(\lambda_g) &= \mu_{n,g}\lambda_g - \alpha_n\left(2\lambda_g - {\lambda_g}^2\right){\sigma_{n,g}}^2 - sgn(\lambda_g)F_g - \lambda_g f_g. \end{aligned} \tag{2}$$

For a single project, the client is willing to agree on every contract with a price below $U(\lambda_g)$, whereby a preferably low price is aspired. If the price exceeds $U(\lambda_g)$, the client would prefer to develop the entire project internally. If the price is equal to $U(\lambda_g)$, the client is indifferent between internal and external development.

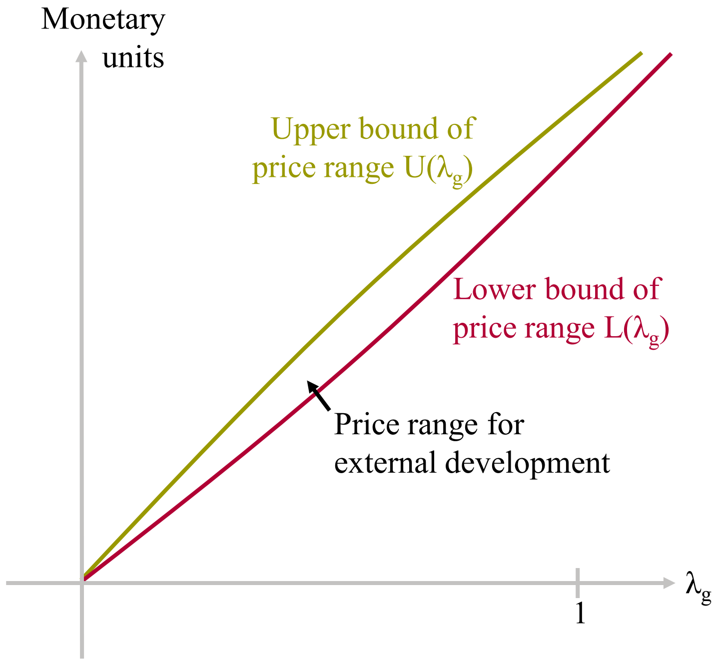
The price's lower bound is determined by the minimum price the service provider must achieve to obtain at least his risk adjusted costs, given the size of the fraction he is going to develop. The specific risk adjusted costs of the service provider are the following.

$$L(\lambda_g) = \mu_{x,g}\lambda_g - \alpha_{x,g}(\sigma_{x,g}\lambda_g)^2 \qquad (3)$$

For a single project, the service provider is willing to agree upon every contract with a price above $L(\lambda_g)$, whereby a preferably high price is aspired. If the price falls below $L(\lambda_g)$, the service provider is not willing to enter the commitment. If the price is equal to $L(\lambda_g)$, the service provider is indifferent whether to close the contract or not.

Since we consider risk averse decision makers, the parameter α is negative. Therefore, $U(\lambda_g)$ is positive as long as fixed transaction costs do not overweigh the advantages of outsourcing and $L(\lambda_g)$ is always positive. If an agreement interval between the two boarders exists, an outsourcing decision is favorable and a room to negotiate can be shared among the involved parties. This is the case only if $\exists\lambda_g$ with $U(\lambda_g) \geqq L(\lambda_g)$. Figure 1 shows the upper and lower bounds and the resulting agreement interval (price range).

Figure 1: Price Range for External Development



Prior research offers different schemes of partitioning agreement intervals (Krapp and Wotschofsky 2004). This, however, goes beyond the scope of our paper. We present a very generic model that can be adapted to map different approaches. Therefore, we introduce the parameter γ_g. $\gamma_g \in]0;1[$ indicates a specific pricing interval share of a party. An agreement with $\gamma_g = 0$ would indicate an outcome at the lower bound, which would be favored by the client, whereas for a single project the service provider would be indifferent between closing the contract or not. An agreement with $\gamma_g = 1$ would indicate an outcome at the upper bound, which would be favored by the service provider, whereas for a single project the client would be indifferent between outsourcing and internal development of the specific project's fraction. These solutions are for the sole benefit of one party and thus not realistic. Therefore, we only consider $0 < \gamma_g < 1$.

Thus, the price $P(\lambda_g)$ of each externally developed fraction is determined by the following formula.

$$\begin{aligned} P(\lambda_g) &= \gamma_g \cdot U(\lambda_g) + (1-\gamma_g) \cdot L(\lambda_g) \\ &= \gamma_g(\mu_{n,g}\lambda_g - \alpha_n(2\lambda_g - \lambda_g{}^2)\sigma_{n,g}{}^2 - sgn(\lambda_g)F_g - \lambda_g f_g) \end{aligned} \tag{4}$$

$$+(1-\gamma_g)\left(\mu_{x,g}\lambda_g-\alpha_{x,g}\left(\sigma_{x,g}\lambda_g\right)^2\right)$$

$P(\lambda_g)$ depends on the existence of an agreement interval, therefore it is only defined for $\{\lambda_g | U(\lambda_g) \geqq L(\lambda_g)\}$. For reasons of simplicity and to avoid case differentiations in the following we presume that $P(\lambda_g)$ is defined for all outsourcing degrees[25] $0 \leqq \lambda_g \leqq 1$.

III.1.3.2 Derivation of the Objective Function

The client's risk adjusted costs of development constitute the objective function which is to be minimized by choosing an optimal λ_g. They consist of the risky internal development costs and risk-free terms for transaction costs and the assessed price. The term of the transaction costs follows equation (1). The price term follows equation (4). We regard these functions and all variables besides λ_g as exogenously given, and integrate them into the objective function.

Thus, the costs of single project g are represented by a normally distributed random variable with distribution parameters

$$\begin{aligned} M(\lambda_g) &= \mu_{n,g}(1-\lambda_g)+P(\lambda_g)+K(\lambda_g) \\ &= \mu_{n,g}(1-\lambda_g)+\gamma_g(\mu_{n,g}\lambda_g-\alpha_n\,(2\lambda_g-\lambda_g{}^2)\sigma_{n,g}{}^2-sgn(\lambda_g)F_g-\lambda_g f_g) \\ &\quad +(1-\gamma_g)\left(\mu_{x,g}\lambda_g-\alpha_{x,g}\,\left(\sigma_{x,g}\lambda_g\right)^2\right)+sgn(\lambda_g)F_g+\lambda_g f_g \end{aligned} \tag{5}$$

as expected value, and

$$S(\lambda_g)=\sqrt{(\sigma_{n,g}(1-\lambda_g))^2} \tag{6}$$

as standard deviation. Therefore, with respect to assumption 2, a single project's risk adjusted costs are modeled according to the following structure.

---

[25]Special calculational cases might occur in boundary areas of the upper and lower bound, thus a pricing interval might not exist. Since the market for specialized and competitive service providers is flourishing, we suppose that in reality an outsourcing vendor willing to provide the service can be found for any outsourcing degree. On this condition, a positive price interval exists for all relevant cases.

$$\Phi(\lambda_g) = M(\lambda_g) - \alpha_n \left(S(\lambda_g)\right)^2$$
$$= \mu_{n,g}(1-\lambda_g) + \gamma_g\left(\mu_{n,g}\lambda_g - \alpha_n\left(2\lambda_g - {\lambda_g}^2\right){\sigma_{n,g}}^2 - sgn(\lambda_g)F_g - \lambda_g f_g\right) \quad (7)$$
$$+\left(1-\gamma_g\right)\left(\mu_{x,g}\lambda_g - \alpha_{x,g}\left(\sigma_{x,g}\lambda_g\right)^2\right) + sgn(\lambda_g)F_g + \lambda_g f_g - \alpha_n(\sigma_{n,g}(1-\lambda_g))^2$$

With multiple projects, the expected costs of the projects, the prices for external development, and the transaction costs are added up to the total portfolio costs. The indices i and j are referencing all projects $(1, \dots, \mathrm{m})$ considered in the portfolio. The vector $\vec{\lambda} = (\lambda_1, \dots, \lambda_m)$ contains the outsourcing degrees of all projects. Therefore, expected total portfolio costs are

$$M(\vec{\lambda}) = \sum_{i=1}^{m} M(\lambda_i) = \sum_{i=1}^{m} \mu_{n,i}(1-\lambda_i) + \sum_{i=1}^{m} P(\lambda_i) + \sum_{i=1}^{m} K(\lambda_i). \quad (8)$$

However, there are dependencies between the different projects' costs that are accounted for using correlation coefficients $\rho_{i,j}$, $0 \leqq \rho_{i,j} \leqq 1$. Please note that we only consider positively correlated projects as a negative correlation of projects is uncommon in reality (this would mean that good performance of one project systematically causes bad performance of another and vice versa). The standard deviation of the total portfolio costs including the diversification effects is

$$S(\vec{\lambda}) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{m} \sigma_{n,i}(1-\lambda_i)\sigma_{n,j}(1-\lambda_j)\rho_{i,j}.} \quad (9)$$

To simplify matters, we do not include diversification effects in the *pricing term* – neither for the outsourcing client, nor the service provider – as this might lead to complex variations of the upper and lower bound. Due to these effects the service provider might be able to offer a lower price and the outsourcing client might be willing to pay a higher price. Thus, the price range would be broader than stated above. Besides, diversification effects in the pricing term would raise questions about the sequence of project investments. Each project would change the portfolio which serves as evaluation basis for the subsequent price negotiation. These effects would amplify the complexity of our model. Since the characteristics of the pricing term would not change severely due to the inclusion of diversification effects, and since it would have low impact on the main results of this paper, we neglect these effects that however might be subject to further research.

Instead, we take the pricing term as given and focus on the client's point of view. Therefore, the price equation for a portfolio of projects is

$$\begin{aligned} P(\vec{\lambda}) &= \sum_{i=1}^{m} P(\lambda_i) \\ &= \sum_{i=1}^{m} \Big(\gamma_i\big(\mu_{n,i}\lambda_i - \alpha_n\big(2\lambda_i - {\lambda_i}^2\big)\sigma_{n,i}{}^2 - sgn(\lambda_i)F_i - \lambda_i f_i\big) \\ &\qquad + (1-\gamma_i)\Big(\mu_{x,i}\lambda_i - \alpha_{x,i}\big(\sigma_{x,i}\lambda_i\big)^2\Big)\Big). \end{aligned} \tag{10}$$

Consequently, the risk adjusted total portfolio costs are modeled according to the following structure.

$$\begin{aligned} \Phi(\vec{\lambda}) = \sum_{i=1}^{m} \mu_{n,i}(1-\lambda_i) \\ &+ \sum_{i=1}^{m} \Big(\gamma_i\big(\mu_{n,i}\lambda_i - \alpha_n\big(2\lambda_i - {\lambda_i}^2\big)\sigma_{n,i}{}^2 - sgn(\lambda_i)F_i - \lambda_i f_i\big) \\ &+ (1-\gamma_i)\Big(\mu_{x,i}\lambda_i - \alpha_{x,i}\big(\sigma_{x,i}\lambda_i\big)^2\Big)\Big) + \sum_{i=1}^{m}(sgn(\lambda_i)F_i + \lambda_i f_i) \\ &- \alpha_n \sum_{i=1}^{m}\sum_{j=1}^{m} \sigma_{n,i}(1-\lambda_i)\sigma_{n,j}(1-\lambda_j)\rho_{i,j} \end{aligned} \tag{11}$$

Before exploring a situation where a company has to determine $\overrightarrow{\lambda^*} = (\lambda_1^*, \ldots, \lambda_m^*)$ for multiple projects in a portfolio view, we initially focus on the determination of a single project's optimal outsourcing degree. Considering a single project's internal development costs and the price paid for an externally developed fraction, the client faces many different internal/external development compositions, i.e. different values for λ_g, to get to the desired outcome of implementing a certain project. Therefore, to provide a basis for the following extensions of our model, a first research question can be posed: *Which degree of outsourcing should a client choose for a single project to minimize the risk adjusted costs of a software development project?*

III.1.3.3 Outsourcing of a Single Project

In this section the client considers only one software development project g. As an equation containing a signum function is not continuously differentiable, we address the fixed transaction costs later on, when we simulate the results for multiple projects. For now, to be able to solve the optimization problem analytically, we set the fixed transaction costs $F_g = 0$.

The formation of the objective function to be minimized for a single project follows the scheme pictured in the previous section. In the first step we neglect that $\lambda_g \in [0,1]$ to obtain a possible minimal solution $\hat{\lambda}_g$. To fulfill the first order condition for optimality, we set the first derivative with respect to λ_g equal to 0.

$$\frac{\partial \Phi(\lambda_g)}{\partial \lambda_g} = (1-\gamma_g)\left(f_g - \mu_{n,g} + \mu_{x,g} - 2\left(\alpha_n(-1+\lambda_g)\sigma_{n,g}^2 + \alpha_{x,g}\lambda_g\sigma_{x,g}^2\right)\right) = 0 \tag{12}$$

We solve the equation for λ_g and get

$$\hat{\lambda}_g = \frac{f_g - \mu_{n,g} + \mu_{x,g} + 2\alpha_n\sigma_{n,g}^2}{2\alpha_n\sigma_{n,g}^2 + 2\alpha_{x,g}\sigma_{x,g}^2}. \tag{13}$$

To fulfill the second order condition, the second derivative with respect to λ_g has to be larger than zero.

$$\frac{\partial^2 \Phi(\lambda_g)}{\partial \lambda_g^2} = 2(-1+\gamma_g)\left(\alpha_n\sigma_{n,g}^2 + \alpha_{x,g}\sigma_{x,g}^2\right) > 0 \tag{14}$$

To obtain a global minimum neglecting that $\lambda_g \in [0,1]$, the first and second order conditions have to be fulfilled. With all exogenous parameters in the previously defined domains, the second order condition (formula 14) is always true. Accounting for $\lambda_g \in [0,1]$, the parameter $\lambda_g^* = \hat{\lambda}_g$ constitutes an optimum, only if $\hat{\lambda}_g \in [0,1]$. If $\hat{\lambda}_g$ takes values below zero or larger than 1, we choose the optimal solutions $\lambda_g^* = 0$ for any $\hat{\lambda}_g < 0$, and $\lambda_g^* = 1$ for any $\hat{\lambda}_g > 1$.

In equation (13) the denominator, consisting of the combined risks of internal and external development, is always negative, since the parameter for risk aversion α is below zero. Regarding the numerator, the algebraic sign can change with a shift in costs. It shows the variable transaction costs, the spread between internal and external development costs, and the risk associated with internal development, adjusted by the parameter for risk aversion. This means that costs caused by outsourcing are compared to costs caused by internal development. If the costs of outsourcing overweigh the costs of internal development, the numerator turns positive. Hence, $\hat{\lambda}_g \leqq 0 \Rightarrow \lambda_g^* = 0$, which means that no outsourcing occurs. Else, if the costs of internal development overweigh the costs of outsourcing, the numerator turns negative. So, $\hat{\lambda}_g$ and λ_g^* turn larger than zero which means that outsourcing occurs. The magnitude of the determined optimal outsourcing degree depends on the risk adjusted cost advantage of either development option.

Figure 2: Optimal Outsourcing Degree of a Single Project

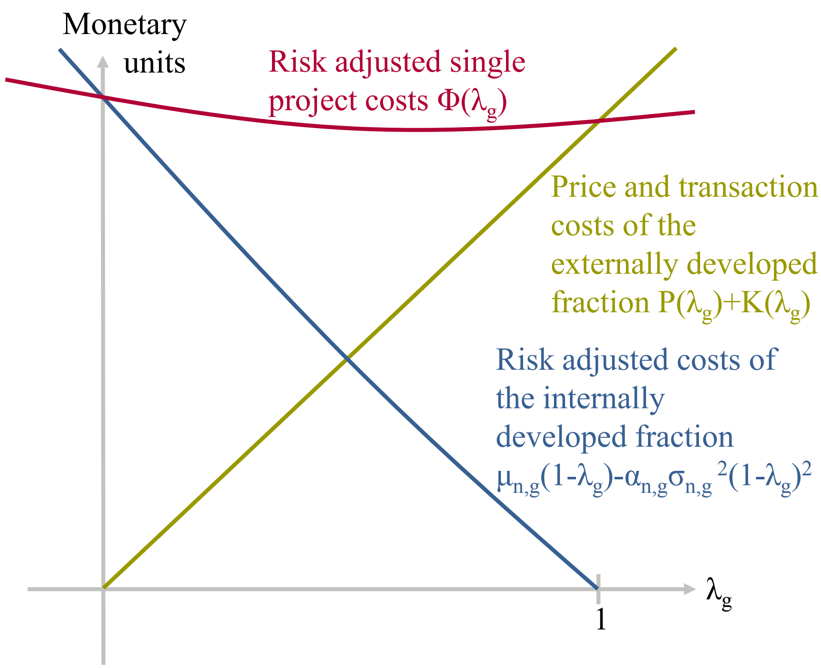


Figure 2 shows the decreasing risk adjusted internal development costs and the increasing price for the outsourced fraction subject to an increasing outsourcing degree. The overall risk adjusted costs of a single project are shown as aggregation of the two slopes, in the upper part of the chart. There, the optimal outsourcing degree can be identified at the curves minimum.

The optimal outsourcing degree is determined by the minimal risk adjusted costs. In the following, we expand our model to identify optimal outsourcing degrees of projects within a portfolio. Since companies conduct multiple projects simultaneously, we capture a multiple projects portfolio in the following section. Therefore, a second research question can be posed:

Which degrees of outsourcing should a client choose for a given multiple projects portfolio to minimize the risk adjusted total portfolio costs?

III.1.3.4 Outsourcing of a Multiple Projects Portfolio

The client considers multiple software development projects in a portfolio. In the following, we want to determine the optimal outsourcing degrees of projects analytically within a portfolio view. As stated in the single project scenario, for reasons of simplicity, fixed transaction costs are not considered. Besides that, the objective function is still built according to the principles stated above.

We now face a multivariate optimization problem with a vector of decision variables $\vec{\lambda} = (\lambda_1, \dots, \lambda_m)$. Again, in the first step we neglect that $\vec{\lambda} \in [0,1]^m$ to obtain the vector $\vec{\hat{\lambda}}$ that contains a possible minimal solution. The first order condition for optimality with respect to every, arbitrary but definite λ_g with $g = (1, \dots, m)$ follows the structure

$$\frac{\partial \Phi(\vec{\lambda})}{\partial \lambda_g} = f_g - f_g\gamma_g + (-1+\gamma_g)\mu_{n,g} + (1-\gamma_g)\mu_{x,g}$$
$$+2\left(\alpha_n\sigma_{n,g}\left((-1+\lambda_g)\gamma_g\sigma_{n,g} - \sum_{j=1}^{m}(-1+\lambda_j)\sigma_{n,j}\rho_{g,j}\right) + \alpha_{x,g}(-1+\gamma_g)\lambda_g\sigma_{x,g}^2\right) \tag{15}$$
$$= 0.$$

Solving this equation[26] for every λ_g we get $\vec{\hat{\lambda}}$. To analyze the curvature, we have to build a Hessian matrix, consisting of all second order partial derivatives of the objective function. Differentiating twice with respect to any λ_g, the second order partial derivatives follow the structure

$$\frac{\partial^2 \Phi(\vec{\lambda})}{\partial \lambda_g^2} = 2(\alpha_n(\gamma_g - \rho_{g,g})\sigma_{n,g}^2 + \alpha_{x,g}(-1+\gamma_g)\sigma_{x,g}^2. \tag{16}$$

---

[26]The equation is obviously difficult to solve for λ_g in general. See equations (19) and (20) for an example on two projects.

They form the main diagonal of the Hessian matrix. The second order partial derivatives of the objective function with respect to any $\lambda_g \lambda_h$, with h as subscript referencing another arbitrary but definite project and $g \neq h$, follow the structure

$$\frac{\partial^2 \Phi(\vec{\lambda})}{\partial \lambda_g \lambda_h} = -2\alpha_n \sigma_{n,g} \sigma_{n,h} \rho_{g,h}. \tag{17}$$

Apart from the main diagonal, they form the lower and upper triangular matrix of the Hessian matrix, which is built according to the following scheme.

$$H = \begin{pmatrix} \frac{\partial^2 \Phi(\vec{\lambda})}{\partial \lambda_1^2} & \cdots & \frac{\partial^2 \Phi(\vec{\lambda})}{\partial \lambda_1 \lambda_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \Phi(\vec{\lambda})}{\partial \lambda_m \lambda_1} & \cdots & \frac{\partial^2 \Phi(\vec{\lambda})}{\partial \lambda_m^2} \end{pmatrix} \tag{18}$$

To obtain a global minimum neglecting that $\overrightarrow{\lambda} \in [0,1]^m$, the first and second order conditions have to be fulfilled. The second order condition demands that the Hessian matrix has to be positive definite, which is always true with all exogenous parameters in the previously defined domains, since $\vec{\lambda}^T H \vec{\lambda} > 0$ for any $\vec{\lambda} \neq \circ$. Accounting for $\overrightarrow{\lambda} \in [0,1]^m$, the vector $\overrightarrow{\lambda^*} = \overrightarrow{\hat{\lambda}}$ constitutes an optimum, if $\overrightarrow{\hat{\lambda}} \in [0,1]^m$. If any element of $\overrightarrow{\hat{\lambda}}$, e.g. $\hat{\lambda}_g$, takes values below zero or larger than 1 the optimal solution is more complex to determine. On independent examination – as stated in the single project view – the solutions $\lambda_g^* = 0$, or $\lambda_g^* = 1$, respectively would be favorable for any individual project g. Nevertheless, due to the form of the objective function $\Phi(\vec{\lambda})$, every element of $\vec{\lambda}$ depends on all other elements of $\vec{\lambda}$ in an optimal portfolio. Therefore, the optimality of the objective function cannot be assured when adapting a single λ_g. As nonlinear optimization goes beyond the scope of this paper we assume for the following two projects example $\overrightarrow{\hat{\lambda}} \in [0,1]^2$. Later on, we overcome this problem and the assumption $F_g = 0$ by using simulation.

III.1.3.4.1 Two Projects Example

We now analyze a two projects setting for the projects g and h, respectively. Figure 3 shows the total risk adjusted costs of a two projects portfolio subject to two outsourcing degrees λ_g and λ_h.

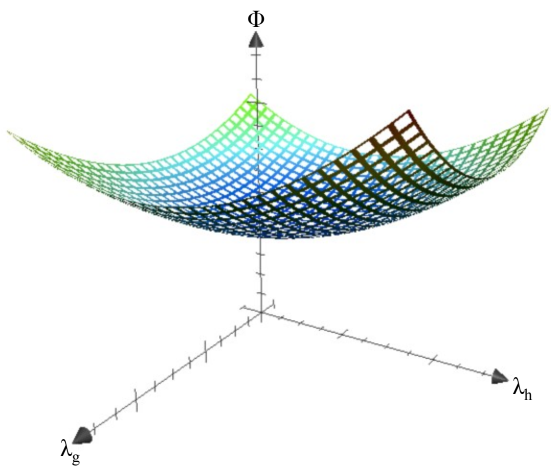
The total risk adjusted costs minimizing outsourcing degrees, λ_g^* and λ_h^*, can be quantified as follows

$$\lambda_g^* = \frac{\left(f_g + \mu_{x,g} - \mu_{n,g}\right)\left(\gamma_g - 1\right) + 2\alpha_n \sigma_{n,g} \left(\sigma_{n,g}(\gamma_g - 1) + \rho_{g,h}\sigma_{n,h}(\lambda_h^* - 1)\right)}{2(\gamma_g - 1)(\alpha_n \sigma_{n,g}^2 + \alpha_{x,g}\sigma_{x,g}^2)} \tag{19}$$

and

$$\lambda_h^* = \frac{\left(f_h + \mu_{x,h} - \mu_{n,h}\right)(\gamma_h - 1) + 2\alpha_n \sigma_{n,h} \left(\sigma_{n,h}(\gamma_h - 1) + \rho_{g,h}\sigma_{n,g}\left(\lambda_g^* - 1\right)\right)}{2(\gamma_h - 1)\left(\alpha_n \sigma_{n,h}^2 + \alpha_{x,h}\sigma_{x,h}^2\right)} \tag{20}$$

Figure 3: Optimal Outsourcing Degrees of Two Projects



As stated in the single project view, the denominators of both, λ_g^* and λ_h^*, are always negative, the extension by constants do not change any findings. The numerator contains the spread in risk-adjusted costs of outsourcing and internal development and is of either sign depending on the profitability of either option.

In the previous sections we do not take fixed transaction costs into consideration. Therefore, our results favor outsourcing even on condition that the fixed transaction costs exceed the savings due to outsourcing. Moreover, with our analytical approach we are not able to assure solutions within the domain of $\overrightarrow{\lambda^*}$ in every case. To eliminate such distortions and to provide more findings, we will use simulations in the following.

III.1.3.4.2 Framework for the Simulations

For the findings shown in the following sections, we generated a set of project parameters and outsourcing reference values to run the simulations, pictured in the graphs below. We suggested the following input parameters for twelve projects: expected costs, standard deviations, parameters of risk aversion, price assessment outcomes, correlation coefficients and fixed/variable transaction costs. For the estimates we adopted the proportions of the expected values and standard deviations of Zimmermann et al. (2008). The values are based on real business case data of a major IT service provider, whose identity is disguised for reasons of confidentiality. The correlation coefficients are randomly generated, equally distributed numbers between 0 and 1. For reasons of comparability we assumed equal returns of all projects. For the outsourcing degrees, we created 24,000 equally distributed reference values for each project. The probabilities of *no outsourcing* and *total outsourcing* were manually set to 5% each. Otherwise, these realistic decisions would be underrepresented in our random numbers.

For simplifying matters of expression, we use the term "efficient" for non-dominated results of our simulation, although we are aware of the fact that they could be dominated by results of a full enumeration or an analytical optimization (either one of them is very difficult to realize, therefore we proceed with a simulation).

III.1.3.4.3 Outsourcing of Multiple Projects within a Fixed Project Portfolio

The client considers $Q = 12$ given software development projects in a portfolio. The expected values and standard deviations of the 12-projects-portfolio with different outsourcing degrees are shown in the following diagram.

Figure 4: Fixed Project Portfolio with Random Outsourcing Degrees

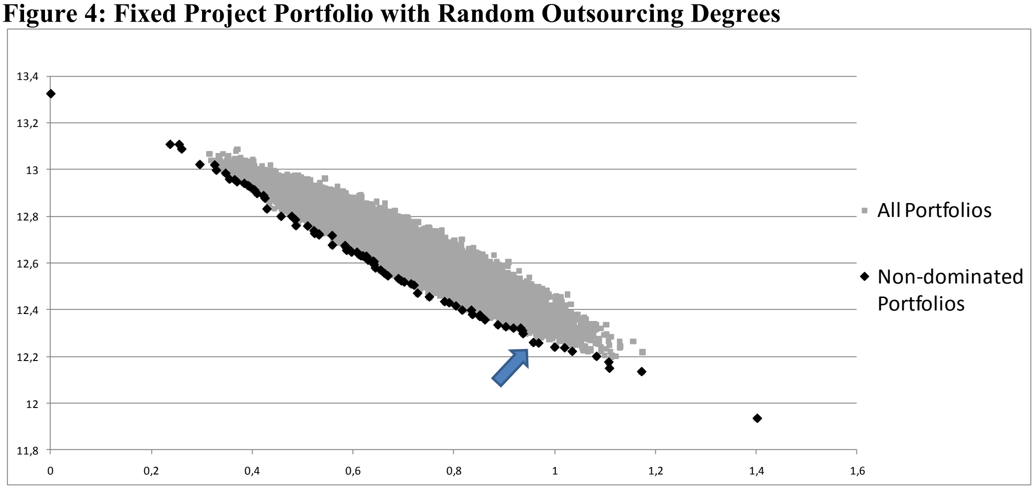


Figure 4 shows a scatter plot of possible outsourcing alternatives for the fixed portfolio. A frontier of efficient portfolios is shown in dark grey. If the outsourcing client considers portfolio dependencies in the selection of the outsourcing degrees, a superior solution can be achieved. The arrow indicates the portfolio with the best allocation of outsourcing degrees identified during the simulation, which is the portfolio with the lowest risk adjusted total costs, amounting to $\Phi = 12.7169$. These solutions are only non-dominated but not necessarily optimal, because results are derived by simulation and not by optimization.

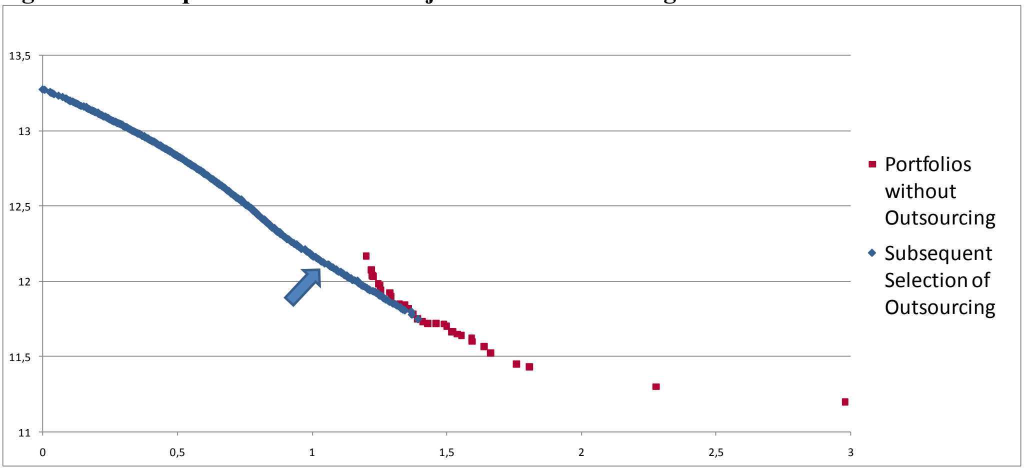
So far, we only considered a given set of projects and combined them into one portfolio and plotted it with multiple outsourcing degrees. However, a client faces multiple options to choose from and to build an efficient project portfolio. Therefore, we will picture the portfolio choice process and show its effects on the best solution. Therefore, a fourth research question can be posed: *Is it more favorable to determine efficient outsourcing degrees for a previously selected optimal portfolio than to simultaneously select both, projects and their respective outsourcing degrees?*

III.1.3.4.4 Outsourcing of Multiple Projects within an ex ante Determined Portfolio

To evaluate the first part of our research question, we consider a selection of an optimal project portfolio with q out of $Q = 12$ projects. We build the portfolios using complete enumeration then we pick the optimal one, which is the portfolio with the lowest risk adjusted total costs. Subsequently, for each project within the optimal portfolio 24,000 random, equally distributed outsourcing degrees are determined by simulation. Amongst all possible outsourcing combinations the best portfolio solution is identified.

In figure 5, the red dots show all efficient expected value- and standard deviation- combinations of portfolio selections without outsourcing. The optimal portfolio has total risk adjusted costs $\Phi = 12.7192$. All efficient portfolio combinations of partially outsourced projects are pictured in blue. One can see that the portfolios of partially outsourced projects dominate several efficient portfolios without outsourcing and therefore might be favored by the client. The best portfolio solution with outsourcing amounting to $\Phi = 12.6664$, is again denoted by an arrow. The portfolio with outsourcing is superior $(+0.42\%)$ to the portfolio without outsourcing.

Figure 5: Subsequent Selection of Projects and Outsourcing



We examined an ex ante portfolio choice with a subsequent selection of outsourcing degrees. We now want to see if a simultaneous portfolio choice and selection of outsourcing degrees will lead to an even better solution.

III.1.3.4.5 Outsourcing of Multiple Projects with Simultaneous Portfolio Selection

In contrast to established business processes where outsourcing decisions are made after the decision on the composition of the project portfolio, we now choose q out of $Q = 12$ projects with their q outsourcing degrees simultaneously.

Figure 6: Simultaneous Selection of Projects and Outsourcing

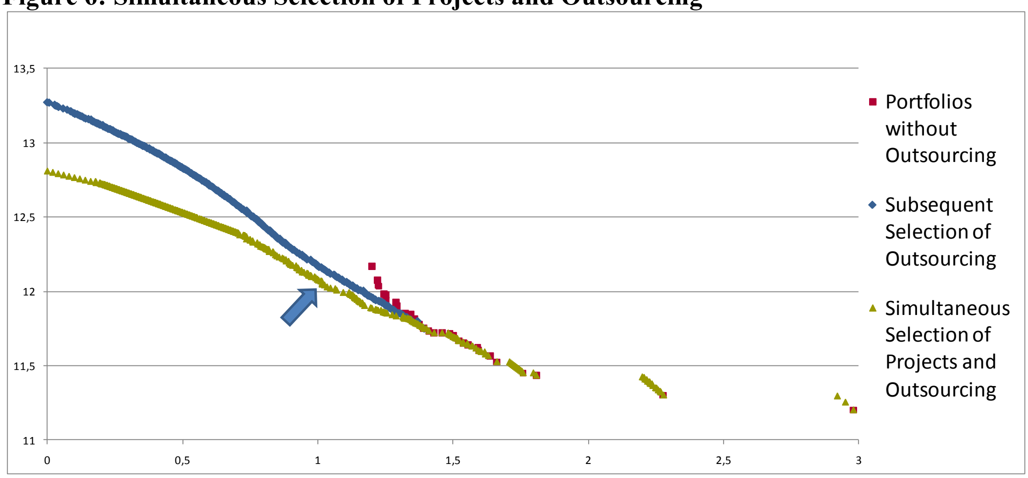


Figure 6 shows a portfolio choice of q out of Q projects and **subsequent** selection of q individual outsourcing degrees for the predetermined portfolio projects as established in the previous paragraph. Furthermore it shows the **simultaneous** selection of q out of Q projects and q associated outsourcing degrees of all possible projects. This leads to portfolio compositions from which the best possible portfolio with $\Phi = 12.5638$ can be determined (indicated by an arrow). The simultaneous selection of projects and outsourcing degrees gets to a superior solution ($+\ 0.82\%$) compared to the subsequent selection, where only the outsourcing degrees of the predetermined portfolio are part of the simulation. Compared to the portfolio without outsourcing, the simultaneous selection of projects and outsourcing degrees is superior, too ($+\ 1.22\%$). Although the improvement might seem small at first sight, the benefit companies might realize should not be underestimated. Above, we compare our finally best portfolio to an already optimized portfolio without outsourcing, but to date, companies rarely use effective portfolio optimization to decide on outsourcing their IT projects. The reference values for comparison would therefore be lower in reality and the potential gains are higher. Furthermore, a major company with a corresponding IT budget might realize substantial absolute savings.

III.1.3.5 Practical Implications, Limitations and Conclusion

Today, companies increasingly realize the relevance of IT portfolio management in general as well as in the context of IT outsourcing. Thereby, they extend their focus from a pure cash-flow oriented view to a more generic one and integrate risk and dependencies into their decisions. Nevertheless, these approaches are often pragmatic and methodically weak. The vision of a value adding quantitative IT portfolio management requires methodically rigor models that deliver initial reasonable results, although they might not be suitable to be applied in practice without adjustments.

Although it bears great cost reduction potential, still little research exists in the field of fixed price outsourcing and its effects on an IT project portfolio. This paper provides a quantitative model to help companies to improve their IT outsourcing strategies. Including interdependencies between projects as well as transaction costs, we find that outsourcing an appropriate fraction of an IT project can enable a company to minimize the risk adjusted costs of a project, as well as of a project portfolio. Moreover, we discover that the simultaneous selection of outsourcing degrees and best project portfolio may lead to even lower risk adjusted total costs than the subsequent determination of the best project portfolio and outsourcing degrees.

This is of special importance as today's IT decision processes mostly feature subsequent decisions only. Companies usually decide on projects first and then evaluate possible outsourcing settings. The restricting assumptions of this paper are necessary to show analytically that this bears optimization potential. Relaxing these restrictions would make an analytical solution impossible. But still, there is no obvious reason, why these effects should not occur. A business oriented model which is directly applicable but still methodically rigor will be the objective of further research in this area. Therefore, every limitation of this paper has to be addressed separately and analyzed profoundly.

First, the exclusion of risk for a fixed price outsourced fraction might not necessarily picture reality, because for example default risks remain. In terms of this paper, these additional risks could be pictured by introducing price and transaction costs as random variables. This leads to a gain in complexity because all correlations between in- and outsourced fractions would have to be considered. This major extension of the model is our current work-in-progress. It will also include the analysis of contract types, other than fixed price outsourcing. Furthermore, we currently neglect varying returns of projects and assume them to be constant regardless of the degree of outsourcing. The implementation of projects by a specialized service provider might however have positive and negative impacts on the return, e.g. through influences outlined in agency theory. This would provide a more eclectic picture of reality.

Also, we assume infinite divisibility of projects to be able to build continuous functions and their derivatives in order to derive our results analytically. However, one has to admit that dividing arbitrary parts of projects might be technically impossible or irrational concerning economical aspects. In contrast, discrete partitioning might lead to inferior absolute outcomes. Nevertheless, the model can be used to heuristically approximate discrete results as a basis for an in-depth analysis. Additionally, the linear relationship of the fraction's size to costs and risk, requested in assumption 3, might lead to a loss in generality, since different parts of a project might entail distinct values of costs and risks. Separate observation of different project parts with different risk/cost structures might be a practical addition. Moreover, we include risk diversification effects in the objective function, but neglect them in the price assessment – for both, the outsourcing client and the service provider. The effects on the price range might as well be subject to further model extensions. Finally, our model pictures ex ante decisions only. The development of an integrated model considering the existing project portfolio as well as the decision on additional projects might be of great significance to practitioners as well as to researchers.

Although the model pictures reality in a constrained way, it provides a basis for firms to plan and improve their outsourcing strategies. Thereby, it is not only of high relevance to business practice, but also provides a theoretically sound economical approach.

III.2 References (Chapter III)

Apte UM, Sobol MG, Hanaoka S, Shimada T, Saarinen T, Salmela T, Vepsalainen APJ (1997) IS Outsourcing Practices in the USA, Japan and Finland: a Comparative Study. Journal of Information Technology 12(4):289-304

Aron R, Clemons EK, Reddi S (2005) Just Right Outsourcing: Understanding and Managing Risk. JMIS 22(2):37-55

Arrow KJ (1971) The Theory of Risk Aversion. In: Arrow KJ (ed) Essays in the Theory of Risk-Bearing, 1 ed. Markham, Chicago, 90-120

Aspray W, Mayadas F, Vardi MY (2006) Globalization and Offshoring of Software. A Report of the ACM Job Migration Task Force. http://www.acm.org/globalizationreport. Access 2010-02/17

Aubert BA, Dussault S, Patry M, Rivard S (1999) Managing the Risk of IT Outsourcing. In: Sprague RH (ed) Proceedings of the 32nd Annual Hawaii International Conference on System Science, HICSS-32, IEEE Computer Society, Maui, Hawaii

Aubert BA, Rivard S, Patry M (2004) A Transaction Cost Model of IT Outsourcing. Information & Management 41(7):921-932

Bernoulli D (1954) Exposition of a New Theory on the Measurement of Risk. Econometrica 22(1):23-36

Boehm B, Abts C, Chulani S (2000) Software Development Cost Estimation Approaches - A Survey. Annals of Software Engineering 10(1):177-205

Bryce DJ, Useem M (1998) The Impact of Corporate Outsourcing on Company Value. European Management Journal 16(6):635-643

Butler S, Chalasani P, Jha S, Raz O, Shaw M (1999) The Potential of Portfolio Analysis in Guiding Software Decisions. In: Proceedings of the First Workshop on Economics-Driven Software Engineering Research, EDSER-1, Los Angeles, California

Conrow EH, Shishido PS (1997) Implementing Risk Management on Software Intensive Projects. IEEE Software 14(3):83-89

Courant R, John F (1965) Introduction to Calculus and Analysis - Volume One. Wiley, New York

Currie WL (1997) Expanding IS Outsourcing Services through Application Service Providers. Growth 8(1):1

Dibbern J, Goles T, Hirschheim R, Jayatilaka B (2004) Information Systems Outsourcing: A Survey and Analysis of the Literature. ACM SIGMIS Database 35(4):6-102

Dibbern J, Heinzl AH, Winkler J (2006) Offshoring of Application Services in the Banking Industry – A Transaction Cost Analysis. Working Paper 16/2006

Dutta A, Roy R (2005) Offshore Outsourcing: A Dynamic Causal Model of Counteracting Forces. JMIS 22(2):15-35

Franke G, Hax H (2004) Finanzwirtschaft des Unternehmens und Kapitalmarkt, 5 ed. Springer, Berlin

Freund RJ (1956) The Introduction of Risk into a Programming Model. Econometrica 24(3):253-263

Hanink DM (1985) A Mean-Variance Model of MNF Location Strategy. Journal of International Business Studies 16(1):165-170

Krapp M, Wotschofsky S (2004) Vorteilhafte Leasinggestaltungen bei asymmetrischer Besteuerung. Zeitschrift für Betriebswirtschaft 74(8):811-836

Lacity MC, Hirschheim RA (1993) Information Systems Outsourcing: Myths, Metaphors, and Realities. Wiley, New York

Lacity MC, Willcocks LP (1998) An Empirical Investigation of Information Technology Sourcing Practices: Lessons from Experience. MIS Quarterly 22(3):363-408

Lacity MC, Willcocks LP, Feeny DF (1996) The Value of Selective IT Sourcing. Sloan Management Review 37(Spring):13-25

Lacity MC, Willcocks LP (2003) IT Sourcing Reflections: Lessons for Customers and Suppliers. Wirtschaftsinformatik 45(2):115-125

Lammers M (2004) Make, Buy or Share Combining Resource Based View, Transaction Cost Economics and Production Economies to a Sourcing Framework. Wirtschaftsinformatik 46(3):204-212

Lee JN, Huynh MQ, Kwok RCW, Pi SM (2003) IT Outsourcing Evolution: Past, Present, and Future. Communications of the ACM 46(5):84-89

Loh L, Venkatraman N (1992) Determinants of Information Technology Outsourcing: A Cross-Sectional Analysis. JMIS 9(1):7-24

Oh LB, Ng BLT, Teo HH (2007) IT Portfolio Management: A Framework for making Strategic IT Investment Decisions. In: Österle H, Schelp J, Winter R (eds) Proceedings of the 15th European Conference on Information System, ECIS, St. Gallen

Patel NR, Subrahmanyam MG (1982) Simple Algorithm for Optimal Portfolio Selection With Fixed Transaction Costs. Management Science 28(3):303-314

Santhanam R, Kyparisis GJ (1996) A Decision Model for Interdependent Information System Project Selection. European Journal of Operational Research 89(2):380-399

Sauer C, Gemino A, Reich BH (2007) The Impact of Size and Volatility on IT Project Performance. Communications of the ACM 50(11):79-84

Slaughter S, Ang S (1996) Employment Outsourcing in Information Systems. Communications of the ACM 39(7):47-54

Standish Group (2006) Chaos Report 2006.

Verhoef C (2005) Quantifying the Value of IT-Investments. Science of Computer Programming 56(3):315-342

Weill P, Aral S (2005) IT Savvy Pays Off: How Top Performers Match IT Portfolios and Organizational Practices. MIT Sloan Research Paper (No. 4560-05)

Willcocks LP, Lacity MC, Kern T (1999) Risk Mitigation in IT Outsourcing Strategy Revisited: Longitudinal Case Research at LISA. Journal of Strategic Information Systems 8(3):285-314

Zimmermann S, Katzmarzik A, Kundisch D (2008) IT Sourcing Portfolio Management for IT Service Providers - A Risk/Cost Perspective. In: Proceedings of 29th International Conference on Information Systems, ICIS, Paris

IV Conclusion and Outlook

This chapter contains in section IV.1 a summary of the main results and in section I.2 an outlook on further areas of research. Both sections are again separated by the two perspectives: IT as an instrument and as an object of risk/return management.

IV.1 Conclusion

Chapters II and III show that information technology plays an important role for risk/return management. The papers not only include relevant models on using IT for risk/return management, but also on managing IT outsourcing projects from a risk/return perspective.

IV.1.1 IT as an Instrument of Risk/Return Management (Chapter II)

In P1, we considered a company that has to decide on the amount of capital reserved to cover potential losses resulting from a risky investment portfolio. The paper shows how to measure the economic value that can be derived from IT supported risk/return management calculations. We developed an optimization model that delivers the optimal amount of computing capacity that should be allocated to risk calculations at a time. Thereby, we restricted our analysis to one well-defined risk/return management problem: Covariances are fundamental and widely used in financial applications. Nevertheless, there are numerous other risk/return management methods and algorithms. Still, most of the basic principles introduced in this paper can be adapted to those scenarios and to more sophisticated and complex surroundings.

In addition, we described in P2 that highly volatile market parameters, as observed during the financial and economic crisis, result in varying demand for computing capacity making a service oriented infrastructure especially advantageous. With a service-oriented infrastructure, an economic optimum regarding the allocated computing capacity can be reached at any time, as resources can easily be reallocated. With a dedicated system where expected values are applied for capacity planning the economic optimum is systematically missed due to market parameters deviating from prior expectations. By analyzing the different cost structures, we were able to provide a rule for decisions on service oriented infrastructures vs. dedicated systems.

P3 is a more technically oriented publication: It contains different grid network topologies and suitable algorithms for covariance calculation on these network structures. Furthermore, I derived the corresponding complexity classes for a distributed calculation on each topology. These results are again highly relevant from a business perspective: When designing a specialized algorithm for a company's (arbitrary) network infrastructure, one can apply the insights of this paper, and benchmark own algorithms against the theoretically derived complexity classes.

Altogether, P1, P2, and P3 provide meaningful insights for the use of IT as an instrument for risk/return management. Nevertheless, this dissertation is only a foundation: Implementing improved risk/return management in today's value networks requires plenty of further research in this area that will be described in section I.2.

IV.1.2 IT as an Object of Risk/Return Management (Chapter III)

Today, most companies start to extend their focus on IT projects from a pure cash-flow oriented view to a more generic one that integrates risk and dependencies into their decisions. Therefore, they increasingly realize the relevance of IT portfolio management in general, as well as in the special context of IT outsourcing. Nevertheless, practical approaches are often pragmatic and methodically weak. P4 follows the vision of a value adding quantitative IT portfolio management. This requires methodically rigor models that deliver initial reasonable results, although they might not be suitable to be applied in practice without adjustments. Relaxing the necessary restrictions would make an analytical solution impossible. Nevertheless, there is no obvious reason, why the identified effects should not occur in a more complex model setting or in reality.

Concerning the field of fixed price outsourcing and its effects on an IT project portfolio, there exists only little literature, although we were able to show that it bears great cost reduction potential. The quantitative model developed in P4 can help companies to improve their IT outsourcing strategies. Outsourcing an appropriate fraction of an IT project can enable a company to minimize the risk adjusted costs of a project, as well as of a project portfolio. Moreover we illustrated, that today's usual process of deciding on the implementation of individual projects first, and deciding afterwards and case by case, if and to what extent a project shall be outsourced, delivers suboptimal results: The simultaneous selection of outsourcing degrees and a best project portfolio can lead to significantly lower risk adjusted total costs, especially for a major company with a corresponding IT budget.

The research area "IT as an object of risk/return management" still offers just as many opportunities for further research as the papers described in chapter II. The following section I.2 will therefore contain outlooks for both major topics of this dissertation.

IV.2 Outlook

Chapters II and III addressed the two perspectives of risk/return management on information technology: IT as an instrument and as an object of risk/return management. The respective papers focused on certain parts of these manifold research areas, only. For this reason, in the following, I will not only describe possible extensions on the presented work, but also on the research area as a whole.

IV.2.1 IT as an Instrument of Risk/Return Management (Chapter II)

Risk/return management has already been supported by corresponding IT before and during the FEC. However, in the majority of cases, the support was implemented in isolated application systems, and therefore neglecting interdependencies, for example between different business sections. To support a value based management and to satisfy regulatory transparency requirements as well as legal reporting obligations, enterprises need a companywide consistent database with return and risk information. Although there are technical approaches, in order to implement integrated management support systems that are usable for different purposes (e.g. data warehouse technology), appropriate financial methods are still lacking: Decisions rules (e.g. on acquiring or accepting a new customer, investing in new asset classes, establishing new business processes, or initiating a data warehouse project) must take into account both dimensions, risk and return. This requirement is often not met in practice nowadays. Furthermore, these decision rules must be consistent and compatible to incentives throughout the company.

Therefore, the vision of a companywide risk/return management depends on an adequately designed IT environment: All assets and their corresponding risk have to be modeled and valuated in a consistent manner and all decisions concerning the risk/return position have to be made according to the same rules, even in different business contexts. Therefore, IT must provide a standardized integration concept throughout the company and its value network.

IT landscapes are often wildly grown, scattered and therefore already difficult to overview. Incorporating risk/return management functionality into such landscapes is a complex task, which will require a flexible architecture that supports mapping and reengineering of existing business processes and application systems. To reduce complexity, software must be decomposed into smaller functional units (e.g. services for calculating reusable intermediate results or different risk and return measures). When designing functional units, factors like usage frequency, corresponding computing time, and protocol overhead have to be taken into account. Furthermore, data which are necessary for risk/return management, but currently not stored within the IT system have to be collected into existing or newly modeled data structures. A lot of data might have been already stored in existing databases that are distributed over the company's information system. Again, as huge sets of data are concerned and as time is a critical factor, existing data warehousing methods might not suffice.

Last but not least, due to dynamic markets, IT must be able to make extensive calculations in very short time. As described in detail in this dissertation P1, P2, and P3 are first contributions to understand the utility of service-oriented infrastructures and especially grid computing to reach this objective. Nevertheless, these approaches can be extended both, from an economic and from a technical point of view. Other methods of measuring risk and other technologies (e.g. In-Memory-Databases) still have to be evaluated.

IV.2.2 IT as an Object of Risk/Return Management (Chapter III)

As described in chapters I and III, IT projects and assets are an interesting object of risk/return management. Although paper P4 necessarily includes some strong assumptions, its results are still able to help companies improve their outsourcing strategies. Accordingly, it is not only rigor, but also relevant for business practice.

Nevertheless, P4 covers effects of IT projects on other IT projects, only. In the context of a companywide risk/return management, dependencies between IT assets and e.g. customers, stocks and all other asset classes need to be analyzed and quantified. For example, the implementation of a customer relationship management system might bear a lot of risk and cost. Still, it might increase customer satisfaction and thus increase return from customers and lower the risk of customer migration.

This also raises questions on the liquidity of assets: Stocks are usually considered very liquid, as they can be sold on stock markets very easily (at least under regular conditions). IT projects cannot easily be stopped or even sold, if they have a negative influence on the company's investment portfolio. To achieve an efficient risk and return position of an enterprise it is necessary to investigate the specific characteristics of liquid and illiquid assets and their interdependencies in an integrated portfolio management approach. Therefore, existing methods for risk measurement and portfolio management have to be enhanced to allow for the specific characteristics of IT projects, e.g. missing arbitrary divisibility/repeatability, long duration or missing market prices. Especially the interaction of liquid and illiquid assets is another interesting topic here: The risk associated with an illiquid IT project could for example be hedged by using liquid financial instruments. Finally, most of the existing models picture ex ante decisions only. The development of an integrated model considering existing IT assets as well as IT project opportunities might be of great significance to practitioners as well as to researchers.

In the near future, IT could serve as nervous system of the economy. Information systems could be enabled to recognize relevant changes of the real world, and to trigger decisions if necessary. As external impacts permanently occur in today's dynamic markets, this requires complex IT projects and assets. Therefore, IT requires an integrated view: Information technology is not only an instrument but also an object of risk/return management.