UNIVERSITY OF LATVIA
FACULTY OF PHYSICS AND MATHEMATICS
DEPARTMENT OF COMPUTING

DARJA ŠMITE

GLOBAL SOFTWARE DEVELOPMENT IMPROVEMENT

DOCTORAL THESIS

FOR PH.D. (DR.SC.COMP.) ACADEMIC DEGREE

FIELD:	COMPUTER SCIENCE
SUB-FIELD:	COMPUTER AND SYSTEM SOFTWARE
SCIENTIFIC SUPERVISOR:	DR.HABIL.SC.COMP., PROF. JURIS BORZOVS, UNIVERSITY OF LATVIA

RIGA, 2007

CONTENTS

PREFACE

This thesis addresses particularities of globally distributed software development (GSD). It is based on a three years long research with industrial background of one of the leading exporting software houses in Latvia.

The author explores the nature of globally distributed software projects in contradiction to common in-house projects and derives a set of particular factors and threats that endanger success of global collaboration. To provide practical application of the results this research also contains a set of practices, methods and tools that help to overcome the unique factors inherited in the heterogeneous nature of globally distributed environment. Global factors, threats and practices are accumulated in a risk-oriented framework for global project outcome prediction and elimination that is supplemented by supporting tools – Knowledge Base and Risk Barometer. Using these tools, managers without prior knowledge about global projects will receive a comprehensive overview of the global factors and threats, frequency of their occurrence and computerized summary of their impact on major project success criteria, including budget, schedule, customer satisfaction and other. This thesis also includes guidelines for global risk management and sustainable process improvement.

The author especially emphasizes importance of awareness of global factors and threats that even experienced managers often overlook. Additional empirical findings from the case studies and field observations provide novel and original insights in how globally distributed software projects are run by one of the major Latvian software houses. The author touches important questions of life cycle partitioning and global or virtual teamwork. In contradiction to many studies conducted from the customer perspective, this work addresses global project problems from supplier perspective.

Research results have a strong practical application for global project managers, team leaders, and other team members and are validated and implemented in the investigated software house.

ACKNOWLEDGEMENTS AND DEDICATION

I express my sincere gratitude to my supervisor professor Juris Borzovs for his guidance, encouragement and the necessary inspiration during times of decrease.

I thank my co-authors Uldis Sukovskis and Fergal McCaffery for interesting discussions and expertise in the field. And I wish to express my special thanks to Nils Brede Moe for interesting collaboration and his methodological assistance and Linda Vītuma for her practitioner's opinion and useful pieces of advice.

I am grateful to all my co-located and geographically distant friends and colleagues for their supportive interest and encouragement during these years.

I owe my deepest and warmest thanks to my parents to whom I dedicate this thesis. I express my dearest gratitude to my mother and farther, who always believed in me and kept me going even in the hardest times. I thank my darling husband for inspiring me and forgiving my frequent traveling and long working evenings.

This work was carried out in a research and development institute, headquartered by one of the leading software houses in Latvia, during the years 2003 – 2007. I am thankful for an opportunity to investigate industrial projects and publish the results that uncover organization's best practices along with faults, because only a very confident organization allows observing and analyzing its operation from inside.

This study was financially supported by the European Social Fund under grant "Doctoral Student Research and Post Doctoral Research Support for University of Latvia" and the Latvian Council of Science within the project Nr. 02.2002 "Latvian Informatics Production Unit Support Program in the Area of Engineering, Computer Networks and Signal Processing".

1. INTRODUCTION

1.1. RESEARCH BACKGROUND AND MOTIVATION

Global Software Development (GSD; also known as global software engineering (GSE), and globally distributed software development (GDSD)) has become the key trend in the area of software engineering. It is motivated by the opportunities of reaching mobility in resources, obtaining extra knowledge, speeding time-to-market and increasing operational efficiency. And yet, GSD is accompanied by both opportunities and problems. Many specialists recognize globally distributed software development as more complex than even the most complex project managed entirely in house [KAR98], [IES04]. Practitioners claim that they have to experiment and quickly adjust their tactical approaches for leveraging global software development risks [CAR01]. Researchers admit that although a body of knowledge on global software development has been crafted over time, the art and science of organizing and managing globally distributed software development is still evolving [DAM06].

Motivated by the market demand and industrial background a research on global software development improvement has been initiated in one of the biggest software houses in Latvia. The investigated software house was established in the late 80s and was reshaped several times. It has been orientated towards the international market, focusing on providing software development outsourcing services for the public sector, telecommunications, insurance and banking, as well as tourism and logistics. The software house has accomplished more than 200 projects both in Latvia, Western Europe and Scandinavia. At the present time the company represents a joint venture with over 380 employees. Industrial background of the research enables access to the current industrial experience and supports practical validation of the research results by the practitioners. In addition, input from a number of other Latvian software houses was received during certain steps of the research.

The research reported in this thesis started in August 2003 by an overview of the current research in the GSD area and process performance in the investigated software house, and finished in April 2007 by finalizing the research results.

1.2. RESEARCH QUESTIONS

Global software development is relatively new and unexplored [SAH03], [LEE03], [LOH95], [DAM06]. International Conference on Global Software Engineering[1] (ICGSE) is the first series of international conferences that in 2006 initiated dialog between academia and industry in this emerging area of research and practice. According to the ICGSE 2006 and ICGSE 2007 calls for papers, topics of interest include:

RT.1. Communication, coordination, collaboration and knowledge management for distributed software engineering;

RT.2. Requirements engineering and distributed client-supplier relationships;

RT.3. Architecture design and testing in distributed software projects;

RT.4. Collaboration infrastructure to support distributed development teams, e.g. change management, quality control, validation, project management, sharing of documents, online reviews, decision support;

RT.5. Process needs and proven solutions for effective and productive distributed software engineering;

RT.6. Managing distributed software projects: Planning, business case, team building, monitoring, reviewing, and reporting;

RT.7. The globalization experience: benefits of global development, effective business models, offshoring versus nearshoring, offshore-outsourcing versus internal offshoring, strategic versus tactical approaches;

RT.8. Managing diversity in global software projects and teams (technical, social and organizational);

RT.9. Working with globally distributed contractors: Vendor selection, contracting, supplier management;

RT.10. Empirical studies of distributed teams;

RT.11. Teaching global software development.

Global software development is said to be different from common in-house software development projects [SAH03], [MWD], [KAR98]. However, peculiarities of globally distributed software projects have not been explicitly formalized. Hence the research reported in

[1] For more detail see www.icgse.org

this thesis focuses on investigation of the phenomenon of global software development by searching the factors that are unique for GSD environment.

Risk analysis concepts have been applied to identify and evaluate particular negative events that might cause globally distributed software project failure. Threats that endanger globally distributed software project success are found to be quite distinctive from in-house project threats. Global risks are proved as just the part of everyday existence that cannot be avoided [SAH03], [PAV05], and must be confronted on a continuous basis. However, empirical results that would help to evaluate the magnitude of consequences of these environmental factors and threats provide contradictious views. In addition, Marvin J. Carr describes [CAR97] that as simple as it sounds, many organizations are unable to manage risks effectively. Accordingly, this research focuses on exploring the unique threats of globally distributed projects, effect of these threats on project performance and ways to overcome these threats before they lead to project failure.

In particular, the research questions are as follows:

RQ1: *What are the factors that distinguish global software development from traditional in-house software development?*

RQ2: *What is the effect of global factors on project performance?*

RQ3: *How to leverage global factors to improve project performance?*

The research addresses theoretical and practical issues and aims to establish a clear vision of risk management in software development projects that adds value to practitioners and supports timely response to the unique threats that appear in globally distributed software development environment. In particular, the research aims to explore application of traditional risk management concepts in an easy and comprehensive way. These activities shall also focus on both project improvement and organizational learning that is required by practitioners that lack knowledge or experience in the area of globally distributed software development.

The following parameters define the phenomenon under study:

- Project type – software development and enhancement projects,
- Location of software development life cycle activities – distributed among the involved partners (not produced within one team and location),
- Distribution – across national and/or organizational borders.

In other words, the author studies software projects that prescribe software or software component development by geographically and organizationally distributed team mates from multiple locations.

Several research steps have been planned and conducted in order to achieve results that would help to answer the research questions. The research steps, questions and results can be seen in Table 1.

Table 1. Research steps, questions and results

Research steps	Related Research Question	Research Results
1. Investigation of the distinguishing factors of GSD projects	RQ1	List of global factors List of global threats
2. Validation of relevance and prevalence of global threats from Latvian perspective	RQ1	Statistical summary of global threats' frequency of occurrence Empirical observations
3. Formalization of the effect of global factors and threats on GSD project performance	RQ2	Risk Barometer outcome predictions regarding different project success criteria
4. Exploration of globally distributed team performance	RQ2	Empirical observations of the reasons and effects of lacking trust
5. Deriving practices for leveraging risks in GSD projects	RQ3	Practices accumulated in a Knowledge Base
6. Developing methods and tools to support GSD project improvement	RQ3	Model for sustainable project performance improvement and Knowledge Base

The author has chosen different methodological approaches for each step of her research. Qualitative research methods are used in steps 1, 4 and 5, in their turn steps 2 and 3 follow quantitative research approaches.

Qualitative research is as scientific and empirically-based as quantitative research and is able to provide unbiased and rigorous results. In fact, specialists admit that carrying out good qualitative research is as at least as complex as conducting good quantitative studies [WIL00].

Survey and survey data statistical analysis as quantitative research methods were chosen to validate the earlier achieved qualitative research results.

1.3. SHORT PRESENTATION OF THE RESEARCH RESULTS

The results of this research address theoretical and practical aspects of globally distributed software project performance. Taking into account industrial background of this research, its major findings address global software project improvement from service supplier perspective.

In this thesis the author presents the following results of her research.

1. Peculiarities of globally distributed software projects

Description:	List of the unique environmental factors and threats that endanger globally distributed projects
Research Method:	Exploratory study using grounded theorizing
Data sources:	Extensive literature analysis and qualitative interviews
Data analysis:	Theoretical sampling (open, axial and selective coding)
Validation:	Relevance and prevalence of the evolved list of global factors and threats was validated through a survey on 38 globally distributed software projects in Latvia

2. Empirical observations

Description:	Data gathered during different research steps provided a ground for empirical observations considering globally distributed software project lifecycle management, work partitioning, and global teamwork.
Research Method:	Exploratory study
Data sources:	Extensive literature analysis, qualitative interviews, surveys, project problem analysis and post mortem analysis sessions
Data analysis:	Observation elaborated from quantitative and qualitative data analysis, theoretical sampling and data categorization
Validation:	Findings are based on or validated through case studies

3. Practices for globally distributed software project improvement

Description:	List of practices that provide guidelines for eliminating or decreasing the effect of global threats, accumulated in a Knowledge Base.
Research Method:	Exploratory study based on grounded theorizing
Data sources:	Extensive literature analysis, qualitative interviews and surveys

Data analysis: Theoretical sampling (open, axial and selective coding)

Validation: The practices are based on the empirically proven experiences. Independent validation is not performed.

4. GSD Knowledge Base and Risk Barometer

Description: GSD Knowledge Base accumulates practices for globally distributed software project improvement and is developed according to the concepts also referred to as Experience Factory [DIN00] and Knowledge Repositories [LIE98]. It supports organizational learning and prevents from the loss of explicit knowledge and experience.

It also contains a function called Risk Barometer, which enables project outcome predictions based on historic data of the effect of global threats and frequency of their occurrence. This is an experience-based guidance for risk evaluation especially for inexperienced project managers.

5. Approach for sustainable process improvement

Description: The author's developed approach for sustainable process improvement contains a set of methods and techniques aiming to compensate diversity and lack of team cohesion in globally distributed software projects. It addresses risk management and process improvement activities at the beginning, during and at the end of a project, as well as supports continuous organizational learning. This approach can be applied both in distributed and co-located teams.

1.4. SIGNIFICANCE OF THE RESULTS AND RESEARCH NOVELTY

The results of the research reported in this thesis support conclusion that globally distributed software development significantly differs from in-house software development. The author derives the list of peculiar global factors and threats that demonstrate the nature of globally distributed environment (published in related publication [SMB06]). These lists provide a valuable ground for effective risk identification supplemented by project outcome predictions that support further risk analysis for practitioners. In contradiction to many studies conducted from the customer perspective, this study investigates and includes global project problems

from supplier by this providing a useful support for Latvian and other software houses that operate as outsourcing service providers.

The results also provide the demanded empirical observations touching the following previously discussed topics of interest in the field of global software development: RT 1, RT 2, RT 5 and RT 10 (see also chapter 1.2). In particular, the research provides insights into globally distributed teamwork addressing communication, coordination, and collaboration perspectives (published in related publication [SMT04]); requirements engineering (published in related publications [SMI06] and [SMT06]); process needs and proven solutions for effective and productive distributed software engineering forming an empirical material based on the practices from the investigated global software development supplier house in Latvia.

The performed multi-case study on understanding the reasons and effect of lacking trust in globally distributed software teams (published in related publication [MOE07]) covers an unexplored topic that the existing literature did not discuss, which is also confirmed by Edwards and Sridhar [EDW03].

GSD project case studies range from announcements of tremendous success and total failure. No research so far provided a clear vision of the true amount of investments necessary to make global software projects work. The author developed a tool called Risk Barometer as a ground for an experience-based risk-oriented approach for GSD project outcome evaluation (published in related publication [SMI07]). The results of Risk Barometer performance include observations of budget, schedule and customer satisfaction threats – their significance and historical frequency of occurrence. Risk Barometer extends the traditional risk analysis approach and provides automatic prediction calculations on the basis of previous project data. This approach is an original attempt to formalize the effect of global threats.

The results of the research also include an approach for sustainable process improvement that includes a particular selection of software process improvement (SPI) methods that answer the needs of the investigated software house and has no scientific contribution

1.5. DISSEMINATION OF THE RESEARCH RESULTS

Research results have been reported and discussed in a set of international and local conferences and seminars:

- 11 international conferences and workshops:
    - Int. conf. DB & IS (2004 - Latvia);
    - Int. conf. EuroSPI (2004 - Norway, 2005 - Hungary, 2006 - Finland);

  - Int. conf. SPICE (2005 - Austria);
  - Int. conf. PROFES (2005 - Finland, 2006 - Netherlands, 2007 - Latvia);
  - Int. workshop LSO+RE (2006 - Germany);
  - Int. conf. Project Management Added Value (2006 - Latvia);
  - Int. conf. ICGSE (2007 - Germany).
- 3 local conferences and seminars:
  - LU Research Conference (2004, 2005);
  - LAT-SOFTWARE [2] SPI Seminar (2006);

Research results are reported in the following 13 papers published in the journals and proceedings of the international conferences:

2007 D.Šmite, "Project Outcome Predictions: Risk Barometer Based on Historical Data", accepted for publication in proceedings of the Int. Conf. on Global Software Engineering (ICGSE) by IEEE Computer Society, August 2007, Germany

2007 N. B. Moe, D. Šmite, "Understanding of Trust in Global Software Teams: A Multi-Case Study", In proceedings of the Int. Conf. on Product Focused Software Process Improvement (PROFES) in Lecture Notes in Computer Science (LNCS) by Springer Verlag, July 2007, Latvia, pp. 20-34

2006 D. Šmite "Requirements Management in Distributed Projects" Journal of Universal Knowledge Management, Vol. 1, No. 2, pp. 69-76. Accessible on-line: http://www.jukm.org/jukm_1_2/requirements_management_in_distributed

2006 D. Šmite, J. Borzovs „A Framework for Overcoming Supplier Related Threats in Global Projects", In Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), published in LNCS by Springer Verlag, October 2006, Finland, pp. 49-60

2006 D. Šmite, N. B. Moe „An ISO 9001:2000 Certificate and Quality Awards from Outside – What's Inside? – A Case study", In Proceedings of the Int. Conf. on Product Focused Software Process Improvement (PROFES), published in LNCS by Springer Verlag, June 2006, the Netherlands, pp.208-221

2006 F. McCaffery, D. Šmite , F. G. Wilkie, D. McFall „A Proposed Way for European software Industries to Achieve Growth Within the Global Marketplace", In journal of Software Process Improvement and Practice (SPIP), published by Wiley, 2006; Vol.11 Nr.3, pp.277-286

2006 D. Šmite „Global Software Development Projects in One of the Biggest Companies in Latvia: Is Geographical Distribution a Problem", In journal of Software Process Improvement and Practice (SPIP), published by Wiley, 2006; Vol.11, pp.61-76

[2] Hereinafter the name "LAT-SOFTWARE" is used to describe the investigate software house. The real name is changed due to confidentiality.

2006 D. Šmite „Requirements Management in Distributed Projects", In Proceedings of the Int. Workshop on Learning Software Organizations and Requirements Engineering (LSO+RE), published by the University of Hannover, March 2006, Germany, pp. 9-16

2005 F. McCaffery, D. Šmite , F. G. Wilkie, D. McFall „How European Software Industries can prepare for growth within the Global Marketplace. Northern Irish Strategies", In Industrial Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), November 2005, Hungary, pp.3.23-3.32.

2005 D.Šmite, U.Sukovskis „Knowledge Management in Distributed Environment", In Industrial Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), November 2005, Hungary, pp.5.15-5.22.

2005 D.Šmite, „A Case Study: Coordination Practices in Global Software Development", In Proceedings of the Int. Conf. on Product Focused Software Process Improvement (PROFES), published in LNCS by Springer Verlag, Finland, June 2005, pp.234-245

2004 D.Šmite, "Global Software Development Project Management – Distance Overcoming", In Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), published in LNCS by Springer Verlag, November 2004, Norway, pp. 23-33

2004 D. Šmite, J. Borzovs, "Global Software Development Process Management: Problem Statement", In Proceedings of the Int. Baltic DB & IS Conf. Doctoral Consortium, June 2004, Latvia, pp. 198-207

In addition, the author was involved in the activities of the Quality department in the investigated software house during the period 2004 – 2006, performing internal quality audits and consulting on software process improvement, developing and introducing a Knowledge Base containing knowledge and experience practices on globally distributed software development. The performed activities include:

- Quality audits in globally distributed projects;
- 3 project risk identification sessions;
- 1 project problem review with improvement suggestions;
- 1 project post-mortem reviews with lessons learned;
- 10 project effort estimations for initiated globally distributed projects.

1.6. OUTLINE OF THE THESIS

This thesis focuses on global software development improvement and describes a three years long research in this field. Research questions and background is described in the first chapter. The rest of the thesis is organized as follows.

Chapter 2 – GLOBAL SOFTWARE DEVELOPMENT: URGENT, DIFFERENT, COMPLEX AND UNEXPLORED – provides an insight into the history of outsourcing and global software development, and its importance to Latvian software market. It also highlights the uniqueness and complexity of GSD projects in contradiction to in-house software development. Novelty of the research area and related research overview concludes chapter 2.

In chapter 3 – PARTICULARITIES OF GLOBAL SOFTWARE DEVELOPMENT – the author describes exploratory study aiming to derive the distinguishing factors of globally distributed software development projects and threats particular for this type of environment.

Chapter 4 – OUTCOME PREDICTIONS FOR GLOBAL PROJECTS – is dedicated to the proposed risk-oriented approach to evaluated outcome of a global project. It provides insight in the basic concepts and calculations, and describes a tool called Risk Barometer for project outcome prediction automation.

Chapter 5 – THE EFFECT OF GLOBAL FACTORS AND THREATS – summarizes conclusions on the effect of global factors and threats on the major project success criteria explored by different means. It provides an overview of prevalence and relevance of the previously derived global threats; highlights the most popular threats, rare threats from Latvian perspective, major sources of budget and calendar deviations, and undermined morale. The author demonstrates the results provided by Risk Barometer performance as an attempt to formalize evaluation of the effect of global threats with the help of principles prescribed by risk management activities.

Chapter 6 and 7 touches particularly demanded topics of SOFTWARE LIFECYCLE MANAGEMENT AND WORK PARTITIONING and GLOBAL SOFTWARE TEAMWORK. Based on empirical material from a survey (chapter 5) and a multi-case study (chapter 6) the author derives her findings regarding the reasons and effect of problems connected to globally distributed collaboration models and teamwork. Both chapters provide recommendations to practitioners. Conclusions are especially valuable due to originality in contradiction to related studies.

Chapter 8 – GSD KNOWLEDGE BASE – describes the author's developed tool for the gathered research knowledge accumulation. This chapter highlights the necessity of knowledge bases for organizational learning and gives the corresponding recommendations for practitioners.

Chapter 9 – APPROACH FOR SUSTAINABLE PROCESS IMPROVEMENT – is dedicated to describe activities necessary for sustainable improvement of project performance. The described methods and techniques can be applied in different kinds of projects, by globally distributed and co-located teams and therefore are general. However, selection of these methods intended to specifically address globally distributed software project faults. The chapter concludes with

discussion of cornerstones in implementing software process improvement initiatives and recommendations to practitioners.

The author repeats the achieved results in chapter 10 – DISCUSSION OF THE RESULTS– providing estimation of their importance, limitations, application in practice and recommendations to practitioners.

The thesis concludes with chapter 11 – CONCLUSION, after which follows APPENDIXES with enclosed research related material. This thesis contains 8 appendixes: A.1: Global project survey template; A.2: Illustration of survey data records with variable representation of the SPSS tool; A.3: Instruction for SPI approach application; A.4: An example of risk management results; A.5: An example of project problem review results; A.6: An example of project post-mortem review results; A.7: Post-mortem review meeting evaluation by participants; A.8: Knowledge Base user guide.

2. GLOBAL SOFTWARE DEVELOPMENT: URGENT, DIFFERENT, COMPLEX AND UNEXPLORED

2.1. HISTORY

In the era of globalization outsourcing has been recognized as a natural evolution of how the global marketplace operates today [MIN05]. Tight budgets, shortage in resources and time has motivated many enterprises to start looking for partners outside. Outsourcing and especially offshoring (relocation of business processes to most frequently a lower cost country) have become components of a new global paradigm that is based on the selection of appropriate and strategic technologies, skills and resources with the strongest potential and the lowest cost within the global marketplace.

This phenomenon started a while ago with the growth of developing markets and the linking of value chains for global economies [MIN05] and continuous to be a stable trend for future. IT outsourcing originated from the professional services and facility management services in the financial and operation support areas during the 1960s and 1970s [LEE03]. Causing a slow down in the popularity of IT outsourcing, information system development was then considered a valued in-house function [LEE03]. Interest in outsourcing resurfaced in the early 1990s along with an increasing necessity for software development professionals due to lacking internal resources [KAR98], and not just for contract programming, specific processing services and resource in-sourcing also known as body-leasing, but for a wide range of services, such as [LEE03], [SAH03]:

- network and telecommunication management,
- data processing,
- system integration,
- software development,
- systems operations, including maintenance services and data recovery, and
- special services as software training and hotline support.

Motivated by resources available at lower costs IT outsourcing spread globally [KAR98]. International business environments and organizational forms are being significantly reshaped as part of a new scenario that has variously been labelled as the “new economy”, the “digital economy”, the “network society”, or the “information age”, enabling global work [SAH03].

Future of IT outsourcing is secure and by various opinions lies in building strong strategic partnerships [LEE03], [KAR98] providing new opportunities. According to Gartner Inc.[3] and IDC[4] market for offshore IT services continues to grow [MEA06]. Gartner expects offshore IT services spending to growth annually by 4.3% through 2010 [SHI07] and IDC predicts that collaborative product development applications market forecast through 2011 is relatively stable, with a five-year compound annual growth rate of 6.9% [WIL07].

The major driving forces for IT outsourcing include interest in deploying talented people, expansion through acquisitions, reduction in development costs, global presence, staying close to customers, and reduction in time-to-market [CAR99]. While at the very beginning companies decided to outsource their functions mainly due to increasing costs of the in-house personnel, now, according to the latest researches from the field [MIN05], there is a clear recognition that the driving forces in IT outsourcing are speed-to-market and quality, not just cost of savings. Moreover, specialists warn to look at more than service costs [SCA02]. Blind cost reduction deals tend to fail, because organizations do not realize all the risks and costs associated with the managing outsourcing contracts.

2.2. IMPORTANCE OF THE TOPIC IN LATVIA

Latvia with each year proves its stable position in the global IT market. The survey among 24 software houses in Latvia showed that the software industry has been an economically effective element in the national economy [ANS04], therefore it has won government attention and continuous to be a high priority sphere.

According to the latest research [MIN05], Latvia is recognized as an excellent destination for outsourcing. One quarter of the Latvian software houses is active exporters with export volumes which represent more than 15% of overall turnover [ANS04].

Its accession to the EU makes it easier to work with clients in “Old Europe” [IES04], [PAV05] – well-educated and multinational workforce [PAV05], one of the most efficient tax systems in Europe [MIN05]; rule-based, liberal economy modelled after Anglo-Saxon economies [MIN05]; great affinity with Nordics [MIN05], [PAV05], identical laws and equal rules on data protection and intellectual property [IES04].

[3] Gartner Research is the global leader in technology-related research and advice - www.gartner.com.

[4] IDC is the premier global provider of market intelligence, advisory services, and events for the information technology, telecommunications, and consumer technology markets - www.idc.com.

Future predictions say that together with its neighbours Estonia and Lithuania, Latvia is a major outsourcing centre for Northern/continental Europe [MIN05]. However, there are some areas of concern, such as aging population that is getting smaller too [MIN05], limited labour pool [MIN05], and increasing costs [MIN05], [IES04].

2.3. PECULIARITY

The concept of globally distributed or global software development (GSD) addresses transition of common way of producing software to software life cycle activities distributed among teams separated by various boundaries, such as contextual, organizational, cultural, temporal, geographical, and political. Global software development is said to be an unexplored form of work that is enabled through organizational forms quite distinctive from traditional global arrangements followed by large multinational corporations [SAH03]. It extends the concept of traditional outsourcing (the practice of subcontracting manufacturing work to outside [MWD]) through involving complex interdependencies between the teams involved in a joint software development life cycle. Accordingly software products are now being developed by multiple teams separated by national borders. Particularities of globally distributed projects often appear to be not connected with the technical nature of a software project [KAR98] that managers are used to monitor and avert. Therefore the unique pressures of the GSD projects are often overlooked by even capable project managers [KAR98]. Practitioners also claim that the reason of global project failure is not related to the abilities of project managers, but mainly to lack of awareness of the unique threats inherited in globally distributed environment [DEL05].

2.4. COMPLEXITY

The process of globalization introduces conceptual changes and increases complexity of software development. GSD has significant challenges with respect to communication, coordination and control issues, because of temporal, geographical and socio-cultural distance between members of the joint development team [AGE06]. Examples of difficulties introduced by the threats inherited in global environment are as follows:

Communication:

- Lack of effective communication tools makes binding among remote team difficult.
- Lack of language skills leads to fear to speak over phone or through video conferencing.
- Lack of personal contact leads to lacking trust and commitment.

- Temporal distance leads to dominant use of asynchronous communication tools.
- Dominance of asynchronous communication leads to considerable time delays.
- Cultural differences lead to misunderstandings.

Coordination:

- Multiple supplier involvement increases coordination effort.
- It also requires advanced tools with centralized access.
- Lack of proximity and next door closeness makes work partitioning, allocation and poorly defined task further specification more difficult and effort-intensive.
- This means increased requirements of accuracy of task definitions and requirements documentation.
- Diversity in process maturity and inconsistency in work practices can ruin schedules.
- Temporal distance introduces coordination problems.
- Weak telecommunications are source of unexpected coordination breakdowns.

Control:

- Lack of proximity and next door closeness means lack of transparency.
- Lack of transparency leads to increasing requirements to report on every activity taken "on the other side of the line".

These are only few examples of challenges that are brought by various aspects of distance and diversity – threats inherited in global software development environment. However, even these examples illustrate the difference between in-house software development projects and globally distributed software development. Therefore, globally distributed or virtual product development and especially project management are recognized as considerably more complex, than even the most complex project managed entirely in house [KAR98], [IES04].

Industrial case studies show that the global projects if not well-managed perform at (or even below) performance levels of those projects that are co-located at the company's home country [CAR99]. Erran Carmel writes that various managers are experimenting and quickly adjusting their tactical approaches for leveraging global software development risks [CAR01]. However, running and managing the projects intuitively according to the project managers' "gut feeling" can hardly bring to a predictable success. On the other hand, practices that are proven as effective in in-house software projects are often not applicable in globally distributed environment due to geographic and temporal distribution. This increases complexity of global project management and emphasizes the importance of in-depth research in this area.

2.5. NOVELTY

L.Loh and N.Venkatraman in 1995 emphasized, that despite its popularity, no research could determine the exact recipe for effective outsourcing performance [LOH95]. After 8 years J.-N. Lee and his colleagues in 2003 outlining the evolution of IT outsourcing claim that though IT outsourcing has long played an important role in the field, outsourcing trends are little understood [LEE03]. And even in 2006 Damian and Moitra introducing the IEEE Software special issue on GSD declare: although some theories and practices have been researched and developed, the art and science of global software development is still evolving [DAM06].

Despite the fact that global work is not a new phenomenon, distributed or virtual software development work is relatively new. Evaluation of the benefits of global software development is balancing between the claims of tremendous success and absolute failure. The existing studies therefore provide conflicting results that require in-depth research and analysis. There are many areas of concern that lack validated methods, techniques and tools for effective performance in globally distributed environment.

2.6. RELATED RESEARCH OVERVIEW

Evolution of research questions on IT outsourcing and the evolving trend of global software development started with the answers to such questions as whether to outsource or not ([WIL94], [ROY00]), how to manage relationship risk management questions ([BAT01], [CAR01], [EBE01]), contractual problems and advices ([AUB03], [LAC02]), success factors that will help to survive starting outsourcing relationship ([GRO96], [LAC02], [LAC95], [LIG01]), and case studies from the field ([LAC95], [LOH95], [LEE00], [CAR01]), supplemented with a great number of commercial whitepapers on how to select an adequate outsourcing supplier.

The more or less mature discipline relates to business process outsourcing (BPO) or IT related process outsourcing and IT service supplier management in general including non-software development companies. Such services are standardized in such frameworks as COBIT[5] and ITIL[6].

[5] COBIT is an IT governance framework (www.isaca.org/cobit/)

[6] ITIL® (the IT Infrastructure Library) is an approach to IT service management that provides a cohesive set of best practice (http://www.itil.co.uk/)

The vast majority of research in the area of global software development was conducted mainly from the customer's perspective because the objective of outsourcing was to self-maximize their internal resources without taking into account the service provider's situation [KAR98]. To answer the question of how to outsource, in other words, how to perform in a globally distributed environment series of conferences and journals have announced their call for papers. However, the number of serious research works in this area is relatively small, increasing only in the last few years. This remains the most demanded question.

To be more particular, the latest trends for research in the area of global software development brings up the questions of software lifecycle management and work partitioning, distributed infrastructure implementation, project management approaches and techniques, team communication and coordination strategies, and specific software process and practice improvements. These questions remain a relatively unexplored topic that demands examination in depth. There is only a handful of research highlighting the fact that the elements like trust, shared knowledge, mutual dependency, organizational linkages and cultural unity are also important and influence the success of the relationship [EBE01], [LEK03], [LEE00], [MOE07].

While processes and practices that in in-house software development seem obvious and are covered by various standardized approaches, in heterogeneous environment were engineers are separated thousands of kilometres from each other, having different languages and cultures, to achieve success is quite a challenge [KAR98], [EBE01]. It has been also found that the reason of failure of global projects is not the lack of capability, but lack of awareness of issues, problems, and barriers associated with global work [DEL05]. Likewise Sahay and Nicholson describe that the unpredictable nature of the risks in a global environment heightens the potential for unintended consequences [SAH03].

These and other findings drives to a conclusion that a company should consider pros and cons of collaborating over borders and never start a distributed collaboration unprepared. Awareness of the threats of global collaboration will help to avoid many problems of joint collaboration. Accordingly, the importance of timely risk management in the extremely dynamic and diverse environment of global software development grows.

2.7. CONCLUSIONS

Motivated by interest in deploying talented people, expansion through acquisitions, reduction in development costs, global presence, staying close to customers, and reduction in time-to-market IT outsourcing has occupied a strong position dictating the way how software is

developed nowadays. Globally distributed software development is a reality and a major trend in research that lacks empirical results. Despite the fact that global work is not a new phenomenon, distributed or virtual software development work is relatively new.

Globally distributed software development expands the concept of traditional outsourcing and addresses transition of common in-house manner of software development to more complex software life cycle activities distributed among teams separated by various boundaries, such as contextual, organizational, cultural, temporal, geographical, and political. This type of development environment can therefore be characterized by its heterogeneity, virtualness and inter-organizational collaboration that are impediments for effective communication and cooperation of the teams involved in completion of a joint project. New unique pressures of project management that appear to have nothing to do with the technical nature of the project and at the same time are reasons that can doom a virtual project is something even capable managers often overlook [KAR98]. Due to an industrial demand, global software engineering becomes an urgent research topic requiring empirical in-depth investigation.

Latvia with each year proves its stable position in the global IT market as one of the key outsourcing destinations for customers from Northern and continental Europe. Therefore, importance of increasing the competence and organizational learning grows.

3. PARTICULARITIES OF GLOBAL SOFTWARE DEVELOPMENT

3.1. OVERVIEW

The nature of global software development brings forward new areas of concern that require careful attention from project managers. Practitioners that have for a long time successfully managed in-house projects, now face new challenges that make them struggle to bring the projects to the end within the budget, time schedule and with the satisfied customers. One may think that the influence of globalization on software development remains limited by distributing end customers from their software development suppliers and at the same time having no effect on the life cycle processes. However, the concept of globally distributed software development prescribes separated teams from different organizations and/or locations work together on a joint project execution. These organizations form supply chains of different complexity by this increasing the complexity of software process distribution. Therefore, if the discipline of in-house software development life cycle management has already achieved some maturity (of course, with still open questions), globalization requires new processes, methods, and tools to be implemented.

In this research the author has explored particularities of global software development, by deriving a list of unique global factors and associated threats that endanger global project performance. The uncovered factors form a set of characteristics demonstrating uniqueness of global software development in contradiction to in-house software projects. The derived threats include those that are inherited in the nature of globally distributed environment and those that are caused by the environmental characteristics (global factors). Practices to overcome global factors and threats were derived within the exploratory study on deriving global project characteristics and sources of negative outcome.

3.2. METHODOLOGY

3.2.1. Grounded Theory As a Method for Deriving Global Factors and Threats

Grounded theory building methodology developed by Glaser and Strauss [GLA67] was chosen as the basis for the study. This methodology introduces a qualitative approach that generates theory from observation [STR90]. Theory-creating studies are very suitable for exploratory investigations, i.e., when there is no prior knowledge of a part of reality or a

phenomenon [JAR01]. Grounded theories, because they are drawn from data, are likely to offer insight, enhance understanding, and provide a meaningful guide to action [STR98].

Understanding of global factors and threats evolved grounded by systematically gathered and analyzed data about the phenomenon. The data was gathered from a variety of sources, including qualitative interviews and enhanced analysis of related research literature. Data analysis was performed according to principles prescribed by grounded theory through applying open, axial, and selective coding techniques [STR90], also called as theoretical sampling. A Lotus Notes-based database was used for data maintenance facilitating in easy categorization.

3.2.2. Data Sources

Various data sources were used for building the theory, including qualitative interviews with experienced project managers from the investigated software house, research literature (journal articles, papers form conference proceedings), and books on global software development.

The author conducted 13 interviews with 9 project managers, who represented all software development departments of the investigated software house running projects with customers from different remote locations and were appointed by the heads of departments as the most experienced ones. The interviews were held by means of semi-structured interviewing and open questions. The interviews were written down for further analysis.

The author performed an extensive literature analysis using input from 33 research articles on global software development published in the related conference proceedings and journals such as IEEE Software, and Communications of ACM.

Literature analysis and interviews with experienced project managers provided a representative input regarding the phenomenon under study.

3.2.3. Data Analysis

The author created a Lotus Notes based database to maintain the gathered data items and support data analysis. Sources of information and each data analysis iteration results were kept within the certain item's history for traceability opportunities.

Data analysis started with an open coding for data breaking down, examining, comparing, conceptualizing and categorizing. While examining data sources, expressions related to particular project characteristics, different negative events, consequences or practices were identified and labelled. Data analysis resulted in total of 253 GSD related issues, which were then stored into the database. Open coding then continued with categorizing. Each issue at the beginning represented a single category, the existing labels then were analyzed in order to

identify issues that are similar in meaning. Those were then grouped under more general concepts called "categories".

> E.g., the labels “Cultural barriers”, “Cultural distance”, and “Poor cultural fit” were coded under a joint category “Poor cultural fit”.

This reduced the number of GSD related categories to 163.

Examination of the existing categories showed that many issues are interrelated and form cause-effect interconnections. Axial coding was used for deriving connections between the existing categories and the risk management concepts, during which the identified GSD related issues from open coding were categorized into a hierarchy of sub-categories as follows:

- Global factors – root of global threats, that distinguish global projects;
- Global threats – items or activities that have potential for negative consequences and result from one or a combination of global factors;
- Consequences – negative outcome of a threat;
- Practices – recommendations for leveraging the risks.

Selective coding was used for systematically validating relationships and filling in categories that need further refinement and development. Axial coding showed that some of the existing categories have to be reconsidered. For some categories identified during open coding this meant dividing into two or even more categories.

> E.g., the category “E-mail communication causes time delays and misunderstandings” was divided into “E-mail communication” – a threat, and “Time delays” and “Misunderstandings” – consequences. “E-mail communication” was then united with one of the more general existing categories – “Asynchronous communication”. The relations between these three categories were then produced.

Refining the dependences between the new issues and tagging the categories with the sources were performed through selective coding.

To strengthen the results of this study regarding global factors and threats, only items that appeared more than once were selected, i.e. the threats that are strongly dependent on particular environment were omitted. Due to the industrial background of the research, customer related issues were also omitted after data analysis. New versions of records were processed, saving the history and notes reflecting the decisions within the database.

3.2.4. Results

Grounded theorizing resulted in 7 global factors, 32 supplier related threats, 7 supplier related major consequences and 32 supplier related practices (for more detail see the following chapters). As the theory was built, the author concludes that the most valuable results refer to global factors and threats. In its turn consequences and relationships between the global factors, threats and consequences are weak and inconsistent. Therefore, the author conducted a survey on 38 globally distributed projects to empirically validate these considerations and improve the theory.

3.3. GLOBAL FACTORS

The major distinguishing factors of globally distributed software projects are the following:

- **Multisourcing** – multiple distributed member involvement in a virtual team that develops software by joint effort, characterized by a number of collaboration partners;
- **Geographic distribution** – distance between the partners involved in the project;
- **Temporal diversity** – characterized by the level of working hours overlay, which most frequently differs from exact time zone differences;
- **Socio-cultural diversity** – level of social, ethnic, and cultural fit that can differ even between the teams from one national location;
- **Linguistic diversity** – language difference, characterized by the level of language skills of the project members;
- **Contextual diversity** – level of organizational fit or heterogeneity, characterized by diversity in process maturity and inconsistency in work practices;
- **Political and legislative diversity** – level of legislative consistency and sources of political threat.

These factors are recognized as roots of global threats that can endanger the success of a global project. They indeed demonstrate the peculiar nature of globally distributed software development and indicate the forces that act as impediments during a project. Each of the global factors and their combination causes various threats and conditions for negative outcomes. To illustrate global project environment in contradiction to co-located software development consider the following two examples.

Meet Jānis from a CO-LOCATED PROJECT – a bit idealistic but rather believable picture.

Multisourcing	At the beginning of the project Jānis was introduced to other project members in a kick-off meeting. With many of them he has already worked together, which was a sign for a good team atmosphere.
Geographic distribution	When Jānis comes to work he usually drinks a cup of coffee together with his colleagues while discussing work plans for the day.
Temporal distribution	If a problem arises, other team members immediately join Jānis to work on its solution. Therefore, it usually brings no delay and impact on the team.
Contextual diversity	And what is also important, everyone always knows what to do and act at one. Understanding of work processes and expectations are established a long time ago and are familiar to anyone in the company.
Socio-cultural diversity and Linguistic diversity	Jānis and his colleagues have traditional topics for discussion, as well as objects and subjects of jokes. They catch the meaning of their manager at once and it is reversible.

Meet Andris from a GLOBALLY DISTRIBUTED PROJECT – a common picture.

Multisourcing	At the beginning of the project Andris and his colleagues were told that they will be working on this software project together with a remote team from a far off location. This announcement established a sort of uncertainty on what to expect further in the project. Some colleagues gossiped about the real motives under such a decision.
Geographic distribution	During the whole project nobody has ever met the remote team members and their involvement would have caused no effect if only Andris and his colleagues wouldn't be so dependent on their work results.

	If a problem arises it is the primary question to search for the guilty person. If the remote team is the one that has caused the problem, this is immediately reported to the upper management. This often causes a downtime for Andris and his colleagues, which brings even more dissatisfaction.
Temporal distribution	In contradiction to other projects, downtimes are usual for this project. Many other delays are introduced by time zone differences. It takes at least half of a day for problem turnaround. Andris already knows that an email sent today will be answered only tomorrow. If answered at all – some emails get lost, and unluckily it always happens to the most important ones.
Contextual diversity	Since the remote team has distinctive education, Andris often spends time to explain what he expects them to deliver. This refers to both the work product and the way they perform.
Linguistic diversity	All attempts to discuss problematic issues over phone are doomed to fail, because neither Andris speaks English well, nor does the remote colleagues. After all, poor language skills cause frequent misunderstandings.
Socio-cultural diversity	Continuing the bad luck, during the last deadline Andris and his colleagues were left alone due to several national vacations at the remote partner side.

Although not every project has such a bad luck as the project that Andris works for, global factors tend to bring many problems that shall be monitored with a careful attention. The two pictures demonstrate the level of distinction between co-located and globally distributed software development. In particular, if the in-house project is characterized by one organization, its resources and work practices (see Fig.1) then globally distributed project is a transformation of a co-located intra-organizational collaboration into distributed inter-organizational collaboration (see Fig.2).

Fig. 1. Intra-organizational projects

Fig. 2. Inter-organizational globally distributed projects

Inter-organizational projects involve joint inter-organizational resources and are developed by global software teams also referred to as virtual teams. Accordingly, software processes are distributed between the remote team members and are affected by organizational work practices and habits.

The author emphasizes the uniqueness of globally distributed environment and marks that awareness of global factors that are inherited in the nature of globally distributed project environment can help practitioners to either reduce the probability or the magnitude of unexpected negative outcomes. However, if the global factors exist, they cannot be avoided.

> E.g., if geographic and temporal distance exists between the remote teams involved in the project, it can be neither avoided nor diminished. What the teams can do is work towards reducing the negative outcome of these factors.

3.4. GLOBAL THREATS

Global factors characterize different impediments for joint collaboration grounded in different types of diversity existent between the remote partners. These factors have considerable impact on the software life cycle processes. To limit or avoid the impact of global factors, project managers require knowledge on what to be aware of. Accordingly, the author has collected information on global threats that endanger global projects, to support project

managers in timely risk management. Global threats discovered within the research are as follows:

- Customer has complex hierarchy and/or several problem escalation levels
- Supplier has complex hierarchy and/or several problem escalation levels
- Diversity in process maturity and/or inconsistency in work practices
- Lack of understanding of each other's context of decision making
- The customer believes that the work cannot be done from a far off location
- Lack of trust and commitment
- Increased cost of logistics of holding face to face meetings
- Increased level of reporting on project progress to the customer
- Increased virtualness
- Lack of language skills by supplier
- Terminology differences
- Customer's employees unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode
- Faulty effort estimates
- Increased level of complexity of project management
- Increased level of unstructured poorly-defined tasks
- Increased complexity of spreading awareness and knowledge
- Lack of common goals
- Lack of experience and expertise of the customer with outsourcing projects
- Lack of experience and expertise of the supplier with outsourcing projects
- Lack of joint risk management
- Lack of team spirit
- Poor or disadvantageous distribution of software development activities
- Relatedness with other suppliers
- Poor cultural fit
- Dominant use of asynchronous communication with the customer
- Time zone difference
- Lack of clarity about responsibility share
- Poor or complex project measurement
- Increased complexity of project, activity, human resources and delivery planning
- Poorly defined or inconsistent SRSs
- Poorly defined or inconsistent software design and/or architecture
- Poor artefact version control

The identified global threats are note categorized according to their root factors, because a threat can be caused by a combination of global factors. These threats also tend to be general. The author aimed to avoid too detailed categorization of the threats to prevent the complexity of

correlated threat hierarchy. It also relieves the process of threat identification – too long checklists with odd issues are rarely used.

Accordingly, this list of threats doesn't contain all possible negative events that can endanger a global project. However, it is a useful guide to risk management that is based on previous experiences. In further chapters the author describes the results of investigation of these threats, regarding their frequency of occurrence and effect evaluation in global software projects with Latvian software house involvement.

3.5. CONCLUSIONS

The derived lists of global factors and threats make the peculiarity and complexity of globally distributed software development obvious. Global software development puts new demands on the software processes stressed by an increased complexity of project coordination (through temporal and geographical distances), communication (lacking proximity and cultural diversity), cooperation (lacking trust and commitment), infrastructure management (uniting heterogeneous contexts) and other aspects of distributed software development. The global factors characterize the distinguishing nature of globally distributed software development projects by emphasizing unavoidable elements that are inherited in this kind of work environment and shall be analyzed throughout the project. The list of global threats provides guidance for effective risk identification and demonstrates the various ways that global factors may act. This knowledge is especially demanded by practitioners that lack previous experiences in developing software with globally distributed partners.

Furthermore, practices applied for global risk mitigation shall act as a counterforce against global threats and reduce the effect of global factors and threats on project results (see Fig.3).

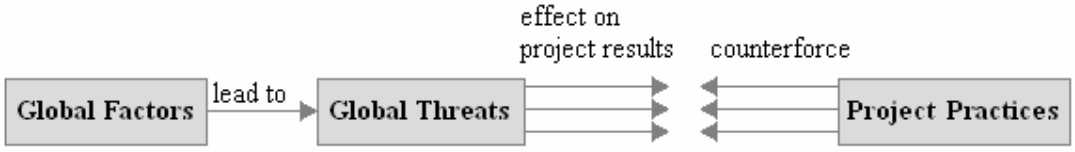


Fig. 3. Project practices as a counterforce for global threats

Global factors and threats derived by the author are used in further steps of the research to explore the outcome of these threats, their interrelation, importance and occurrence in projects run in Latvian software houses.

4. OUTCOME PREDICTIONS FOR GLOBAL PROJECTS

4.1. OVERVIEW

As previously emphasized, awareness of global factors and threats is essential for global project success. However, knowing about possible threats doesn't mean that organizations can evaluate the extent of each factor and threat. Limited experience and expertise in globally distributed software development often drives organizations to sudden problems due to underestimation of the hidden threats. Accordingly, an important role in estimating the severity of each factor plays awareness of their negative outcomes.

In this research the author offers an experience-based risk-oriented approach to leverage global threats. Traditional risk management concepts in this approach are introduced by components that characterize the effect of global threats on project performance. These are: probability of a threat to endanger a certain project success criteria, and the magnitude of the negative outcome of a threat. The author additionally calculates probability of negative outcome for each threat based on global project survey data, which extends traditional risk analysis concepts and introduces an approach to calculate future outcome predictions.

Experience data for effect evaluation was collected through a survey of global software projects from Latvian software houses. The author gathered data from 38 globally distributed software projects that provide a representative insight in what and how endanger global projects considering Latvian specifics.

4.2. BASIC CONCEPTS

Software risk management can be defined as an attempt to formalize risk oriented correlates of development success into a readily applicable set of principles and practices [BOE91]. However, practitioners often misuse risk terminology. Therefore the basic concepts and rules are defined as follows:

1. Term ***threat*** is used to describe possible negative events that can lead a project to its failure. E.g. Lack of experience with outsourcing projects.
2. Each threat has its ***probability of occurrence*** evaluated through the frequency of occurrence within the surveyed projects.

3. Each threat is evaluated for its ***negative outcome***. The following criteria is used in negative outcome evaluation for this research [7]:
    - Budget overrun;
    - Unexpected management costs;
    - Customer cost escalation;
    - Time delays;
    - Late product delivery;
    - Customer dissatisfaction;
    - Supplier team's undermined morale;
    - Disputes and litigations.
4. A threat can cause different level of negative outcome. E.g. dominant asynchronous communication may cause considerable time delays, but insignificant temporal distance only minor delays.
5. Evaluation of the level of negative outcome of the threat is called ***magnitude of the negative outcome***. To conform to traditional risk management concepts, magnitude of the negative outcome is calculated for each pair [threat; consequence]. In other words, the threat of poor cultural fit can cause e.g. minor time delays, considerable customer dissatisfaction, disastrous undermined morale of the supplier team and none effect on other success criteria.
6. Magnitude of the negative outcome and frequency of occurrence are evaluated according to a quantitative scale with an equivalent qualitative scale for interpretation as seen in Table 2.

[7] Project compliance with budget and schedule, customer satisfaction and software product quality are the major success criteria for the projects according to related literature [LIS05]. However, software product quality was not included in the list of indicators due to high risk of bias of the given evaluation. On the other hand, the list of project success criteria was extended due to the following reasons:

- Differentiation of causes of budget overrun (unexpected management costs, customer cost escalation, budget overrun);
- Time delays has been emphasized as a source of downtime, which doesn't obligatory drive to late product delivery;
- Supplier team's undermined morale is an important success criteria considering the industrial research background (supplier side of the project);
- Disputes and litigations are also possible negative outcomes that were additionally explored as a possible cause of project cancellation.

Table 2. Rating scales

Magnitude of the negative outcome		Frequency of occurrence	Probability
0	None	0	Improbable
1	Negligible	(0-10%]	Doubtful
2	Minor	(10-20%]	Unlikely
3	Moderate	(20-40%]	Possible
4	Significant	(40-80%]	Probable
5	Disastrous	(80-100%]	Certain

7. The combination of Magnitude of the negative outcome and frequency of its occurrence (for each pair [threat; consequence]) form ***risk exposure*** [BOE91] that is widely used in traditional risk comparison and prioritization. Multiplication can be used for quantitative evaluations, and matrixes for qualitative evaluations. Accordingly it helps to identify threats that have the most severe effect on the project performance separately for budget overrun, time delays, customer dissatisfaction, etc.

4.3. APPROACH TO CALCULATE OUTCOME PREDICTIONS

Threat prioritization is complex process. The described concepts contain two frequencies – frequency of a threat to occur and a frequency of a threat to cause certain level of negative outcome. Therefore, threats can be prioritized according to their frequency of occurrence but not according to the magnitude of consequences of the threat (due to their complexity). However, if it is necessary to include an evaluation of the effect of each threat, overall risk exposure in conditional units of measurement can be calculated according to the following equation:

$$RE(t) = \sum_{j=1}^{n} \left[Freq(t) \sum_{i=1}^{5} (i\ Freq(t, c_{i,j})) \right] \qquad (1)$$

Variables and functions:

t – threat;

RE (t) – risk exposure for a certain threat;

Freq (t) – frequency of occurrence of the threat;

i – quantifiable evaluation of level of the negative outcome (in this case 1..5)

Freq ($t, c_{i,j}$) – frequency of occurrence of the certain level of the certain negative outcome

Threat prioritization using the described evaluation of risk exposure in conditional units shall be used as the first step for further analysis of each negative outcome of a threat.

Derived from the equation above, the following equation can be used to determine the **risk exposure for each pair [threat; consequence]** – RE (t, $c_{i,j}$):

$$RE(t, c_{i,j}) = \sum_{i=1}^{5} (i \, Freq(t, c_{i,j})) \qquad (2)$$

This value can be used to compare different threats for magnitude of consequences on a certain project success criteria.

In order to support risk management activities for practitioners **Probability of negative outcome** is evaluated using frequency of occurrence of the negative outcome of the threat on the certain level by computing frequencies of lower effect levels with those of higher according to the following equation:

$$Prob(t, c_{i,j}) = \sum_{k=j}^{5} Freq(t, c_{i,k}) \qquad (3)$$

Variables and functions:

t – threat;

ci,j – outcome, where first index indicates the certain negative outcome (budget overrun, time delays, etc.) and the second – its level (1, 2, 3, 4 or 5)

Freq (t) – frequency of occurrence of the threat;

Freq (t, $c_{i,j}$) – frequency of occurrence of the negative outcome of the certain level of the certain threat;

Prob (t, $c_{i,j}$) – probability of the negative outcome of the certain level of the certain threat.

This equation can be illustrated with the following example:

> If the stroke of lightening in the past has once burned the whole tree, once burned leaves and sprigs but didn't cause any defect to the bole, and once burned only the leaves – the probability that the leaves will be burned based on historic information on condition that the lightening strikes the tree is 100%, the probability of the tree leaves and sprigs being burned is 66,6%, and the probability of the tree being burned is 33,3%.

Examples of application of these concepts follow in further chapters.

4.4. SURVEY OVERVIEW

The previously discovered list of threats was offered to different project managers and team leads for evaluation. A representative data set was collected using a survey instrument by mailing the developed questionnaire to a selected sample of employees in the investigated company, whose job title is project manager or equivalent, e.g., development manager or development team leader. In addition, the questionnaire was made accessible in other 4 small software houses, where the project managers could participate in the survey if interested. The survey template is given in Appendix 1.

The complexity of lifecycle distribution in the investigated projects varied from direct subcontracting to a complex chain of 10 subcontractors involved in completion of a joint project. The respondents experience varied from 3 to 30 years. Other characteristics considering the investigated projects under study are given in Table 3.

Table 3. Characteristics of the Surveyed Projects

Characteristics	Survey results		
Collaboration type Describes entities involved in the joint project, e.g. customer→ supplier (1→1), or customer → multiple suppliers (1→N)	1→1→1	13 projects	
	1→1	10 projects	
	1→N	7 projects	
	1→1→N	6 projects	
	1→N→N	2 projects	
Number of partners	2	11 projects	
	3	16 projects	
	4	3 projects	
	more than 4	5 projects	
Project success Subjective evaluation given by the project managers considering budget and calendar compliance, and customer satisfaction, using the scale 1-10.	Successful: 15,8%	10	4 projects
		9	2 projects
	Somewhat successful: 50,0%	8	7 projects
		7	12 projects
	Unsuccessful: 34,2%	6	5 projects
		5	4 projects
		4	2 projects
		3	1 project
		1	1 project

The following data was gathered during the survey:

- Project characteristics (collaboration model, project activity distribution, location of partners, project type, project status, success evaluation, etc.);
- Report of frequency of occurrence of the listed threats in the projects;
- Evaluation of the impact of each experienced threat on the project results.

4.5. SURVEY DATA ANALYSIS

Survey data was kept in and analyzed with the help of SPSS ® 14.0 tool[8], which provided a broad range of capabilities for the entire analytical process, including easy data search and categorization, powerful statistics, tabular and graphical representation of the results. Data was recorded within 316 variables; an example of SPSS variable representation view can be found in Appendix A.2.

Quantitative analysis of 38 globally distributed project survey data was performed to evaluate the effect of global factors and threats on GSD project performance. Survey responses have been statistically analyzed to compute the following values for each threat:

- Frequency of occurrence;
- Average outcome;
- Probability of certain level of the certain negative outcome.

Frequency of occurrence is based on the historic information from the survey data.

Average outcomes of a threat are minimum conditions that practitioners have to be taken into account while collaborating in the globally distributed project environment. Survey data contains evaluations of the magnitude of the negative outcome of each threat. Magnitude of the negative outcome of each threat is evaluated using a linear scale: [0, 1, 2, 3, 4, 5] or equivalent [None, Negligible, Minor, Moderate, Significant, Disastrous] as described earlier.

Probability of certain level of the negative outcome or **negative outcome predictions** are evaluated using frequency of occurrence of each threat to cause certain level negative outcome by computing frequencies of lower levels of impact with those of higher according to the definition given in the previous chapter.

A Bivariate Correlation analysis was performed in order to explore the **interdependence of the global threats**. The Bivariate Correlations procedure computes Pearson's correlation

[8] SPSS Software Solutions Online – http://www.spss.com/

coefficient with its significance levels. Correlations measure how variables are related. It is suggested to apply the cutting level for a statistically significant loading at the lowest, 0.30, when sample sizes are 50 and over [HAI79]. Therefore, as the sample size is only 38, the author applied a more precautionary cutting level of 0.40 to strengthen the results of the correlation analysis. The analysis uncovered cause-effect dependencies between the related threats and is used in characterization of the global threats and other observations of global projects.

4.6. RISK BAROMETER

Considering the length of the list of global threats and complexity of risk analysis, the author developed a tool that computerizes project outcome predictions correspondingly labelled as a "Risk Barometer" (reported in related publication [SMI07].

Risk Barometer is developed as a Lotus Notes based function aiming to support outcome predictions in global projects especially for project managers who lack awareness of possible negative events and their consequences in globally distributed environment. Risk Barometer performs its predictions on the basis of historic data from post-project risk evaluation reports. Risk Barometer and historic data is integrated in the Knowledge Base, which in more detail is described later in chapter 8. The author provided the first input for outcome predictions from survey data gathered during the research that is anonymously kept within the Knowledge Base. New project experiences can be added to continuously support Risk Barometer prediction improvements.

Risk Barometer provides risk analysis for each threat in the Knowledge Base upon the following historical data:

- frequency of occurrence,
- percentage distribution of magnitude of consequences,
- calculations of probability of negative outcome in future,
- risk exposure in conditional units.

This information shall be used by project managers to evaluate global project threats, considering the probability of occurrence and possible negative impact that can be compared with historic data from other projects. The author foresees that hidden threats and their outcomes, such as hidden costs, unobvious sources of time delays and customer dissatisfaction, will help inexperienced project managers in preparations for impediments inherited in the nature of globally distributed projects.

An example of Risk Barometer predictions for a threat of lacking experience and expertise in outsourcing projects can be seen on Fig.4.

Threat:	Lack of experience and expertise of the supplier team with outsourcing projects
Frequency of occurrence:	22 % or 8 of 36

Report: Historical Data for Risk Analysis

	Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
5 – Disastrous				12 %	12 %	12 %	12 %	12 %
4 – Significant	12 %		12 %			12 %	12 %	
3 – Moderate				38 %	12 %		12 %	
2 – Minor	12 %	25 %	12 %	-> 12 %	-> 25 %	12 %	->	12 %
1 – Negligible	->	-> 25 %	->	38 %		->	12 %	->
0 – None	75 %	50 %	75 %		50 %	62 %	50 %	75 %

The sign -> points the average outcome of the threat on a certain project success criteria

Report: Probability of the Negative Outcome (if the threat has occured)

	Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
5 – Disastrous	0 %	0 %	0 %	12 %	12 %	12 %	12 %	12 %
4 – Significant	12 %	0 %	12 %	12 %	12 %	24 %	24 %	12 %
3 – Moderate	12 %	0 %	12 %	50 %	24 %	24 %	36 %	12 %
2 – Minor	24 %	25 %	24 %	62 %	49 %	36 %	36 %	24 %
1 – Negligible	24 %	50 %	24 %	100 %	49 %	36 %	48 %	24 %
0 – None	75 %	50 %	75 %	0 %	50 %	62 %	50 %	75 %

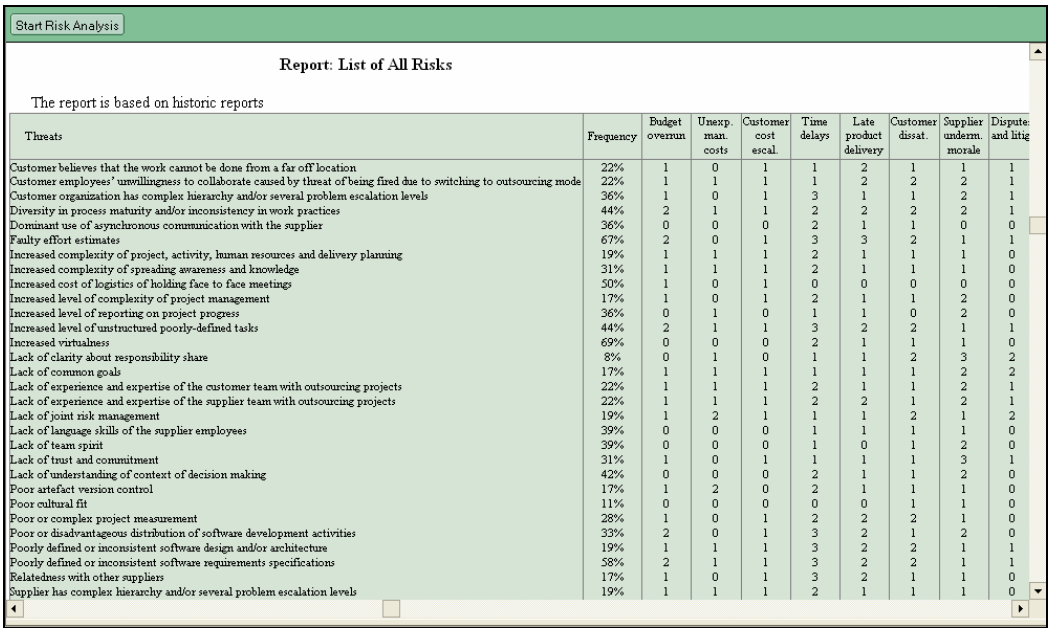
Report: Risk Exposure Level

Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
72	75	72	236	146	132	156	84

Risk Exposure evaluation [0; 1500].

Fig. 4. Example of predictions by the Risk Barometer: for a threat

Risk Barometer also provides an overview of all risks as seen on Fig.5.



Start Risk Analysis

Report: List of All Risks

The report is based on historic reports

Threats	Frequency	Budget overrun	Unexp. man. costs	Customer cost escal.	Time delays	Late product delivery	Customer dissat.	Supplier underm. morale	Dispute and litig
Customer believes that the work cannot be done from a far off location	22%	1	0	1	1	2	1	1	1
Customer employees' unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode	22%	1	1	1	1	2	2	2	1
Customer organization has complex hierarchy and/or several problem escalation levels	36%	1	0	1	3	1	1	2	1
Diversity in process maturity and/or inconsistency in work practices	44%	2	1	1	2	2	2	2	1
Dominant use of asynchronous communication with the supplier	36%	0	0	0	2	1	1	0	0
Faulty effort estimates	67%	2	0	1	3	3	2	1	1
Increased complexity of project, activity, human resources and delivery planning	19%	1	1	1	2	1	1	1	0
Increased complexity of spreading awareness and knowledge	31%	1	1	1	2	1	1	1	0
Increased cost of logistics of holding face to face meetings	50%	1	0	1	0	0	0	0	0
Increased level of complexity of project management	17%	1	0	1	2	1	1	2	0
Increased level of reporting on project progress	36%	0	1	0	1	1	0	2	0
Increased level of unstructured poorly-defined tasks	44%	2	1	1	3	2	2	1	1
Increased virtualness	69%	0	0	0	2	1	1	1	0
Lack of clarity about responsibility share	8%	0	1	0	1	1	2	3	2
Lack of common goals	17%	1	1	1	1	1	1	2	2
Lack of experience and expertise of the customer team with outsourcing projects	22%	1	1	1	2	1	1	2	1
Lack of experience and expertise of the supplier team with outsourcing projects	22%	1	1	1	2	2	1	2	1
Lack of joint risk management	19%	1	2	1	1	1	2	1	2
Lack of language skills of the supplier employees	39%	0	0	0	1	1	1	1	0
Lack of team spirit	39%	0	0	0	1	0	1	2	0
Lack of trust and commitment	31%	1	0	1	1	1	1	3	1
Lack of understanding of context of decision making	42%	0	0	0	2	1	1	2	0
Poor artefact version control	17%	1	2	0	2	1	1	1	0
Poor cultural fit	11%	0	0	0	0	0	1	1	0
Poor or complex project measurement	28%	1	0	1	2	2	2	1	0
Poor or disadvantageous distribution of software development activities	33%	2	0	1	3	2	1	2	0
Poorly defined or inconsistent software design and/or architecture	19%	1	1	1	3	2	2	1	1
Poorly defined or inconsistent software requirements specifications	58%	2	1	1	3	2	2	1	1
Relatedness with other suppliers	17%	1	0	1	3	2	1	1	0
Supplier has complex hierarchy and/or several problem escalation levels	19%	1	1	1	2	1	1	1	0

Fig. 5. Example of predictions by the Risk Barometer: for all threats

4.7. APPLICATION IN PRACTICE

Risk Barometer provides a general overview of the outcome of each threat and probability of its occurrence. Since global projects are so different and the extent of global factors may influence occurrence of global threats in particular circumstances, project managers may not ground their risk predictions only on personal experience. It is therefore recommended to use an experience-based approach to analyse global risks and monitor them on a regular basis among different projects in an organization.

Application of Risk Barometer is feasible in any global project despite its size and complexity. Global factors and threats inherited in the nature of globally distributed environment will not vanish if the project will long only a month or consider a well-known task that shall be performed by well-trained developers. Project managers shall use outcome predictions to see what kind of effect they may cause and report on real situation after the end of the project. Regular feedback will strengthen further predictions and statistical significance of the results.

4.8. CONCLUSIONS

Answering the demand of inexperienced project managers, the author offers an experienced-based risk-oriented approach to evaluate globally distributed software project outcome prediction. The developed Risk Barometer automates historic information generalization by statistical analysis of the gathered experience reports and facilitates an easy to use tool for risk evaluation and analysis. The Risk Barometer aims to acknowledge inexperienced project managers about possible negative events and their consequences that influence overall project success. This will also prevent hidden threats and help managers in preparations for impediments inherited in the nature of globally distributed projects.

Compensating the impediments of effective risk management such as inadequate management infrastructure and lack of systematic and repeatable methods to identify, analyze and plan risk management [CAR97], this approach answers the needs of practitioners. Indeed, Risk Barometer supports preclusion of risk-averse culture and provides a comprehensive mean to operate risk analysis in an easy to use manner.

5. THE EFFECT OF GLOBAL FACTORS AND THREATS

5.1. OVERVIEW

Despite the fact that existence of some of the global factors can be obvious – e.g. two different organizations distributed by national, organizational and temporal distance have to work together – the effect of global factors and resulting threats is often hidden for those who have never worked in globally distributed environment. Therefore, organizations shall get acquainted with various consequences and ways that global factors and threats may act before starting global collaboration.

The gathered experience-based information from 38 global projects in the investigated software house and other organizations in Latvia served as a ground for many empirical observations on globally distributed software development. In this chapter the effect of global factors and threats is discussed based on the global project outcome predictions provided by Risk Barometer.

The following conclusions are derived upon analysis of the gathered survey data:

- Most frequent global project threats;
- Rare threats from Latvian perspective;
- Specific threats that endanger
    - project budget,
    - schedule,
    - customer satisfaction and
    - developer team psychological comfort.

5.2. STATISTICS OF EFFECT OF GLOBAL THREATS

The detailed statistics of the effect of global threats can be seen on Fig. 4. The first column contains global threats, the second column contains overall risk exposure of each threat in conditional units (according to equation 1), the next column describes frequency of occurrence of the threats in the investigated globally distributed projects, and the following columns contain risk exposure of each pair [threat; consequence] (according to equation 2).

Probability of the threats can be summarized as follows: there is 1 doubtful threat, 10 unlikely threats, 14 possible threats, and 7 probable threats. However the effect of each threat shall be also taken into account. The results show that although the frequency of occurrence of these threats may be low, evaluation of the magnitude of their consequences is not in direct dependence and may be considerable. Overview of the most dangerous threats follows.

	RE in conditional units	Frequency of occurrence	Overall Outcome	Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
Increased virtualness	420	69%	608	40	36	20	192	112	104	92	12
Faulty effort estimates	825	67%	1231	205	32	96	282	253	154	141	68
Poorly defined or inconsistent software requirements specifications	688	58%	1187	165	60	90	301	221	186	112	52
Increased cost of logistics of holding face to face meetings	162	50%	323	65	45	84	23	30	30	40	6
Diversity in process maturity and/or inconsistency in work practices	525	44%	1194	169	99	99	236	187	168	170	66
Increased level of unstructured poorly-defined tasks	518	44%	1178	197	66	88	285	171	167	132	72
Lack of understanding of context of decision making	307	42%	731	48	42	33	162	120	120	173	33
Lack of language skills of the supplier employees	195	39%	499	49	14	21	121	63	91	105	35
Lack of team spirit	190	39%	487	21	21	21	113	49	56	171	35
Customer organization has complex hierarchy and/or several problem escalation levels	298	36%	828	68	30	62	254	113	62	175	64
Terminology differences	215	36%	596	48	24	48	162	99	85	122	8
Increased level of reporting on project progress	197	36%	548	24	55	48	100	76	38	183	24
Dominant use of asynchronous communication with the supplier	181	36%	503	23	23	15	210	100	69	48	15
Poor or disadvantageous distribution of software development activities	384	33%	1165	182	40	84	326	201	143	156	33
Lack of trust and commitment	251	31%	810	72	45	54	72	72	135	252	108
Increased complexity of spreading awareness and knowledge	240	31%	774	81	72	54	234	126	90	99	18
Poor or complex project measurement	244	28%	870	110	30	80	180	170	170	80	50
Customer employees' unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode	256	22%	1165	99	75	138	135	162	200	248	108
Lack of experience and expertise of the customer team with outsourcing projects	228	22%	1037	108	124	125	172	125	123	173	87
Lack of experience and expertise of the supplier team with outsourcing projects	214	22%	973	72	75	72	236	146	132	156	84
Customer believes that the work cannot be done from a far off location	170	22%	774	76	48	85	100	158	108	137	62
Poorly defined or inconsistent software design and/or architecture	235	19%	1238	126	56	115	285	227	199	101	129
Lack of joint risk management	199	19%	1046	142	154	56	100	126	156	128	184
Increased complexity of project, activity, human resources and delivery planning	154	19%	808	127	56	70	230	112	129	84	0
Supplier has complex hierarchy and/or several problem escalation levels	136	19%	717	98	72	56	225	70	98	84	14
Time zone difference	35	19%	185	0	0	0	101	28	14	42	0
Relatedness with other suppliers	166	17%	976	136	17	68	299	218	119	85	34
Lack of common goals	165	17%	973	102	66	116	102	102	119	216	150
Increased level of complexity of project management	132	17%	778	68	0	102	170	136	119	166	17
Poor artefact version control	131	17%	772	85	167	34	151	66	119	133	17
Poor cultural fit	30	11%	275	50	0	0	0	0	100	125	0
Lack of clarity about responsibility share	74	8%	924	0	66	0	99	99	231	264	165

Fig. 6. Statistics on effect of global threats

5.3. THE MOST DANGEROUS GLOBAL THREATS

5.3.1. Overview

The report on the major global threats is based on evaluation of risk exposure levels. The author describes the severity of each threat by characterizing its frequency of occurrence and effect on different project success criteria: budget, schedule, customer satisfaction and other. The offered tables contain outcome predictions based on the data from 38 globally distributed projects. Each cell refers to the probability of the threat to cause certain level of impact for certain project success criteria.

> E.g. there is a 25% probability that faulty effort estimates will cause significant impact (level 4) on late product delivery (see Table 4).

The collared cells indicate particular areas of concern, where impact evaluation values exceed 33% (1/3 of the projects) and related to moderate negative consequences or higher.

Outcome predictions also illustrate the developed Risk Barometer performance (in more detail described in chapter 4.6).

5.3.2. Nr. 1: Faulty effort estimates

Occurred in 69% of the investigated globally distributed projects. This threat causes serious impact on the project performance. Lacking awareness of global threats, project managers often ignore considerable sources of time delays brought by global distribution and involvement of multiple suppliers while performing effort estimation. At the end, faulty effort estimates lead to considerable budget overruns and affects time schedule. Neither customers, nor supplier teams are happy to work under budget and schedule pressure, which sometimes leads to disputes and litigations. Outcome prediction evaluation for faulty effort estimates is given in Table 4.

Table 4. Outcome predictions (threat 1)

Impact	Budget overrun	Unexpected manag.costs	Customer cost escal.	Time delays	Late product delivery	Customer dissatisf.	Team under-mined morale	Disputes and litigations
5	4%		4%			4%		
4	16%		8%	29%	25%	12%	8%	8%
3	41%	8%	20%	79%	58%	29%	29%	16%
2	70%	8%	32%	87%	83%	46%	50%	16%
1	74%	16%	32%	87%	87%	63%	54%	28%
0	25%	83%	67%	12%	12%	38%	46%	71%

5.3.3. Nr. 2: Increased virtualness

Occurred in 69% of the investigated globally distributed projects. This threat is caused by temporal and geographical borders and is inherited in the nature of globally distributed projects. Correlation analysis of the threats shows that virtualness is related with an increased use of asynchronous communication and lack of personal meetings, accordingly affecting the overall quality of collaboration. Accordingly the most affected area due to increased virtualness is time schedule. Outcome prediction evaluation for increased virtualness is given in Table 5.

Table 5. Outcome predictions (threat 2)

Impact	Budget overrun	Unexpected manag.costs	Customer cost escal.	Time delays	Late product delivery	Customer dissatisf.	Team under-mined morale	Disputes and litigations
5								
4				12%	4%	4%		
3	4%	8%		36%	24%	20%	16%	
2	8%	8%	8%	68%	36%	28%	32%	4%
1	28%	20%	12%	76%	48%	52%	44%	8%
0	72%	80%	88%	24%	52%	48%	56%	92%

5.3.4. Nr. 3: Poorly defined or inconsistent software requirements specifications

Occurred in 58% of the investigated globally distributed projects. Requirements specifications are sources of threat in most of the software projects. In global environment requirements specifications are often developed by a geographically distributed customer team without programming team involvement, and then sent off-site to one or several supplier teams. Distribution in time and space makes it complicated to communicate incomprehensible or inconsistent requirements. Therefore, the magnitude of consequences of inconsistent requirements is so severe. Outcome prediction evaluation for poorly defined or inconsistent software requirements specifications is given in Table 6.

Table 6. Outcome predictions (threat 3)

Impact	Budget overrun	Unexpected manag.costs	Customer cost escal.	Time delays	Late product delivery	Customer dissatisf.	Team under-mined morale	Disputes and litigations
5	14%		5%	10%	10%	5%	5%	
4	24%	5%	10%	39%	29%	15%	10%	
3	34%	15%	20%	68%	48%	44%	29%	14%
2	44%	15%	25%	87%	62%	54%	29%	14%
1	49%	25%	30%	97%	72%	68%	39%	24%
0	52%	76%	71%	5%	29%	33%	62%	76%

5.3.5. Nr. 4: Diversity in process maturity or inconsistency in work practices

Occurred in 46% of the investigated globally distributed projects. As a result of multiple partner involvement in the completion of a joint project, it always means facing some level of diversity in the way each partner works. However, partners often lack time for negotiation of detailed work procedures. Consequences caused by organizational diversity are therefore often ignored. As a result this threat increases the and as a result act in heterogeneity of the working environment and appears to be one of the major sources of disputes affecting customer and supplier physiological comfort, time schedule and accordingly budget. Outcome prediction evaluation for this threat is given in Table 7.

Table 7. Outcome predictions (threat 4)

Impact	Budget overrun	Unexpected manag.costs	Customer cost escal.	Time delays	Late product delivery	Customer dissatisf.	Team undermined morale	Disputes and litigations
5				6%	6%	6%	6%	6%
4	19%		6%	31%	25%	25%	18%	12%
3	44%	25%	31%	37%	44%	25%	30%	12%
2	50%	31%	31%	81%	56%	50%	55%	12%
1	56%	43%	31%	81%	56%	62%	61%	24%
0	44%	56%	69%	19%	44%	38%	38%	75%

5.3.6. Nr. 5: Increased level of unstructured poorly-defined tasks

Occurred in 43% of the investigated globally distributed projects. In the distributed environment of global projects, poorly-defined tasks that travel from one partner to another without an easy way to be discussed and understood cause huge problems for developers, affecting project time scale and budget, bringing customer and the supplier teams dissatisfaction and disputes.

Table 8. Outcome predictions (threat 5)

Impact	Budget overrun	Unexpected manag.costs	Customer cost escal.	Time delays	Late product delivery	Customer dissatisf.	Team undermined morale	Disputes and litigations
5	6%			12%	6%		6%	
4	12%	6%		31%	18%	25%	18%	
3	43%	12%	19%	62%	30%	37%	30%	12%
2	62%	18%	25%	87%	49%	43%	36%	24%
1	74%	30%	44%	93%	68%	62%	42%	36%
0	25%	69%	56%	6%	31%	38%	56%	62%

These and other threats are discussed from different perspectives in the next chapters.

5.4. GLOBAL THREATS: ENVIRONMENT SPECIFIC OR GENERAL?

Analyzing the nature of global threats the author concludes that the major threats in globally distributed projects have strong relatedness to the derived global factors. The factors that distinguish globally distributed software projects from the co-located projects by involving multiple suppliers facing various levels of diversity, such as geographic and temporal distribution, socio-cultural and linguistic diversity, along with the contextual diversity of the companies involved in the joint project bring lots of threats that are not familiar to in-house projects. Increased virtualness, diversity in process maturity, inconsistency in work practices, relatedness with other suppliers and other threats indeed characterize the nature of the global project environment that project managers usually face while collaborating over borders.

Nonetheless, the major global threats are not only environment specific. Threats such as faulty effort estimates, poorly defined or inconsistent software requirements specifications, and increased level of unstructured poorly-defined tasks are also met by the majority of in-house projects. However, the significance of consequences of these threats in the GSD projects is affected by the global factors. Therefore, these threats can more easily jeopardize a project and inter-organizational collaboration.

5.5. SOURCES OF BUDGET THREATS

Survey data separates findings related to threats causing overall budget overrun, unexpected management costs and customer cost escalation. Yet, all these threats have a negative effect on project budget despite the question regarding who pays for what.

Analysis of the budget threats shows that global threats are able to cause disastrous impact on project budget. However, the average evaluation of each threat doesn't exceed moderate consequences.

Analyzing the major sources of budget overrun, the author concludes that poorly established or managed processes (such as effort estimation, task partitioning, requirements engineering and design, life cycle process distribution, and risk management) require extra resources including possible rework. Processes such as project measurement, artefact version control, and the already mentioned ones, often require additional investments into infrastructure and tools. Lack of experience with outsourcing projects and lack of joint risk management lead to unexpected management costs that are not duly addressed from the very beginning of the project.

5.6. SOURCES OF SCHEDULE THREATS

Since time and space is the major media for distributed team collaboration, it naturally appears to cause considerable time delays. Survey data differentiates threats leading to time delays and threats that cause late product delivery, referring to the importance and significance of the effect of these threats.

Survey data analysis indicates that project schedule is the most affected project area in globally distributed software projects. Nearly every threat (94% of the threats) has been to some extent evaluated as sources of schedule overrun. Most of these threats have caused moderate impact in average among the investigated projects.

Average evaluation of the time delays caused by relatedness between the remote teams within complex supplier chains and poor distribution of the activities between these teams are evaluated as significant. Likewise budget threats, poorly established or managed processes require extra time resources. Lack of flexibility and adoptability accompanied by complex organizational structures, several problem escalation levels, diversity in process maturity and inconsistency in work practices, negatively affect project schedule. In addition, disputes, misunderstandings, and poor communication require extra time. Time zone difference and dominance of asynchronous communication lead to time delays and breaks in joint work.

Survey findings indicate that the major sources of schedule deviations are environment specific. Multisourcing, geographical, temporal, socio-cultural and organizational distances brings additional challenges for the teams to continue working within the planned time schedule. Although some of the threats act as chains of negative events and may have moderate or minor effect evaluation, these threats shall not be underestimated.

5.7. SOURCES OF TEAM UNDERMINED MORALE

Survey data differentiates customer dissatisfaction and supplier team's undermined morale.

There is a correlation between the customer dissatisfactions and budget and schedule threats. Budget overruns and schedule deviations naturally affects customer satisfaction. Survey findings show that customers are also dissatisfied with poor performance and outputs of the software development processes. The major sources of customer dissatisfaction according to the average evaluation among the investigated projects are lacking clarity about responsibility share and customer employees' unwillingness to collaborate caused by the threat of being fired due to switching to outsourcing mode. Deeper analysis of individual cases indicates that diverse and untrustworthy atmosphere has caused significantly negative influence on customer satisfaction.

Diversity, unwillingness to collaborate, lack of cognition from the customer and low trust and team spirit are the main reasons of supplier team psychological discomfort. Projects with low coherence show increasing problems with team morale. Possibly due to industrial background of the survey, the results point to dominating effect of the global threats on supplier team morale than customers.

Taking into account environmental peculiarities of the globally distributed projects, the findings indicate that global teamwork is affected by various threats. Achievement of team coherence requires effort. Global teamwork related findings are in more detail described in chapter 7.

5.8. THREAT INTERDEPENDENCES

The correlation analysis of the threats uncovered interdependencies between the related threats. These interdependences characterize cause-effect and kindred relations. Threats that partly or fully overlap with global factors (such as "Diversity in process maturity" and "Organizational diversity", "Poor socio-cultural fit" and "Socio-cultural differences") can lead to other negative events. The correlation analysis also shows that along with the independent occurrence some threats can form chains of negative events. However, in further analysis the author doesn't use these chains because they entangle risk identification for practitioners.

5.9. RARE THREATS FROM LATVIAN PERSPECTIVE

Some threats that are strongly emphasized in GSD related research and industrial literature however appear to be rare for the Latvian software developers. The most occasional threats are caused by political, temporal and cultural diversity.

Latvian suppliers have a limited prevalence in global market, focusing their effort on collaboration with Scandinavian and West European companies. Political diversity is rarely a problem between these partners since Latvian legislation is adequate to European Union principles.

Temporal distance has also limited outcome on project goals. The biggest time zones difference between European countries is two hours. Therefore, most of the project managers do not consider temporal distance as a threat. However, others report that "One hour difference leads to five instead of eight working hours overlap". One project manager emphasized, that "Even one hour is sufficient for a question that is sent shortly before the end of the working day, to be answered only the next day's morning". However, even these comments do not refer to problems comparable with those faced in collaboration e.g. over Atlantic.

Likewise the previous factors cultural diversity has been recognized as a rare problem for Latvian software development suppliers. World wide case studies show that putting together teams with different cultural backgrounds cause frequent misunderstandings that in some cases lead to more serious problems that have considerable impact on project performance. However, in contradiction to these studies, Latvian suppliers appear to avoid such experience because of sharing comparable cultural background with their partners.

Among other rarely met threats with frequency of occurrence less than 20 % are the following threats:

- poorly defined software design specifications (19%),
- lack of experience of the customer with outsourcing projects (19%),
- lack of joint risk management (19%),
- customer employees' unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode (19%),
- lack of common goals (17%),
- increased level of complexity of project management (17%),
- relatedness with other suppliers (17%),
- poor artifact version control (17%),
- lack of clarity about responsibility share (8 %).

Despite the fact that some of the threats have rarely occurred in the majority of the investigated global projects in Latvia, the historic information shows that the possible negative effect of these threats should not be underestimated.

Poorly defined software design specifications are rare because in contradiction to software requirements specifications that are most often developed without supplier involvement, design specifications in the majority of the cases are developed by the suppliers themselves. However, in other cases poor design specifications has had a very considerable negative effect on project performance, especially time schedule.

Projects where customers lacked experience in outsourcing projects, experienced frequent time delays and caused customer dissatisfaction. One project manager described the effect of this threat as follows: "Customer sometimes wrongly assumes that off-shored or outsourced projects in Latvia can be managed in exactly the same way as the local projects in their native country".

Projects that lacked joint risk management report on considerable increase in unexpected management costs, which are understandable – partners that do not work on project risks by joint effort tend to ignore the hidden costs.

Customer employees' unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode is one of the symptoms that indicate project failure. No collaboration can survive without commitment. This threat causes considerable negative effect on both customer satisfaction and supplier psychological comfort. One project manager admitted, that "indisputable evidence for this (threat) does not exist, however it may account for what appears to be an unwillingness or slowness by the customer to act on suggestions by the supplier for process improvements". Another project manager reported that once customer management noticed employee dissatisfaction and unwillingness to proceed, the outsourcing project was cancelled.

Relatedness with other suppliers along with poor project activity planning had negatively affected project schedule by bringing time delays and at the end causing late deliveries in a number of the investigated projects.

5.10. CONCLUSIONS

The results of the survey on 38 globally distributed projects with Latvian suppliers and foreign customers mainly from Scandinavia and Western Europe uncovered the frequency of occurrence and the effect of previously derived global threats. The collected survey data forms ground for outcome predictions that can be used by practitioners during the process of risk management. The achieved results are an attempt of the author to formalize the evaluation of effect of global threats, considering certain project criteria such as budget, schedule and team morale. Because even capable project managers that are inexperienced in GSD often struggle with uncertainty of the global projects, awareness of experience-based negative outcomes of global factors and threats is essential.

Despite the fact that some of the threats have rarely occurred among the investigated global projects in Latvia, the historic information shows that the possible negative effect of these threats should not be underestimated.

Analysis of global project outcome predictions shows that the major threats in globally distributed projects are not only environment specific. However, the significance of consequences of global threats in the globally distributed software projects is affected by the global factors – multiple team involvement, diversity of the teams involved in the project, distribution of the software development activities and other factors. Accordingly, if it is difficult e.g. to perform accurate requirements specification in in-house projects that often cause problems in the further phases of software development, then in global projects the

consequences of poor requirements specifications are many times more severe than for in-house projects that involve co-located cohesive team members.

These findings drives to conclusion that the major areas of concern that shall be paid careful attention in globally distributed software projects are as follows:

- Software process distribution – which work partitioning strategies are advantageous?
- Global software teamwork – how to make remote team members work effectively?
- Global project management – how to manage remote resources and processes?

Because of the unique pressures of globally distributed environment, organizations switching to outsourcing mode shall consider all pros and cons of global collaboration and get prepared by tailoring current processes and methods for distributed collaboration. The author offers findings of additional in-depth analysis of the software process distribution and global software teamwork as areas of major concern and emphasizes the effect of global factors and threats as an impediment for successful globally distributed collaboration. Complexities of global project management can not be appropriately evaluated due to limitations of this study on only supplier related threats. Observations of global project management shall especially address the prime contractor's role and is seen as future research work in this area.

6. SOFTWARE LIFECYCLE MANAGEMENT

6.1. OVERVIEW

The process of globalization introduces conceptual changes and increases complexity of software development by introducing collaboration interfaces between the distributed partners involved in completion of a joint project. Theory describes and practice illustrates examples of projects where increasing number of participants in one co-located team doesn't make the product delivery faster. Instead of making the process faster and easier, increased number of developers increases the number of problem sources. Likewise, process distribution between the geographically distributed teams makes it much more complicated.

However, lacking stabilized process models for outsourced projects and previous experience 1/3 of the investigated distributed projects suffered from poor or disadvantageous distribution of software development activities. Thus this chapter aims to illustrate lifecycle management and work partitioning among the investigated projects and discuss factors that affect partner choice and the effect of poor distribution.

6.2. COLLABORATION MODELS

The author conducted two studies based on survey data considering collaboration models and life cycle management based on experience of 19 global projects in the investigated company [SMI04] and 38 globally distributed projects from the last survey. Analysis of the project data helps identifying different collaboration models and evidences of their suitability. Collaboration models were analyzed according to the following criteria (see Table 9).

Table 9. Collaboration model evaluation criteria

Evaluation criteria	Description and possible values
Process distribution	Describes collaboration models by indicating which party was involved in what process.
Project success	Project managers' evaluation using scale: 1-10. Values are then categorized as unsuccessful projects (1-6), somewhat successful (7-8), and successful projects (9-10)
Evaluation of process distribution	Whether project managers have evaluated process distribution as poor or not.
Quality of requirements analysis	Whether SRSs are poor / inconsistent, or not.

Survey data analysis uncovered 19 different collaboration models. This indicates that conclusions on evaluation of process distribution cannot be used due to statistical insignificance. However, investigation of different collaboration models supports conclusions on types of collaboration and their variations based on the level of supplier involvement. The author has developed a classification model for different global collaboration types, which is based on four quadrants. The polar cases within the investigated collaboration models are full outsourcing and full partnership. These types of collaboration depend on the involvement and independence of process performance by the involved partners as visualized on Fig. 5.

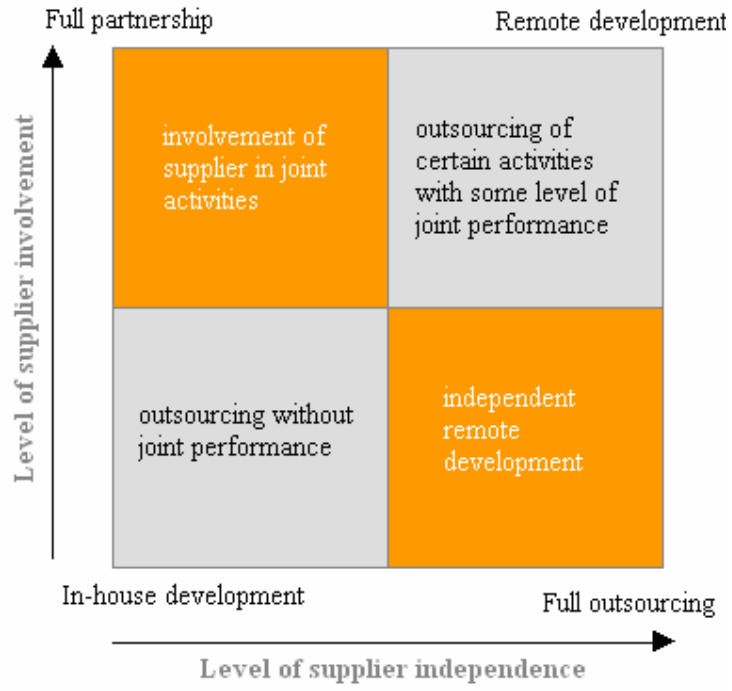


Fig. 7. Distribution of collaboration models according to the types

In fact, some of the collaboration models stand between certain types of collaboration.

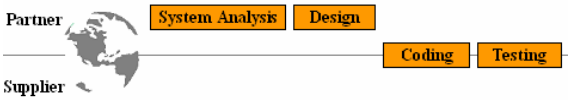
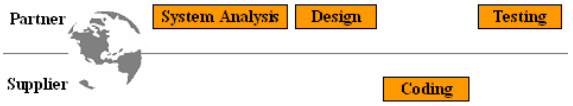
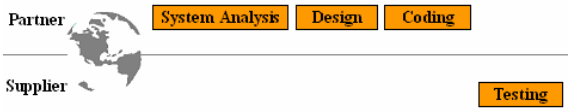
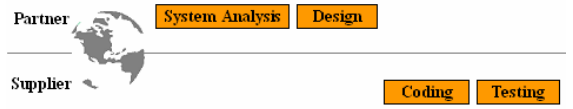
> E.g. there are projects with neither partnership nor outsourcing models, but something mixed with a tendency to more independence or more joint performance.

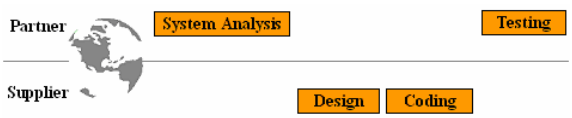
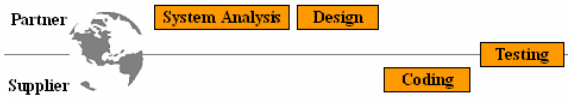
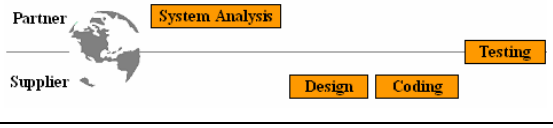
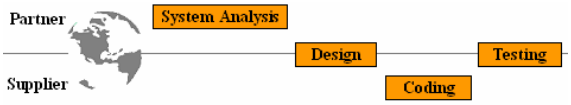
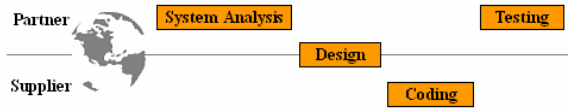
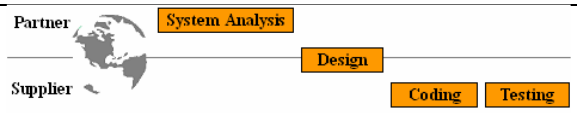
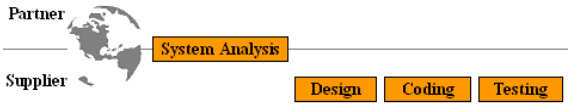
The author categorizes these collaboration models into four groups:

- Involvement of suppliers in joint activities;
- Outsourcing of certain activities with some level of joint performance;
- Outsourcing without joint performance;
- Independent remote development.

Table 10 describes different models of partner collaboration in the investigated globally distributed projects.

Table 10. Collaboration model evaluation

Nr.	Life cycles	No. of projects	Successful	Somewhat successful	Unsuccessful	Poor process distribution	Poor SRSs
Involvement of supplier in joint activities							
M1		3	✓				✓
				✓		✓	✓
				✓			
M2		1			✓		
M3		2		✓		✓	
				✓		✓	
M4		1		✓		✓	✓
Outsourcing without joint performance							
M5		2		✓			✓
					✓		
M6		1			✓	✓	✓
M7		5			✓		✓
			✓				✓
			✓				
				✓			
			✓				
M8		1		✓			
M9		3			✓		✓
				✓			
					✓		
M10		1		✓		✓	✓

M11		5		✓		✓	✓
					✓		
				✓		✓	✓
				✓			✓
				✓			
Outsourcing of certain activities with some level of joint activities							
M12		1	✓			✓	✓
M13		2			✓	✓	✓
				✓			
M14		3			✓		✓
					✓	✓	✓
				✓			
M15		2			✓		✓
					✓		✓
M16		1	✓				
M17		1		✓			✓
Independent remote development							
M18		1			✓	✓	✓
M19		1		✓			✓

The place of each of these collaboration models within the square of collaboration types can be seen on Figure 6.

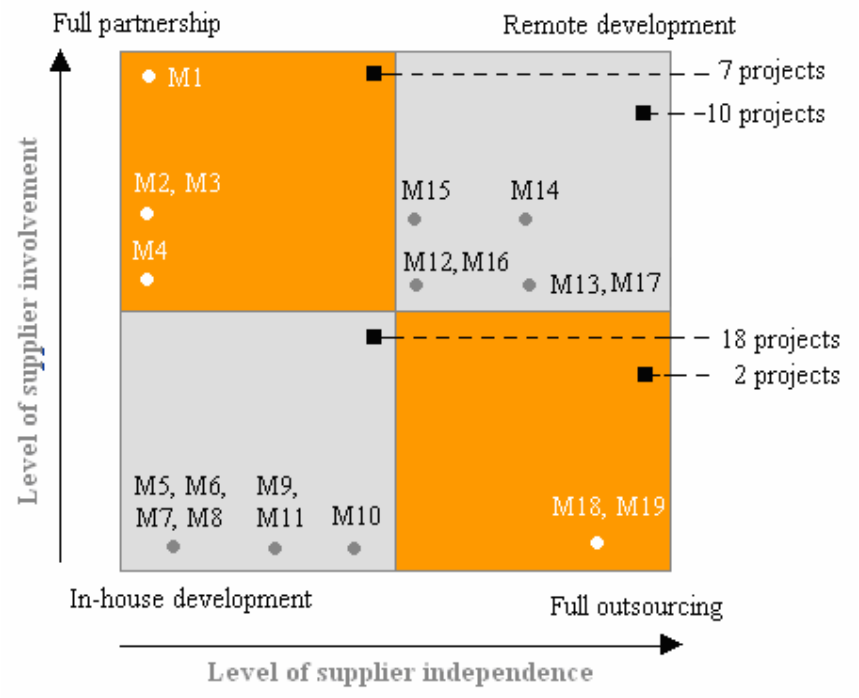


Fig. 8. The place of each model within the square of collaboration types

Motivated by market pressures, cost saving strategies and lack of awareness of global threats, the most frequently met collaboration type is outsourcing without joint performance. Accordingly, the most frequent collaboration model is sending coding activities to a remote supplier (followed by 5 projects – M5, M6, M7) or sending one or several other activities to a remote supplier (followed by 5 projects – M8, M9, M10, M11). Evaluating the success of this type of collaboration admittedly 40% of the projects reported poor distribution of the activities. Nonetheless, M7 is a pretender to be successful way to collaborate (followed by 5 projects) – no project reported poor distribution of activities and only 1 project was unsuccessful.

There are many projects with mixed distribution of activities – joint and independent performance. There are 6 projects with dominance of the customer independence (M2, M3, M4, M12, and M16) and 8 projects with supplier independence dominating (M13, M14, M17, M18, and M19). Low level of supplier distribution in 2/3 of the projects is reported as poor, and also supplier dominance is negatively reported by 3 out of 8 projects. However, M16 and M17 can be classified as successful collaboration type, but are validated only in 1 project.

Full outsourcing projects, especially the most extreme representatives of this type of collaboration, don't appear among the investigated projects, because there is no process distribution. Accordingly these projects do not fit the phenomenon under study. However, variations of independent remote development appear (followed by only 2 projects). Both cases were unsuccessful but didn't report on poor distribution of activities.

The last type of collaboration, polar to full outsourcing, prescribes fully joint performance between the partners (followed by 3 projects – M1). Success of joint collaboration in all activities is not indisputable.

The reasons for selection of lifecycle distribution are discussed in the following sub-chapter.

6.3. FACTORS AFFECTING LIFECYCLE DISTRIBUTION

Analysis of the collaboration models and project data helps to derive the following list of factors affecting lifecycle distribution in global projects:

- Motives for going global – the major motives for organizations to switch to remote software service supplier often determine the selection of lifecycle distribution. These motives can be as follows: software houses aiming to decrease costs, end customers aiming to purchase cheaper services, software houses aiming to gain extra knowledge or workforce, software houses aiming to gain proximity to the Baltic market;
- Level of experience with outsourcing.

Driving factors of going global often influence the future style of globally distributed collaboration during the project, which is illustrated by the following examples.

- Cost saving strategies

Software houses aiming to decrease costs choose outsourcing one or even several life cycle activities without joint performance due to extra investments that is required for joint collaboration, such as special tools, management effort and travelling expenses. These organizations expect work increments or pieces to be developed independently and be later integrated into the end product or delivered for further implementation.

- End customers aiming to purchase cheaper services

End customers aiming to purchase cheaper services either act through the internal information system (IS) or information technology (IT) departments, involve consulting companies or directly collaborate with remote suppliers. This type of collaboration can be both with and without joint performance.

- Software houses aiming to gain extra knowledge or workforce

Due to increasing costs or lack of knowledge and experience, some software houses are focusing their competence on certain lifecycle activities, such as requirements analysis, design and/or testing. Keeping the core activities and project management in-house, the other

activities are sent to an outsourcing provider. This type of collaboration often prescribes involvement of one or several remote suppliers in joint performance on the project.

- Software houses aiming to gain proximity to market

Software houses aiming to gain proximity to market often establish their offshore offices or joint ventures prescribing close collaboration and joint activities.

- Level of experience with outsourcing

Work partitioning within the projects with mixed (joint and independent) performance often depends on the experience of the customer with outsourcing projects. Many organizations start outsourcing because everybody does. They might have heard of tremendous success in decreasing costs and speeding time-to-market, however are not prepared to face the threats of collaborating over borders. Experienced and knowledgeable organizations invest in their collaboration by this increasing the probability of successful project performance.

- Attempts to perform similarly to in-house projects

Outsourcing and global software development is not a phenomenon. And yet not every company understands that globally distributed software development requires adjustment of processes and practices of the partners involved in the project. This may result in inconsiderate decisions regarding process distribution and end with poor performance.

6.4. EFFECT OF POOR DISTRIBUTION

- Extra documentation and effort for project management

Attempts to perform the same way as before switching to outsourcing mode are doomed to fail. Process distribution among the geographically remote partners requires careful planning, preparations and investments, additional tools to support software development. Independent performance requires extra effort for documentation to avoid inconsistency, misunderstandings, and rough transfer between the phases.

- Poor performance

Lack of proximity and next door closeness between the developers involved in the project makes it difficult to expect the remote developers acting as a joint co-located team without additional team building activities. The conducted multi-case study on trust in global teams (described in the next chapter) points out that although remote team members ought to form a joint team they consider distribution as team separator, and after all there might be

committed teams in each location and lack of teamness between them [MOE07]. Poor process distribution may result in strong relatedness between the perceived team members that do not associate themselves as a team, causing poor performance.

- Poor results of the software development processes

Lifecycle distribution and especially level of joint performance indicates the maturity of global collaboration between the partners. Such threats as poorly defined or inconsistent software requirements specifications often faced by the investigated projects (58%) and poorly defined or inconsistent software design or architecture (19%) indicate that the related software processes have faults especially regarding poor distribution.

However, 80% of the projects reported that requirements analysis has been performed without supplier involvement. This means that the customer team performs analysis, develops a requirements specification and sends it via email to a supplier, expecting a cheap, qualified and timely development. However, it almost never happens. According to the survey data, only 15% of the mentioned projects were successful.

6.5. RECOMMENDATIONS TO PRACTITIONERS

As earlier described, despite the popularity of the GSD topic, no research could determine the exact recipe for effective outsourcing performance [LOH95]. On the other hand, the results of the research described in this thesis show that there is a great variety of reasons and ways to outsource. Therefore, the exact recipe such as “one fits all” will not serve as a saving answer. Moreover, it is well know that cakes cooked by 10 people in accordance with one and the same recipe differ.

Nonetheless, conclusions on what has worked well and what didn’t work can be made:

- Sending independent pieces of work to a cheap supplier tend to fail;
- Lack of joint performance introduces sources of threat for transition of work products between the remote teams;
- Independent performance requires accurate documentation of requirements and tasks.

Interviews with experienced project managers and field observations form a set of practices proved to be effective in an industrial context of the Latvian software house under study. Supplemented by other industrial case studies and research findings, the recommendations for lifecycle management and work partitioning are as follows.

- Discuss preferable process distribution considering possible risks

Evaluate the risks associated with transition of work results between process input and output interfaces from one remote team to another, if not performed jointly and minimize hidden threats of misunderstandings and time delays by considering involvement of the remote team members in joint activities.

Discussions between the partners are essential for successful collaboration.

- Consider joint performance and close collaboration

Remote supplier involvement in the on-site activities of the customer or partner, such as requirements analysis and design, helps to eliminate the unpredictable outcome of transition between the project phases. GSD supplier practitioners recommend sending a representative to the customer/partner side for certain activity performance, or constantly for the entire project to provide effective communication and awareness of work results and associated problems.

- Establish common philosophy and approach

To mitigate diversity between the partners, each project shall establish common philosophy and approach. This is related to work practices, as well as perception of certain aspects of lifecycle management, such as quality, process inputs, outputs and entry/exit criteria etc. Due to differences in education and background the perceptions of the remote partners may vary. A common process workshop can be used to create this [DIN04].

Establishing written procedures is feasible in large projects.

- Pay attention to flexibility and adaptability

Diversity and inconsistency, in other words heterogeneity of the teams and their backgrounds may lead to various problems regarding collaboration. Despite the fact that organizations are naturally resistant to change, those that involve subcontractors should experiment and adjust the global product delivery models by decreasing the processes and interaction layers that “steal” time and consider changes focussing on improvements that would enable effective cooperation of distributed teams.

- Plan and perform small increments

Threats such as lack of trust and commitment between the remote teams, lack of belief in ability to perform from a far off location, and other threats may negatively affect teams working in a globally distributed environment. Planning and performing small increments

with frequent deliverables will provide supplier a chance to demonstrate ability to perform and assure project success.

- Improve performance during the project, if it is lower than expected

Ineffective global project and remote team members performing below the expected productivity level is more than just a headache. Yet practice shows that productivity problems can be caused not only by poor performance and lack of knowledge of the supplier. Root of many problems is inappropriate project lifecycle management.

Explore the actual sources of problems and mitigate them on a continuous basis. Solving problems by joint effort improves mutual trust. Improvements initiated by suppliers provide a vision of competence and by this improves customer attitude, trust, and commitment.

6.6. CONCLUSIONS

Analysis of the survey shows that 1/3 of the investigated distributed projects suffered from poor or disadvantageous distribution of software development activities. Such threats as poorly defined or inconsistent software requirements specifications often faced by the investigated projects (58%) and poorly defined or inconsistent software design or architecture (19%) indicate that the related software processes have faults especially regarding process distribution.

Survey data analysis uncovered a great variety of collaboration models. The author derives 4 types of collaboration dependent on the level of partner interdependence. The investigated collaboration types vary between full outsourcing and full partnership categorized into 4 major groups: involvement of supplier in joint activities; outsourcing of certain activities with some level of joint performance; outsourcing without joint performance; independent remote development. The major reasons for selection of process distribution are motives for going global and level of experience with outsourcing projects.

Due to market pressures, cost saving strategies and lack of awareness of global threats the most frequent collaboration type is outsourcing without joint performance. Many organizations start outsourcing because everybody does. They might have heard of tremendous success in decreasing costs and speeding time-to-market, however are not prepared to face the threats of collaborating over borders. And yet, practice shows that attempts to perform similarly to in-house projects leads to inconsiderate decisions regarding process distribution and failure.

Although, there is no undisputable evidence of the relation between collaboration models and project success based on the survey data, interviews with experienced project managers show that process distribution matters. However, many of them admit that suppliers have also limited influence on the work partitioning and are often slaves of incompetent decisions.

7. GLOBAL SOFTWARE TEAMWORK

7.1. OVERVIEW

A global software team is a team whose members collaborate on a common software project while working across geographic, temporal, cultural, and organizational boundaries to accomplish an interdependent task. A global software team can also be characterised as a Virtual Team [MAR04]. Team cohesion is recognized as a factor that influences team productivity [BOE00], in other words, teamwork effectiveness, and accordingly the success of the project.

However, being a "real" team is not an easy question. A "real" team is characterized by the following behavioural qualities [CAR99], [SAL05]: "real" team members are perceived to be a team by its members, has collective responsibility for products, share responsibility for managing its work through back-up behaviour, has a common goal or set of tasks, shares its rewards and other qualities. Nonetheless, related studies show that global software teams satisfy relatively few of these [CAR99].

Survey on 38 globally distributed projects show that lack of team spirit is faced in 39% of the investigated projects, but lack of trust and commitment in 31% of the projects. Trust is a recurring problem in global teams, because of geographical, temporal, organizational, cultural and political differences among the team members. Trust functions as the glue that holds and links virtual teams together [KAN02]. In addition to trust, effective communication, coordination and cooperation (abbreviated as the 3Cs) form the stronghold for teamwork.

The topic of dispersed teams is a new ground [CAR99], especially in relation to software teams. The performed related literature overview concludes that existing researches do not describe in-depth the importance of trust in global software development and understanding of what leads to lacking trust, and the effect of lacking trust on teamwork. This is why, investigation of global software teamwork addresses knowledge and experience outside of the software engineering field focusing on e.g. sociological and psychological aspects of co-located and virtual teams.

Because lacking trust is recognized as a major reason for loss of team cohesion and impediment of building a "real" team, the author believes that understanding problems related to lacking trust in global software teams will also provide insights in reasons and effects of poor communication, coordination, cooperation driving to loss of team cohesion. The findings presented here are based on an empirical multi-case study in four software projects in the

investigate software house that was conducted in cooperation with a senior researcher from SINTEF, Norway and reported in a research paper [MOE07]. Additional practices related to communication, coordination and cooperation among the remote global software team members are derived from interviews with experienced project managers and literature analysis as described earlier.

7.2. MULTI-CASE STUDY

Four globally distributed software projects that faced trust problems were chosen for in-depth investigation. Conclusions of this multi-case study are based mainly on qualitative interviews and results from post-mortem meetings held during and at the end of the project. A post-mortem meeting [DIN05] focuses on describing project benefits and problems, main reasons for both benefits and problems are then explored by performing a root-cause analysis. The results of post-mortem meetings in the investigated projects supported the root-causes of problems related to trust. Project problems were also recorded using previously developed problem checklists – lists of global threats. To avoid bias and misunderstandings, conclusions on the case studies were then sent to the interviewed team members for approval.

The following are characteristics of the projects under study:

- Project A is an ongoing software product development and maintenance project that started in 1995. It involves a customer from Germany, direct supplier from Germany (3 team members) and Latvian supplier team (5 team members). The size of the project is 46080 hours. Data sources: interviews with current project manager, previous project manager and one developer; problem checklists.
- Project B is a software product development project that started in 2002 and ended in 2006. It involved a customer from the UK, direct supplier from the UK (13 team members), and Latvian supplier team (16 team members). The size of the project is 40480 hours. Data sources: interviews with project manager and 3 team leaders; post-mortem analysis and problem checklists.
- Project C is a software pilot product development that started in 2006 and was not prolonged afterwards. It involved a customer from Sweden and Latvian supplier (3 team members). The size of the pilot project is 320 hours. Data sources: interviews with project manager and problem checklists.
- Project D is an ongoing software product development project that started in 2005. It involves a customer from Norway (2 team members), and a Latvian supplier that is divided into two remote teams – Riga team (6 team members) and a remote programmers team from

a small town situated around 250 km from the city (5 team members). The size of the project is 1460 hours. The attention in this case study was focused particularly on collaboration between two separate teams within one country and one organization not separated by country borders. Data sources: interviews with project manager; post-mortem analysis and problem checklists.

7.3. FACTORS AFFECTING TEAM PERFORMANCE

Virtual and global team research literature analysis describes several factors that lead to lack of trust between team members that are also related to team communication, coordination and cooperation. These are the following:

- Problems related to cognitive-based trust [KAN02]

Cognitive dimension of trust can be build upon competence, reliability, and professionalism of the team members. Without awareness of the remote team members' background, education and competence development of cognition-based trust is impossible. This factor is also affected by the threat of lack of belief in ability to perform from a far off location.

- Lack of face-to-face meetings [BHA06], [CAR99], [PIC03]

Face-to-face collaboration is considered irreplaceable for both developing and repairing trust in virtual teams. It is well known that "trust needs touch". If there is no face-to-face communication in a virtual team, this tends to hinder effective communication.

- Too little communication [JAR04]

Virtual teams in a low trust situation need frequent communication to increase the level of trust. In a distant collaboration frequent communication is important for providing constant confirmation that team members are still there and still working.

- Unpredictability in communication [PIC03]

Inexperienced virtual team members may experience anxiety or trust decline due to negative interpretations of silence or delays associated with temporal distribution.

- Poor socialization [JAR99], [KAN02]

Socialization is more than just face-to-face meetings. Webster's dictionary defines socialization as adoption of the behavior patterns of the surrounding culture. In the globally distributed environment where remote team members have differences in

national, social, cultural, organizational, etc. backgrounds, socialization plays a very important role. Socialization strategies may help managers develop trust also in virtual teams.

- No conflict handling [KAR98], [JAR99], [KAN02]

It is hardly possible to collaborate in global environment without conflicts, and it is often difficult to maintain trust when conflicts among team members emerge. Accordingly, missing joint conflict handling is a threat against building and maintaining trust.

- Increased monitoring [PIC03], [JAR04]

The use of behavioural controls, such as having members file weekly reports and assigning specific tasks, has been found to be associated with a decline in trust among virtual team members. Also too much communication might raise suspicions of team members that others are monitoring them and this decreases the trust.

- Poor socio-cultural fit [DUA01]

Since culture cannot be visually recognized, its nature can be linked with a hidden iceberg. Culture affect people's assumptions, behaviours, and expectations about teamwork and diversity in leadership practices, work habits, and team norms tend to decrease trust among the remote team members.

These reasons of lacking trust were validated in the four investigated projects within the multi-case study. The occurrence of these can be seen in Table 11.

Table 11. Reasons for lacking trust in the project

Reason for lacking trust	Projects			
	A	B	C	D
Cognitive-based trust	✓	✓		✓
Poor socialization	✓	✓	✓	✓
Missing face-to-face meetings	✓	✓	✓	✓
No conflict handling		✓	✓	✓
Too little communication		✓	✓	✓
Unpredictability in communication		✓		✓
Increased monitoring	✓	✓		✓
Poor socio-cultural fit	✓	✓	✓	✓

The described multi-case study shows that poor socialization, lack of face-to-face meetings and poor socio-cultural fit were reported by all the projects. Lack of face-to-face meetings and poor socialization are probably related since it is difficult to socialize if you seldom or never meet. Poor socio-cultural fit may also be strengthened due to lack of face-to-face interaction and poor socialization.

Other symptoms and factors related to lacking trust and having effect on teamwork that were uncovered during the interviews are lack of language skills, organizational differences, and belief that the work cannot be done from a far off location. Lack of language skills also leads to socialization problems and decrease in communication, because employees with poor language skills tend to be afraid to speak over the phone. Inconsistency in work practices may lead to a lack of cognitive-based trust, misunderstandings and again cause increased monitoring. Involvement of unenthusiastic employees who lack previous experience in outsourcing projects can lead to a belief that the work cannot be done from a far off location. This negatively affects mutual socialization, communication and trust. Increasing the number of collaboration partners involved in the project results in more complex communication, coordination and cooperation. This again increases the number of sources of threat for effective teamwork. Geographic distribution leads to increased virtualness, communication problems, troubled socialization, and knowledge and awareness share. Level of organizational fit characterized by diversity in process maturity and inconsistency in work practices acts as a counterforce for shared environment development. Team members who do not share background and work habits seem to have less commitment and team cohesion.

7.4. EFFECT OF POOR TEAM COHESION

The effect of lacking trust based on related virtual team literature analysis is as follows.

- Decreased information exchange and feedback [BAN01], [DIR01], [SAL05]

  A low level of trust is associated with suspiciousness of information, and therefore decreased information exchange and feedback.

- Competition and not cooperation [DIR01], [BAN01]

  If one does not trust a partner, it might be difficult to work toward common goals and it is likely that the employees will pay more attention to competitive motives and not to cooperation, and even avoid participation because they feel insecure.

- Self-protection [DIR01]

If a team member does not trust the manger, he or she finds it worrisome to behave as expected; and the management's request is likely to exert a much weaker effect on the individual's behaviour, as he or she diverts resources to self-protection. This hinders the team leader from effectively managing the team.

- Doubt negative feedback from manager [DIR01], [SAL05]

When there is a negative feedback from a manager with low trust, it is likely that the employee will doubt the accuracy of the feedback.

- Relationship conflict [DIR01], [SAL05]

Under low trust, task conflict within a group is interpreted negatively and subsequently results in relationship conflict.

- Individual goals over group goals [DIR01]

In contradiction to shared goals and responsibility that characterizes a "real" team individuals under low trust direct their efforts toward individual goals, instead of the group's goals.

- Team not self-correcting [SAL05]

Low trust decreases the mutual performance monitoring, which means the ability to develop common understandings of the team environment and possibility to accurately monitor team member performance. This is essential to identifying mistakes and lapses in other team members' actions, and providing feedback regarding team member actions to facilitate self-correction.

- Not shifting workload among members [SAL05], [CAR99]

Decrease in the mutual performance monitoring will again affect the backup-behaviour. This is the ability to anticipate other team members' needs through accurate knowledge about their responsibilities. This includes the ability to shift workloads among members to achieve balance during high periods of workload or pressure. Inability of team members to work together on tasks that are interdependent is also recognized as an impediment of "real" team achievement.

- Productivity and quality decrease [BAN01], [DIR01], [SAL05]

Since the lack of trust reduces team performance this reduces the productivity and quality.

The occurrence of these negative outcomes was validated among the four investigated global projects (see Table 12). The multi-case study also uncovered additional effects of lacking trust:

- Increased monitoring

Recognized as one of the reasons of lacking trust it is also found to be an effect of lacking trust by this forming a closed loop. Managers lacking trust and belief in ability of the remote team members to perform tend to increase monitoring of their performance that can grow into a pressing control.

- Undermined morale of the team members

Lack of trust between the team members is found to have a negative influence on psychological comfort of individuals. Undermined morale of the team members is obviously a serious impediment of team cohesion.

- Threat of project cancellation

Multi-case study has also uncovered that lack of trust cannot be associated with a long term relations and therefore, may lead to project cancellation.

Table 12. The effect of lacking trust

The effect of lacking trust	Projects			
	A	B	C	D
Decreased information exchange and feedback	✓	✓		✓
Competition and not cooperation	✓		✓	
Self-protection	✓			✓
Doubt negative feedback from manager	✓	✓		✓
Relationship conflict	✓		✓	
Individual goals over group goals	✓		✓	
Team not self-correcting				✓
Not shifting workload among members				
Productivity and quality decrease	✓	✓	✓	✓
Increased monitoring	✓	✓		✓
Undermined morale of the employees	✓	✓	✓	✓
Threat of project cancellation	✓		✓	

The effect of lacking trust illustrates symptoms of poor teamwork and loss of team cohesion. Likewise related studies the described multi-case study uncovered that lacking trust may cause significant problems with performance of the team members. All of the studied projects reported that lack of trust resulted in a decrease in quality and productivity. This indeed proves the importance of trust for overall project performance.

The other frequently reported effects of lacking trust were decreased information exchange and feedback, and team members doubting negative feedback from their manager.

Issues such as team members not shifting their workload and not self-correcting were barely mentioned or not mentioned at all by the interviewed project members. This can be explained by the problem with information exchange and feedback, self-protection, competition and lack of cooperation. Because of these problems the team probably never had the chance to consider shifting their workload and self-correcting.

This and other comments point that although remote team members ought to form a joint team, a "real" team, they consider distribution as team separator. After all, the investigated projects demonstrate that there might be committed teams in each location and lack of team spirit between them.

7.5. RECOMMENDATIONS TO PRACTITIONERS

Building a "real" team in globally distributed environment appears to be particularly difficult. Recipes that work for co-located teams can hardly be helpful in global teamwork, due to environmental factors such as barriers associated with organizational, socio-cultural, geographic, temporal, etc. diversities between the remote team members and collaboration strategies that often prescribe decrease of project costs. Recommendations for improving global teamwork derived from the multi-case study findings, research literature and interviews with experienced project managers from the investigated software house are the following:

- Establish common philosophy and approach

  Create a common understanding of the work process, and how to cooperate in this process. This can be achieved by creating some common process elements. A common process workshop can be used to create this [DIN04].

- Share common goals

  Communicate expectations early and establish initial rules, in the form of a contract or trust structure, to spell out performance parameters for the team as a whole and for individual team members.

- Develop awareness among the team members

The effect of lack of awareness about team members' background diversity shall not be underestimated. Go through the list of "reasons" and "effects" of lacking trust, discuss this early with the team, and identify actions to meet these threats. Do not start collaboration unprepared.

- Provide wise coordination

Dealing effectively with global factors requires much effort and a deep competence in what may be labelled "distributed organizing" – the capability of operating effectively across the temporal, geographical, political, and cultural boundaries. However, the coordination processes shouldn't transform into pressuring monitoring and control. Decentralize part of the project management duties and coordinate the remote performance with mutual trust.

- Invest in one or several face-to face meetings

Since "trust needs touch" face-to-face meetings are considered irreplaceable for both developing and repairing trust also in virtual teams. Meetings in person improve communication. One project manager explained: "Regular personal meetings are very necessary. When our travelling was restricted by the customer due to project budget economy, the relationship started to turn for the worse".

- Invest in socialization

Managers shall consider investing in socialization activities for the whole team together. Similarly to team-building seminars implemented for co-located teams, team-building activities shall be organized to improve team cohesion in global software teams as well. Socialization strategies may help managers develop trust. Team members shall travel to remote sites to engage in team-building activities to engender lasting trust. Cultural diversity shall be addressed and discussed in socialization meetings.

- Provide effective teamwork infrastructure

Invest in groupware packages to provide remote team members with effective means of communication and compensate lack of personal contact during the project. Develop a 360º view by establishing a team intranet; facilitate publishing and updating individual, team, status and task information; encourage personal touches including personal pages.

Provide lightweight and reversible project management tools, including performance monitoring rules, to avoid stress and suspicion of being controlled.

- Encourage regular communication

Virtual teams in a low trust situation need frequent communication to increase the trust level. While lacking next door closeness, frequent communication is important for providing constant confirmation that team members are still there and still working. The frequency and predictability of communication, and the extent, to which feedback is provided on a regular basis, improve communication effectiveness leading to higher trust and improving team performance.

7.6. CONCLUSIONS

The conducted multi-case study explores the reasons and effects of lacking trust from teamwork perspective and is the first study addressing trust problems in global software teams. The study proves the importance of trust for overall teamwork and concludes that lack of trust can drive to many impediments of successful team performance, such as decreased productivity and low quality, undermined morale, relationship conflict, project cancellation, and many other problems that form a counterforce for building a "real team". However, as simple as it sounds, building trust in a global software team is not a trivial thing, especially when organizations seek to decrease their costs by outsourcing. Besides, activities implemented in co-located teams are often hard to establish in the geographically, temporally, organizationally and culturally distributed environment. Trust in a global team requires more than a team-building seminar and shall be maintained throughout the project.

An important finding of this study refers to the question of existence of "real teams" in globally distributed environment, since it is found that remote members consider global distribution an impediment for team cohesion. Moreover, there might be united teams in each location and lack of team spirit in between. Accordingly, organizations developing software in a global environment shall take into account two alternatives:

1. Project is developed by multiple teams with corresponding consequences:
    - Time-consuming communication;
    - Limited information exchange and feedback;
    - Possible competition instead of cooperation;
    - Lack of self-correcting action and workload shifting among the teams;
    - Productivity and quality decrease in comparison with a co-located team;
    - Relationship conflicts and dissatisfaction.
2. Organization invests in building a "real team" by:

- Developing awareness among the team members;
- Establishing common philosophy and approach;
- Providing wise coordination instead of pressing monitoring and control;
- Investing in face-to face meetings and team socialization;
- Providing effective teamwork infrastructure;
- Encouraging regular communication and personal touches.

8. GSD KNOWLEDGE BASE

8.1. OVERVIEW

To support sustainable process improvement and organizational learning, knowledge and experience from global projects and related research studies has been accumulated for further utilization in a database that is developed according to the concepts also referred to as Experience Factory [DIN00] or Knowledge Repositories [LIE98].

Knowledge repositories are defined as an "on-line computer-based storehouse of expertise, knowledge, experiences, and documentation about a particular domain of expertise. In creating a knowledge repository, knowledge is collected, summarized, and integrated across sources" [LIE98].

The author developed a GSD Knowledge Base called “PraDis” (risks and practices for distributed environment”) as a Lotus Notes application. Lotus Notes is selected according to the corporate software technological platform within the investigated software house. The main users of the Knowledge Base are global project managers and SPI consultants also called “coaches”. The interaction of users with the Knowledge Base can be also described with the following context diagram (see Fig.9).

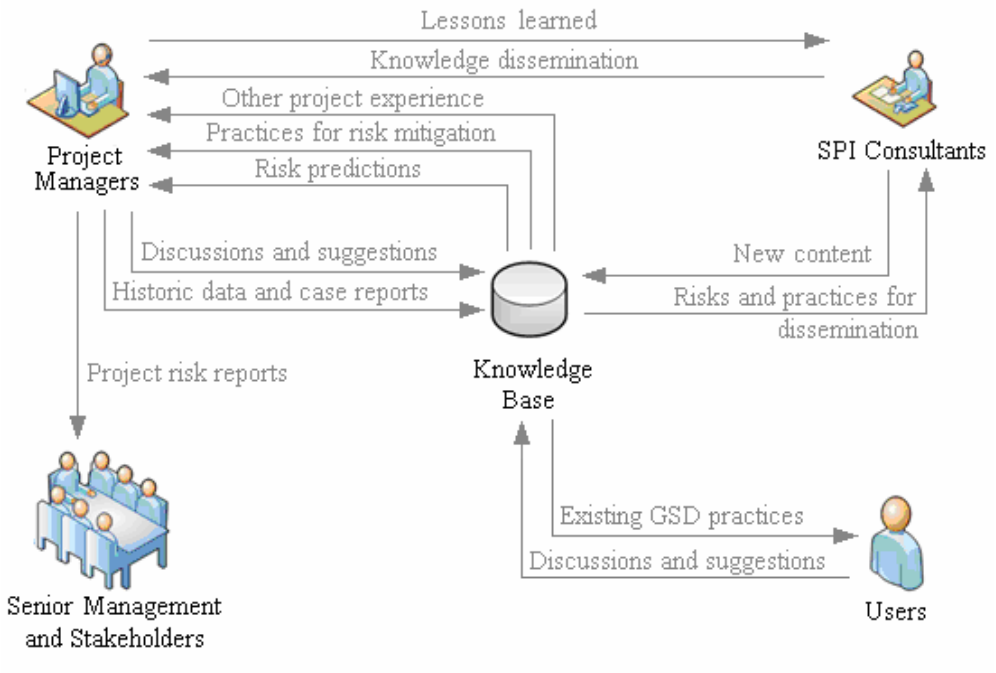


Fig. 9. Knowledge Base and its users

Project managers are the major target group. Project managers shall read the knowledge and experience items, suggest new items, suggest and participate in discussion topics, use templates, read and add case study reports and historic data from previous projects. Other users shall use Knowledge Base as a source of existing practices from GSD projects. Though senior management and stakeholders are not seen as a user target group, reports provided by the Knowledge Base shall be used by project managers for reporting on project risks. Project SPI consultants shall perform maintenance of the content: monitor discussions, new item, improvement proposals and case study reports, gather, pack and integrate new items into the Knowledge Base, and disseminate knowledge by different means.

8.2. CONTENT AND FUNCTIONALITY

The Knowledge Base provides the users various functions as follows (see also Fig.10):

- Description of global factors and threats;
- Experience generalization – practices to overcome global factors and threats;
- Threat outcome predictions with the help of so called Risk Barometer (see also chapter 4.6);
- Item categorization and search facilities;
- Quality document templates addressing global specifics;
- Case study overviews;
- Discussions;
- Personal folders and notifications;
- New knowledge and experience item proposal;
- Suggestions for Knowledge Base improvement.

Global practices, the main content of the Knowledge Base, were derived through grounded theorizing as described in chapter 3.2.1. The practices were generalized, tested for repetition, assigned to the mitigated threats and integrated into the Knowledge Base by the author. Grounded theorizing provided 32 supplier related practices. Some additional practices were then added from personal experience of the author from conducting post mortem reviews and risk management consultancy in globally distributed software projects in the investigated software house, and additional literature reviews and survey data analysis. This also refers to the process of sustainable knowledge and experience transfer. The total number of practices in the Knowledge Base is 37.

Global factors and threats accumulated in the Knowledge Base help project managers to identify sources of risk in global projects during risk identification activities. Historic data and threat outcome predictions support risk analysis and evaluation. Knowledge and experience items containing the proven practices also form recommendations for risk mitigation activities.

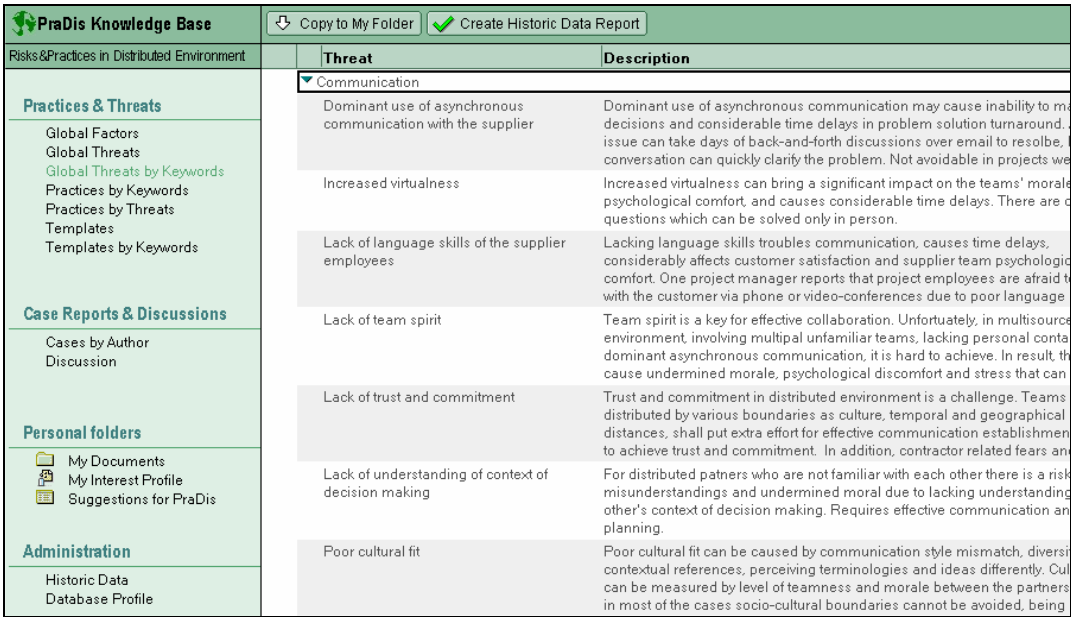


Fig. 10. Knowledge Base interface

An example of a threat entry is seen on Fig. 11.

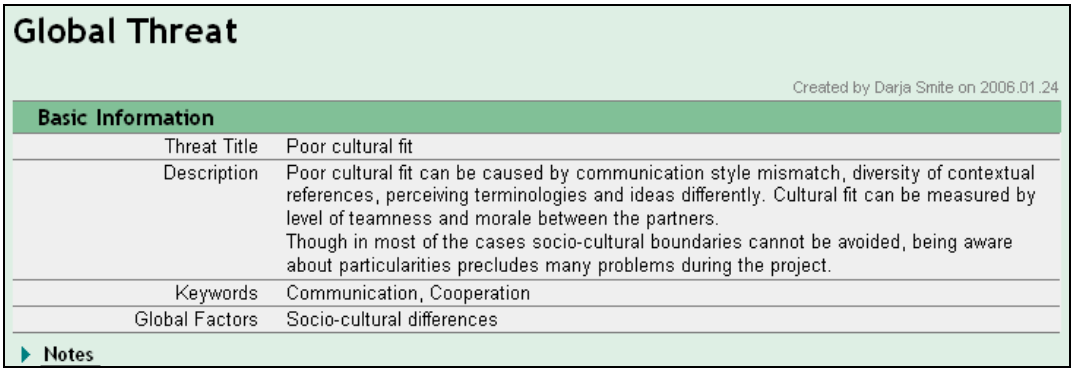
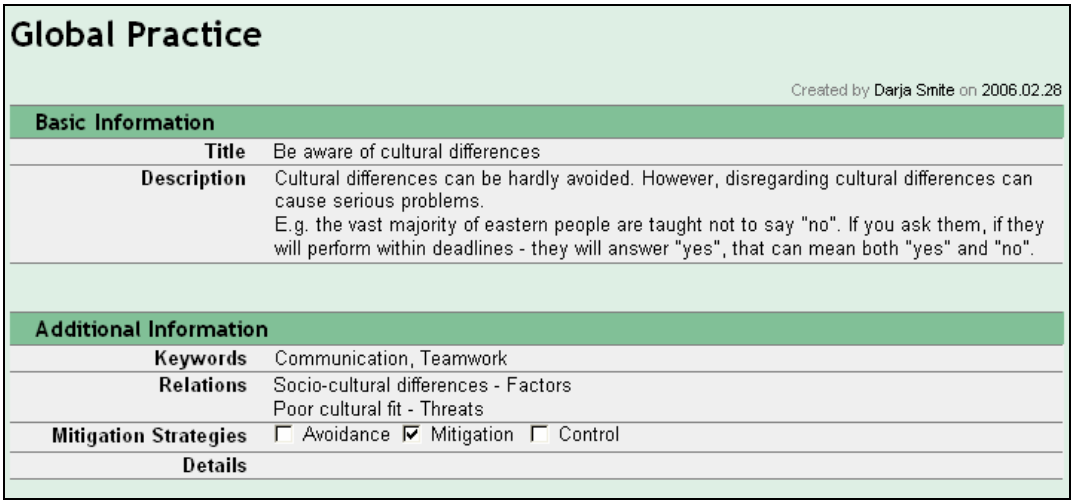


Fig. 11. Example of a global threat's entry

An example of a practice entry is seen on Fig.12.



Global Practice

Created by Darja Smite on 2006.02.28

Basic Information	
Title	Be aware of cultural differences
Description	Cultural differences can be hardly avoided. However, disregarding cultural differences can cause serious problems. E.g. the vast majority of eastern people are taught not to say "no". If you ask them, if they will perform within deadlines - they will answer "yes", that can mean both "yes" and "no".

Additional Information	
Keywords	Communication, Teamwork
Relations	Socio-cultural differences - Factors Poor cultural fit - Threats
Mitigation Strategies	☐ Avoidance ☑ Mitigation ☐ Control
Details	

Fig. 12. Example of a global practice's entry

For more detail on Knowledge Base functionality see also user guide in Appendix A.8.

8.3. OBJECTIVES FOR KNOWLEDGE BASE IMPLEMENTATION

Practice and related studies show that Quality systems and its components are rarely used without the following preconditions of implementation and maintenance of such systems:

- Top management involvement [PIT03];
- Employee involvement [MOE04], [MOD05];
- Usefulness of the content [DYB04], [ROG03];
- Training and promotion;
- Corporate culture.

An exploratory study has been performed to learn about attitude of the investigated company's employees towards existing quality system to learn how to implement new software process improvement initiatives [SMM06]. The study uncovered low top management interest and employee involvement in process improvement initiatives, and failure of implementing the current quality system. It also referred to a questionable percept of the Quality Department initiatives by the employees.

With respect to these findings, the new Knowledge Base for globally distributed projects has been developed according to the following requirements:

- Employee involvement and usefulness of the content – experience was based on the interviews with the experienced global project managers;

- Training and promotion – Knowledge Base is introduced to project managers in the annual internal quality seminar and promoted via email notifications.

Functionality of the Knowledge Base also aimed to be designed as easy to use for intuitive operation without user training. It follows basic principles of Lotus Notes applications and is not overloaded with hidden or complex elements.

8.4. APPLICATION OF THE KNOWLEDGE BASE

Currently, the Knowledge Base is accessible in user mode by any employee from the investigated software house; however, the major audience are global project managers. It was introduced to project managers during internal training seminars. Additional promotion was done through emailing the news after major changes. Knowledge Base administration mode is available by the author, acting as a SPI consultant responsible for Knowledge Base maintenance from Quality Department. The author used so called "coaching activities" based on SPI consultancy for regular dissemination of the research results and accumulation of new knowledge and experience.

"Coachers" also referred to as "Experience brokers" [DIN00] prescribe centralized maintenance of knowledge and experience items through involvement of SPI consultants who perform knowledge dissemination among the projects during internal quality audits, seminars, consultations, etc., and gather new knowledge and experience from projects, packaging and storing it into the Knowledge Base. The author was involved in the activities of the Quality Department in the investigated software house during the period 2004 – 2006, performing "coaching activities" during internal quality audits and consulting on process improvement. The performed activities include:

- quality audits in globally distributed projects;
- 3 project risk identification sessions;
- 1 project problem review with improvement suggestions;
- 1 project post-mortem review with lessons learned;
- 10 project effort estimations for initiated globally distributed projects.

In addition, the Knowledge Base was recently installed in another software house that is a new competitor on the global market and was therefore interested in research results.

8.5. RECOMMENDATIONS TO PRACTITIONERS

- Organizational learning

Knowledge Bases are essential for management that pays attention to organizational learning. Employee knowledge and experience can be easily lost with the loss of human resources. Consider turning knowledge from implicit into explicit by maintaining Knowledge Bases for experience accumulation and transfer. These also provide ability to introduce new employees with organizational philosophy.

- Sharing experiences with a Risk Barometer

Risk management is an essential part of globally distributed software projects. Due to variety of global factors and threats complexity of project risk identification and analysis increases. Experience maintained from project to project throughout the organization provides valuable support for practitioners. Therefore, it is recommended to accumulate risk reports as a part of the Knowledge Base.

- Sustainable improvement and updates

The content of the Knowledge Base shall be updated and supplemented with new experience reports in order to provide sustainable improvement of Risk Barometer predictions and reflect the actual situation in globally distributed software projects.

- Knowledge "coachers" or "brokers" for content maintenance

If Knowledge Bases do not prove themselves and are rarely used by practitioners, consider "coaching activities". Knowledge "coachers" or "brokers" may act for both knowledge and experience dissemination and accumulation by this maintaining the content of the Knowledge Base on a regular basis.

8.6. CONCLUSIONS

Along with the scientifically significant theoretical results of answering the question of what distinguishes globally distributed projects from the co-located projects and how to deal with that, global factors, threats and practices accumulated in the Knowledge Base provides a practical application of the research results. It is implemented to provide a long life for the research results and continue to accumulate new factors, threats and practices discovered by practitioners. The content provides support in various ways:

- risk identification, analysis and management,
- project improvement planning,
- post-project reporting,
- best practice accumulation, and
- new employee introduction.

Knowledge Base and the previously described Risk Barometer form a valuable toolset for practitioners and support learning on the project and organizational levels. The major users of these tools are global project managers, especially those lacking experience and expertise with outsourcing projects. Implementation of tools as Knowledge Base also helps to introduce new software development projects and project managers with earlier experiences and prevents the loss of organizational implicit knowledge with the loss of experienced and knowledgeable human resources.

9. APPROACH FOR SUSTAINABLE PROCESS IMPROVEMENT

9.1. OVERVIEW

The growing complexity of software products, software development processes and development environments with dynamically changing circumstances emphasize necessity for new tools and techniques improving performance. The need for and interest in software process improvement (SPI) is widely recognized and becomes even more popular in recent years. Software process improvement has become the primary approach to improving software quality and reliability, employee and customer satisfaction, and return on investment [MAT05], [NIA06. Since software organizations have been pressured or required to conform to certain standards, such as CMMI [SEI02] or ISO 9001 [ISO00], organizations have been focusing and investing in software process improvement and quality system implementation.

There are two types of the SPI approaches mainly used [MUN06]:

- Continuous SPI approaches (such as the Quality Improvement Paradigm, etc.) – these focus on solving selected problems and usually involve improvement cycles based on an initial baseline;
- Model-based SPI approaches (such as ISO 9001, CMMI, etc.) – these compare the current processes and practices of a development organization against a reference model or a benchmark.

The weakness of model-based SPI approaches is that there is no "one fits all" solution. Related investigations show that many SPI initiatives have not proven successful [CON02]. Indeed, as earlier described, the results of the performed exploratory study investigating the attitude of employees from the investigated software house towards the existing quality system (published in [SMM06]) uncovered several serious problems, such as:

- low top management involvement in motivating employees to work in accordance with the quality management practices;
- lack of employee involvement in the early stages of quality system implementation and subsequent low motivation to use the system;
- lack of training and awareness of quality initiatives;
- overwhelming content of the existing quality system;
- incoherence of the quality system with real work practices;

- dissatisfaction of the employees with the existing procedures in some cases resulting in faking quality documentation before the audits.

This situation indicates the necessity of renewal of the existing approaches to process quality assurance with respect to the following requirements – these shall:

- focus on reflecting real work practices;
- be agile in nature to attract active project member involvement;
- help practitioners to document the required information with minimal effort;
- be beneficial and provide added value for practitioners;
- ensure sustainable organizational learning.

The author supports opinion that SPI initiatives shall address both continuous and model-based and be complementary in order to identify problem areas and potential improvement options, as well as implement and optimize solutions as recommended by Münch and Vierimaa [MUN06]. In order to provide a long life for the achieved research results in the investigated software house, the author additional compiled a set of software process improvement activities that prescribe application of the developed Knowledge Base and Risk Barometer and answers the needs of the investigated organization with respect to the findings discussed here.

9.2. SELECTION OF THE METHODS AND TECHNIQUES

Immaturity of global software development discipline and lack of proven approaches leads to a necessity of continuous software process improvement method application. Because of the high level of uncertainty and hidden risks in globally distributed environment, the selected approach includes many risk-oriented activities. Knowledge Base and Risk Barometer in this approach serves as a supporting toolset to make the process of risk identification, evaluation and mitigation planning easier and encourage reuse of previous experiences. In contradiction to the existing situation, this also prescribes updating existing information and storing new practices by practitioners.

The major impediments of successful performance in globally distributed projects that may have influence on software process improvement activities are the following:

- Diverse inter-organizational environment – leads to collision of interest, goals, processes and work practices of each organization involved in the project;
- Socio-cultural, temporal, and geographic distances – lead to negative effect on global software teamwork.

Accordingly, selection of the methods and techniques for the developed approach for globally distributed software process improvement in addition aimed to compensate diversity and lack of team cohesion by focusing on process tailoring and active project member involvement. The author selected basic methods are process workshops described in [DIN04] and post-mortem review meetings described in [DIN05]. These methods prescribe active project member involvement including developers, not only managers and stakeholders. Developers are the ones most likely to see what is possibly wrong and therefore SPI approaches shall not be top-down directed [CON02]. Besides, these approaches are easy to follow and agile in nature and therefore are better accepted by practitioners. These activities are described in the next chapter.

9.3. APPROACH FOR SUSTAINABLE PROJECT IMPROVEMENT

Due to an unexplored nature of globally distributed projects, lack of awareness of global factors and threats often leads to project failure. The author proposes an approach to sustainable project improvement in order to support improvement on project level and organizational learning. The proposed approach does not bring any scientific contribution and is based on recommendations from related studies on software process improvement initiatives [CON02], [SMM06], [NIA06], [MUN06]. However, it is necessary to bring research results to life and eliminate some of the existing problems with respect to quality initiatives. The proposed approach involves the following activities (see Fig. 13).

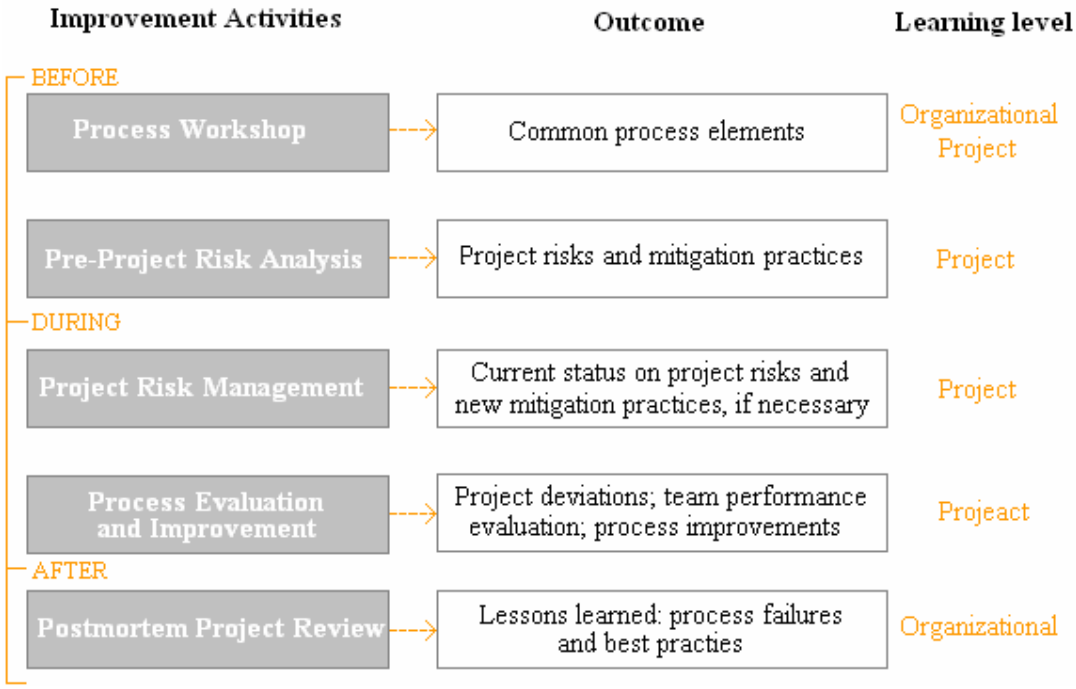


Fig. 13. Sustainable project improvement activities

This approach is advisable to address active collaboration between the remote teams, but can be also applied locally. It consists of different methods and techniques proposed by experienced software process improvement experts and empirically evaluated as efficient. The developed Knowledge Base and Risk Barometer supplement implementation of the SPI approach.

SPI activities in more detail are described in the following sub-chapters. Instruction of its application in practice is also included in Appendix A.3.

9.3.1. Process Workshop

Description: Process workshops shall be organized to form the processes that are corresponding real work practices. This refers to organizational learning and can be performed for certain process areas. Further projects may organize only process tailoring workshops.

Due to the fact that formal modelling of processes may easily be overdone and is anyhow not enough to ensure developer motivation and hence process conformance [DIM04], process workshops aim to tailor project processes in a form of structured, workflow-oriented, electronic reference documents, in order to support participants with a common understanding of project processes and an electronic process guide as a part of the quality system. Common process elements, such as activities, artefacts, roles, tools and techniques for the project can be developed during a joint workshop [DIN04]. Compliance with existing model-based SPI frameworks shall be ensured. Active involvement of potential project participants during the development and introduction of formal routines is proved to play a very important role [CON01]. Collaborative, social processes shall be organized to promote effective infusion and project member interest in such guides.

Inputs: Existing process descriptions (if exist).

Outputs: Process handbook, electronic guide and/or project intranet.

Benefits: Process participants receive an effective self-developed particularly tailored guidance especially necessary when process conformance is important, when a process changes frequently, and when new personnel join a project. For globally distributed projects working on a common process guide means developing a common understanding of how the project will proceed. Effectiveness of process workshops has been empirically validated and proven to have positive impact on process performance [DYB04], [MOE05].

9.3.2. Pre-Project Risk Analysis

Description: Pre-project risk analysis aims to identify all possible negative events and threats that can endanger project and partner collaboration. It can be organized in a way of either project meeting for all project members (if such are already known) or a joint task for project

stakeholders or steering committee. The author proposes to use the contents of Knowledge Base to decrease complexity of risk identification and historical information from previous projects (Risk Barometer performances) for risk analysis especially for inexperienced project members.

Inputs: Risk checklists, risk mitigation practices, historic information from previous projects.

Outputs: Description of possible project risks, risk mitigation plan.

Benefits: Early discussions on project risks may prevent it from unexpected negative events. Using experience from previous projects provides useful information on the proven practices that help to avoid or mitigate project problems.

9.3.3. Project Risk Management

Description: Project risk management shall be held on a regular basis throughout the project involving all project members. Involvement of project personnel brings awareness of what may happen and what to do to prevent it. Project team members shall be able to report considerations of possible sources of project risks, since they often see more symptoms of project threats than project managers. Joint project risk management may be especially critical to provide coordinated action when the problem symptoms occur. All project members shall be aware of dependences and relatedness with the remote team members. Therefore, risk management shall be performed both on the internal level for each partner involved in the project, and on the inter-organizational level, involving at least steering committee and major stakeholders.

To motivate sustainable SPI and knowledge transfer between different projects and project teams risk management process is integrated with application of an Experience Factory or Knowledge Base (for more detail see 8) according to the following scheme (see Fig. 14).

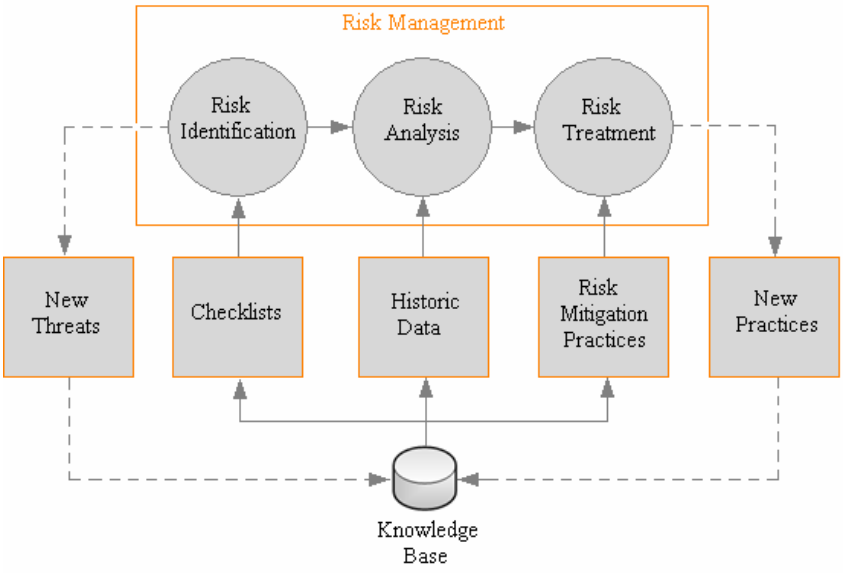


Fig. 14. Risk management scheme

Inputs: Risk checklists, risk mitigation practices, historic information from previous projects.

Outputs: Description of possible project risks, risk management plan. Example of a global project risk management report is given in Appendix A.4 (project details are hidden due to confidentiality)

Benefits: Regular risk management meetings with active project member involvement ensure awareness of possible negative events and coordinated action on their symptom occurrence.

9.3.4. Process Evaluation and Improvement

Description: Poor process performance is more than a headache for a project. Process evaluation and improvement aims identifying weak areas of the project before it turns into a big problem and implementing improvement solutions. It is particularly feasible in large and/or long-term projects.

Process guide developed in the beginning of the project shall be evaluated and updated on a regular basis. In addition, process performance evaluation shall be done regularly throughout the project. This can be done by organizing a project review meetings in accordance to post-mortem review method [DIN05]. Project members shall a) evaluate existing processes in relation to the process guides and identify updates; b) evaluate effectiveness of process performance and suggest improvements. This provides sustainable process improvement and prevents from problems in relation to following the outdated processes.

Inputs: Existing process guides

Outputs: Improved and/or updated process guides, additional report on project problems and risks, corrective activity plan. An example of project problem review results is given in Appendix A.5.

Benefits: Process improvements are essential for effective performance. The updates prevent from following outdated processes. Joint process workshops and review meetings strengthen team spirit.

9.3.5. Post-mortem Project Review

Description: Post-mortem project reviews based on the project team meetings [DIN05] aim to focuses on describing project benefits and problems. Such meetings can be held at the end of the project (post-mortem reviews) and also during the project for process efficiency evaluation and project problem analysis.

Post-mortem project reviews are traditionally managed by a leader. The author suggests involving organizational SPI experts for this purpose. The meeting starts with a brainstorming about what went well and what went wrong in the project. Main reasons for both benefits and

problems are then explored by performing a root-cause analysis. The results of post-mortem meetings support root-cause analysis and lessons learned. At the end of the meeting, participants suggest improvements in the current project processes or future projects to ensure sustainable organizational learning.

Inputs: No pre-prepared input is required.

Outputs: List of risks faced by the project, best practices applied to overcome project problems, and improvement suggestions to be stored in the Knowledge Base. Example of a post-mortem review results is given in Appendix A.6.

Benefits: Active project participant involvement motivates commitment to SPI initiatives and provides an opportunity to derive lessons learned. Organizational SPI expert involvement helps to gather experience for further centralized dissemination.

9.4. APPLICATION IN PRACTICES

Instruction for SPI approach application in practice describing in more detail each activity can be found Appendix A.3. During her involvement in the Quality department activities and SPI consultancy, the author was involved in a set of activities and workshops in global projects prescribed by the sustainable process improvement approach. The author performed the following activities: 3 project risk identification sessions, an in-depth problem analysis investigations aiming in process improvement, a globally distributed project post-mortem review meeting and a globally distributed project process improvement meeting held in accordance to the post-mortem review method. An example of the results of the performed globally distributed project process improvement meeting can be found in Appendix A.4 and an example of a post-mortem project review can be found in Appendix A.6.

The effectiveness and acceptance of these post-mortem reviews have been evaluated through participant surveys. Analysis of the survey data indicates that participants have evaluated post-mortem review results, content and method positively, expressing desire to participate again and recommend it for other projects. A participant evaluation survey overview is given in Appendix A.7.

Application of these methods among all participating teams in global projects in the investigated software house was not possible due to a limited access of the author to the remote prime contractors. However, a field study between two remote teams in Latvia distributed by 250 km was performed. It prescribed parallel risk analysis sessions, joint management of the identified problems and a post-mortem analysis meeting at the end of the project.

9.5. RECOMMENDATIONS TO PRACTITIONERS

Importance of software process improvement in globally distributed environment is emphasized in the following recommendations to practitioners. Most of these address joint performance, but can be also established locally as intra-organizational SPI strategies.

- SPI initiatives demonstrate competence and maturity

Only mature organizations invest in software process improvement. Projects personnel under stress of deadlines and chaotic management have no time to spend for process improvements. In contradiction, well established processes can be further tailored and improved for certain needs.

In addition, related studies show that well-considered SPI activities have a positive effect on project and organizational performance [CON01].

- Adoptability and flexibility

The dynamic environment of globally distributed software development requires frequent changes. Processes by their nature, contribute to change in a company, and, thus, a good process must be readily changeable [WAR01]. These processes shall be the subject for software process improvement activities. Pay attention to adoptability and flexibility of organizational processes that play an important role in the inter-organizational collaboration.

- Involve remote partners

Process workshops and process improvement sessions with remote team involvement help to achieve a common understanding of how to perform by joint effort. Common philosophy and approach are especially necessary to overcome threats of diversity and lack of commitment to the joint project.

- Use SPI activities for co-located teams

The proposed SPI approach and its activities can be also applied locally for a co-located team. If the inter-organizational processes cannot be changed due to limited access, SPI activities shall address intra-organizational processes.

- Involve stakeholders and senior management in SPI implementation

Although SPI results shall not be to-down directed [CON02], experts warn that without high levels of senior management commitment, most quality improvement efforts are doomed to

fail [SMM06], [PIT00]. Therefore, stakeholder and senior management involvement and interest in SPI initiatives shall exist.

- Employee involvement

Processes defined by employees that are actually working according to these procedures, exclude unnecessary detail and are tailored for their direct needs.

There is also another purpose of employee involvement. Joint workshops, meetings and sessions provide so necessary socialization for team members performing across borders. This increases team cohesion for both co-located and distributed team members and effectively influences team performance.

- Begin and continue with risk management

Every project shall start with possible risk identification and continue with monitoring these risks and identifying new ones. Experiences from other projects that are accumulated for organizational learning shall help managers choosing risk mitigation strategies. In its turn proven solutions shall be also packed and stored in the Knowledge Base for active knowledge transfer.

- Finish with lessons learned

It is essential to provide continuous organizational learning based on previous experiences. Therefore, organization of a post-mortem review is important. It shall focus on identification of drawbacks and problems, and accumulation of best practices. Lessons learned shall be then packed and stored in a Knowledge Base to provide organizational learning.

9.6. CONCLUSIONS

Various researches have proved that software process improvement has become the primary approach to improving software quality and reliability, employee and customer satisfaction, return on investment and team performance. Thus the majority of process improvement initiatives usually die a long way before there implementation, there are ways to attract practitioners with SPI activities. One of the motives used for involving the practitioners is the agile style of the SPI activities.

> E.g. process definition activities and post mortem / problem analysis using seminars use democratic techniques as brainstorming and discussions; additional materials used for recording the results (post-it

notes and paper sheets are recorded with the help of a digital camera) saves lots of time for results documentation.

Another motive is wide participant involvement that transforms practitioners from the role of the guided and "guinea-pigs" into experimentalists. This ensures that processes reflect the real life processes and do not come "from above". At the end, developers are the ones most likely to see what is possibly wrong and therefore SPI approaches shall not be top-down directed.

With respect to global specifics, joint workshops, meetings and sessions provide so necessary socialization for team members performing across borders. Activities selected by the developed approach aim to compensate diversity and lack of team cohesion in globally distributed software projects. Thus, these can be also applied in co-located teams.

Although approach for sustainable process improvement is in the process of implementation in the investigated software house, methods and techniques proposed by this approach have been selectively validated by the author in a set of projects. Effectiveness of these approaches and acceptance by users have been estimated through participant survey (Appendix A.7.) and positively approved by practitioners.

10. DISCUSSION OF THE RESULTS

10.1. REPETITION OF RESULTS AND ESTIMATION OF IMPORTANCE

This doctoral thesis describes the author's three years long empirical research in the field of globally distributed software development within one of the leading software houses in Latvia. The research aimed to achieve results according to the following research questions:

RQ1:	What are the factors that distinguish global software development from traditional in-house software development?
RQ2:	What is the effect of global factors on project performance?
RQ3:	How to leverage global factors to improve project performance?

Through application of various qualitative research methods based on empirical data gathered from the software house under study the following results have been achieved:

An overview of global factors and threats (for RQ1): The author has gathered a wide amount of sources including literature overview and empirical material that helped to derive global factors and threats that characterize peculiarities of globally distributed software projects. The results exceed the known descriptions of global project peculiarities and can be used as a reference model. Validation of relevance and prevalence of the explored global threats has been performed by means of global project survey from Latvian perspective.

Empirical observations of the effect of global factors and threats (for RQ2): In contradiction to the studies that conclude with global threat or risk identification, the author performed till now unachieved attempt to formalize the effect of global threats on GSD project performance using an experience-based risk-oriented approach. Statistical analysis of the gathered data provides the list of undisputable sources of major concern for successful global project performance with respect to budget overrun, time delays and project schedule, customer dissatisfaction and team undermined morale. These areas of concern shall be taken into account by practitioners while starting globally distributed collaboration.

The author extended evaluation of the effect of global factors and threats by conducting an in-depth analysis of the demanded topics of process distribution and teamwork in globally distributed environment.

Process distribution was found to have impact on overall partner collaboration, team performance and uncovered additional documentation needs for artefacts that are transferred over the borders. An original linkage between process distribution and collaboration types was derived forming a magic quadrant.

Additional multi-case study was performed in collaboration with a researcher from SINTEF, Norway, aiming to investigate till now unexplored question of the reasons and effects of lacking trust on global software teams applying social research findings. Results of this study proved the importance of trust in globally distributed projects, uncovered additional unexplored effects of lacking trust in global teams and derived a set of recommendations to practitioners to improve global teamwork.

Practices, methods and tools for GSD project improvement (for RQ3): The author developed a toolset for sustainable process improvement including Knowledge Base, Risk Barometer and a set of SPI methods and techniques united in a joint approach. This approach prescribes to support globally distributed project improvement and organizational learning. Research material has been integrated into the Knowledge Base in the form of global practices and historic data reports. Risk Barometer has been developed to automate the estimates of the effect of global threat for practitioners performing risk management activities. Approach for sustainable process improvement includes methods that act as a counterforce for global factors and mitigate diversity and poor team cohesion.

These findings are especially valuable due to lack of research results in this research area. Empirical material gathered during the research forms significant and unique insight into how globally distributed projects are run from perspective of one of the major Latvian software development outsourcing service providers. In contradiction to various studies conducted from the customer perspective, this research focuses on supplier related threats and practices by offering original findings in this area. Research uncovered rare threats from Latvian perspective. Recommendations to practitioners derived by the research can be characterized as empirically based, harmonized, categorized for different user groups and addressing Latvian GSD export project particularities.

10.2. LIMITATIONS

The first limitation of this research is its focus on only supplier related threats and practices. The author expects that conducting a mirror research from the customer perspective may lead to similar findings regarding frequency of occurrence of global threats and different results

regarding the effect of global threats. However, this limitation is acceptable due to the industrial background of the research.

The second limitation of this research is the number of empirical data sources:

- 13 interviews were conducted with 9 project managers to derive factors, threats and practices (in addition to research literature overview);
- 38 projects were investigated for global factor and threat validation and further observations.

This is dictated by limited access to industrial projects within the investigated software house. More projects and interviews could strengthen significance of the results. Thus, the author considers this acceptable and over expected.

10.3. APPLICATION IN PRACTICE

Results of the research covered in this thesis have strong practical nature. These are demanded not only in the investigated software house, but also outside it.

During the period 2004 – 2006, the author was involved in the activities of the Quality department in the investigated software house, performing internal quality audits and SPI consulting on globally distributed software development. The performed activities include different SPI consultancy and activity organization for globally distributed projects.

The GSD Knowledge Base and Risk Barometer are accessible by all employees in the investigated software house and have been used by Quality Department and project managers. These are also being implemented in another software house that is interested in research results and demanded access to the gathered practices.

Finally, practices used to overcome global factors and threats were selectively published and presented in a set of international conferences, and were approved by the research and industrial audiences.

10.4. RECOMMENDATIONS TO PRACTITIONERS

The results of the research on global software development are recommended for application in practice by various types of users. The following table describes the target groups for research result application with the corresponding purposes as follows (see Table 13).

Table 13. Recommendations to practitioners

Target Group	Results	Purpose
Project managers and team leaders	Knowledge on global factors and threats; risk mitigation practices	Use this information in risk management to identify environmental sources of risks, and plan risk mitigation activities.
	Knowledge Base	Utilize Knowledge Base content as guidance in GSD projects. Store lessons learned from existing and finished projects.
	Risk Barometer	Use threat outcome predictions for risk analysis. Update current knowledge by submitting historic information. Use risk prediction reports for reporting on project risks to senior management and stakeholders. Do not assume that outcome predictions certainly come true.
	Approach for sustainable process improvement	Use methods and techniques proposed by the approach to avoid known problems, improve the project in progress, and compensate the effect of global factors and threats such as various diversities and lack of team cohesion.
SPI consultants	Knowledge on global factors and threat; mitigation practices	Use all these results to disseminate existing knowledge and experience and to accumulate, pack and store new experience.
	Knowledge Base	
	Approach for sustainable process improvement	
Senior management	Knowledge on global factors and threat; mitigation practices	Plan globally distributed collaboration with respect of global factors and threats. Learn how to overcome the risks
	Recommendations for SPI initiative implementation	Learn how to properly implement SPI initiatives and what problems to avoid
Project members	Knowledge Base	Use Knowledge Base to introduce new project members with existing practices and lessons from previous projects. Existing employees can use it as guidance for GSD projects.

11. CONCLUSIONS

An emerging but still poorly explored area of globally distributed software development puts new demands on the software processes stressed by increased complexity of global projects. Lacking experience and expertise in global environment managers are balancing between spectacular success and immediate failure. Since practitioners are forced to experiment for leveraging emerging and dynamic sources of global software development risks, global project success becomes unpredictable. Software industry demands empirical investigation of the nature of global projects to provide answers to many questions.

Global factors that have been derived by the author scientifically prove that global software development projects are distinctive from in-house projects. The distinguishing nature of distributed software development is emphasized by unavoidable elements, called global factors, which shall be analyzed throughout the project. Research results demonstrated considerable effect of these factors on project performance, in particular causing budget and calendar deviations, customer dissatisfaction and undermined morale of the development team, resulting in poor team performance.

Many global projects have failed not because they were run by incapable project managers, but because the managers were not aware of global threats that endanger project performance in globally distributed environment. Research results provide guidance for effective risk identification, analysis and management in globally distributed projects and help practitioners to avoid unexpected negative outcomes of the factors that are inherited and often hidden in the nature of global project environment. The developed Risk Barometer and Knowledge Base implementation formalizes the evaluation of outcome predictions and supports practitioners with computerized experience-based reports.

In addition to peculiar global factors and threats, the author provides research results touching important topics that demanded empirical investigation, such as software life cycle distribution and global software teamwork. Empirical material gathered during the research forms significant and unique insight into how globally distributed projects are run from perspective of one of the major Latvian software development outsourcing service providers.

Research results have strong practical application. Recommendations derived from empirical observations can be used by project managers, team leaders, team members, software process improvement consultants. The results of the research have been widely published in international conferences and approved by industrial and research audiences.

LIST OF TABLES

LIST OF FIGURES

LISTS OF ABBREVIATIONS

CMMI - Capability Maturity Model Integration

GSD - Global Software Development

IS - Information System

ISO - International Organization for Standardization

IT - Information Technology

LNCS - Lecture Notes in Computer Science

SPI - Software Process Improvement

BIBLIOGRAPHY

11.1. AUTHOR'S PUBLICATIONS

[SMB04] D. Šmite, J. Borzovs, "Global Software Development Process Management: Problem Statement", In Proceedings of the Int. Baltic DB and IS Conf. Doctoral Consortium, June 2004, Latvia, pp. 198-207

[SMI04] Šmite D., „Global Software Development Projects in One of the Biggest Companies in Latvia: Is Geographical Distribution a Problem", Software Process Improvement and Practice, published by Wiley, 2006; Vol.11, pp.61-76

[SMT04] Šmite D., "Global Software Development Project Management – Distance Overcoming", In Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), published in LNCS by Springer Verlag, November 2004, Norway, pp. 23-33

[SMT04] Šmite D., "A Case Study: Coordination Practices in Global Software Development", In Proc. of the 6th Int. Conf. on Product Focused Software Process Improvement (PROFES), Springer, Oulu, Finland, June 2005, pp. 234-244

[SMS05] D.Šmite, U.Sukovskis „Knowledge Management in Distributed Environment", Industrial Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), November 2005, Hungary, pp.5.15-5.22.

[SMB06] D.Šmite, J.Borzovs „A Framework for Overcoming Supplier Related Threats in Global Projects", In Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), published in LNCS by Springer Verlag, October 2006, Finland, pp. 49-60

[SMI06] D.Šmite "Requirements Management in Distributed Projects" Journal of Universal Knowledge Management, 2006, Vol. 1, No. 2, pp. 69-76. Accessible on-line (29.03.2007):

http://www.jukm.org/jukm_1_2/requirements_management_in_distributed

[SMI07] D.Šmite, "Project Outcome Predictions: Risk Barometer Based on Historical Data", In proceedings of the Int. Conf. on Global Software Engineering (ICGSE) by IEEE Computer Society, August 2007, Germany, pp.103-112

[SMM06] Šmite D., and Moe N.B. „An ISO 9001:2000 Certificate and Quality Awards from Outside – What's Inside? – A Case study", In Proceedings of the Int. Conf. on Product Focused Software Process Improvement (PROFES), published in LNCS by Springer Verlag, June 2006, the Netherlands, pp.208-221

[SMT06] D.Šmite „Requirements Management in Distributed Projects", In Proceedings of the Int. Workshop on Learning Software Organizations and Requirements Engineering (LSO+RE), University of Hannover, March 2006, Germany, pp. 9-16

[MCC06] McCaffery F., Šmite D., Wilkie F.G., McFall D. „A Proposed Way for European software Industries to Achieve Growth Within the Global Marketplace", In journal of Software Process Improvement and Practice, published by Wiley, 2006; Vol.11 Nr.3, pp.277-286

[MCC05] F. McCaffery, D.Šmite , F.G Wilkie, D.McFall „How European Software Industries can prepare for growth within the Global Marketplace. Northern Irish Strategies", In Industrial Proceedings of the Int. Conf. on European Software Process Improvement (EuroSPI), November 2005, Hungary, pp.3.23-3.32.

[MOE07] Moe N.B. and Smite D. „Understanding Lacking Trust in Global Software Teams – A Multi-Case Study". In Proceedings of the 8th Int. Conf. on Product Focused Software Process Improvement (PROFES), July 2007, Riga, Latvia, pp. 20-34

11.2. OTHER PUBLICATIONS

[AGE06] Agerfalk, P.J. and Fitzgerald B., "Flexible and distributed software processes: Old petunias in new bowls?" Communications of the ACM, 49(10) 2006, pp. 26-34.

[ANS04] Anspoks A. "Software Industry Trends in Latvia", Baltic IT&T Review, Nr.3 (34), 2004, pp.32-34

[AUB03] Aubert, B., Houde, J.F., Patry, M. and Rivard, S. "Characteristics of IT Outsourcing Contracts. In Proceedings of the 36th Annual Hawaii International Conference on System Sciences, 2003, 9 pp.

[BAN01] Bandow, D., Time to create sound teamwork. The Journal for quality and participation, Vol. 24, Issue 2, 2001, pp. 41-47.

[BAT01] Battin, R.D., Crocker, R., Kreidler, J. "Leveraging Resources in Global Software Development", IEEE Software, Vol. 18, No. 2, 2001, pp. 70-77

[BHA06] Bhat, J.M., M. Gupta, and S.N. Murthy, Overcoming requirements engineering challenges: Lessons from offshore outsourcing. IEEE Software, Vol. 23, No. 5, 2006. pp. 38-44.

[BOE91] Boehm B.W., "Software Risk Management: Principles and Practices", IEEE Software, Vol. 8, No.1, 1991, pp. 32-41.

[BOE00] Boehm B.W., Horowitz E., Madachy R., Reifer D., Clark B.K., Steece B., Brown A. W., Chulani S. and Abts C. "Software Cost Estimation with COCOMO II", Prentice Hall, 2000

[CAR97] Carr M.J., "Counterpoint: Risk Management May Not Be for Everyone", IEEE Software, Vol.30 No.5, 1997, pp. 21-24

[CAR99] Carmel E., „Global Software Teams: Collaborating Across Borders and Time Zones", Prentice-Hall, 1999

[CAR01] Carmel, E., Agarwal, R. "Tactical Approaches for Alleviating Distance in Global Software Development", IEEE Software, Vol. 18, No. 2, 2001, pp.22-29

[CON01] Conradi, R., and Dybå, T., "An Empirical Study on the Utility of Formal Routines to Transfer Knowledge and Experience", In Proceedings of the European Software Engineering Conference (ESEC), published by ACM/IEEE CS Press, 2001, pp. 268-276

[CON02] Conradi R., and Fuggetta A. "Improving Software Process Improvement", IEEE Software, Vol. 19, Issue 4, 2002, pp. 92-99

[DAM06] Damian D. and Moitra D., "Global Software Development: How Far Have We Come?", IEEE Software, Vol. 23, No.5 2006, pp.17-19.

[DEL05] DeLone, W., et al. Bridging global boundaries for IS project success. in 38th Hawaii International Conference on System Sciences. 2005. Big Island, HI, United States: Institute of Electrical and Electronics Engineers Computer Society, Piscataway, NJ 08855-1331, United States

[DIN00] Dingsøyr T., "An Evaluation of Research on Experience Factory", Workshop on Learning Software Organisations at the 6th Int. Conf. on Product Focused Software Process Improvement (PROFES), Oulu, Finland, 2000, pp. 55 – 66

[DIN05] Dingsøyr, T., Post-mortem reviews: purpose and approaches in software engineering. Information and Software Technology, 2005, Vol.47 No.5, pp. 293-303.

[DIM04] Dingsøyr T., Moe N.B., Dybå T., and Conradi R., "A workshop-oriented approach for defining electronic process guides - A case study," NOKOBIT, Stavanger, Norway, 2004

[DIN04] Dingsøyr, T., Moe N.B., Dybå T. and Conradi R., "A workshop-oriented approach for defining electronic process guides - A case study", in Software Process Modelling, S.T. Acuña and N. Juristo, Editors. Kluwer Academic Publishers: Boston, 2004, p. 187-205.

[DIR01] Dirks, K.T. and D.L. Ferrin, "The role of trust in organizational settings", Organization Science, 2001, Vol.12 No.4, pp. 450-467.

[DUA01] Duarte D.K. and Snyder L.T., "Mastering Virtual Teams: Strategies, Tools, and Techniques that Succeed", 2nd edition, Jossey-Bass, A Wiley Company, 2001

[DYB04] Dybå T, Moe N.B., and Mikkelsen E.M. "An Empirical Investigation on Factors Affecting Software Developer Acceptance and Utilization of Electronic Process Guides" in Proceedings of the 10th International Symposium on Software Metrics (METRICS'04), 2004, pp. 220-231

[EBE01] Ebert, C. and De Neve P. "Surviving Global Software Development", IEEE Software, Vol.18, No 2, 2001, pp.62-69

[EDW03] Edwards, H.K. and V. Sridhar, "Analysis of the effectiveness of global virtual teams in software engineering projects", In Proceedings of the 36th Annual Hawaii international Conference on System Sciences, 2003, IEEE Computer Society, Washington, DC, 19.2.

[GLA67] Glaser B., Strauss A. "The discovery of grounded theory: Strategies of qualitative research", Wiedenfeld and Nicholson, London, 1967

[GRO96] Grover V., Cheon M.J., and Teng J.T.C., „The Effect of Service Quality and Partnership on the Outsourcing of Information Systems Functions", In Journal of Management Information System, Vol.12 No.4, 1996, pp.89-116

[HAI79] J. Hair, R. Anderson, R. Tatham, and B. Grablowsky, "Multivariate Data Analysis". Tulsa, Okla.: PPC Books, 1979

[IES04] Iesalnieks J., Gulbe B. "Old and New Europe as IT Partners", Baltic IT&T Review, Nr.2 (33), 2004, pp.38-40

[ISO00] ISO, "ISO 9001:2000 Quality management systems - Requirements", 2000

[JAR98] Jarvenpaa, S.L., K. Knoll, and D.E. Leidner, "Is anybody out there? Antecedents of trust in global virtual teams", Journal of Management Information Systems, 1998. Vol.14 No.4, pp. 29-64

[JAR99] Jarvenpaa, S.L. and D.E. Leidner, "Communication and trust in global virtual teams", Organization Science, 1999. Vol.10 No.6, pp. 791-815

[JAR01] Jarvinen P. "On research methods", Opinpajan Kirja, Tampere, 2001

[JAR04] Jarvenpaa, S.L., T.R. Shaw, and D.S. Staples, "Toward contextualized theories of trust: The role of trust in global virtual teams", Information Systems Research, 2004. Vol.15 No.3, pp. 250-267.

[KAN02] Kanawattanachai, P. and Y. Yoo, "Dynamic nature of trust in virtual teams", Journal of Strategic Information Systems, 2002, 11(3-4), pp. 187-213.

[KAR98] Karolak D.W., "Global Software Development: Managing Virtual Teams and Environments", IEEE Computer Society, 1998

[KAR04] Karnitis E. "Internet Governance or Regulation: The Position of the Baltic States in Advance of the Second Phase of WSIS", Baltic IT&T Review, Nr.2 (33), 2004, pp.46-49

[LAC95] Lacity M.C., Willcocks L.O., and Feeny D.F., "IT Outsourcing: Maximize Flexibility and Control", Harvard Business Review, May-June 1995, pp.84-93

[LAC02] Lacity, M.C. "Lessons in Global Information Technology Sourcing". IEEE Computer, Vol.35, No.8, 2002, pp.26-33

[LEE00] Lee, J.N.; Huynh, M.Q.; Kwok, C.W. and Pi S.M. "The Evolution of Outsour cing Research: What is the Next Issue?", In Proceedings of the 33rd Hawaii International Conference on Systems Sciences, Hawaii, January 2000, pp.1-10.

[LEE03] Lee J.-N., Huynth M.Q., Kwok R.C.-W., Pi S.-M. "IT oursourcing Evolution – Past, Present, and Future", Communications of the ACM, 2003, Vol.46, No.5, pp.84-89.

[LEK03] Lee, J.N. and Kim, Y.G. "Exploring a Causal Model for the Understanding of Outsourcing Partnership", Proceeding of the 36th Hawaii International Conference on System Sciences, Hawaii, January 2003

[LIE98] Liebowitz J. and Beckman T., "Knowledge Organizations: What Every Manager Should Know". Boca Raion, FL: CRC Press, 1998

[LIG01] Light M., Matlus R., and Berg T., „Strategic Analysis Report Application Development Contracting: Lifeline or Noose?" R-14-38791, 28 September 2001

[LIS05] List B., Bruckner R.M., Kapaun J. "Holistic Software Process Performance Measurement From the Stakeholders' Perspective" Proceedings of the Sixteenth International Workshop on Database and Expert Systems Applications, IEEE Press, 2005, pp. 941 – 947

[LOH95] Loh L., Venkatraman N., "An empirical study of information technology outsourcing: Benefits, risks, and performance implications". In Proceedings of the 16th International Conference on Information Systems, Dec. 10-13, 1995, Amsterdam, the Netherlands, pp.277-288

[MAR04] Martins, L.L., L.L. Gilson, and M.T. Maynard, "Virtual teams: What do we know and where do we go from here?" In Journal of Management, 2004. Vol.30 No.6, pp. 805-835.

[MAT05] Mathiassen L., Ngwenyama O.K., and Aaen I. "Managing Change in Software Process Improvement", IEEE Software, Vol.22, No.6, 2005, pp. 84-91

[MIN05] Minevich M. and Richter F.J., "Global Outsourcing Report 2005", March 2005

[MEA06] Mears J. "Offshoring remains hot issue for IT", Network World, 12/06/04. Available online (09.05.2006):

http://www.networklifemag.com/news/2004/120604offshore.html

[MOE04] Moe, N. B. and Dybå, T., "The Adoption of an Electronic Process Guide in a Company with Voluntary Use," In Proceedings of the European Software Process Improvement Conference (EuroSPI), Trondheim, Norway, 2004, pp.114-125

[MOE05] Moe N.B., Dingsøyr T., Nilsen K.R., and Villmones N.J., "Project Web and Electronic Process Guide as Software Process Improvement", In Proceedings of the 12th European Conference EuroSPI, Budapest, Hungary, November 2005, pp.175-186

[MOD05] Moe, N. B. and Dingsøyr, T., "The Impact of Process Workshop Involvement on the Use of an Electronic Process Guide: A Case Study," Presented at EuroMicro, Porto, Portugal, 2005, IEEE, pp.188-195

[MUN06] Münch J., and Vierimaa M. Preface for the proceedings of the 7th International Conference on Product-Focused Software Process Improvement, published in LNCS by Springer Verlag, June 2006, Amsterdam, The Netherlands

[MWD] Merriam-Webster Online Dictionary. Available online www.m-w.com

[NIA06] Niazi M, "Software Process Improvement: A Road to Success", In Proceedings of the Int. Conf. on Product Focused Software Process Improvement (PROFES), published in LNCS by Springer Verlag, June 2006, the Netherlands, pp.395-401

[ORL02] Orlikowski W.J., Knowing in Practice: Enacting a Collective Capability in Distributed Organizing, INFORMS, May/June, 2002, pp. 249-273

[PAV05] Pavlovs S., “Рига интересна, ибо мультикультурна” Коммерсант Baltic daily / 25.04.2005, Nr.78, pp.5

[PIC03] Piccoli, G. and B. Ives, “Trust and the unintended effects of behavior control in virtual teams”, MIS Quarterly, 2003, Vol.27, Nr.3, pp. 365-395.

[PIT00] Pitterman, B., "Telcordia technologies: The journey to high maturity," IEEE Software, Vol.17, No.4, 2000, pp.89-96

[ROC98] Rocco, E. “Trust breaks down in electronic contexts but can be repaired by some initial face-to-face contact”. Los Angeles, CA, USA: ACM, New York, USA, NY, 1998

[ROG03] Rogers, E. M., “Diffusion of Innovations”, vol. Fourth Edition. New York: The Free Press, 2003

[ROY00] Roy, V., Aubert, B. “A Resource-Based Analysis of IT Sourcing”, Scientific Series, CIRANO, Montreal, 2000

[SAH03] Sahay S., Nicholson B., Krishna S., “Global IT Outsourcing: Software Development across Borders”. Cambridge University Press, 2003

[SAL05] Salas, E., D.E. Sims, and C.S. Burke, “Is there a "big five" in teamwork?”, Small Group Research, 2005, Vol.36, Nr.5, pp. 555-599

[SCA02] Scardino L. “Improving Sourcing Deals: Look at More Then Service Costs”, Gartner Inc., 11.03.2002. Available online (21.08.2003):
http://www3.gartner.com/DisplayDocument?doc_cd=105022

[SEI02] SEI, "Capability Maturity Model ® Integration (CMMISM), Version 1.1", 2002

[SHI07] Shiffler G.III, Hahn Wm. L., Hale K., Biscotti F., “Gartner Dataquest Market Databook”, Gartner Inc., 15 February 2007

[STR90] Strauss A., Corbin J. “Basics of qualitative research – Grounded theory procedures and techniques”, Sage Publications, Newbury Park Ca, 1990

[STR98] Strauss, A., Corbin, J., “Basics of qualitative research: Techniques and procedures for developing grounded theory”. Thousand Oaks, CA: Sage Publications, 1998

[WAR01] Ward R.P., Fayad M.E., and Laitinen M., “Thinking objectively: software process improvement in the small”, Communications of the ACM, Vol.44, No.4, 2001, pp.105-107.

[WIE99] Wiegers K.E., “Software Requirements”, Microsoft Press, 1999

[WIL94] Willcocks, L.; Fitzgerald, G. “Guide to Outsourcing Information Technology”. Business Intelligence, 1994

[WIL93] Willis, J. (2000). "Defining a field: Content, theory, and research issues". Contemporary Issues in Technology and Teacher Education [Online serial], 1 (1), pp.209-219. Available (29.03.2007):

http://www.citejournal.org/vol1/iss1/seminal/article1.htm

[WIL07] Wilson G.D. "Worldwide Collaborative Product Development Applications 2007–2011 Forecast: A First Look at a Sound 2006", IDC, March 2007

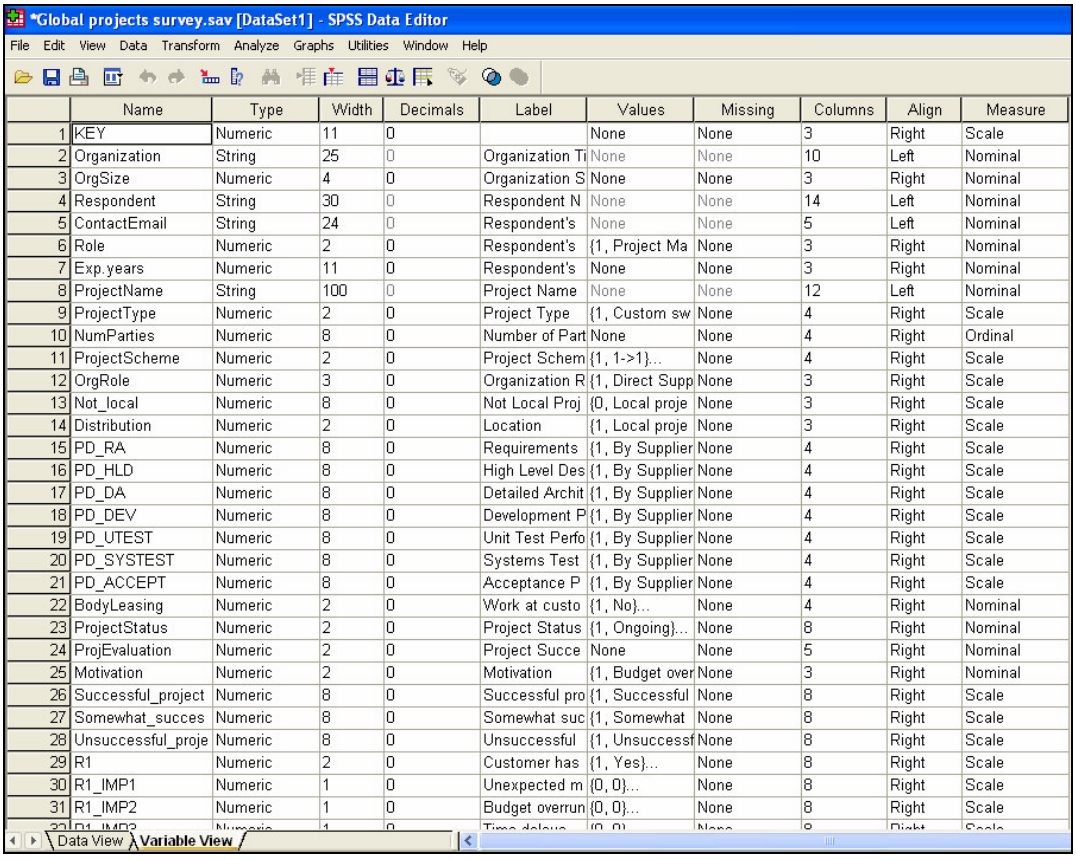
APPENDIXES

A.1. SURVEY TEMPLATE

Organization Information				
Organization Name				
Organization Focus	A. Custom Software Development B. Product Development C. Software Maintenance D. Other –			
Organization Size	number of employees –			
Respondent's Name				
Contact E-mail (optional)				
Role in the Projects	A. Project Manager B. Other –			
Years of Experience in IT				
Project Information				
Project Name				
Project Description	A. Custom Software Development B. Software Product Development C. Software Maintenance D. Improvement of Existing Software E. Other –			
Project Scheme	A. Customer → Direct Supplier B. Customer → Direct Suppliers C. Customer → Direct Supplier → Subcontracted Supplier D. Customer → Direct Supplier → Subcontracted Suppliers E. Other –			
Number of Direct Suppliers				
Number of Subcontracted Suppliers				
Your Role in the Projects	A. Direct Supplier B. Subcontracted Supplier			
Partner Countries of Location	Customer – Direct Supplier(s) – Subcontracted Supplier(s) –			
Project size	A. > 2 manyear B. 1 - 2 manyears C. < 1 manyear			
Project Phase Distribution	*Phases*	*By You*	*By Other Supplier*	*By Customer*
	Requirements Analysis	□	□	□
	High Level Design	□	□	□
	Detail Architecture	□	□	□
	Development	□	□	□
	Unit Testing	□	□	□
	System and Integration Testing	□	□	□
	Acceptance Testing	□	□	□
Did the supplier team work at the customer premises?	A. Yes B. No C. Sometimes If C, then specify when –			
Project Status	A. Planned B. Ongoing C. Finished D. Canceled E. Other –			
Project Success Evaluation		*Scale: 1 – 10, where 1 is very unsuccessful and 10 is very successful.* *Considering 3 factors – budget, calendar deviations and customer satisfaction*		
Motivate Your Evaluation				
Notes				

Threats	Y/N	Effect evaluation						
		0-None, 1-Negligible, 2-Minor, 3-Moderate, 4-Significant, 5-Disastrous						
	Did the threat materialize in your project?	Unexpected manag.costs	Supplier budget overrun	Time delays	Late product delivery	Your dissatisfaction	Supplier undermined morale	Disputes and litigations
Customer has complex hierarchy and/or several problem escalation levels								
Supplier has complex hierarchy and/or several problem escalation levels								
Diversity in process maturity and/or inconsistency in work practices between the partners								
Lack of understanding of each other's context of decision making								
The customer believes that the work cannot be done from a far off location								
Lack of trust and commitment								
Increased cost of logistics of holding face to face meetings								
Increased level of reporting on project progress to the customer								
Increased virtualness								
Lack of language skills by supplier								
Terminology differences								
Customer's employees unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode								
Faulty effort estimates								
Increased level of complexity of project management								
Increased level of unstructured poorly-defined tasks								
Increased complexity of spreading awareness and knowledge								
Lack of common goals								
Lack of experience and expertise of the customer with outsourcing projects								
Lack of experience and expertise of the supplier with outsourcing projects								
Lack of joint risk management								
Lack of team spirit								
Poor or disadvantageous distribution of software development activities								
Relatedness with other suppliers								
Poor cultural fit								
Dominant use of asynchronous communication with the customer								
Time zone difference								
Lack of clarity about responsibility share								
Poor or complex project measurement								
Increased complexity of project, activity, human resources and delivery planning								
Poorly defined or inconsistent software requirements specifications								
Poorly defined or inconsistent software design and/or architecture								
Poor artifact version control								

A.2. SURVEY DATA RECORDS WITHIN SPSS – VARIABLE REPRESENTATION



*Global projects survey.sav [DataSet1] - SPSS Data Editor

File Edit View Data Transform Analyze Graphs Utilities Window Help

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
1	KEY	Numeric	11	0		None	None	3	Right	Scale
2	Organization	String	25	0	Organization Ti	None	None	10	Left	Nominal
3	OrgSize	Numeric	4	0	Organization S	None	None	3	Right	Nominal
4	Respondent	String	30	0	Respondent N	None	None	14	Left	Nominal
5	ContactEmail	String	24	0	Respondent's	None	None	5	Left	Nominal
6	Role	Numeric	2	0	Respondent's	{1, Project Ma	None	3	Right	Nominal
7	Exp.years	Numeric	11	0	Respondent's	None	None	3	Right	Nominal
8	ProjectName	String	100	0	Project Name	None	None	12	Left	Nominal
9	ProjectType	Numeric	2	0	Project Type	{1, Custom sw	None	4	Right	Scale
10	NumParties	Numeric	8	0	Number of Part	None	None	4	Right	Ordinal
11	ProjectScheme	Numeric	2	0	Project Schem	{1, 1->1}...	None	4	Right	Scale
12	OrgRole	Numeric	3	0	Organization R	{1, Direct Supp	None	3	Right	Scale
13	Not_local	Numeric	8	0	Not Local Proj	{0, Local proje	None	3	Right	Scale
14	Distribution	Numeric	2	0	Location	{1, Local proje	None	3	Right	Scale
15	PD_RA	Numeric	8	0	Requirements	{1, By Supplier	None	4	Right	Scale
16	PD_HLD	Numeric	8	0	High Level Des	{1, By Supplier	None	4	Right	Scale
17	PD_DA	Numeric	8	0	Detailed Archit	{1, By Supplier	None	4	Right	Scale
18	PD_DEV	Numeric	8	0	Development P	{1, By Supplier	None	4	Right	Scale
19	PD_UTEST	Numeric	8	0	Unit Test Perfo	{1, By Supplier	None	4	Right	Scale
20	PD_SYSTEST	Numeric	8	0	Systems Test	{1, By Supplier	None	4	Right	Scale
21	PD_ACCEPT	Numeric	8	0	Acceptance P	{1, By Supplier	None	4	Right	Scale
22	BodyLeasing	Numeric	2	0	Work at custo	{1, No}...	None	4	Right	Nominal
23	ProjectStatus	Numeric	2	0	Project Status	{1, Ongoing}...	None	8	Right	Nominal
24	ProjEvaluation	Numeric	2	0	Project Succe	None	None	5	Right	Nominal
25	Motivation	Numeric	2	0	Motivation	{1, Budget over	None	3	Right	Nominal
26	Successful_project	Numeric	8	0	Successful pro	{1, Successful	None	8	Right	Scale
27	Somewhat_succes	Numeric	8	0	Somewhat suc	{1, Somewhat	None	8	Right	Scale
28	Unsuccessful_proje	Numeric	8	0	Unsuccessful	{1, Unsuccessf	None	8	Right	Scale
29	R1	Numeric	2	0	Customer has	{1, Yes}...	None	8	Right	Scale
30	R1_IMP1	Numeric	1	0	Unexpected m	{0, 0}...	None	8	Right	Scale
31	R1_IMP2	Numeric	1	0	Budget overrun	{0, 0}...	None	8	Right	Scale

Data View | Variable View

A.3. INSTRUCTION FOR SPI APPROACH APPLICATION

APPROACH FOR SUSTAINABLE PROCESS IMPROVEMENT

INTRODUCTION

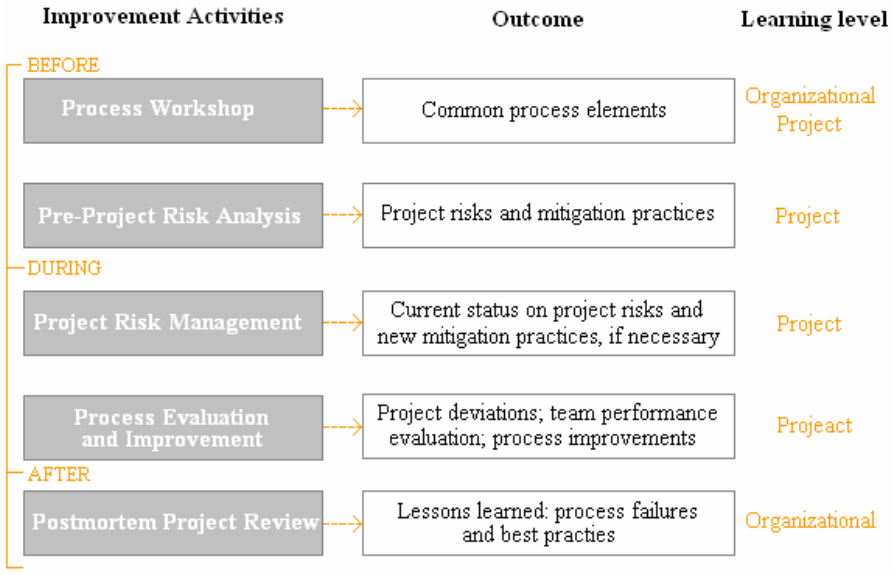
This document describes a compilation of known software process improvement methods and techniques composed by Darja Šmite in an approach for sustainable process improvement for software development projects. It aims in compensating the existing problems with improvement initiatives by direct project personnel involvement and agile style of the prescribed activities.

Software process improvement (SPI) aims in improving software quality and reliability, employee and customer satisfaction, and return on investment. SPI activities performed by project management on a regular basis ensure organizational learning and prevents from a loss of explicit knowledge and experience. It also supports the ISO quality system improvement.

Activities described in this document shall be part of every project. Thus, due to different reasons can be applied on a selective basis too.

OVERVIEW

The described approach is based on a set of methods used in continuous software process improvement models. The activities are seen on the following scheme.



This approach focuses on an active collaboration between the team members. Application of this approach shall be supported by Risk Management Repository and Knowledge and Experience Repository. Each activity in more detail is described in the following chapters.

PROCESS WORKSHOP

1. OBJECTIVES

Due to the fact that formal modeling of processes may easily be overdone and is anyhow not enough to ensure developer motivation and hence process conformance, process workshops aim to tailor project processes in a form of structured, workflow-oriented, electronic reference documents, in order to support participants with a common understanding of project processes and an electronic process guide as a part of the quality system.

2. PARTICIPANTS

The more project members are involved the better the process guide will reflect real process needs and will further be accepted by developers.

3. CONTENT OF A PROCESS GUIDE[9]

A process guide should include the following basic elements:

- Activities: descriptions of "how things are done", including an overview of the activities and details regarding each individual activity.
  Description of tasks for the most important roles in a Project
  Checklists for each main process
- Artefacts: details regarding the products created or modified by an activity, either as a final or intermediate result of the activity or as a temporary result created by one of the steps.
  Templates for all documents produced
  Descriptions of best practice
- Roles: details regarding the roles and agents involved in performing the activities.
- Tools and Techniques: details regarding the tools and techniques used to support or automate the performance of an activity.
  Access to all tools needed in the project (e.g. a requirement and a bug track system)

A common way to describe processes is to describe process entry, tasks, verification and exit, where entry and exit are criteria needed to be fulfilled and the tasks describe activities, roles, artefacts, tools and techniques.

[9] Description is based on "A Workshop-Oriented Approach for Defining Electronic Process Guides. A Case Study" by Torgeir Dingsøyr, Nils Brede Moe, Tore Dybå and Reidar Conradi

4. DESCRIPTION OF A PROCESS WORKSHOP

The following activities shall be performed to defined processes for a project process guide:

1. Decide on process(es) to define. Processes such as “Project initiation”,

2. Prepare for the workshop. Each workshop shall have a leader. Running a process workshop requires a meeting room, collection of stickers in different colours, and walls covered with paper for attaching stickers and drawing figures. A digital camera is useful to document the results of the workshop. Preparation concludes with participant invitation.

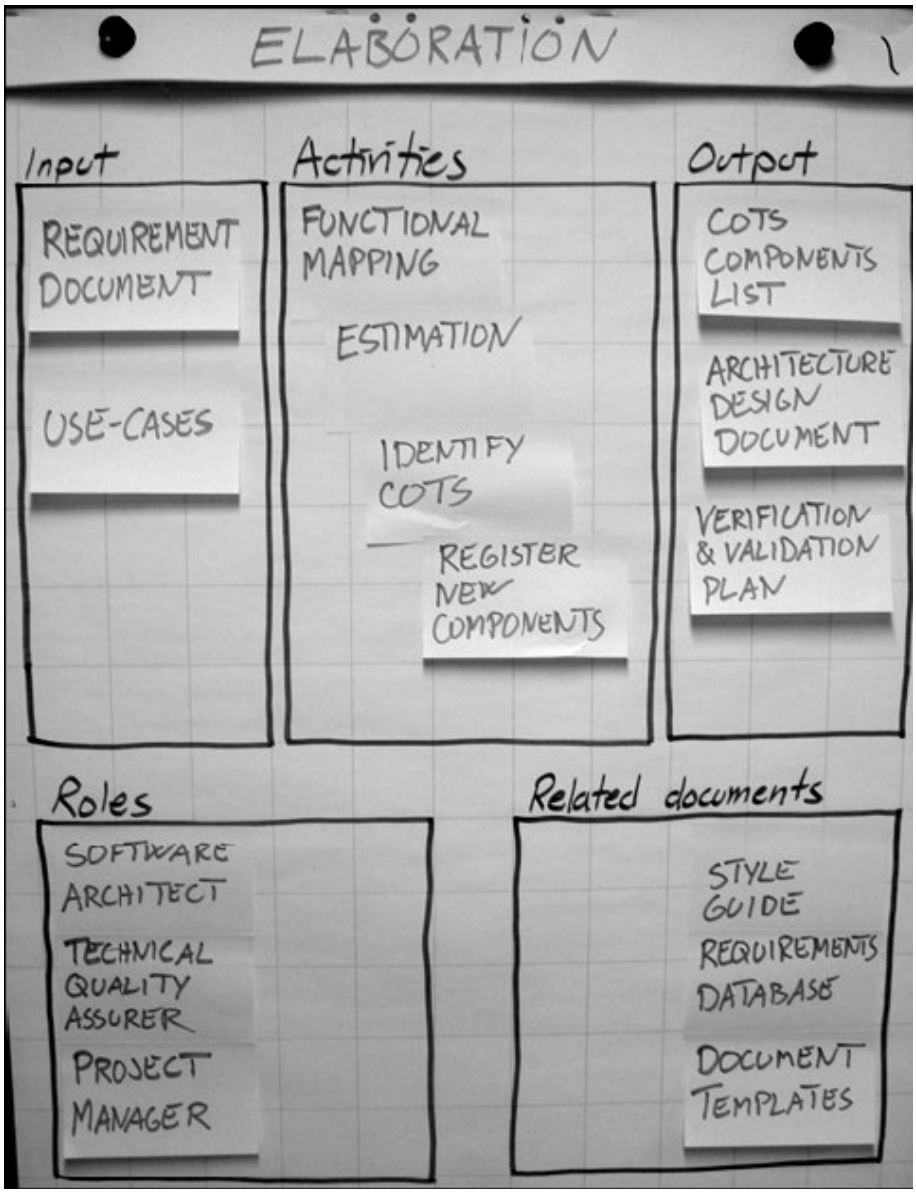
3. Process workshop usually lasts half of a day and includes one process definition. It starts with a 15 min introduction about procedure. Discuss what you are going to do and what is expected from participants. To describe each process, the following steps shall be followed:

> 3.1. Identify activities by brainstorming. Each participant receives yellow stickers and writes 5-10 ideas suggestions for activities on each sticker in large letters. Then each participant presents each idea and attaches it to the wall. All suggestions afterwards shall be grouped and assigned a heading.
>
> 3.2. Define the sequence of the activities. A suitable workflow between the activities shall be found, drawing a time scale and attaching the stickers.
>
> 3.3. Define input and output. Find documents or artefacts that must be available to start the process, and documents that mark the end of the process. Use stickers with different colours to mark inputs and outputs, and attach them on the process worksheet on the wall together with the activities. Conditions that must be satisfied to begin or exit the process can be described in checklists.
>
> 3.4. Define roles. Organize a brainstorm for roles that should contribute in each activity.
>
> 3.5. Find related documents. Identify documents that either already exist in the company, or new documents that would be helpful in carrying out the activities, for example, templates, checklists and good examples of input or output documents.

4. Validate resulting processes. The documented processes shall be revised by participants to validate that there was no mistake or misunderstanding.

5. Implement the process in an electronic process guide. An electronic process guide, e.g. in the form of a project intranet shall be established upon the results from the process workshops.

An example of workshop results can be illustrated by the following picture.



PRE-PROJECT RISK ANALYSIS

1. OBJECTIVES

Pre-project risk analysis aims to identify all possible negative events and threats that can endanger project and partner collaboration. Awareness of the project threats succeeds timely risk mitigation and avoidance.

2. PARTICIPANTS

Pre-project risk analysis shall be performed by project manager and project main stakeholders. It is advisable to discuss results of the risk analysis within the steering committee of the project, including remote stakeholders.

3. DESCRIPTION

Pre-project risk analysis can be organized in a way of either a project meeting for all project members or a joint task for project stakeholders. Risk analysis shall follow the following steps:

1. Risk identification. Use historic information from previous projects accumulated in the Knowledge Base for more convenient risk identification. Go through the list of possible project threats and evaluation if these threats are feasible in your project.

2. Risk evaluation. Evaluate probability and impact of the threat, if it comes true. Historic information from previous projects shall help you with evaluation. Your estimates can be either quantitative (consider exact cost estimation) or qualitative (using the following scales for probability: Improbable, Doubtful, Unlikely, Possible, Probable, Certain; and impact: None, Negligible, Minor, Moderate, Significant, Disastrous).

3. Risk mitigation activity selection. Go through recommendations given in the Knowledge Base for certain risk mitigation and select appropriate solutions. Prioritize the list of mitigation activities for further planning.

4. Assigning responsibility and further action planning. Discuss risk analysis results with the stakeholders and/or project steering committee. Assign responsibilities for implementing risk mitigation activities and document them in an action plan.

PROJECT RISK MANAGEMENT

1. OBJECTIVES

Project risk management shall be held on a regular basis throughout the project involving all project members. Involvement of project personnel brings awareness of what may happen and what to do to prevent it. This may be especially critical to provide joint action when the problem symptoms occur. All project members shall be aware of dependences and relatedness with the remote team members. Therefore, risk management shall be performed both internal level for each partner involved in the project, and inter-organizational level, involving at least steering committee and major stakeholders.

Risk monitoring shall be seen as an integral part of regular project meetings.

2. PARTICIPANTS

Project risk management shall be performed by the project manager with project team and/or project stakeholder involvement.

3. DESCRIPTION

The results of pre-project risk analysis shall be monitored and maintained on a regular basis as an integral part of project meetings. The following activities shall be performed.

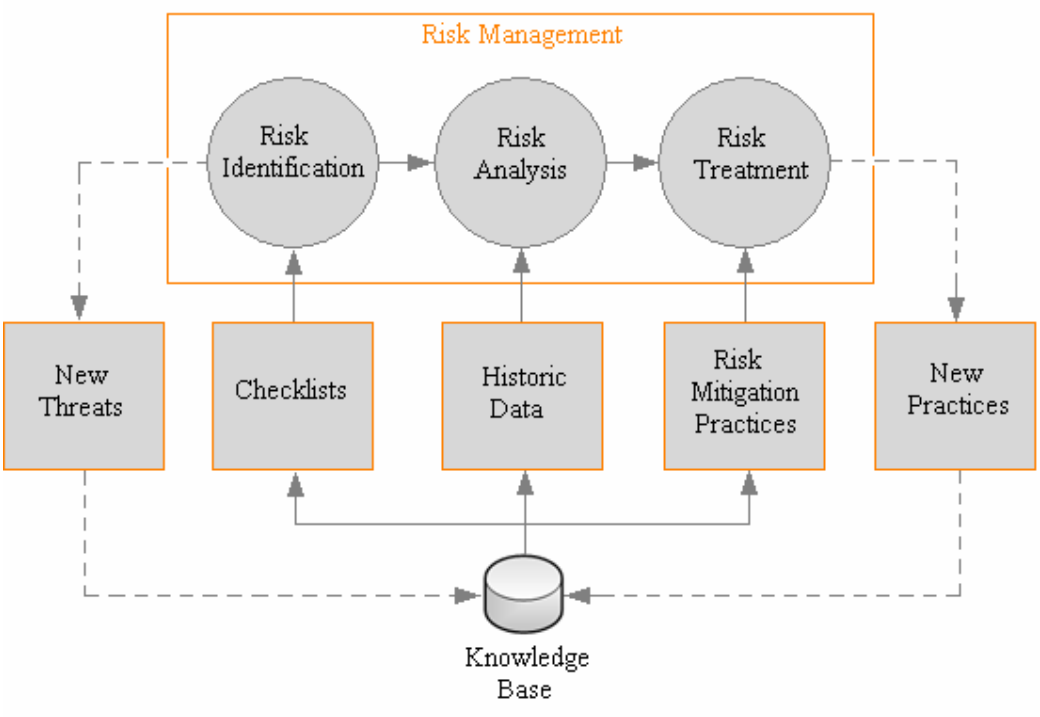
1. Risk identification. Project team members shall be able to participate in brainstorming and report considerations of possible sources of project risks. Team members may often see more symptoms of project threats than project managers.

2. Risk evaluation. These considerations shall be then discussed and evaluated. Evaluation shall follow the same procedure as selected for pre-project risk analysis.

3. All sources of project risks shall be documented for further maintenance and reported to project stakeholders. Use templates to document and maintain project risk statuses for more convenience.

4. Mitigation activities shall be then selected and planned with responsibility assignments. Use Knowledge Base for best practice selection.

To motivate sustainable software process improvement and knowledge transfer between different projects and project teams risk management process is integrated with application of an Experience Factory or Knowledge Base according to the following scheme.



PROCESS EVALUATION AND IMPROVEMENT

1. OBJECTIVES

Poor process performance is more than a headache for a project. Consider organizing process evaluation and improvement sessions to revise your project performance. This process aims identifying weak areas of the project and implementing improvement solutions. It is particularly feasible in large and/or long-term projects.

Process improvements are essential for effective performance. The updates prevent from following outdated processes. Joint process workshops and review meetings strengthen team spirit.

2. PARTICIPANTS

Process Evaluation and improvement shall be performed by the project manager with or without project stakeholder involvement. Evaluation and improvement meetings may also involve all project members. This will provide a wider view of project problem and assure acceptance of the implemented solutions.

3. DESCRIPTION

Process guide developed in the beginning of the project shall be evaluated and updated on a regular basis. In addition, process performance evaluation shall be done regularly throughout the project. This can be done by organizing a project review meeting in accordance to post-mortem review method (see below). Project members shall a) evaluate existing processes in relation to the process guides and identify updates; b) evaluate effectiveness of process performance and suggest improvements. This provides sustainable process improvement and prevents from problems in relation to following the outdated processes.

The output of this process shall include improved and/or updated process guides and corrective activity plan.

POST-MORTEM PROJECT REVIEW

1. OBJECTIVES

Post-mortem project reviews based on the project team meetings aim to focuses on describing project benefits and problems, and provide knowledge and experience accumulation after the end of a project. Such meetings can be held at the end of the project (post-mortem reviews) and also during the project for process efficiency evaluation and project problem analysis.

2. PARTICIPANTS

Post-mortem project reviews are traditionally managed by a leader, e.g. an SPI expert. All members of the project shall participate, if possible.

3. AGENDA

Meeting agenda for a 3 hours seminar is the following:

00.00 Introduction too PMA

00.10 Brainstorming/KJ: What in the project is a success?

Presentation of experiences using stickers

00.30 Structure and priority of experiences

00.45 Brainstorming/KJ: What in the project is not a success?

Presentation of experiences using stickers

01.10 Structure and priority of experiences

01.25 Small break

01.30 Root cause analysis: What was the cause of the main problem(s)

(1-2 fishbone diagrams)

02.00 Suggest actions to solve the problems. Solution prioritization.

02.15 Root cause analysis: what was the cause of the success?

02.40 Suggest actions to repeat the success. Prioritizing these

02.50 Summing up of the seminar and further plans.

03.00 End

4. DESCRIPTION

The following steps shall be performed to run a post-mortem review[10].

1. Select a chairman for the meeting. It may be wise to use a chairman from outside the project. In this way it may become easier to focus on "what should a new project team do" instead of making the meeting a more or less unreflected review of "what we did".

2. Go over project purposes and project delivery. Start the meeting with a short review of "what should have been done" and "what was actually done". Also pay attention to whether time limits were kept or not, and if the quality of the delivery met with the expectations. If the customer is there, ask: "Did you get what you had expected?"

3. Ask: "What in the project was a success?" Always start with the positive. We all want to use our successes as a foundation for future work and avoid repeating mistakes. Ask: "What were the successful steps to achieve project objectives?", "What in the project was really a success?", "What were the causes for the successes?" Identify the success factors so that they can be repeated in the future. Try sticking to the facts. Opinions may be good, but recommendations to future projects should as far as possible be based on facts that the projects members can agree on. Focus on specific advice that can easily be implemented into new projects. If you are running out of time, ask: "What was the most important success factor?"

4. After describing successful aspects of the project, ask: "What could have been better?" There certainly are areas where the project could have been more successful, where pitfalls were identified to late and where the process was anything but optimal. Ask: "What was not so successful?" Do not allow any sort of blaming to take place, rather encourage people to say: "My opinion of what happened is different." Remember that all opinions are just as valuable.

5. Find out what the problem really was. Identify pitfalls and obstacles so that they can be avoided in the future. Ask: "What were the causes for the less successes parts?" Given our present knowledge, what could we have done better? What advise would you give a future project team based on the experience from this project?"

6. Identify actions. If a project team is about to start a new project similar to the one they just have finished, it could be useful to follow up the PMA with a start-up meeting for the new

[10] Description is based on „Process improvement in practice. A handbook for IT companies" by Tore Dybå, Torgeir Dingsøyr, and Nils Brede Moe. Published by Fagbokforlaget Vigmostad & Bjørke AS, 2002

project. In this way you can close the learning loop. Action is important. No improvement is possible without planning certain actions and assigning responsibilities.

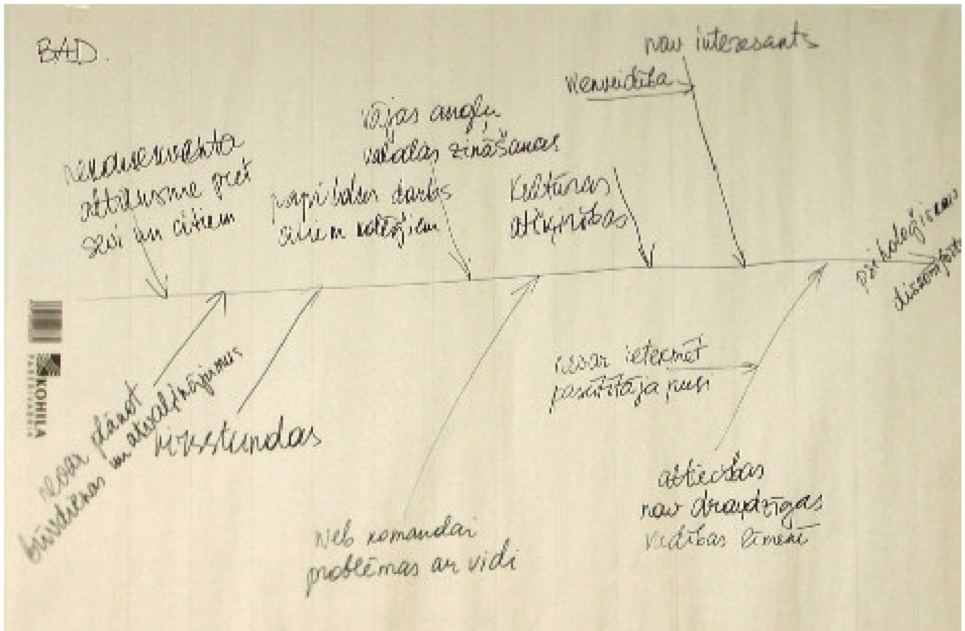
7. Experience must be documented. It is important to make a good summary from the closing learning meeting in the project. Moments to consider are: advice and hints, experiences, casualties, actions, illustrating examples, name of project members and references to documents, models, guidelines, checklists and key personnel. It is a good idea to use quotes of what has been said to make the presentation livelier. The acid test on whether the summary is useful or not, ask yourself: “If I were a project leader for the next project, would this summary be useful to me?" It is therefore important to identify actions, for example to incorporate the new knowledge into updated processes, procedures, checklists, models or Knowledge Base.

5. RESULTS

The main results from a PMA can be the following:

- knowledge about the process that has been carried out, knowledge that can be reused;
- hints and experience notes in a knowledge base;
- new and improved checklists;
- updated and improved development model;
- better understanding for the participants of what was good and what could have been better.

The results of the PMA shall be documented in a protocol. Camera shots can included into the protocol as pieces of evidence (see examples below).


BAD.


GOOD!
Laba komanda

A.4. PROJECT RISK ANALYSIS

SE-SOFTWARE AND LAT-SOFTWARE COOPERATION

PROJECT RISK ANALYSIS

1. INTRODUCTION

This document describes the results of informal threat analysis performed on October 10th by LAT-SOFTWARE project manager KR and LAT-SOFTWARE consultant Darja Šmite.

The purpose of this threat analysis is to identify areas of concern that may become more serious if not addressed in the early stages of the relationship. Conclusions based on this risk analysis should be kept in mind when meeting with SE-SOFTWARE representatives responsible for cooperation planning

2. OVERVIEW

SE-SOFTWARE is an international consultancy firm within the IT, ERP systems and business development sectors. SE-SOFTWARE major services are focused strongly on ERP systems including mySAP Business Suite, mySAP All-in-One and Microsoft Business Solutions – Axapta ERP systems.

Lately, SE-SOFTWARE has been planning switching its operation to distributed mode, outsourcing coding of software components to an external provider. LAT-SOFTWARE has been seen as a strategic partner for providing outsourcing services of software coding. Cooperation scheme involving end customers, SE-SOFTWARE and LAT-SOFTWARE can be seen in the figure below.

Fig.1. Cooperation scheme

Cooperation has been started by developing a small piece of software – adapter for SETMTS SYSTEM integration with external data sources. This pilot will show LAT-SOFTWARE's ability to perform, as well as cooperation issues to work on. In time, this cooperation has

potential to grow into long term relationship where distributed teams from SE-SOFTWARE and LAT-SOFTWARE will work with joint effort on larger software development projects.

3. RISK ANALYSIS OVERVIEW

3.1. Approach

SE-SOFTWARE and LAT-SOFTWARE cooperation risk analysis has been performed on October 10th by LAT-SOFTWARE project manager KR and consultant Darja Šmite. A list of specific factors and threats, that are peculiar for multi-sourced projects in geographically distributed environment, has been used for risk identification. Areas of concern and issues for monitoring have been derived for further mitigation and monitoring.

3.2. Risk Overview

Cooperation between SE-SOFTWARE and LAT-SOFTWARE has a good potential. It is started by small steps, therefore preparing the ground for better understanding and planning of the further relationship. SE-SOFTWARE stakeholders' involvement and interest in the cooperation is an essential factor for success. Nevertheless, special attention needs to be paid to personal contact and joint performance in strategic planning.

The factors that can threaten possible long term cooperation are twofold. First, these are LAT-SOFTWARE related areas of concern that are critical for successful cooperation. Second, these are SE-SOFTWARE related areas of concern that require careful attention, initiative and consultancy from LAT-SOFTWARE side.

The area of concern in this and other distributed projects is increasing complexity of globally distributed team management. This threat is connected with necessity for joint procedures, methods and tools for effective collaboration in distributed environment. Lack of effort estimation procedures for distributed projects may cause faulty effort estimates and drive to budget and calendar overrun. Incomplete requirements specifications delivered by the customer may cause time delays for requirements clarification and trouble software development. Differences in work practices between SE-SOFTWARE and LAT-SOFTWARE may result in unexpected costs for LAT-SOFTWARE and SE-SOFTWARE and cause delivery delays due to project managerial problems.

The risk of lacking resources to fulfil the needs of SE-SOFTWARE in further cooperation requires timely planning. In addition, developers with good English language skills shall be employed, while forming the development team. Lack of language skills may create problems in communication, causing misunderstandings and time delays.

Other threats can be divided into the following groups:

- Increased complexity of working in distributed environment – shall be addressed jointly
    - Communication barriers due to language differences
    - Increased cost and logistics of holding face to face meetings
    - Uncoordinated communication
    - Inefficient and ineffective communication due to geographic distribution
    - Poor distribution of tasks in the development lifecycle across distributed teams
    - Necessity for collaboration tools in distributed environment
    - Technological discontinuity
    - Terminology differences
- Customer and supplier fears – shall be monitored in long term
    - Poor cultural fit
    - Lack of trust and commitment
    - Customer belief that the work cannot be done from a far off location
    - Lack of direct control over outsourced resources due to increased virtualness
    - Necessity for organizational changes
    - Active opposition against collaboration with external parties due to organizational changes

Some of the threats need joint planning for mitigation, other are yet not observed and need further monitoring.

4. LIST OF RISKS IN DETAIL

This chapter provides description of the threats identified during the process of project risk identification, followed by mitigation recommendations. This table can be further used in continuous project risk management.

The prioritized list of areas threats is given in the table below. Priorities have been assigned by the project manager in accordance with possible impact on project results. Priority values: H – high, M – medium, L – low, considering LAT-SOFTWARE interests.

Table 1. Detailed list of risks

Area of concern	Priority	Threats / Solutions	Owner(s)	Comments
Lack of resources to fulfil the needs of SE-SOFTWARE	H	**Threat:** • LAT-SOFTWARE will not have enough developer resources for the upcoming projects **Solution** • Employ more .NET programmers	LAT-SOFTWARE	No comment
Increased complexity of distributed multiple team management	H	**Threat:** • Multisourcing requires new approaches in project management. o Lacking risk management can cause late reaction to deviations. o Lacking conflict management procedures can cause time delays. o Lack of effort estimation procedures for distributed projects (see below) o Necessity for increased level of requirement elaboration (see below) **Solution** • Risk management procedures should be developed and implemented • Joint conflict management procedures should be developed and implemented	LAT-SOFTWARE, SE-SOFTWARE	SE-SOFTWARE is aware and further meetings with LAT-SOFTWARE are planed

Lack of effort estimation procedures for distributed projects	H	**Threat:** • Faulty effort estimation will be performed if a method is not provided, causing budget and calendar overrun. **Solution** • Develop and implement a reliable method for project effort estimation in distributed environment. This methodology should be adopted by SE-SOFTWARE and LAT-SOFTWARE jointly. • SRS should be written in detail for further usage by formal effort estimation methods	LAT-SOFTWARE	Has to be developed
Necessity for increased level of requirement elaboration	H	**Threat:** • Incomplete requirements specifications delivered by SE-SOFTWARE will cause time delays for requirements clarification and trouble software development. **Solution** • Templates for requirements specification and supporting instructions are being developed by LAT-SOFTWARE. • SE-SOFTWARE is also reviewing the method for SRS development.	LAT-SOFTWARE, SE-SOFTWARE	SRS template is being developed and hopefully will be adopted by SE-SOFTWARE.

Differences in work practices	H	**Threat:** • Inconsistency in work practices and diversity of process maturity between the partners may result in unexpected costs for LAT-SOFTWARE and SE-SOFTWARE. • This can also cause delivery delays due to project managerial problems. **Solution:** • Requires initiative from LAT-SOFTWARE and SE-SOFTWARE. • Assign responsibility for planning joint procedure development and implementation.	LAT-SOFTWARE, SE-SOFTWARE	Has already been discussed with SE-SOFTWARE and will need to be further elaborated.
Communication barriers due to language differences	M	**Threat:** • Lack of English language knowledge troubles communication. • This may threaten understanding of the requirements specification during software development. **Solution** • Involve one or two systems analysts-programmers (in case of further cooperation) with good language skills for software development team supervision.	LAT-SOFTWARE	Need to ensure that we involve team members with good English language skills

Increased cost and logistics of holding face to face meetings	M	**Threat:** • Distribution in space causes extra costs for holding face to face meetings. • The project is lacking agreement on meeting logistics and budget. **Solution** • An agreement about budget and calendar planning for face to face meetings shall be achieved between SE-SOFTWARE and LAT-SOFTWARE .	LAT-SOFTWARE, SE-SOFTWARE	Needs to be discussed in upcoming meetings with SE-SOFTWARE
Uncoordinated communication	M	**Threat:** • Chaotic communication due to multiple lines of communication. **Solution** • Communication lines should be defined and adhered to.	LAT-SOFTWARE, SE-SOFTWARE	Has already been discussed with SE-SOFTWARE and will need to be further elaborated
Inefficient and ineffective communication due to geographic distribution	M	**Threat:** • Time delays due to excessive use of email. **Solution** • Implementing use of phones, messengers, and/or video-conferences. • Find more effective ways of using email.	LAT-SOFTWARE, SE-SOFTWARE	Has already been discussed with SE-SOFTWARE and will need to be further elaborated

Poor distribution of tasks in the development lifecycle across distributed teams	M	**Threat:** • Lack of unit testing activities by the LAT-SOFTWARE team can cause low quality of the product delivered and customer dissatisfaction • Lack of LAT-SOFTWARE participation in project planning may cause budget and calendar deviations due to inaccurate effort estimates. **Solution** • Unit testing needs to be performed by LAT-SOFTWARE. SE-SOFTWARE needs to supply LAT-SOFTWARE with valid data to be used for unit testing. • Possibly create a Steering Committee (SC) and Project Change Control Board (PCCB) with both SE-SOFTWARE and LAT-SOFTWARE involvement to perform strategic (SC) and operational (PCCB) planning.	LAT-SOFTWARE, SE-SOFTWARE	Template for SRS will contain section for unit test data. SC, PCCB development needs to be discussed in future meetings with SE-SOFTWARE.
Necessity for collaboration tools in distributed environment	M	**Threat:** • Lack of collaboration tools for information storage, version control, configuration management, etc. may cause misunderstandings, time delays, and loss of information. **Solution** • A project environment for information storage may be implemented • A tool for configuration management may be implemented • A tool and procedure for version control may be implemented • A tool for problem report tracking may be implemented	LAT-SOFTWARE, SE-SOFTWARE	SE-SOFTWARE is currently investigation this

Technological discontinuity	M	**Threat:** • Time delays have already been caused by technological discontinuity due to external entities that provide software interfaces to the system being developed. **Solution** • Timely access to the external sources should be provided by SE-SOFTWARE.	SE-SOFTWARE	Inform the Customer and monitor to ensure that they react
Terminology differences	L	**Threat:** • Terminology differences can cause misunderstandings and result in rework of the implemented software. **Solution** • A terminology dictionary may be developed.	LAT-SOFTWARE, SE-SOFTWARE	Template for terminology dictionary is being developed
Poor cultural fit	L	**Threat:** • Poor cultural fit may cause misunderstandings and psychological discomfort. Yet not experienced. **Solution** • Watch.	LAT-SOFTWARE	Not yet observed if this is real
Lack of trust and commitment	L	**Threat:** • Lacking personal contact in distributed environment may reduce trust between the partners and cause psychological discomfort and stress. **Solution** • Face to face meetings are very desirable in the beginning of the cooperation.	LAT-SOFTWARE, SE-SOFTWARE	Not yet observed if this is real

Customer belief that the work cannot be done from a far off location	L	**Threat:** • This is the first project for SE-SOFTWARE involving a geographically distributed supplier. Therefore, the customer has fears about supplier ability to perform. Causes desire to proceed by small steps. **Solution** • Requires careful attention and initiative from the LAT-SOFTWARE to develop trust and commitment, and to prove the competence.	LAT-SOFTWARE	Does not appear to be a concern for SE-SOFTWARE at this point
Lack of direct control over outsourced resources due to increased virtualness	L	**Threat:** • Customer fears connected to lack of direct control may cause unexpected effort for progress reporting. • In addition, may cause lack of trust. **Solution** • Implement project management environment with joint access, including project documents, plans, progress reporting, meeting minutes, etc.	LAT-SOFTWARE, SE-SOFTWARE	Not yet observed if this is real
Necessity for organizational changes	L	**Threat:** • Organizational changes are needed by SE-SOFTWARE due to lack of experience in outsourcing projects. The organization is yet in the process of adaptation for managing outsourcing projects. This has already caused time delays in information turnaround. **Solution** • Requires strategic planning from SE-SOFTWARE side. • LAT-SOFTWARE representatives can be used for knowledge sharing.	SE-SOFTWARE	The Customer seemingly is aware of this risk. We need to monitor and mitigate in case of no action.

Active opposition against collaboration with external parties due to organizational changes	L	**Threat:** • SE-SOFTWARE employees that are under threat of being fired due to organizational changes can cause active opposition against collaboration with LAT-SOFTWARE. This can result in psychological discomfort and stress, troubled access to core information whenever necessary, and time delays in problem solution. **Solution** • Requires careful attention and SE-SOFTWARE key personnel involvement	LAT-SOFTWARE, SE-SOFTWARE	Not yet observed if this is real

A.5. PROJECT PROBLEM REVIEW RESULTS

„ASC[11]" PROJECT PROBLEM ANALYSIS SEMINAR PROTOCOL

PROCEDURE

Place:	LAT-SOFTWARE Ganību dambis 17, Conference Room, 14.03.2006 13:00-16:30
Leaded by:	Darja Šmite, RITI
Participants:	Project Manager and all sub-project team leaders.
Method:	Problem analysis seminar is organized according to principles prescribed by the selected postmortem review controlled procedure.
Documentation:	The results of the seminar are being documented. The seminar notes are saved, recorded with a dictophone and a camera and transcribed in this protocol.

GOALS

Project postmortem review is initiated by the project manager in order to identify and discuss project related problems and search for improvements in the project internal atmosphere.

SEMINAR PLAN

13:00 Introduction to seminar procedure

13:10 Brainstorm: What is project successful experience? Presentation of the results, using post-it notes (see Figure 1). Discussion and grouping.

13:50 Brainstorm: What is project unsuccessful experience, project failures? Presentation of the results, using post-it notes (see Figure 1). Discussion and grouping.

14:40 Coffee break

14:45 Failure root-cause analysis. Development of the recommendation plan.

16:20 Coffee break

16:25 Success root-cause analysis. Development of the recommendation plan

16:40 Closure.

RESULTS – PROJECT SUCCESSFUL EXPERIENCE

- Team
  - Good team
  - Appreciation of LAT-SOFTWARE employee professionalism
- Order
  - Well known for everyone what to do, well-assigned responsibility

[11] Project name is changed due to confidentiality

- An opportunity to coordinate own team work
- Processes are stable (2x)
- Quality increased

- Professional growth
  - Opportunity to learn new technologies
  - Opportunity to learn XML data bases application
  - Work processes, know-how experience can be gained in the work environment in the process
  - Opportunity to gain experience about a large project
- Cooperation with the customer
  - Collegiate collaboration in Designer → Developer relations
  - Improved mutual responsiveness
  - Calendar plans became more predictable
  - No conflicts about deadlines (in one of the teams)

RESULTS – PROJECT NEGATIVE EXPERIENCE

- Organizational problems
  - Task assignment procedure is often ignored
  - Team work is very dependent on other team tasks
  - Problem reports are delivered to wrong teams
  - Obviously inadequate problem reports
- Cooperation with the customer
  - Impact on customer team is unlimited
  - Lack of the joint view on the project
  - No overview of customer success / failure
  - Dislike the management culture
- Planning problems
  - No overview of the customer plans
  - Work organization – task flow is not even
  - The customer sometimes makes fault effort estimates
  - The deadlines are sometimes communicated just couple of days / weeks in advance (not enough)
  - Downtime in CORE API and EJB teams
  - Overload and overtime in WEB form team
- Technological problems
  - WEB forms for the team

  - „ICS Framework" is complicated – training: 2 month per programmer
  - EBase – veri uncomfortable tool. It drives to several quality problems, that could be solved during compilation, but with EBase it is not possible.
  - Work environment through Citrix – unpleasantly slow and sometimes unstable
  - Unit testing is not automated. Integration testing can also be performed only by delivering the work to QA team.
  - Code transparency problems with EBase
    - Problems with work environment installing / maintenance
- Motivation problems
  - Project is not interesting any more
  - Hopeless situation with the customer
  - Standardization, templates – for the whole project it is good, for quality good, but for team members to be interested in the project – it is bad.
  - Customer assigns only uninteresting tasks

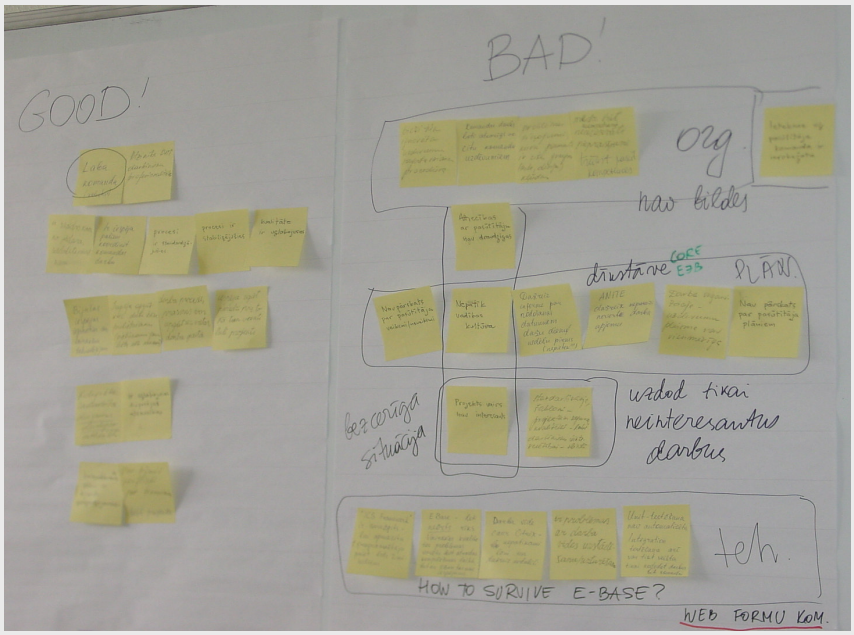


Figure 1. Project positive and negative experience ideas

DECISIONS AND PROPOSALS

- Internal meetings
  - Organize internal project meetings involving all 3 teams once in 2 weeks
  - Agree on the certain day (Fridays are proposed)
  - Organize unformal social events afterwards

- Cooperation with the customer
  - Continue to propose improvements
  - Use upper management from the Steering Committee to enounce improvements and proposals
  - Invite the customer management to Latvia
- Process improvements
  - Continue improvements of internal project tasks
  - WEB form team: develop work procedure. Ask the customer to do it.
  - Learn about a possibility to organize English language courses
  - Motivate independent work in WEB form team (due to lack of English language skills)
- Solution for technical problems
  - VPN implementation
  - Monitor and escalate Interneta connection problems
- Atmosphere
  - „Activate" work environment (with aquarium, posters, flowers)
  - Organize social events

A.6. PROJECT POST-MORTEM REVIEW RESULTS

„AT [12]" PROJECT POSTMORTEM REVIEW PROTOCOL

PROCEDURE

Place:	LAT-SOFTWARE Ganību dambis 17, Conference Room, 13.04.2006 13:00-16:00
Leaded by:	Darja Šmite, RITI
Participants:	Project Manager and 10 Project Members [13].
Method:	The seminar is organized according to postmortem review controlled procedure.
Documentation:	The results of the seminar are being documented. The seminar notes are saved, recorded with a camera and transcribed in this protocol.

GOALS

Project postmortem review is initiated by the project manager in order to identify and discuss project successes, failures, the whys and wherefores, as well as produce a list of recommendations for further projects.

SEMINAR PLAN

13:00 Introduction to postmortem seminar procedure

13:10 Brainstorm: What is project successful experience? Presentation of the results, using post-it notes. Discussion and grouping.

13:40 Brainstorm: What is project unsuccessful experience, project failures? Presentation of the results, using post-it notes. Discussion and grouping.

14:40 Coffee break

14:55 Failure root-cause analysis. Development of the recommendation plan.

15:40 Closure; non-formal follow-up.

RESULTS – PROJECT SUCCESSFUL EXPERIENCE

- Project management
  - Close contact with systems analyst during testing and good project management
  - Project manager considered my objections against several customer requirements (due to impossibility of technical realization)
  - Not too "intrusive" and not too careless manager control
- Process order

[12] Project name is changed due to confidentiality

[13] Not mentioned due to confidentiality

- Sufficiently well established project processes
- Work order
- Planing
- Analysis
- Timely developed and sufficient requirements specifications
- Deadlines
- Performance
- Testing (2 times)
- TraceIT tool usage
- Decision to write test cases in the beginning of the project improved specifications
- Opportunity to create and maintain GUI specifications synchronized with real code and visa versa.

- Cooperation with the customer
  - Opportunity directly communicate with the customer without any mediating partner
  - Right-minded customer
  - Good cooperation with the customer
- Team
  - Good colleagues
  - Professional team
  - Project team
  - Team
  - Good team
  - Good working team
  - Cooperative programmers after receiving problem reports from the tester
- Professional growth
  - Actuality
  - Interesting system
  - New business domain learning
  - Modern technology usage
  - Learned a lot of new things. From architecture and development point
  - New experience in component development
  - New experience in Web programming, form development etc.
  - Learned ASP
  - Modern environment selection (ASP.NET), new experience
  - Opportunity to learn automative testing method – IE DOM
  - Experience
  - Technologies
  - Opportunity to learn new development platform

- Meetings
  - Weekly meetings
  - Weekly conferences
  - Friday team meetings
- Different
  - Cooperation in remote mode
  - Opportunity to work in Rezekne for a week
  - Extra project that uses TraceIT
  - Well established technological infrastructure
  - Everything came off well
  - Project funeral

RESULTS – PROJECT NEGATIVE EXPERIENCE

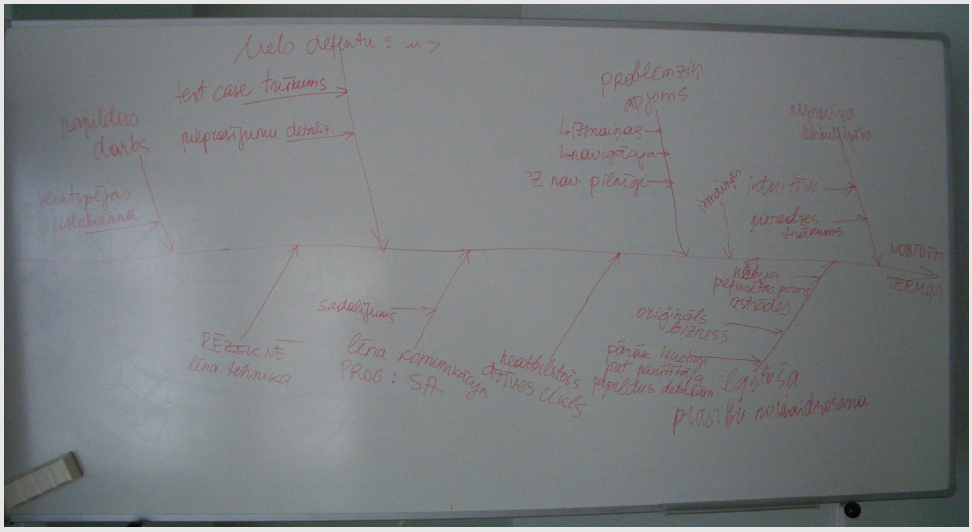
- Requirements
  - Too many changes in requirements during the development process
  - Additional requirements
  - Late and inconsistent customer requirements
  - Changes in requirements
  - Protracted ambiguity in requirements
  - Last changes are not included in specifications
- Deadlines
  - Once again project with overdue deadlines
  - Work during weekends
  - Underestimated project effort
  - Sometimes there was an awfully great work amount
  - No time for performance tests
- Number of defects
  - To my mind, high number of defects
  - High number of defects
- Technological problems
  - Slow computer – timely compilation, testing etc.
  - Low-powered line with the customer (technical resources)
  - Equipment (2 times)
- Technical architecture
  - Navigation
  - Insufficient consideration and weak realization of navigation between the forms
  - Several technology unjustified usage (e.g. sessions)
  - Architecture failures (presentation); session, cache usage
  - Procurement documents didn't have suitable architecture

- Cooperation with the customer
  - Suspicion that the customer didn't fully pay for the work
  - Customer prohibited to program part of the specification
  - Customer didn't require sufficiently detailed application usage manual
- Infrastructure
  - GD 17, 112 room (2 times)
- Different
  - Didn't have an opportunity for business trip to Norway
  - New unknown technology in the beginning
  - Insufficient unit testing
  - Insufficient responsibility in completion of the tasks
  - Supplier team was distributed (Rīga – Rēzekne)
  - Not enough controlled requirement specification
  - Problem reports are not well considered

PROBLEM REASONS

Root-cause analysis was performed to uncover the reasons for overdue deadlines:

- Overdue deadlines:
  - Underestimated effort
    - Due to intuitive estimation
    - Due to lacking experience
  - Changes
  - Timely requirement specification
    - Due to undiscovered requirements
    - Due to specificity of business domain
    - Due to too tolerant attitude towards customer extra requirements
  - Great amount of problem reports
    - Due to changes
    - Due to problems with navigation
    - Due to gaps in problem reports
  - Inappropriate life cycle
  - Slow communication between programmers and systems analysts
    - Due to geographic distribution
  - Increased number of defect
    - Due to lacking test cases
    - Due to inadequate requirement detalization
  - Slow technology infrastructure in Rēzekne
  - Extra work
    - Due to necessity of performance improvement



Picture 1. Fishbone diagram – Reasons of overdue deadlines

FURTHER PROJECT IMPROVEMENT RECOMMENDATIONS

- Repetition of successful experience
  - Cooperation with the customer
  - Template usage decreases probability of failure / defects
- Avoidance of negative experience
  - Life cycle selection (Agile)
  - Problem reports classification implementation (for further analysis)
  - For new requirements –
    - Changes in SRS or
    - Test cases
  - Assure that all team members have sufficient equipment and technologies in the beginning of the project

FUTURE PROJECT IMPROVEMENT SUGGESTIONS AND LESSIONS LEARNED

- Positive experiences shall be repeated
  - Collaboration with the customer
  - Usage of templates decreases errors occurrence
- Avoid identified problems
  - Life cycle selection (Agile)
  - Unclassified problem reports – classification shall be established for further analysis possibilities
  - New requirements shall be described in –
    - SRS changes or
    - Test cases
  - Project initialization shall address requirements and validation of sufficient hardware and technological support

A.7. POST-MORTEM REVIEW MEETINGS EVALUATION

Survey Questions	Participant 1	Participant 2	Participant 3
1. Would you like to participate in a post-mortem meeting again?	Yes. It would be useful to refine and put attitudes in order.	Yes	Yes
2. Would you recommend it for other projects?	Yes	Yes	Yes
3. Did you like the method used?	Yes. I have never participated in such meetings before. It seemed interesting and also useful, because it resulted in a list of possible project improvements!	Quite interesting. Allows considering different aspects of the project.	Since it was my first experience to participate in this type of meetings, in the beginning I was sceptical, but at the end I liked it.
4. Did the meeting uncover new findings for you?	No, there were no surprises for me.	I was surprised that I am not the only person that is not interested in the project any more.	I discovered more positive experiences in the project that I could imagine myself.
5. What are the benefits that you gained?	A wider view on what and how happens in the project. An opportunity to compare this project to my personal observations and experience from other projects.	So far difficult to say.	Time will show.
6. Would you suggest any change to the method?	So far it seems that there's no need for improvements	I think that it would be useful to analyze not only current situation, but also imagine expectations deriving ways to achieve them.	So far I have no advices

A.8. KNOWLEDGE BASE USER GUIDE

PraDis Knowledge Base

1. Objectives

PraDis Knowledge Base is an Experience Factory that aims to provide useful information about global software development projects. It prescribes active knowledge and experience transfer between different globally distributed projects in the organization and a central point for new employees to be introduced with the current practices and previous experiences.

2. Users

PraDis Knowledge Base mainly serves as a tool for globally distributed software project managers. However, it can also attract other user groups as seen on Figure 1.

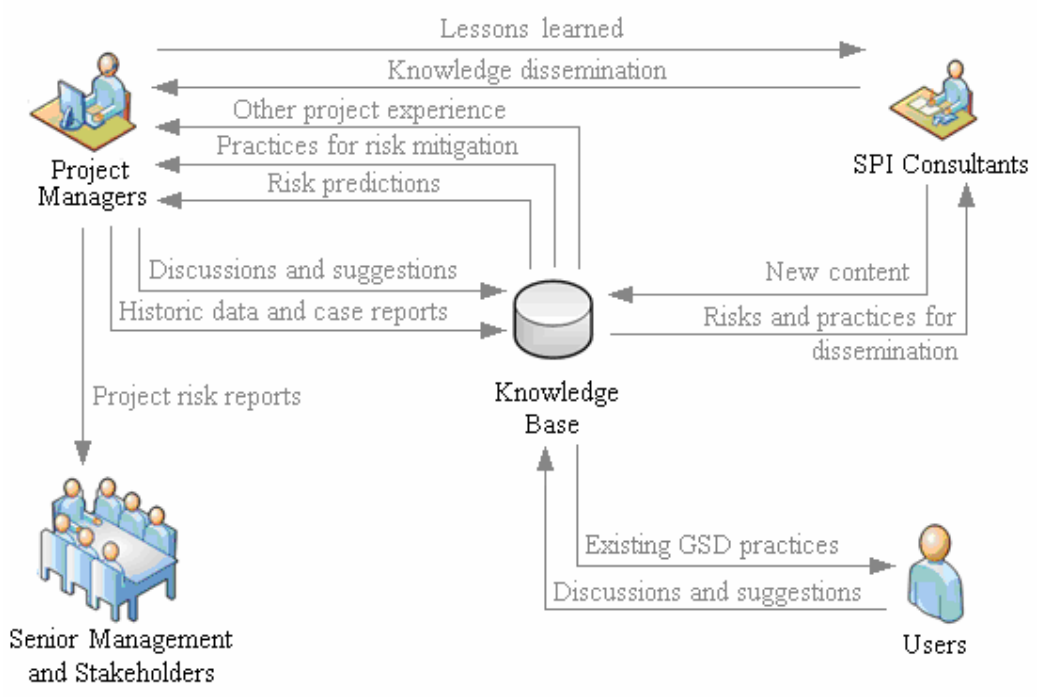


FIGURE 1. Knowledge Base and its users

Project managers can read the knowledge and experience items, suggest new items, suggest and participate in discussion topics, use templates, read and add case study reports and historic data from previous projects.

Other users can use Knowledge Base as a source of existing practices from GSD projects.
Though senior management and stakeholders are not seen as a user target group, they can require and receive reports about project risks from project managers.
Project SPI consultants shall perform maintenance of the content: monitor discussions, new item, improvement proposals and case study reports, gather, pack and integrate new items into the Knowledge Base, and disseminate knowledge by different means.

3. Major functions

PraDis Knowledge Base provides the following functions:

- Description of global factors and threats;
- Experience generalization – practices to overcome global factors and threats;
- Threat outcome predictions with the help of so called Risk Barometer;
- Item categorization and search facilities;
- Quality document templates addressing global specifics;
- Case study overviews;
- Discussions;
- Personal folders and notifications;
- New knowledge and experience item proposal;
- Suggestions for Knowledge Base improvement.

This user guide first provides basic information about PraDis Knowledge Base principles of use; and then offers a detailed description of different use case scenarios.

4. Using PraDis Knowledge Base

When a user opens PraDis Knowledge Base, the following screen is provided (see Figure 2).
The PraDis main screen provides user the following opportunities:

- Switching between database views to select different content from the database (1);
- Opening documents displayed within database views by a double clicking it (2);
- Using different functions under action buttons available in the action bar (3).

PraDis functions based on these principles in detail are described in the following chapters.

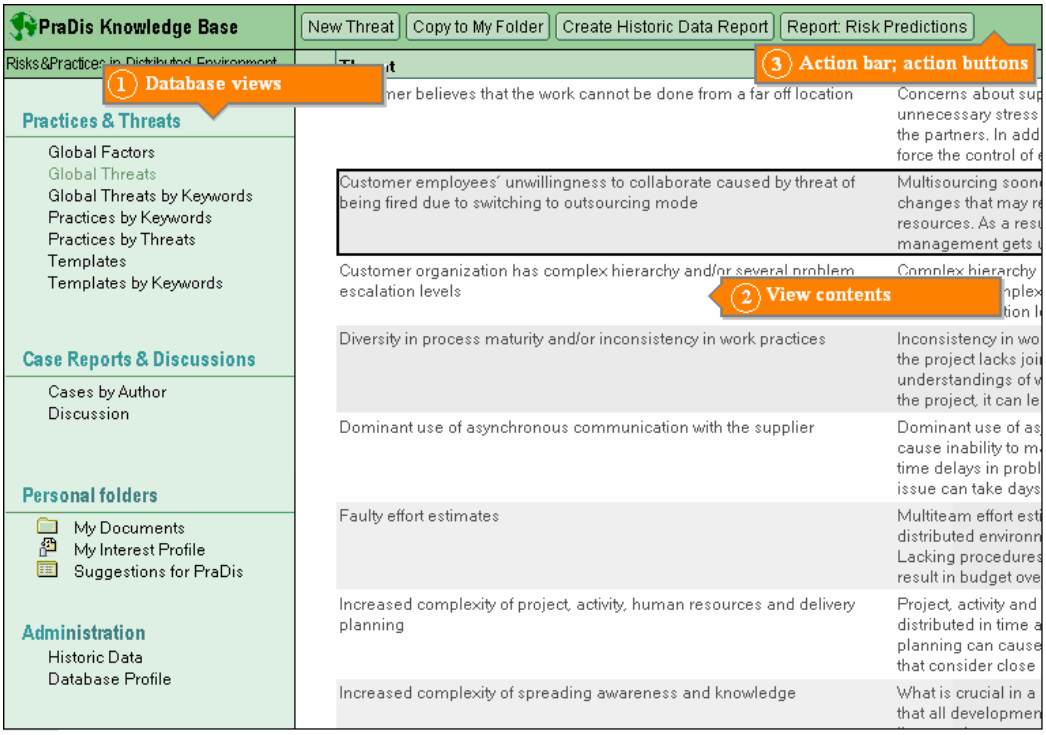


FIGURE 2. Opening the Knowledge Base

4.1. Global factors

PraDis contains the following information about global factors:

- Title – title of a global factor;
- Description – detailed description of a global factor including its effect on a project;
- Notes – may contain additional data, attachments or references to related documents;
- Document creation information - user and creation date.

An example of a global factor can be seen on the following figure.

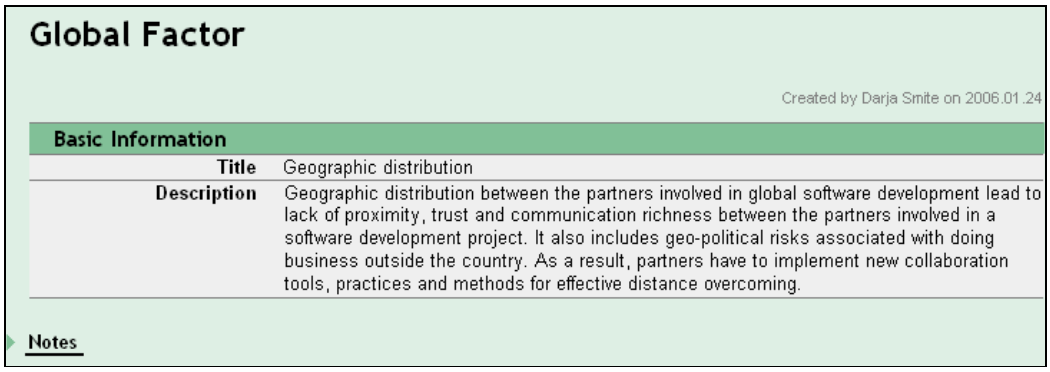


FIGURE 3. A Global Factor

Global factors can be accessed from the view *Global Factors*.

Action buttons available in this view:

New Factor - opens a new factor for creation;

Copy to My Folder - copies the selected document or documents in the user's favorites;

Create Historic Data Report - opens a new historic data report for creation;

Report: Risk Predictions - performs risk predictions (described in chapter 4.5).

4.2. Global threats

PraDis contains the following information about global threats:

- Threat Title – short title of a threat;
- Description – detailed description of a threat and its effect on a project;
- Keywords – keywords for threat categorization and further searching facility;
- Global Factors – roots of a threat;
- Notes – may contain additional data, attachments or references to related documents;
- Document creation information - user and creation date.

An example of a global threat can be seen on the following figure.

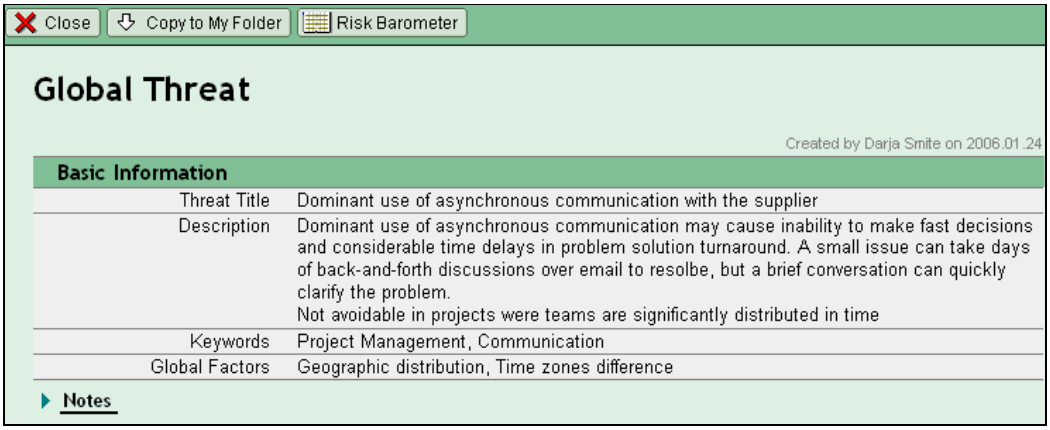


FIGURE 4. A Global Threat

Available action buttons:

Close - closes the document;

Copy to My Folder - copies this document to the user's favorites;

Risk Barometer - performs risk analysis (described in chapter 4.5).

Global threats can be accessed from the following views:

- *Global Threats* – sorted alphabetically;

- *Global Threats by Keywords* – categorized by different keywords;

In the categorized views, documents appear under different categories.

Action buttons available in these views:

New Threat - opens a new threat for creation;

Copy to My Folder - copies the selected document or documents in the user's favorites;

Create Historic Data Report - opens a new historic data report for creation;

Report: Risk Predictions - performs risk predictions (described in chapter 4.5).

4.3. Practices

PraDis contains the following information about global practices:

- Title – short title of a practice;
- Description – short description of a given recommendation;
- Keywords – keywords for practice categorization and further searching facility;
- Relations – contain factors and threats that can be eliminated by this practices;
- Mitigation strategies – description, whether this practice helps to avoid, mitigate or control related threats;
- Details – detailed description of a practice and its implementation;
- Notes – may contain additional data, attachments or references to related documents;
- Document creation information - user and creation date.

An example of a global threat can be seen on the following figure.

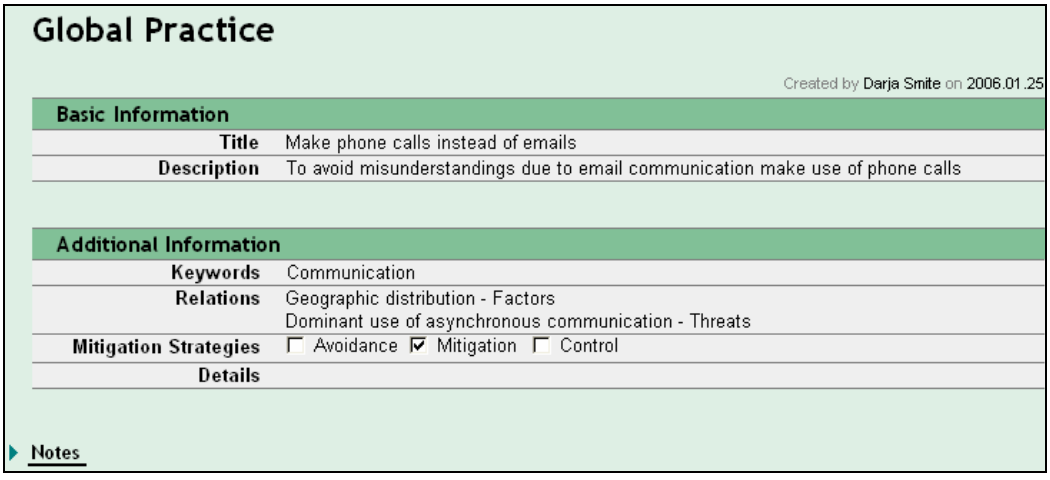


FIGURE 5. A Global Practice

Global practices can be accessed from the following views:

- *Global Threats by Keywords* – categorized by different keywords;
- *Global Practices by Threats* – categorized by related threats.

In the categorized views, documents appear under different categories.

Action buttons available in these views:

New Practice - opens a new practice for creation;

Copy to My Folder - copies the selected document or documents in the user's favorites;

4.4. Templates

Templates are different quality documents forms that can be used for project purposes.Templates contain the following information:

- Title – title for a template;
- Purpose – what shall a template be used for within a project;
- Keywords - keywords for template categorization and further searching facility;
- Notes – template attachments and additional instructions for template application;
- Document creation information - user and creation date.

An example of a template can be seen on the following figure.

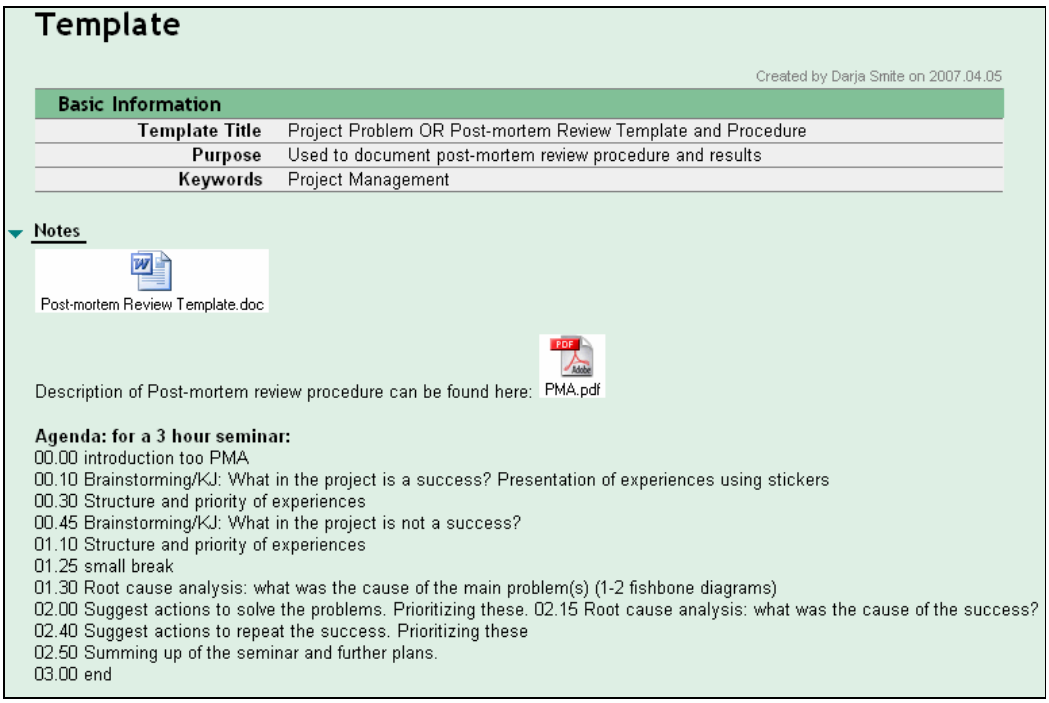


FIGURE 6. A Template

Templates can be accessed from the following views:

- *Templates* – sorted alphabetically;
- *Templates by Keywords* – categorized by keywords.

In the categorized views, documents appear under different categories.

Action buttons available in these views:

- opens a new template for creation;

- copies the selected document or documents in the user's favorites.

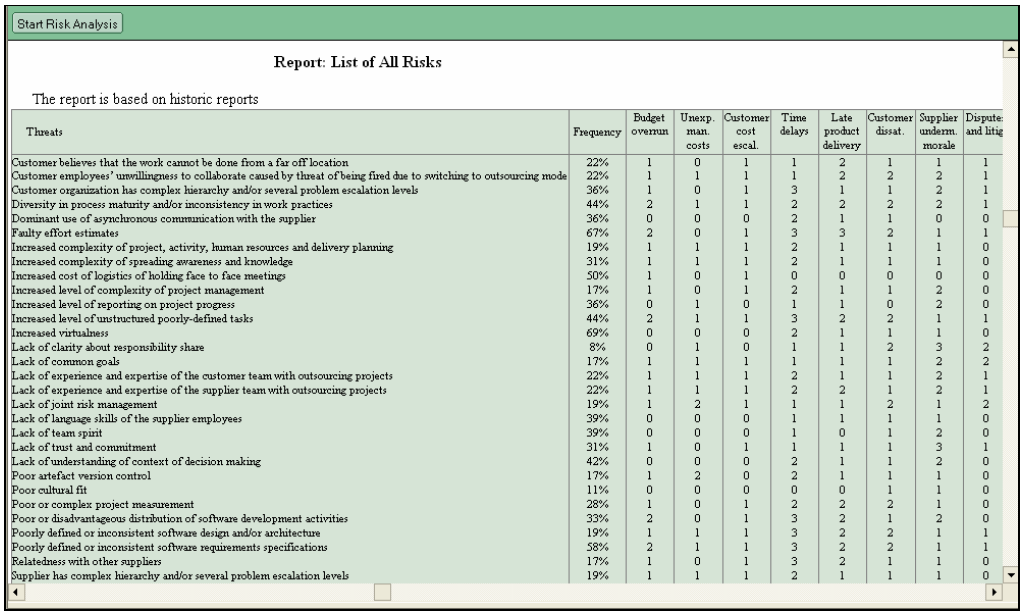
4.5. Risk Barometer

Risk Barometer provides a set of functions that serve users in the process of risk analysis. Risk Barometer contains two major outputs: Risk Prediction report on all threats and each threat Risk Barometer report.

Risk Prediction report can be accessed in all threat views using appropriate action button - 

that opens a new empty report form. To run a risk prediction report the user shall push the button 

that performs risk analysis using the historical reports from the database. When the process is finished user can see the following information: threats, frequency of occurrence of each threat, and average values for the main project success criteria such as budget overrun, unexpected management costs, customer cost escalation, time delays, late product delivery, customer dissatisfaction, supplier team's morale, disputes and litigations.


Start Risk Analysis

Report: List of All Risks

The report is based on historic reports

Threats	Frequency	Budget overrun	Unexp. man. costs	Customer cost escal.	Time delays	Late product delivery	Customer dissat.	Supplier underm. morale	Dispute and litig
Customer believes that the work cannot be done from a far off location	22%	1	0	1	1	2	1	1	1
Customer employees' unwillingness to collaborate caused by threat of being fired due to switching to outsourcing mode	22%	1	1	1	1	2	2	2	1
Customer organization has complex hierarchy and/or several problem escalation levels	36%	1	0	1	3	1	1	2	1
Diversity in process maturity and/or inconsistency in work practices	44%	2	1	1	2	2	2	2	1
Dominant use of asynchronous communication with the supplier	36%	0	0	0	2	1	1	0	0
Faulty effort estimates	67%	2	0	1	3	3	2	1	1
Increased complexity of project, activity, human resources and delivery planning	19%	1	1	1	2	1	1	1	0
Increased complexity of spreading awareness and knowledge	31%	1	1	1	2	1	1	1	0
Increased cost of logistics of holding face to face meetings	50%	1	0	1	0	0	0	0	0
Increased level of complexity of project management	17%	1	0	1	2	1	1	2	0
Increased level of reporting on project progress	36%	0	1	0	1	1	0	2	0
Increased level of unstructured poorly-defined tasks	44%	2	1	1	3	2	2	1	1
Increased virtualness	69%	0	0	0	2	1	1	1	0
Lack of clarity about responsibility share	8%	0	1	0	1	1	2	3	2
Lack of common goals	17%	1	1	1	1	1	1	2	2
Lack of experience and expertise of the customer team with outsourcing projects	22%	1	1	1	2	1	1	2	1
Lack of experience and expertise of the supplier team with outsourcing projects	22%	1	1	1	2	2	1	2	1
Lack of joint risk management	19%	1	2	1	1	1	2	1	2
Lack of language skills of the supplier employees	39%	0	0	0	1	1	1	1	0
Lack of team spirit	39%	0	0	0	1	0	1	2	0
Lack of trust and commitment	31%	1	0	1	1	1	1	3	1
Lack of understanding of context of decision making	42%	0	0	0	2	1	1	2	0
Poor artefact version control	17%	1	2	0	2	1	1	1	0
Poor cultural fit	11%	0	0	0	0	0	1	1	0
Poor or complex project measurement	28%	1	0	1	2	2	2	1	0
Poor or disadvantageous distribution of software development activities	33%	2	0	1	3	2	1	2	0
Poorly defined or inconsistent software design and/or architecture	19%	1	1	1	3	2	2	1	1
Poorly defined or inconsistent software requirements specifications	58%	2	1	1	3	2	2	1	1
Relatedness with other suppliers	17%	1	0	1	3	2	1	1	0
Supplier has complex hierarchy and/or several problem escalation levels	19%	1	1	1	2	1	1	1	0

FIGURE 7. A Risk Prediction report

If the user requires investigating a certain threat, use Risk Barometer function from the threat document by pushing the action button- 

. The results of risk analysis contain the following information (see also Fig. 7):

- Threat;
- Frequency of occurrence;
- Report: Historical Data for Risk Analysis – statistical information based on the historic data from the database with the labelled average values;
- Report: Probability of the Negative Outcome – statistical information based on the historic data from the database;
- Report: Risk Exposure Level – statistical information based on the historic data from the database.

Threat:	Lack of experience and expertise of the supplier team with outsourcing projects
Frequency of occurrence:	22 % or 8 of 36

Report: Historical Data for Risk Analysis

	Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
5 – Disastrous				12 %	12 %	12 %	12 %	12 %
4 – Significant	12 %		12 %			12 %	12 %	
3 – Moderate				38 %	12 %		12 %	
2 – Minor	12 %	25 %	12 %	-> 12 %	-> 25 %	12 %	->	12 %
1 – Negligible	->	-> 25 %	->	38 %		->	12 %	->
0 – None	75 %	50 %	75 %		50 %	62 %	50 %	75 %

The sign -> points the average outcome of the threat on a certain project success criteria

Report: Probability of the Negative Outcome (if the threat has occured)

	Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
5 – Disastrous	0 %	0 %	0 %	12 %	12 %	12 %	12 %	12 %
4 – Significant	12 %	0 %	12 %	12 %	12 %	24 %	24 %	12 %
3 – Moderate	12 %	0 %	12 %	50 %	24 %	24 %	36 %	12 %
2 – Minor	24 %	25 %	24 %	62 %	49 %	36 %	36 %	24 %
1 – Negligible	24 %	50 %	24 %	100 %	49 %	36 %	48 %	24 %
0 – None	75 %	50 %	75 %	0 %	50 %	62 %	50 %	75 %

Report: Risk Exposure Level

Budget overrun	Unexpected management costs	Customer cost escalation	Time delays	Late product delivery	Customer dissatisfaction	Supplier team's undermined morale	Disputes and litigations
72	75	72	236	146	132	156	84

Risk Exposure evaluation [0; 1500].

FIGURE 8. Risk Barometer output

4.6. Case reports

Case reports aim to accumulate specific experiences from different project managers. Case reports are accessible from the view Cases by Authors and contain the following information:

- Case field information – project that this case reports about:
  - Project Name;
  - Project Manager;
  - Project Customer;
  - Project Description;
- Problem Area – description of the problem and its solution:
  - Problem History;
  - Solution;
  - Practices Used (if any);
- Notes – template attachments and additional instructions for template application;

- Document creation information - user and creation date.

An example of a case report can be seen on the following figure.

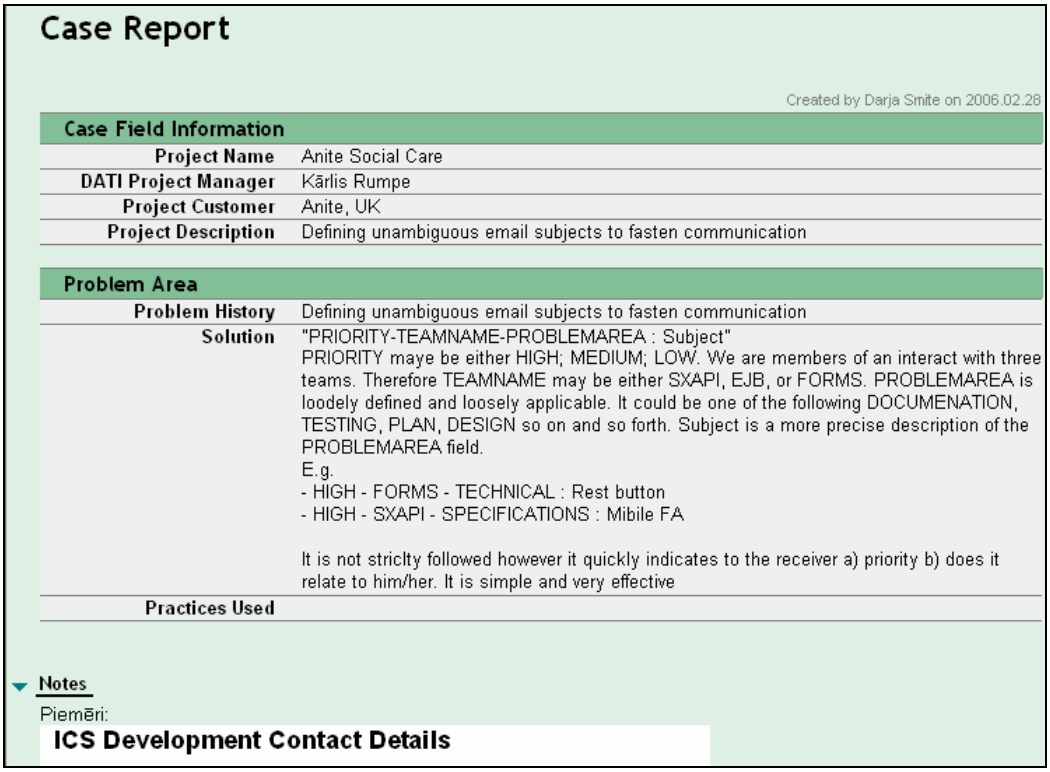


FIGURE 9. Case Report example

Action buttons available in the Cases by Author view:

New Case Report - opens a new case report for creation;

Copy to My Folder - copies the selected document or documents in the user's favorites.

4.7. Discussions

Users can participate in different discussions within the PraDis Knowledge Base. Discussions are accessible in the view Discussion under "Case Reports&Discussions" on the database view outline.

Using action button New Discussion Topic users can propose a new topic for discussion by filling the following form (see Figure 10).

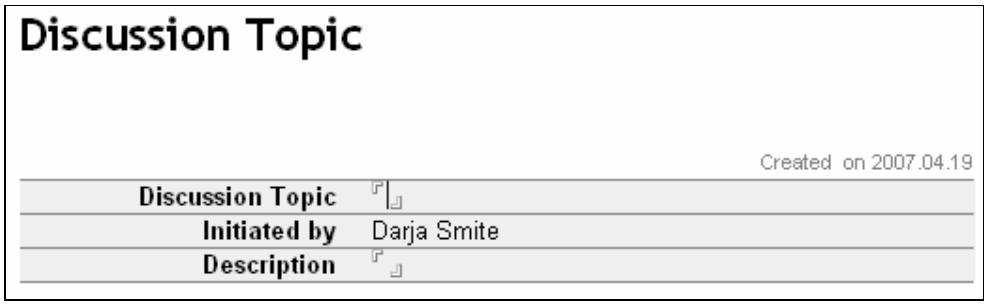


FIGURE 10. Discussion topic form

For replying an existing discussion topic use action button New Comment. Note that discussion comments can be created for main topics and for other comments as well. An example of a discussion comment form can be seen on figure 11.

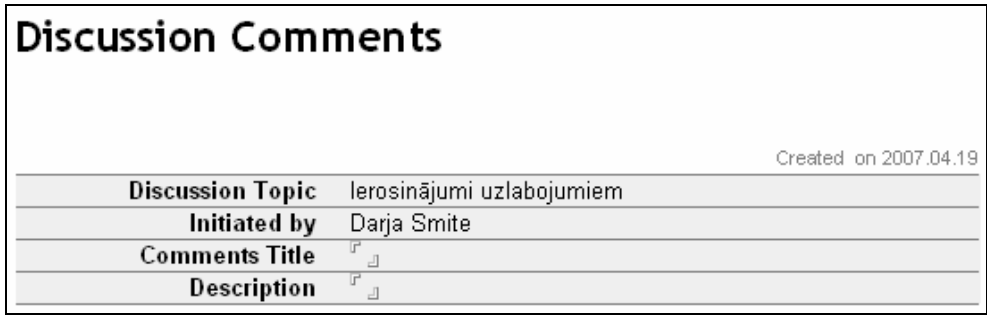


FIGURE 11. Discussion comments form

4.8. Personal folders and notifications

Users can select documents of different types as favorite and collect them in a personal folder "My documents" using the action button - Copy to My Folder.

If a user would like to be notified about new documents created by certain authors or containing certain keywords, an Interest Profile shall be created using reference - My Interest Profile from the database views outline (see the figure below).

FIGURE 12. Interest Profile

Suggestions for PraDis Knowledge Base can be filled by all users from the view – *Suggestions for PraDis*, using the action button - Suggestion that creates a new form (see the form in the figure below).

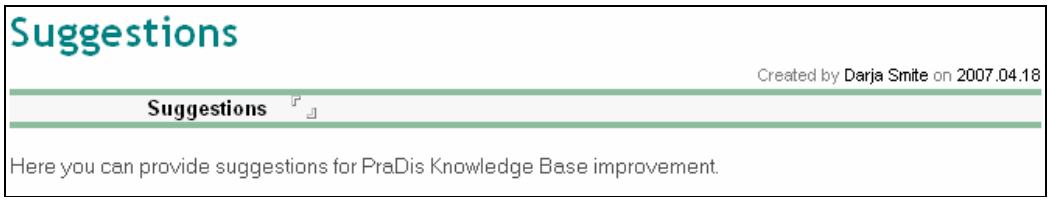


FIGURE 13. Suggestions form

5. Use Case Scenarios

Major function provided by the PraDis Knowledge Base are described in more detail within the following use case scenarios:

5.1. Organizational knowledge management

5.2. Global project risk management

5.1. Organizational knowledge management

PraDis Knowledge Base provides a basis for maintaining the implicit organizational knowledge that is essential to prevent knowledge assets from loss on a long term. Project managers shall use the Knowledge Base for the following purposes:

- Select problem solutions from experiences recommended by the Knowledge Base;
- Provide recommendations for others by storing best practices into the Knowledge Base;
- Ground project risk evaluation on historical data to prevent global threat underestimation;
- Use the Knowledge Base as a central point for describing how global projects are run;

- Introduce new employees with organizational philosophy regarding global projects accumulated in the Knowledge Base.

5.2. Global project risk management

It is recommended to integrate the usage of PraDis Knowledge Base into the process of project risk management as described in the following figure.

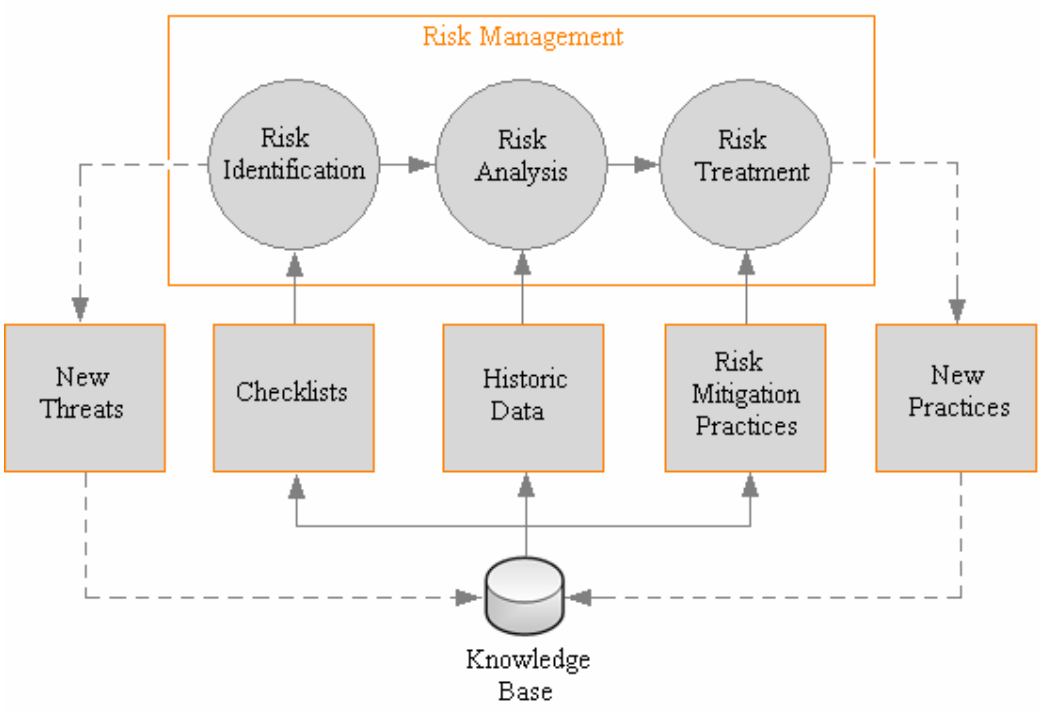


FIGURE 14. Risk management scheme

Risk management activities car be described as follows:

- Risk identification: Project team members can use global factors and threats from the Knowledge Base as support in risk identification.
- Risk analysis. Risk outcome predictions performed by Risk Barometer can provide historic information as a support in analyzing risks – impact and probability of occurrence.
- Risk mitigation. Corresponding mitigation practices can be selected from the Knowledge Base by threats. Consider using previous experiences for better risk mitigation.

In order to provide sustainable Knowledge Base content improvement and actualization, it is recommended to invest some time for the following activities:

- In case of new threat identification – store threat description into the database;
- In case of new threat mitigation practice experience – store practices or case study reports into the database;
- In case of post mortem analysis – consider filling a historical data report to improve Risk Barometer predictions.