Reporting Empirical Research in Global Software Engineering: a Classification Scheme

Darja Šmite[1], Claes Wohlin[2], Robert Feldt[2], Tony Gorschek[2]
1: University of Latvia; 2: Blekinge Institute of Technology
Darja.Smite@lu.lv, Claes.Wohlin@bth.se, Robert.Feldt@bth.se, Tony.Gorschek@bth.se

Abstract

Increased popularity of global software engineering (GSE) has resulted in quite a number of research and industrial studies. As the area matures, an increased focus on empirically supported results leads to a greater potential impact on future research and industrial practice. However, since GSE scenarios are diverse, what works in one context might not directly apply in another. Thus it is necessary to understand, how GSE-related empirical findings should be reported to be useful for practitioners and researchers. Furthermore, it's important to summarize progress and get the big picture of published research to identify gaps and commonalities. In this paper we analyze differentiating factors of GSE scenarios and offer a classification scheme for describing the context of a GSE study. In addition, we report initial results of a systematic review on GSE-related empirical literature using papers from ICGSE 2006 and 2007, at the same time illustrating and evaluating the proposed scheme.

1. Introduction

Many researchers and practitioners have recently turned to exploring the emerging field of Global Software Engineering (GSE). Resulting from globalization, recognized as the phenomenon of the 21st century [1], GSE underlines a transition from a common way of producing software to life cycle activities distributed among teams separated by various boundaries. Because of new factors such as geographic, temporal, and socio-cultural distance, virtual product development is recognized as considerably more complex, than even the most complex project managed entirely in-house [2].

The demand for research results in GSE is growing with an increasing number of internationally distributed software organizations. Carmel and Agarwal [3] report that practitioners are experimenting and quickly adjusting their tactical approaches for leveraging global software development risks, because methods used in collocated projects have limited applicability in the GSE environment. However, the concept of the GSE environment is still unclear. Global software work is enabled through various collaborative forms such as inter-organizational outsourcing, intra-organizational offshoring, intra-national nearshoring, to name only a few. Thus it is reasonable to argue that what works in one context might not directly apply in another. Although some theories and practices have been researched and developed, the art and science of global software development is still evolving [4]. As the area matures, there is an increased focus on empirically supported results, which also leads to a greater potential impact on future research and industrial practice. Hence we aim to answer the following research questions:

- How should GSE-related empirical research results be reported so that
  a) Practitioners can understand the context and thus applicability of the reported results?
  b) Researchers can understand the relation between different research results and the level of their generalization?
- What is known about GSE practices and what are the gaps and comonalities in existing empirical results?

Answering these research questions, we developed a classification scheme for GSE-related empirical studies that can be used to understand the context of existing studies and guide the process of reporting and structuring future studies. We have used the scheme in an ongoing systematic review of empirical studies in the area of global software engineering and present initial results revealing the trends and implications of existing state of the practice.

The rest of the paper is organized as follows. In Section 2 we describe the proposed scheme for classifying empirical research in global software engineering and motivate it with the demand from

related studies. Section 3 provides an overview of the research methodology and outline of the systematic review. We then introduce initial results of the systematic review in Section 4 followed by a discussion and conclusions.

2. Classification of Empirical Research in Global Software Engineering

In this chapter we present a set of factors that are necessary to evaluate applicability of and ability to generalize GSE-related empirical findings. Together these factors constitute a classification scheme. We provide a motivation for selecting each factor based on related research literature. These factors are grouped into three main categories:

- GSE background – factors that differentiate GSE scenarios and thus classify the origin of reported findings;
- Empirical background – factors that refer to the origin and perspective of empirical data used to derive reported findings;
- Study background – characteristics and aspects of software engineering work used in investigated studies.

2.1. GSE Background

There are many ways to organize and manage global development [5]. International business environments and organizational forms are being significanly reshaped as part of a new global scenario [6]. GSE can be conducted in different scenarios implemented in different organizational forms. Based on an empirical investigation, Poikolainen and Paananen [7] argue that, it is fair to assume that the findings and theories made for intra-organizational GSD projects may not directly apply in global inter-organizational software development projects [7]. Therefore, we emphasize the necessity to describe **collaboration mode** for reported studies (inter-organizational; intra-organizational).

The complexity of communication, coordination and control in GSE projects is dependent on the **number of distributed sites**. Increasing the number of collaboration partners increases the number of sources of threat and also complexity of trust achievement [8].

Geographic, temporal and socio-cultural factors are recognized as the major barriers for successful GSE team performance [9]. Distributed software development ranges from team members being distributed over adjacent buildings to being distributed over different continents [10]. Work that spans sites takes longer than work that does not cross sites [11]. Temporal distance is found to be affecting both accuracy of work and team productivity [12]. Socio-cultural distance is a complex dimension involving organizational culture, national culture and language, politics, and individual motivations and work ethics and inherents the challenge of creating mutual understanding between people with different backgrounds [9]. Scaling these distances from nearshore to farshore makes a difference on how the teams interact. Therefore, in order to understand the applicability of reported findings, **geographic locations of the sites involved in the study** shall be specified.

For a long time the vast majority of research in the GSE area was conducted mainly from the customer's perspective because the objective of outsourcing was to selfmaximize the internal resources without taking into account the service provider's situation [13]. For the past few years studies from supplier [14, 15] and global software team [10, 16] perspective start to emerge. However, because of different goals, practices and recommendations on how to survive global projects among these studies may differ. It is therefore important to differentiate the **perspective** of reported empirical findings that may vary between suppliers, originators or collaboration in general. The latter classification is used when there is no particular perspective used in the study, i.e. it concerns both the suppliers and originators or neither.

Studies show that there is a dependency between reasons for implementing GSE scenarios and actual success of collaboration, e.g. experienced managers warn that blind cost reduction strategies tend to fail [17]. Therefore, it is important to report the **reasons for distributed software development** (costs, resources, speed, new markets, etc.).

2.2. Empirical Background

It is important for both practitioners and researchers to be able to evaluate the sigificance of empirical results of a reported study. Therefore, we emphasize the necessity of describing the **methods** used to gather and analyze empirical data (e.g. survey, case study, interviews, controlled experiment or other). We describe the methods used on two levels, one is the **main method** used, and **sub-methods** are any methods that are used to support the main method. An example would be a case study (main) where interviews and questionnaires are both used as sub-methods.

The issue of differences between students and professionals used in experimental studies has been discussed in the literature raising the question of the representativeness of experimental results from academia for an industrial setting [18]. We therefore

aim to compare studies with different **backgrounds** of laboratory or real-world software development projects capturing the record of students vs. practitioners used as **subjects of investigation**. This enables an analysis of proportion of industrial versus academic studies and thus evaluation of experimental realism. Comparison of these studies can deepen our understanding of whether student experiments provide different results from industrial studies and provide conclusions on how to do student studies in a more effective way instead of disregarding them.

Empirical studies may have different objectives. For example, a study may aim at exploring a situation or explaining something. As an example, in an exploratory case study, data is collected prior to defining research questions and hypotheses. This means that the study is run to generate an increased understanding. The intention is to explore, which means that in global software engineering this type of study is often used to create an understanding of challanges and other issues in relation to distribution. This type of paper base its results and conclusions on empirical data. Thus, we refer to studies of exploratory nature as being **empirically-based**. In an explanatory study, the aim is to explain something and in many cases to try to identify a causal relationship. Case studies can also be exploratory, where the objective is to explain or evaluate something. For example, the objective may be to explore the influence of the introduction of a new tool or method in global software engineering. Thus, the objective is to evaluate the new way of working in relation to the situation before we introduce the new tool or method. We refer to studies of explanatory nature as being **empirically-evaluated**. More information about for example different types of case studies can be found in the book by Yin [19].

2.3. Study Background

In order to classify existing studies in the GSE area and identify the gaps and comonalities, we also focus on the background of different studies. In particular, we aim to understand what we know and what we do not know about global software engineering. Thus we record the **focus of studies**, whether it is about collaboration in general, a particular development phase or a single practice, e.g. requirements engineering, coordination, or other.

It is also important to record different **development methodologies** applied in the GSE projects since there is no standardized model for global development and the methodology used can greatly affect how the distribution of work and interactions happen.

An overview of GSE project case studies shows that the observed results can vary from tremendous successes to total failures. Thus it is necessary to classify existing practices in terms of clear success stories, success of the practices described, clear failure stories, failure of the practices described, or discussions about the consequences working in a globally distributed environment, in other words, an evidence of GSE-related problems. In order to further accommodate practitioners in their understanding of the applicability of the GSE-related studies, we propose to record the **application domain**, in which the studies were carried out (e.g. telecom, automotive,

Finally we record the major **claims** derived from these studies and supported by empirical evidence.

Capturing information about the background of GSE empirical studies we further aim to understand, whether lessons learned in different GSE scenarios can really apply in different contexts or are unique for the original background.

2.4. Application of the Classification Scheme

The developed classification scheme facilitates an analysis of existing empirical literature in the area of global software engineering. It can be used by readers who try understand the applicability of reported findings and compare these from different studies, or by authors as a guidance in describing the context of GSE-related studies.

We have used our classification scheme in an ongoing systematic review of GSE-related empirical studies. A special review form was developed to facilitate documentation of the review results (see Table 1). To keep track of unclarities and evaluate the accurateness of the context of reported studies we added „Unclear“ as a possible value for most of the scheme's factors. Factors that were irrelevant due to specific nature of the studies, were marked as „Irrelevant“.

We have also developed a list of definitions for each factor used in the scheme to ensure a mutual agreement between the reviewers. An overview of the process and initial results of the systematic review are described in Section 3 and 4.

3.Research Methodology

The findings reported in this paper correspond to the initial step of an ongoing systematic review of GSE-related empirical studies. In particular, in this paper we focus on the research questions related to developing and evaluating the classification scheme for GSE-related empirical studies. Validation and

Table 1: Systematic Review Form (core questions)

Empirical background	
Main Method	☐ Survey ☐ Case Study ☐ Interviews ☐ Controlled experiment ☐ Other: …
Sub-Methods	☐ Survey ☐ Case Study ☐ Interviews ☐ Controlled experiment ☐ Other: …
Background	☐ Laboratory ☐ Industry/Real world
Subjects of investigation	☐ Students ☐ Industry/Real world
Empirical focus	☐ Empirically based ☐ Empirically evaluated
GSE background	
Collaboration mode	☐ Inter-organizational ☐ Intra-organizational ☐ Unclear ☐ Irrelevant
Number of sites	… ❍ Unclear ❍ Irrelevant
Location of the originator	… ❍ Unclear ❍ Irrelevant
Location of the suppliers	… ❍ Unclear ❍ Irrelevant
Perspective	❍ Supplier ❍ Originator ❍ Collaboration in general ❍ Unclear ❍ Irrelevant
Reason for outsourcing	☐ Costs ☐ Extra knowledge ☐ Extra people ☐ Unclear ☐ Irrelevant ☐ Other …
Study	
Development methodology	… ❍ Unclear
Focus of the study	❍ Collaboration in general:… ❍ Single practice(s):… ❍ Development phase(s):… ❍ Other:…
Success or failure?	☐ Evidence of the GSE-related problems ☐ Unclear ☐ Other: … ☐ Clear success story ☐ Success of the practices described ☐ Clear failure story ☐ Failure of the practices described
Application domain	☐ Telecom ☐ Automotive ☐ Web ☐ Finance ☐ Automation ☐ Other: … ☐ Unclear
Claims	

improvement of the scheme was organized as an initial review, for which we have chosen to use studies reported at the International Conference on Global Software Engineering (ICGSE) in 2006 and 2007. Since ICGSE is the first specific conference on global software engineering, we consider articles reported at ICGSE to be representative in terms of state of the art in this research area.

The process of developing the classification scheme involved four researchers and started by brainstorming ideas of differentiating factors of GSE scenarios (the final version is reported as GSE Background in Section 2). In order to validate the applicability of the scheme it was further evaluated and improved in several review stages:

1. Stage 0: selection of relevant ICGSE articles, using a search strategy, which is further described below.
2. Stage 1: joint review session of 3 papers, read by all 4 reviewers), followed by improvement suggestions and scheme refinement. The scheme was restructured and supplemented by questions regarding Empirical Background and Study background (described in Section 2).
3. Stage 2: separate review of 5 papers, each paper read by 2 reviewers, followed by improvement suggestions.
4. Stage 3: using the refined scheme in reviewing the rest of the relevant ICGSE articles. Each paper was read by 2 reviewers. Previous reviews are refined and classified in accordance with the latest version of the scheme.

The final version of the core questions is presented in Table 1. Additional questions motivated by our research questions (e.g. do the authors mention GSE-related term definitions) and necessity to evaluate the quality of the reviewed studies (the strength of software engineering, GSE and empirical focus in the study) were added to the systematic review template.

Aiming to answer the research questions, we have been searching for research studies that have an empirical background from studies with either professionals or students with the main focus on global software engineering. Our search strategy defined the scope of the investigated studies and the search string consisted of the following terms:

((global software development) <OR> (global software engineering) <OR> (distributed software development) <OR> (distributed software engineering)) <AND> ((empirical) <OR> (industrial) <OR> (experiment) <OR> (case study))

Although the kewords selected for indentifying the empirical component of the investigated papers do not cover all possible research methods, we believe that these are sufficient to cover in-depth empirical studies. Because we were looking for papers with the type of depth that can be expected from a case study or an

experiment, general survey papers were out of our target orientation.

The search was performed using the IEEE Xplore database and applied to the full text. In result we found 24 articles – 11 articles from ICGSE'06 and 13 from ICGSE'07.

A quality evaluation was performed in two steps according to the systematic review procedure. During the first step the initial screening was performed through reading the titles and abstracts and evaluating the relevance of the articles in terms of empirical evidence and GSE focus. At this point, 3 articles were excluded as irrelevant. The second evaluation was based on the full text of the articles. In result, 9 articles of the remaining 21 articles were excluded from the analysis as weak in either empirical evidence or GSE focus.

In the next section we describe the initial results of our systematic review focusing on the evidence gathered from the studies presented in earlier ICGSE conferences and discuss the importance of reporting GSE-related studies according to the proposed classification scheme.

4. Systematic Review: Initial Results

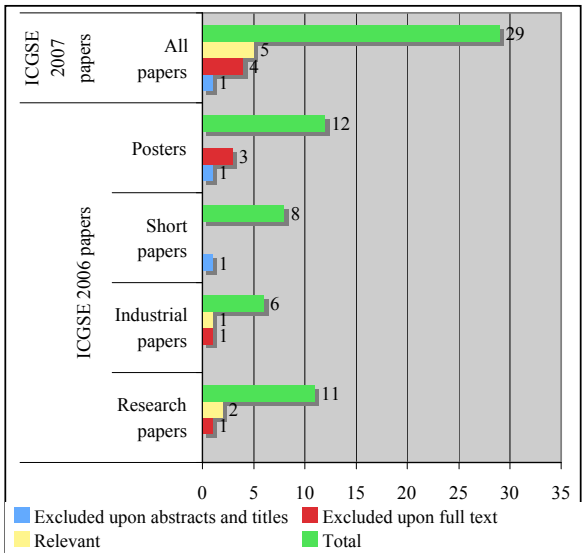

Fig.1. Statistics of the found studies

To our surprise out of 66 papers and posters in total presented in the ICGSE 2006 and 2007 conferences our search found only 24 papers using the search strings. Three papers were then excluded upon titles and abstracts and nine papers were excluded upon full text review from the further analysis mainly because of insufficient empirical evidence. These papers mostly contain claims or lessons learned without presenting empirical evidence supporting these claims and thus were recognized to be weak in terms of empirical background. Two of the nine excluded papers were recognized as weak in terms of GSE focus. These papers provide observations of non-software engineering processes with evidence from non-software engineering studies and explore collaborative work in general. In result, only a half of the initially found papers (namely 12) were relevant for the further analysis. Statistics of the searched and reviewed papers is presented in Fig.1.

The remaining twelve articles are unfortunately insufficient to provide a representative overview of what we know and what we do not know about GSE according to one of our research questions. However, we use the existing data to illustrate diversity of the reported studies and GSE scenarios and validate the scheme proposed for evaluating empirical study conducted in the GSE context. Existing data also helps us to identify several trends that we discuss later in this paper.

In Table 2, 3 and 4 we present our findings according to the classification scheme and its categorization.

Table 2. Review Findings: Empirical Background

Methods used	Survey	1
	Case Study	8
	Interviews	9
	Controlled experiment	1
	Grounded Theory	4
	Archives	3
	Observations	1
Background	Laboratory	2
	Industry	10
Subjects of investigation	Students	2
	Practitioners	10
Empirical focus	Empirically validated	2
	Empirically based	10

According to our results, the majority of the studies are exploratory in nature. Empirical evidence is mainly gathered through case studies using interviews, grounded theory and archive documentation as sub methods. In contradiction to our expectations, only few student studies were found among the relevant articles. Hence, the majority of the investigated studies came from the industry.

The proportion between empirically validated and empirically based studies is surprising. There are ten empirically based studies that focus on mainly capturing evidence from the industry and further derive the lessons learned. This shows that we are still

learning about the challenges of globally distributed environment. Few papers have actually evaluated proposed improvements, methods or practices. This proves that the research in this area is still maturing.

Table 3. Review Findings: GSE background

Collaboration mode	Intra-organizational	8
	Inter-organizational	4
	Unclear	1
Geography of the studied locations	US	6
	India	5
	Brazil	3
	Canada	3
	Germany	3
	Ireland	2
	Malaysia	2
	Poland	2
	China	1
	England	1
	France	1
	Latvia	1
	Portugal	1
	Russia	1
	Unclear	2
Perspective	Supplier	1
	Originator	5
	Collaboration in general	6
Reason for outsourcing	Cost reduction	2
	Competitiveness	1
	Irrelevant	2
	Unclear	8

There are more intra-organizational GSE studies from multi-national companies. Some of the studies report findings from investigating several intra-organizational and inter-organizational cases.

Investigated studies are widely spread across different continents. However, it is worth to mention that it was often difficult to deduct what role each site has. Several articles provided a wider area, e.g. Europe, in order to make investigated multi-national companies more anonymous.

Most of the studies are conducted either from the global team or originator perspective. Only one study is conducted from GSE supplier perspective, which clearly indicates a gap in knowledge.

Reasons for starting global collaboration are very rarely mentioned, however, cost reduction is the most frequently met among the investigated studies. Our observations show that reasons for outsourcing are widely mentioned in the introduction to almost every article, discussing the benefits of global software engineering, however, these have no empirical support.

Table 4. Review Findings: Study background

Development methodology	Microsoft Solution Framework with RUP	1
	Model-based testing with UML	1
	Unclear	9
Focus of the study	Collaboration in general	5
	Coordination	2
	Communication	1
	Risk management	1
	Requirements engineering	1
	Testing	1
	Awareness	1
	Reference model	1
Success or failure	Clear success story	0
	Success of the practices described	2
	Clear failure story	0
	Failure of the practices described	1
	Evidence of GSE-related problems	8
	Unclear	3
Application domains	Hardware	4
	Finance	3
	Telecom	3
	Automotive	3
	Airline	1
	Healthcare	1
	Research	1
	Logistics	1
	Tailored DB systems	1
	Process management	1
	Biomed	1
	Unclear	2

We have observed that even the most accurately described cases often disregard reporting about development methodologies followed in the investigated studies. On the other hand, there is a dominance of studies focussing on interviews with experienced project managers as the prime and only source of empirical evidence. Thus we identify a gap in reporting project that would share software engineering experiences and different development method application.

Classification of existing empirical evidence provided a diverse thematical distribution of the investigated studies. These studies come from different application domains. Unfortunately, due to statistical insignificance, we cannot build any generalizable conclusion using these findings.

It is interesting, that eight out of twelve articles address GSE-related problems, two - success of the practices described and four studies could not be classified in terms of success or failure. None of the studies provided clear success or failure studies. Because of the lack of clear success stories of the reported projects and the dominance of describing GSE-related problems and success of a few separate practices, we conclude that GSE is a problematic environment and promised benefits of global collaboration are still not clear. This is in line with related research (Moe and Smite 2007), (ConchúCir et al. 2006), (Karolak 1998).

5. Discussion

5.1. Necessity for Clarity of Reported Findings

The conflict between the widely promoted benefits of global collaboration in contrast to the reviewed articles that discuss GSE-related challenges led us to the question of how practitioners can learn from the existing literature. And thus, whether we should care about the clarity and sufficiency of empirical study descriptions.

While organizations tend to think that outsourced software work can be built cheaply and quickly, recent empirical studies show that blind cost reduction deals are doomed to fail. Upon the results of 232 interviews between 2004 and 2007, Lacity and Rottman [20] identified four major benefits of offshore outsourcing, where cost reduction is not mentioned at all. In fact, they refer that project managers experience significant hidden transaction costs and many other „problems“, „headaches“ and „crises“ from offsore outsourcing. Why shall we then outsource? What shall we do to avoid similar problems?

We argue that studing the existing empirical research, practitioners come across comparison of the results from significantly different contexts and backgrounds. However the average recipe or one-fits-all solution is barely applicable and therefore causes confusion for the reader. Moreover, new studies argue that what suites one scenario may not directly apply in another [7]. Accordingly, unclearities inherited in the reported studies, may not only burden the reader's understanding, but also drive to „lessons learned“ applied in a wrong way.

We therefore emphasize that building the body of knowledge on how to manage GSE projects we need to classify experiences and practices that help to achieve positive results, in order to understand in what circumstances they can be applied.

5.2. Classification of GSE Experience

To enable industry and researchers to understand the actual state of the art in GSE, we must start reporting studies more consistently. This is a prerequisite to actually know what we know and what we do not know about global software engineering. A systematic review relies on being able to deduce data from a large number of articles.

The validation of the scheme proposed for evaluation of comparability and applicability of the GSE-related empirical studies has shown that only few studies contained well-structured background descriptions. The reviewers often had to deduce the implicit information related to the origin of the reported findings.

The observations also mark out the necessity of more focussed and clear studies. Many studies are conducted from an organizational perspective. The authors of this article have repeatedly discussed and argued about applicability of the findings and best practices derived from interviewing several project managers from a multi-national organization with unclear, hidden or diverse project portfolios. Evaluation of the background of these studies according to the proposed scheme was thus problematic.

The numerous discussions about interpretations of different papers shows both the difficulty in reporting as such, but it also shows the obstacles for true progress in the field of global software engineering.

5.3. Recommendatios for the Authors

To motivate the clarity of further studies and thus enable evaluation of the applicability of reported findings, we offer the following recommendations.

Authors shall in detail specify the background of conducted studies. Descriptions shall contain characteristics of specific GSE scenarios in terms of organizational forms, work being distributed, as well as geography of distribution and complexity of delivery chains. General company profiles are not sufficient to judge about the origin of recommendation proposed by the authors. Descriptions shall be explicitly stated in e.g. a background or case overview sections to ease the understanding for the reader. The developed classification scheme can be used as a guide for structuring these descriptions.

Authors shall be careful in reporting lessons learned from a set of diverse studies. We recommend to clearly state the background and origin of each case study and experiment used in the synthesis of the reported findings.

We encourage practitioners to address the gaps identified by this research, such as:

- Studies focusing on evaluation of methods, practices and solutions for GSE;
- Studies conducted from supplier perspective;
- Studies focusing on different development methodologies;
- Studies of inter-organizational collaboration;
- Clear success and clear failure stories.

In conclusion we agree with observations derived from other systematic reviews of empirical literature that emphasize the necessity to improve the way how we write our papers, adopting a consistent form, structured abstracts, meaningful titles and keyword schemes [21].

6. Conclusions and Future Work

Because of the diverse scenarios enabled by different forms of global collaborative work, lessons learned in one context may not directly apply in another. Practitioners aiming to learn from existing literature may thus misapply recommendations. This motivated us to develop a classification scheme for GSE-related empirical research analysis. These scheme can be further used to understand the context of existing studies and guide the process of reporting and structuring future studies.

In this paper we also offer initial findings from a systematic review of ICGSE 2006 and 2007 articles that we consider to be representative. To our surprise, out of 66 papers and posters, we have found only a few empirical studies relevant for our review. According to our results, the majority of the investigated studies are exploratory in nature. This includes in particular case studies conducted mainly in an intra-organization context from the customer‘s or team’s perspective and report implications inherited in the nature of globally distributed environment. Thus we conclude that the state of the practice in the GSE area is still evolving.

The numerous discussions about interpretations of the investigated papers shows the level of unclearity of the reported studies. A reader still needs to put quite a large effort into understanding the applicability of the offered findings. To facilitate the readers and enable future progress in the area of GSE environment we provide a set of factors to be describe and thus emphasize the necessity of consistent reports and importance of our scheme for GSE empirical research.

Our future research will focus on enlarging the systematic review of GSE-related empirical studies published between 2000 and 2007 in order to find out what we know and what we do not know about practices and methods applied in global software engineering.

7. Acknowledgement

This research is supported by the European Social Fund under grant “Doctoral student research and post doctoral research support for university of Latvia”, the Latvian Council of Science within project Nr. 02.2002 “Latvian Informatics Production Unit Support Program in the Area of Engineering, Computer Networks and Signal Processing” and the Knowledge Foundation in Sweden (under a research grant for the BESQ project).

8. References

[1] Friedman, T.L, “The World is Flat: Brief History of the 21st Century”, Farrar, Straus and Giroux, New York, 2005.

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

[3] Carmel, E., and R. Agarwal, “Tactical Approaches for Alleviating Distance in Global Software Development”, IEEE Software Vol.18, 2001, pp. 22–29.

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

[5] Ebert C. and P. De Neve, “Surviving global software development”. IEEE Software Vol.18 No. 2, 2001, pp.62–69

[6] Sahay, S., Nicholson, B., Krishna, S.. “Global IT outsourcing: software development across borders”. Cambridge University Press, Cambridge (2003)

[7] Poikolainen T. and J. Paananen, “Performance Criteria in Inter-Organizational Global Software Development Projects”, In proceedings of International Conference on Global Software Engineering ICGSE 2007, pp. 60-70

[8] Moe N.B. and 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

[9] Holmstrom H., E.Ó Conchúir, P. J Ågerfalk, and B. Fitzgerald. “Global Software Development Challenges: A Case Study on Temporal, Geographical and Socio-Cultural Distance”. In Proceedings of the Internation Conference on Global Software Engineering (ICGSE‘06), 2006, pp.3-11

[10] Layman L., L.Williams D.Damian, H.Bures, “Essential communication practices for Extreme Programming in a global software development team”. Information and Software Technology 48, 2006, pp.781–794

[11] Herbsleb J.D., A. Mockus, T.A.Finholt, and R.E.Grinter, “An Empirical Study of Global Software Development: Distance and Speed”. In Proceedings of the International Conference on Software Engineering (ICSE’01), 2001, pp.81-90

[12] Espinosa J.A., N.Nan, and E.Carmel “Do Gradations of Time Zone Separation Make a Difference in Performance? A

First Laboratory Study". In Proceedings of the International Conference on Global Software Engineering (ICGSE'07), 2007, vol., no., pp.12-22, Aug. 2007
[13] Lee J.-N. et al. "The Evolution of Outsourcing Research: What is the Next Issue? ", In the Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000
[14] Šmite D. "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
[15] Šmite D. and 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
[16] Damian D., L.Izquierdo, J.Singer and I.Kwan. "Awareness in the Wild: Why Communication Breakdowns Occur". In Proceedings of the internation Conference on Global Software Engineering (ICGSE07), 2007, pp.81-90
[17] Šmite D. "Project Outcome Predictions: Risk Barometer Based on Historical Data", In the Proceedings of the International Conference on Global Software Engineering ICGSE 2007, pp.103-112
[18] Sjøberg D.I.K., J.E.Hannay, O.Hansen, V.B.Kampenes, A.Karahasanovic, N.-K.Liborg, and A.C.Rekdal. "A Survey of Controlled Experiments in Software" Engineering. In IEEE Transactions on Software Engineering, Vol. 31, No. 9, September 2005, pp. 773-753
[19] Yin, R K, "Case Study Research. Design and Methods" (3rd edition) London: Sage 2003
[20] Lacity M.C. and J.W. Rottman, "The Impact of Outsourcing on Client Project Managers", IEEE Computer, January 2008, pp. 100-102
[21] MacDonell S.G. and M.J. Shepperd "Comparing Local and Global Software Effort Estimation Models -- Reflections on a Systematic Review", In Proceedings of Empirical Software Engineering and Measurement (ESEM'07), 2007, pp.401-409