

Empirical Evidence in Global Software Engineering: A Systematic Review

DARJA SMITE, CLAES WOHLIN, TONY GORSCHER, ROBERT FELDT

IN THE JOURNAL OF EMPIRICAL SOFTWARE ENGINEERING

DOI: 10.1007/s10664-009-9123-y

ONLINE: [HTTP://WWW.SPRINGERLINK.COM/CONTENT/947W6533P607U036/](http://www.springerlink.com/content/947w6533p607u036/)

Abstract: Recognized as one of the trends of the 21st century, globalization of the world economies brought significant changes to nearly all industries, and in particular it includes software development. Many companies started global software engineering (GSE) to benefit from cheaper, faster and better development of software systems, products and services. However, empirical studies indicate that achieving these benefits is not an easy task. Here, we report our findings from investigating empirical evidence in GSE-related research literature. By conducting a systematic review we observe that the GSE field is still immature. The amount of empirical studies is relatively small. The majority of the studies represent problem-oriented reports focusing on different aspects of GSE management rather than in-depth analysis of solutions for example in terms of useful practices or techniques. Companies are still driven by cost reduction strategies, and at the same time, the most frequently discussed recommendations indicate a necessity of investments in travelling and socialization. Thus, at the same time as development goes global there is an ambition to minimize geographical, temporal and cultural separation. These are normally integral parts of cross-border collaboration. In summary, the systematic review results in several descriptive classifications of the papers on empirical studies in GSE and also reports on some best practices identified from literature.

Keywords: Empirical Evidence, Empirical Studies, Systematic Review, Global Software Engineering, Distributed Software Engineering

DARJA SMITE

Blekinge Institute of Technology, PO Box 520, SE- 372 25 Ronneby, Sweden
University of Latvia, Raina bulv.19, LV1869, Riga, Latvia
Darja.Smite@bth.se; Darja.Smite@lu.lv

CLAES WOHLIN

Blekinge Institute of Technology, PO Box 520, SE- 372 25 Ronneby, Sweden
Claes.Wohlin@bth.se

TONY GORSCHER

Blekinge Institute of Technology, PO Box 520, SE- 372 25 Ronneby, Sweden
Tony.Gorschek@bth.se

ROBERT FELDT

Blekinge Institute of Technology, PO Box 520, SE- 372 25 Ronneby, Sweden
Robert.Feldt@bth.se

1 Introduction

Globally distributed software development gained momentum as it promised spectacular benefits. However, it also attracted attention due to the complexity and challenges related to globally distributed development teams. The challenges relate to aspects such as economical, technical, organizational, and cultural issues to those arising from different time zones, languages, and geographical locations (Damian and Moitra 2006). Thus, the expected benefits are challenged from many different aspects, and hence this article focuses on providing a systematic review of the area of globally distributed software development to systematize the empirically documented challenges as well as looking for empirical evidence concerning solutions in this type of situation.

The concept of global software engineering (GSE) originates from contract programming, which was a form of outsourcing known from the 1970s (Lee et al. 2000). In contrast to outsourcing, GSE addresses software engineering activities performed by globally distributed teams. Software development became global in the 1990s as a consequence of the PC revolution (Carmel 1999) and sequential problems of tight budgets, shortage of resources and time motivated many companies to start looking for partners or to set up development sites in different countries. As an outcome of this evolution, many companies built joint ventures and relocated their development centers to low-cost countries. However, soon practitioners started to realize that globally distributed development, in particular from a project management perspective, is considerably more challenging than even the most complex project managed entirely in house (Karolak 1998). Therefore, empirical research results are needed to help understand the challenges with the aim of helping practitioners to improve performance of global software teams. This demand for empirical results is growing as a consequence of an increasing number of internationally distributed software organizations.

Despite the popularity of the topic, the art and science of global software development is still evolving (Damian and Moitra 2006). The results presented by empirical studies and expert reports are still controversial. While some show crucial impediments emphasizing the high amount of failed projects (e.g. (Carmel and Abbott 2007)), others report best practices and lessons learned within success stories (e.g. (Ebert and De Neve 2001; Battin et al. 2001)) though often admitting that the success was not easily achieved. The majority of empirical research seems to be exploratory in nature and focuses on identifying the problems caused by distribution, and possible solutions are discussed in quite general terms. Yet, there is still no standard or recipe for successful GSE performance. In addition, lessons learned often focus on structuring success factors, but give little guidance on how to actually achieve success. In addition, due to the diverse scenarios enabled by different forms of global collaborative work, lessons learned in one context may not directly apply in another context (Smite et al. 2008). This means that learning from existing literature may require considerable effort for practitioners to understand the applicability of the offered findings, putting increased demands on researchers to explicitly and clearly state context and circumstances when reporting empirical results.

Independent of state-of-practice, the number of companies starting global software development grows every day and it is today the “normal” way of doing business (Damian and Moitra 2006; Carmel and Abbott 2007). Many of these companies lack the experience and expertise needed and thus managers are experimenting and quickly adjusting their tactical approaches (Carmel and Agarwal 2001). This motivates us to systematically evaluate the status of the field of global software engineering from an empirical perspective, and provide guidance for future progress. It is worth noting that the scope of this systematic review is limited to the boundaries of the software engineering discipline, which is reflected in our search strategy and search terms. To the best of our knowledge no systematic review in the area of global software engineering with a focus on empirical results exists. The overall objective of our systematic review is to capture the current empirically evaluated knowledge in the area and to identify needs and opportunities for future research.

The remainder of the paper is structured as follows. In Section 2, the research questions for the systematic review are presented. Section 3 provides an outline of the methodology used in the systematic review. The results of the review are presented in Section 4, and conclusions based on the findings are presented in Section 5.

2 Research questions

To understand state-of-the-art in global software engineering, in terms of gaps and commonalities in existing empirical results, the following two research questions were formulated.

Global software engineering is not new and has been practiced in industry for quite some time. Nonetheless, related research indicates that the art and science are still evolving (Damian and Moitra 2006). There are indications of success and total failure, indications of vast amounts of variations in contexts and yet no clear guidance for applying the right practices in the right situation. The main driver for conducting the systematic review was to identify gaps and commonalities in empirical research related to GSE and sum up the existing practices to answer the questions: who, where, what, how and why in distributed software work. The research questions are formulated to form a baseline for state-of-the-art and hence the objective is that the systematic review should form a stepping-stone for both future research and for practical use by practitioners. Thus, our first question is as follows:

RQ.1: What is the state-of-the-art in empirical studies of GSE?

Furthermore, we aimed to understand the existing research directions within global software engineering and in particular empirical research on the topic. The latter is particularly important since it provides information about what we actually know in terms of having evidence. Empirical findings may vary due to the strength of the studies, taking aspects such as sources of evidence and research approaches into account. Many researchers exploring the impact of a GSE environment use synthetic studies or student experiments. While some studies try to test hypothesis, by evaluating methods or tools (empirically-evaluated), others present surveys that are more exploratory in nature (empirically-based). The strength of the empirical evidence in the field provides important information when making decision both about future research and how to practice distributed development globally. Thus, the second research question is:

RQ.2: What is the strength of the empirical evidence reflected in the empirical GSE literature?

3 Review Outline

The systematic review can be divided into planning, realization and reporting activities each of which consists of several steps. Four researchers were involved in the review and it took around one year to complete. An outline of the review can be seen in Fig.1, illustrating the planning, realization and reporting processes on a time scale and the outcomes produced as part of each process.

The planning activity is concerned with developing the review protocol as well as deciding how the researchers should interact and work to conduct the systematic review. In addition, improvements in the review process are shown. The overview of the realization activity reflects the steps taken in the process of conducting the systematic review. The reporting activity shows how the pilot report and the final report evolved. Finally, the outcomes are described in terms of protocols, forms and also how the number of relevant papers changed as the systematic review process progressed. A more detailed description of the review process can be found online¹.

¹ http://web.me.com/darja.smite/Site/Systematic_Review.html


	Planning	Realization	Reporting	Outcomes
November 2007	Protocol development	Data retrieval		Review protocol Repository with articles 
December 2007		Study selection upon titles Study selection upon abstracts Consensus meeting		Primary studies screened
January 2008	Process improvement	Pilot: data extraction, 3 papers (all)		3 papers reviewed Draft: data extraction form
		Revisit reviewed papers		
		Pilot: data extraction, 10 papers (in pairs)		13 papers reviewed
	Process improvement			Definition dictionary Refined: data extraction form
		Revisit reviewed papers (in pairs)		
February 2008		Pilot data extraction: 11 papers (in pairs)		24 papers reviewed
		Disagreement resolution (in pairs)		
		Pilot: data synthesis		
March 2008			Pilot Report	
		Review: data extraction remaining papers		109 papers reviewed
June 2008		Data synthesis		
October 2008			Final Report	

Fig. 1 Activities in the systematic review

The systematic review was conducted according to the procedure described by Kitchenham and Charters (2007). We first developed the systematic review protocol that prescribed a controlled procedure for conducting the review. The protocol included research questions, search strategy, evaluation strategy and inclusion/exclusion criteria, data extraction form and synthesis methods. The protocol was revisited and refined in iterations after piloting each of the related review steps.

The second activity – conducting the review (as outlined in Fig.1) – is broken down to four main steps: data retrieval, study selection, data extraction and data synthesis. Our decisions in each of these steps are reported in Sections 3.1-3.4.

3.1 Data retrieval

To answer the research questions, an extensive search for research papers was conducted. During data retrieval, the boundaries of the systematic review were set. First, the keywords for the search were selected. These aimed at finding empirical research regarding GSE within software engineering. Therefore, we intentionally did not include such keywords as “outsourcing” and “offshoring”. Furthermore, performing searches on full text ensured that, even though the terminology in GSE is rather diverse, the papers related to software engineering activities would certainly (in most cases) contain at least one of the terms that were used. In other words, we chose to use rather general terms for both dimensions used in the search (A and B as shown below). The papers were chosen if they reported on any type of empirical evidence or experience in relation to GSE. The subjects of the study could be either professional software developers or students. The final search strings were based on the experience from the pilot searches and consisted of a Boolean expression: (A1 OR A2 OR A3 OR A4) AND (B1 OR B2 OR B3 OR B4), where

A1 - global software development
A2 - global software engineering

B1 - empirical
B2 - industrial

A3 - distributed software development B3 - experiment
A4 - distributed software engineering B4 - case study

The search strategy contained the following decisions:

Searched databases: Compendex, IEEE Xplore, Springer Link, ISI Web of Knowledge, ScienceDirect, Wiley Inter Science Journal Finder, ACM Digital Library.
Searched items: Journal articles, workshop papers and conference papers.
Search applied on: Full text – to avoid exclusion of the papers that do not include our keywords in titles or abstracts, but are still relevant to the review.
Language: The search strategy limited our search to the papers written in English.
Publication period: Since 2000

Papers published before 2000 were not included in the search. The main motivation for this was due to the fact that GSE as an effect of globalization is recognized as a 21st century trend (Friedman 2005), and that studies conducted after 2000 are more relevant than studies being 10 years old or more.

The outcome of a search generally results in a rather high proportion of papers, which later are judged as being out of scope. Therefore it was insufficient to use the search strings as the sole criteria for deciding whether to include or exclude a specific paper. Thus, the researchers jointly decided the limits based on agreed criteria. This is also one reason why it is crucial that several researchers are involved in a systematic review.

All search results were documented carefully.

3.2 Study selection

The objective of the study (paper) selection process was to identify the papers relevant for the objectives of the systematic review in accordance with the agreed scope. The search strings, as discussed in the previous section, were quite broad and hence it was expected that not all studies identified would make it to the final phase in the selection process. Study selection was performed in six relevance analysis phases as outlined in Table 1. Each phase and the related decisions can be found in the online version of the process overview.

Table 1 Study selection

Relevance analysis phase	Involvement	Inclusion criteria	Papers left
1. Selection of studies based on the search	leading researcher	- only English - date of publication: 2000 – present - only published works - contains the search strings	387
2. Screening: exclusion upon titles	leading researcher	- not editorials, prefaces, discussions, comments, summaries of tutorials, workshops, panels or duplicates	367
3. Screening: exclusion upon abstracts 1	three researchers	- empirical background - subjects practitioners or students	222
4. Screening: exclusion upon abstracts 2	the fourth researcher	- main focus on GSE or software development phase, process or other aspect from GSE perspective	171
5. Consensus meeting	all researchers		109
6. Relevance analysis: exclusion upon full text	all researchers	- presence of empirical data in the paper - originality of empirical evidence (only one inclusion for studies with the same results reported multiple times) - sufficient focus on GSE or software development phase / process / aspect from GSE perspective	59

It is worth emphasizing that the relevance analysis performed does not necessarily refer to evaluating the quality of a paper as such. All papers in the systematic review have been published and thus quality assured through peer-review. The relevance evaluation, in relation to the objective of the systematic review, was aimed at ensuring that the papers finally included were indeed focused on GSE and had sufficient details with respect to the empirical studies presented in the papers.

As an illustration of the selection of studies included and the boundaries of the systematic review, some examples are provided. Several papers referred to that a technique being the main topic in the paper could be applied in a global context. This type of papers was excluded from the study, since the researchers had not actually performed the described study in a global environment. Furthermore, some papers were general project management papers that could be applicable in a GSE context, but the application was not specific for GSE. These papers were also excluded. Thus, for a paper to be included it required that the empirical study was clearly conducted in a distributed setting and that the paper had a clear software engineering component. The latter may include a complete development process, or the study of a specific method, technique or tool in a GSE context. Thus, the objective of what to include and exclude respectively was clear already in the retrieval step, although it was infeasible to formulate search strings that excluded the type of papers mentioned here. Thus, these papers were excluded in the steps described below.

3.3 Data extraction

A template for data extraction was developed to ease the process of synthesizing the gathered data. The template will be called *scheme* from now on. The scheme evolved from the analysis of a selected segment of the data that was conducted as a pilot (reported in (Smite et al. 2008)) prior to the systematic review reported in this article. The content of the scheme corresponds to the guidelines provided by Creswell (2003) who advises extracting information as shown in Table 2. The scheme includes capturing data regarding relevance, empirical background, GSE background, and focus of the study presented. In addition, a qualitative evaluation of the papers was made. The latter was done to cross-check the view of the different researchers performing the review. Each of these areas is further elaborated in Table 2. The extraction form is presented in its original form in Appendix 1.

Table 2 An overview of the extracted data from the papers in the systematic review.

Extracted data type	Corresponding section	Description of the data extracted
Technical and methodological flaws of the study	Relevance	<ul style="list-style-type: none"> - A study contains empirical evidence - A study is relevant to GSE - A study is relevant to SE - A study does not repeat other included studies (relation to other papers)
Information about the sample, population or participants	Empirical background	<ul style="list-style-type: none"> - Main method and sub-methods - Background (industry vs. laboratory) - Subjects of investigation - Empirical focus (empirically-based vs. empirically-evaluated)
	GSE Background	<ul style="list-style-type: none"> - Collaboration mode - Number of sites - Locations of the sites - Perspective of investigation - Reason for outsourcing
Central focus of the study and the problem addressed	Study	<ul style="list-style-type: none"> - Dev. methodology - Focus of the study (practice, phase or other) - Evaluation of the study in terms of success - Application domain
Review of the key results	Qualitative Evaluation	<ul style="list-style-type: none"> - Claims - Personal evaluation - Reported recommendations

To gain a common understanding of the data extraction strategy among the researchers, it was first piloted on a limited number of papers and then improved through a series of iterations. During the further review, papers were equally distributed among the four researchers for individual evaluation. The extracted data was documented using a template and then uploaded into an MS Excel tool, which was further used in the data synthesis phase.

3.4 Data synthesis

Data extracted from the reviewed papers was analyzed quantitatively and qualitatively to answer the research questions. The classification scheme used in data extraction assisted in data categorization with respect to study population, empirical background, and findings of the studies.

We first aimed to summarize the main areas of empirical research related to GSE and drive the future directions in this area using quantitative summaries of the results. This approach usually characterizes systematic mapping studies (Kitchenham and Charters 2007). To illustrate the data, we use bar-chart diagrams supplemented by references to the included papers (from Appendix 2).

At the same time, as a part of addressing our research questions a more thorough analysis and narrative synthesis of the studies was performed, thus leading our work towards a systematic review. For this matter systematic maps based on bubble plots were used to illustrate the data (Petersen et al. 2008). We used multiple perspectives to highlight interesting relationships between different categories. However, due to the limited amount of data, statistical analysis was infeasible. Thus, we are limited to discussions about the results based on the systematic maps presented in Section 4. Qualitative data coding techniques were also used to characterize the focus of each study, including shared recommendations, main focus and success of the reported studies. The categories used to characterize the main focus of the studies were synthesized into the list of topics and mapped to the SWEBOK Knowledge Areas (Abran et al. 2004).

Finally, it should be noted that although all 59 papers were judged as relevant for the goals of the systematic review, while representing the answer to the research questions some of these papers were labelled as irrelevant for the specific investigation. For example, papers reporting on studies of students were regarded as irrelevant when we discussed the reasons for starting global collaboration (costs, competitiveness, quality, etc.).

4 Results

In this section, we present the results of the systematic review based on the 59 papers finally selected. The results are structured based on the research questions stated in Section 2.

4.1 The state-of-the-art in empirical studies of GSE (RQ.1)

To answer **RQ.1**, an analysis based on five perspectives of the empirical literature in the GSE field is presented. The analysis is presented from the following five perspectives: Who, Where, What, How and Why in global software engineering. The following data represents the five perspectives:

- Types of global arrangements, including relation between participating organizations and perspectives of investigation, for example whether a study is conducted from the point of view of the originator of the work or the supplier in terms of development organization (who);
- Geographical location of participating organizations (where);
- Topics addressed in the studies (what);
- Categorization of the empirical observations in terms of success and failure, i.e. how successful are the cases reported (how);
- Reasons for engaging in globally distributed development, development methodologies and their relation to success of a project, as well as recommendations gathered in these empirical investigations (why).

4.1.1 Who is involved in GSE?

Based on an empirical investigation, Poikolainen and Paananen (Poikolainen and Paananen 2007) argue that it is fair to assume that the findings and theories made for international companies (intra-organizational projects) may not directly apply in projects run by a chain of collaborating companies (inter-organizational projects). We thus aimed to understand, how different the GSE arrangements are with respect to objects of investigation, location and number of sites involved in collaboration, and whether the reader can easily evaluate the context of empirical findings of the analyzed studies.

First we present an overview of arrangements that were used as empirical sources in the analyzed studies and proportion of unclear or irrelevant studies (see Fig. 2). From this we can see that 2/3 of the studies (the first three categories as illustrated in Fig. 2) were conducted in industry with the majority of studies coming from international companies. This may also indicate that the progress in the field of global software engineering starting with outsourcing in the end of the last century has driven companies to build joint ventures and international corporations as foreseen in related literature (Carmel 1999; Sahay et al. 2003). This includes multi-national companies having development sites in different countries. There are only few studies about inter-organizational collaboration. 10% of the analyzed studies were based on both inter- and intra-organizational distributed projects. About 25% of the studies used students as subjects in the investigation. These studies were in most cases joint projects between universities in different countries. Finally, the studies that were neither conducted in an organizational nor in a student setting were categorized as irrelevant. In particular, there were two papers based on open-source projects and one based on expert opinions.

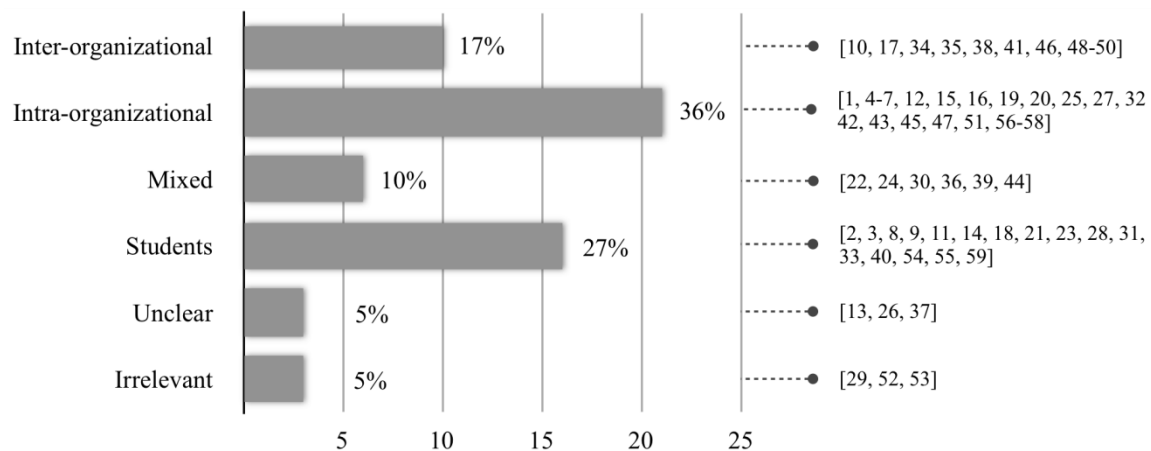


Fig. 2 Global arrangements (references to the papers in Appendix 2 to the right)

An overview of the literature on IT outsourcing history before 2000 conducted by Jae-Nam Lee et al. (Lee et al. 2000) uncovered that most of the research in this area was conducted mainly by companies starting outsourcing collaborations, i.e from the perspective of customers because the objective of outsourcing was to self-maximize their internal resources without taking into account the supplier situation. It is likely that investigating GSE problems from a distance or using only a single perspective may influence the results of the studies, since some of the true reasons behind the problems with GSE can be hidden. Furthermore, some of the solutions applied in only one site may not have the desired effect of improving joint performance. Thus the perspectives of the analyzed studies were also captured (see Fig. 3). For this purpose we selected the following categories:

- Collaboration in general, if the study was conducted at, or addressed problems and solutions relevant for, all involved partner locations;
- Supplier, if the study was conducted at, or addressed problems and solutions specific for, the sub-contractor (in inter-organizational collaboration) or a remote site that is receiving work tasks from the central location (in intra-organizational collaboration);
- Originator, if the study was conducted at, or addressed problems and solutions specific for, the

contractor (in inter-organizational collaboration) or the central location responsible for completion of the work being distributed across remote locations (in intra-organizational collaboration);

Each paper is represented by a single category.

Our results show that the vast majority of the analyzed studies are addressing collaboration between originators and suppliers. At the same time, the supplier perspective dominates over the originator perspective in single-perspective studies. Interestingly, most of the studies conducted from the originator perspective cover intra-organizational collaborations. There are also a number of studies with unclear perspectives; most of these studies had very vague background descriptions in general and the perspective could not be judged. The irrelevant category was common for student studies, since these studies are mostly based on joint project work as part of a course and do not have the intention to develop software used in commercial products.

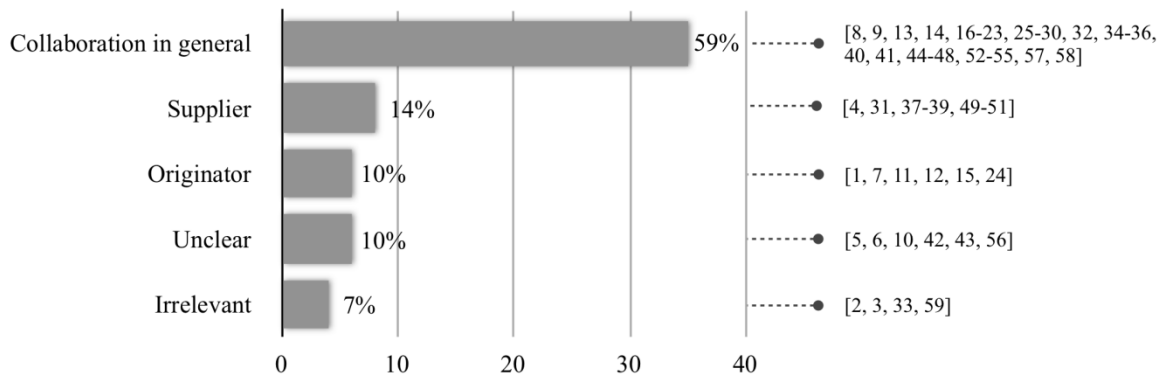


Fig. 3 Perspectives of the analyzed studies

The data also indicates that in contradiction to the previous observations (Lee et al. 2000), there is an increasing number of studies conducted from the team perspective, thus showing the progress in analyzing the problems jointly.

4.1.2 Where are the development sites located?

The origin and location of the sites involved in GSE collaboration may be important to understand the usefulness of the offered findings. The data provides an overview of the population of the analyzed studies. We have uncovered 28 different geographic locations of companies involved in GSE. Our findings show that the most frequently mentioned locations are Unites States (18 studies), India (13 studies), UK (5 studies), Germany (5 studies), and China (5 studies).

Complementing the results of Carmel and Abbott (2007) who analyzed locations of customers and suppliers mentioned in studies on nearshoring, the locations of the sites discussed in the analyzed studies are shown on Fig. 4. Our mapping differs from the one used in the mentioned study, since we looked for the following types of locations:

- △ Location of GSE suppliers from inter-organizational collaboration, in other words sites that were sub-contracted to do work from the originator;
- ▲ Location of GSE originators from inter-organizational collaboration, in other words sites that engaged one or more suppliers in completion of a joint project;
- Location of international companies from intra-organizational collaboration.

Each country can be represented as either a location of particular type or a mixed type, and illustrated by a combined pictogram. Thus, each country has one pictogram and hence Fig. 4 does not capture the number of studies in a specific country.

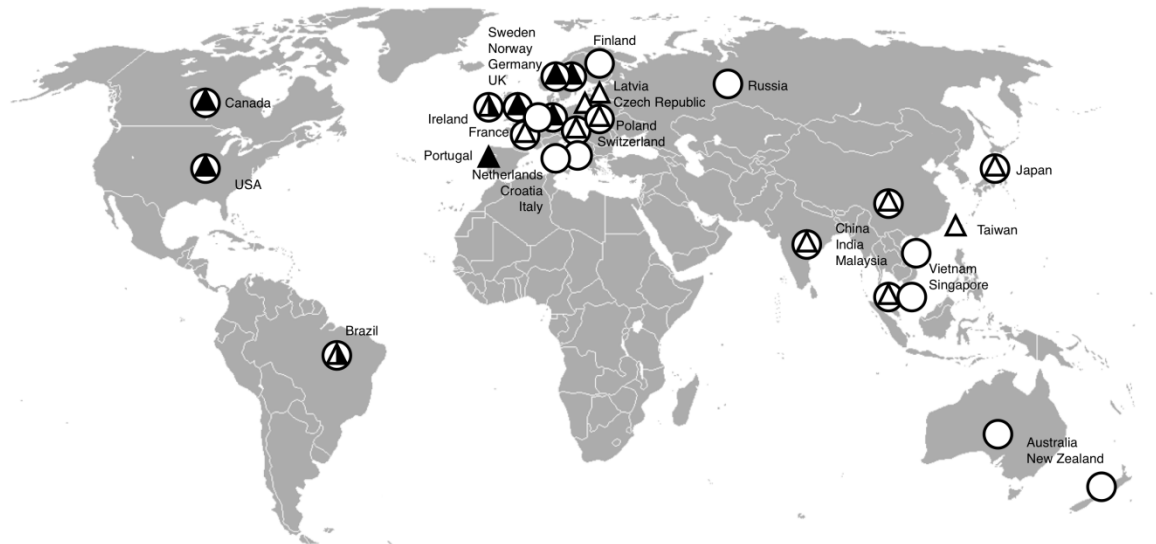


Fig. 4 Location of partners from the analyzed studies

Unfortunately, due to the limited amount of studies, it was impossible to derive any viable correlations of success, topic or even type of arrangement dependent on different locations.

Additionally we analyzed how highly distributed the analyzed GSE projects are by capturing the number of sites involved in the studied collaboration projects. For this case we counted only identifiable sources of empirical evidence – information about each collaboration project or case study that we could identify in the descriptions of empirical background. We identified 92 cases from 34 studies, thus some of the studies are overrepresented. For example, if a multiple case study was based on four student projects each involving two sites, we counted four for the two sites/students category. The categories and the results of the analysis are illustrated in a bubble plot diagram (as suggested by Petersen et al. 2008) with two dimensions to better represent the population of our review (see Fig. 5).

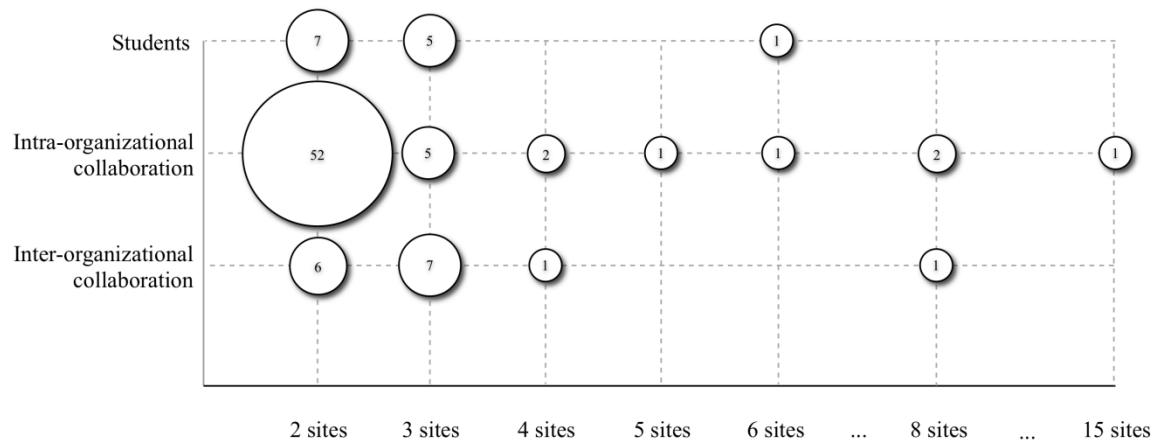


Fig. 5 Number of partners from the analyzed studies

As can be seen, a clear majority of the identifiable sources of empirical evidence are studies on intra-organizational collaboration between two sites of a company. These findings emphasize the lack of inter-organizational studies once again. The amount of studies lacking clear description of the empirical background is quite high - we have counted 25 unidentifiable studies that are not present in Fig. 5; in eight of these studies the empirical background was irrelevant and in 17 studies it was not clearly described.

4.1.3 What is studied in GSE?

To understand what is known and what is not known about GSE we captured the topics of investigation and major focus of each study. This was done through a qualitative analysis of each study during the data retrieval process. Then, the identified topics were classified using broader categories. Additionally, we mapped these categories to the Knowledge Areas defined by SWEBOK (Abran et al. 2004). Each study is ultimately represented by a single category. Fig. 6 shows an overview of the topics addressed by the analyzed studies.

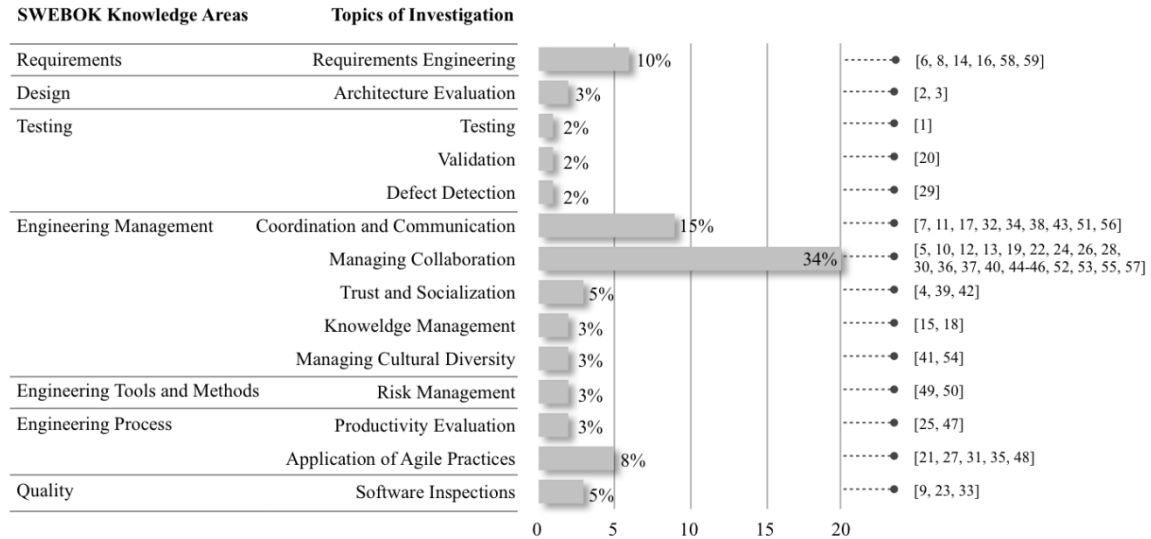


Fig. 6 Detailed list of covered topics

Most of the studies were classified as managerial, since they address different aspects of managing distributed collaboration. Other studies were focused on a particular topic, practice or development phase. From these, the most popular topics of empirical investigation were requirements engineering, coordination and communication, and the application of agile processes.

4.1.4 How successful are the cases reported in literature?

In addition to capturing the topics, we investigated how successful reported cases were. Accordingly, while reading the papers we marked whether it was a success story, a failure story, a report on successful practices, failed practices or a problem report. The difference between success story and successful practice is that a success story refers to the success of a project or the case studied, and a successful practice is more limited in that it may be a reference to something successful within a project even if the project could have failed. The same reasoning is used for failure story and failed practice. Each paper can be represented by multiple categories in case of diverse conclusions or multiple-case studies (see Fig. 7).

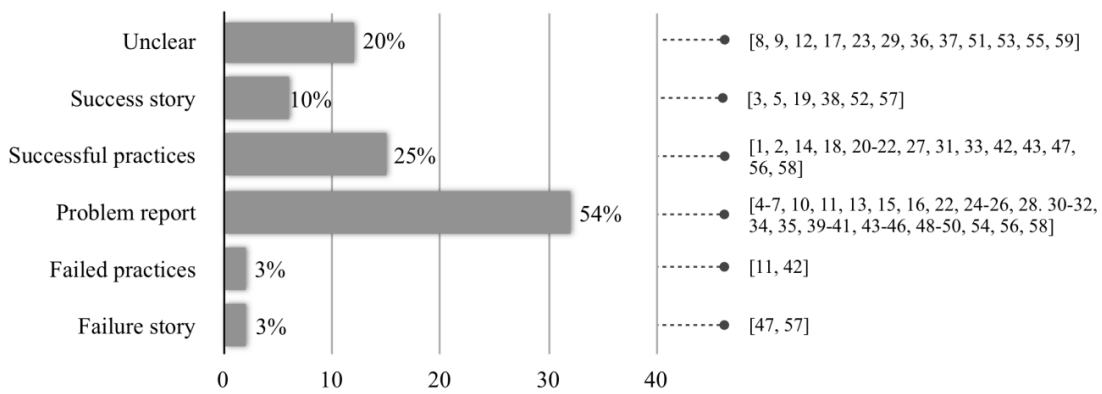


Fig. 7 Success evaluation

Fig. 8 illustrates our findings from multiple perspectives. In particular, we have classified the analyzed studies according to the Knowledge Areas defined by SWEBOK (Abran et al. 2004). The topics in Fig. 8 have been mapped to the Knowledge Areas in SWEBOK and these are illustrated with respect to the success or failure as described above. In addition, the studies are mapped to the Knowledge Areas in relation to the context in terms of inter-organization, intra-organizational, students, unclear or irrelevant.

To avoid multiple representation of a study in this particular case we have given $\frac{1}{2}$ score for studies reporting on success and failure or both intra- and inter-organizational contexts.

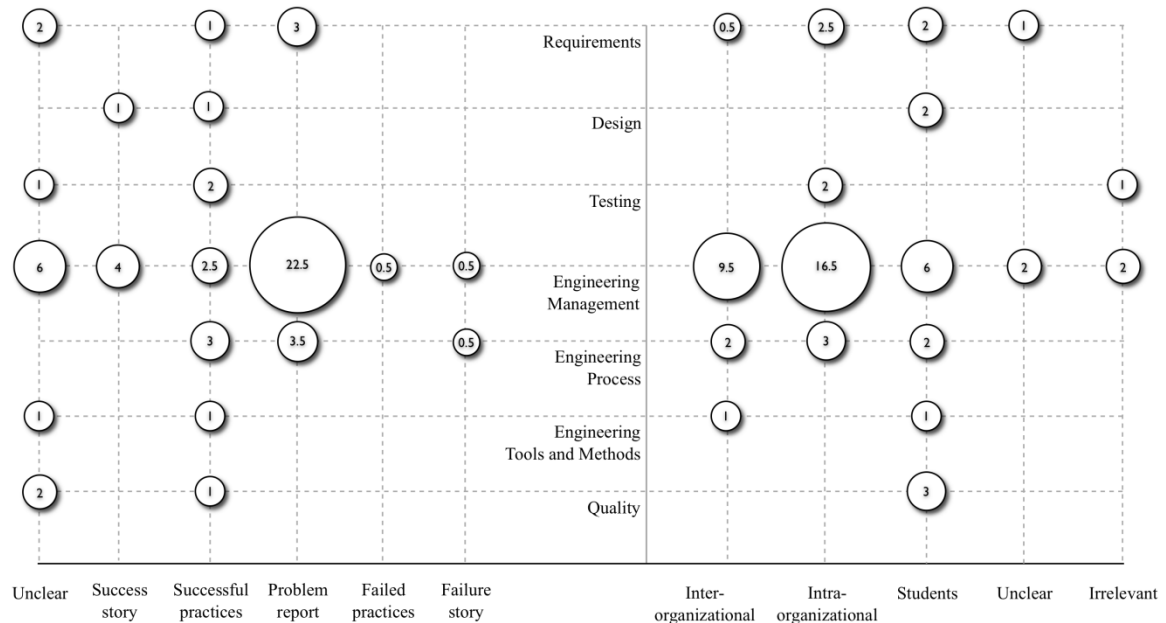


Fig. 8 Distribution of topics

We conclude that the majority of empirical research represents problem-oriented reports that focus on different aspects of GSE management. Interestingly, while student experiments address different knowledge areas, there is a lack of studies conducted in industrial environments that at the same time investigate a particular method, practice or aspect of software engineering knowledge areas. In particular there seem to be a lack of studies related to methods and tools for design, testing and quality. Even more notably there were no studies particularly addressing the SWEBOK knowledge areas of software construction, maintenance and configuration management, and hence these areas have been skipped in the figure.

These findings indicate that the field is still quite immature in terms of being more problem-oriented than focused on solutions and particularly on empirically evaluated solutions. A much smaller group of papers discusses successful practices and shares lessons learned that are devoted to a variety of different topics. There are only a handful of clear success stories or clear failure stories that one can learn from. Several explanations may exist. It is possible that the complexity of globally distributed development makes it hard to identify specific reasons for either success or failure. It is also possible that it is due to the competitiveness, i.e. companies are not able to share failures since it may affect the perception of them.

4.1.5 Why are companies involved in GSE?

Although GSE has become popular for its promised benefits of cheaper, faster and better development of software products, empirical studies have indicated that achieving these benefits is not an easy task (Conchuir et al. 2006). In fact, it has been recognized that collaborations built upon a pure cost reduction strategy tend to fail. To understand the reasons why global software engineering projects fail, we studied the correlation between project success and the motivation behind “going global” decisions. We tried to

deduce the necessary information from the analyzed collaborations. Although most studies provided general reasons for global software development in the introduction of the paper, only a few studies explicitly described the reasons for starting global collaboration in the investigated empirical cases (see Fig. 9). The irrelevant category was common for student studies and other experiments, since these do not involve any commercial decisions for outsourcing.

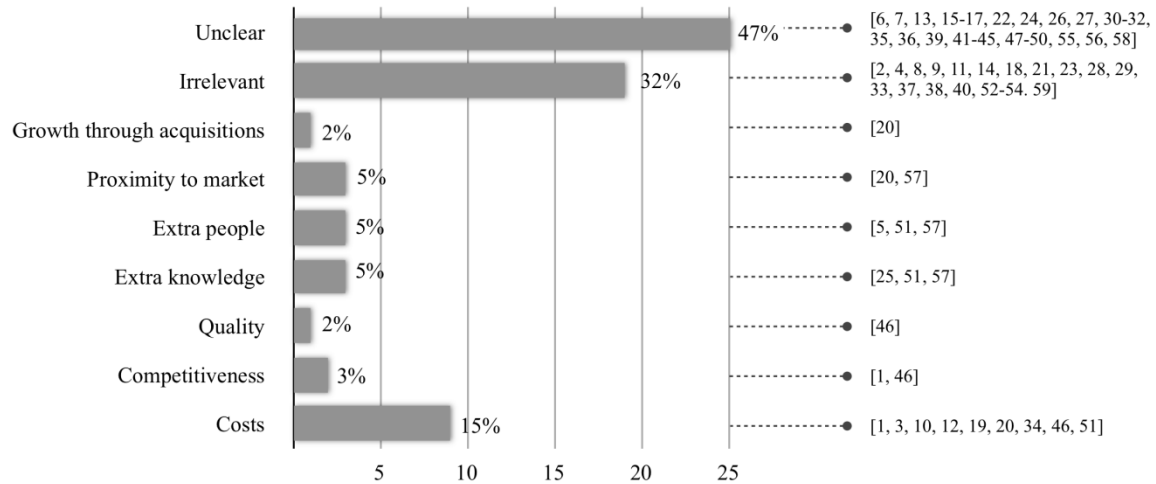


Fig. 9 Reasons for starting global collaboration

Our results show that in line with the related research (Karolak 1998) cost reduction remains the dominating motive for starting global collaboration. Interestingly, though some other reasons such as proximity to market, extra people and knowledge, are also mentioned, they usually supplemented cost reduction motives and only three studies did not mention costs as one of the reason for global collaboration, i.e. after having removed the unclear and irrelevant categories.

Since practitioners are still searching for recipes for success in managing globally distributed projects, we wanted to understand whether any pattern could be identified, for example, in terms of usage of a specific development methodology (see Fig. 10).

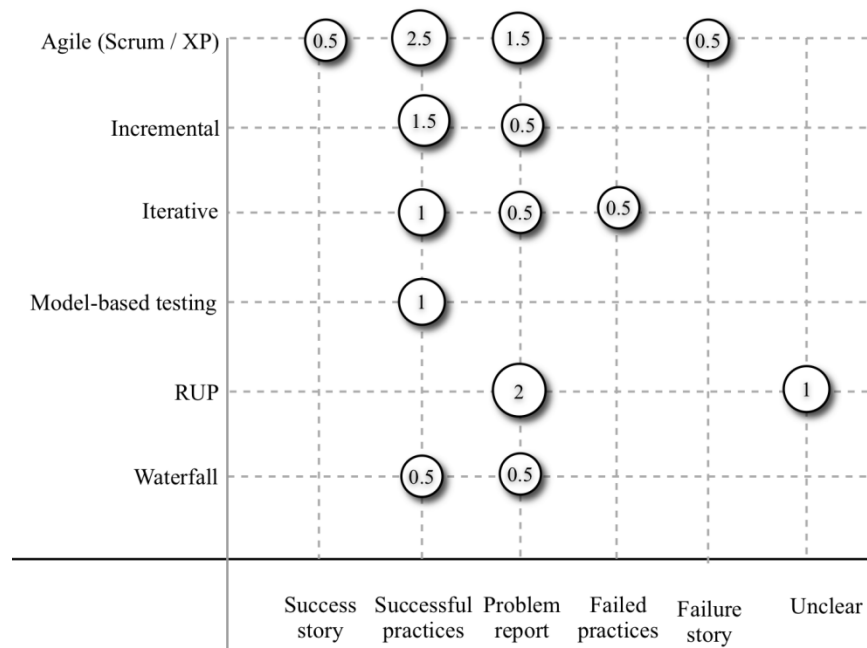


Fig. 10 Reasons for starting global collaboration

To our great disappointment, we found that information about development methodologies was very vague. In 40 cases we could not determine this information, and in five cases it was irrelevant (these were omitted from the figure). Agile methods were the most popular among the identifiable studies, followed by RUP (three cases), incremental (two cases) and iterative development (two cases). Though our data is not sufficient to make any reliable conclusion, it provides an interesting overview of methodologies tried in a distributed context. We may also observe that agile, incremental and iterative development form a group of related methods that are gaining interest in GSE as well. As expected, the data also indicates that other factors than just a methodology determines success or failure of global projects, since the results of the studied projects vary.

As mentioned earlier, there seems to be no recipe for successful global collaboration, although some papers address best practices and give practical recommendations. These practices and their potential benefits and their corresponding constraints are presented in Table 3. Interestingly, the qualitative analysis of these recommendations indicates a conflict between cost saving strategies of the companies starting global collaboration and amount of investments necessary for their implementation (see Table 3). This may also explain why achievement of successful collaboration in a globally distributed environment is problematic (or at best challenging).

Table 3 Best practices

Practice	Benefits	Constraints
F2F meetings, temporal collocation, exchange visits	Trust, cohesiveness, effective teamwork	Requires extra costs
Centralized project repository, common configuration management tool support	Awareness, process transparency	Requires overcoming heterogeneity
Effective and frequent synchronous communication	Trust, cohesiveness	Requires temporal proximity and extra time
Reliable infrastructure, rich communication media	Effective communication	Requires extra costs
Synchronous interaction	Effective teamwork	Requires temporal proximity
Task distribution based on architectural decoupling and low dependencies across remote locations	Effective teamwork	Requires full transition of parts of the work
Incremental short-cycle development	Early feedback, capability evaluation	Requires frequent and transparent communication

Geographical, temporal and cultural differences are found to have a significant effect on how distributed team members interact, in particular, many challenges have been found in relation to communication, coordination and control (Ågerfalk et al. 2005). Our findings indirectly support these observations and suggest that the most frequently discussed practices focus on these areas of concern. These recommendations address different aspects of work: human, technical and process. Human-oriented practices suggest addressing communication, interaction and teamwork. Interestingly, while distributed development is mainly characterized by a geographical separation, empirical findings urge to collocate team members as much as possible to facilitate effective teamwork (Ebert et al. 2001). Technical-oriented practices, on the other hand, focus on implementing effective infrastructure and providing tool support to enable computer-mediated collaboration. At the same time, according to some research findings, specialized tools for the support of e.g. articulation of work in a distributed environment are not so common in practice (Boden et al. 2007). Finally process-oriented practices suggest adopting incremental development and introducing architectural principles for task allocation. Although distributed software development typically relies on formal mechanisms, there is a growing trend towards balancing agile and distributed approaches to meet the challenges of communication, control and trust (Ramesh et al. 2006; Holmström et al. 2006).

It is also notable that through these practices organizations try to minimize the exposure of geographical, temporal and cultural separation by requiring regular visits and synchronous work. While the mentioned practices have been widely discussed as a prerequisite of success, it also means that successful distributed work has limitations. We thus claim, that in order to correspond to the best practices in GSE, organizations shall either be prepared to ensure significant investments into team support (travels, communication and coordination overhead), or alternatively consider collaboration with nearshore locations. This is in line with conclusions of Carmel and Abbott (2007) who state that, when sourcing abroad, a growing number of companies now weigh whether the location is near or far.

4.1.6 Summary

We summarize the answer to the **RQ.1** by concluding that most of the empirical findings presented in the analyzed studies are based on intra-organizational industrial collaboration between two geographically distributed sites. USA and India remain the major players in GSE or at least the main sources of empirical investigation represented in the related literature. There is a clear lack of studies about inter-organizational collaboration and highly distributed projects with more than just two collaborating partners. The amount of studies with mixed, not described or unclear contexts is relatively high. This burdens the understanding of applicability of the results and requires additional effort from the readers. Accordingly, we encourage researchers to thoroughly describe the contexts of the undertaken studies.

Concerning the content of the studies, we conclude that the majority of empirical research represents problem-oriented reports that focus on managerial aspects. There is a clear lack of studies conducted in industrial environments that investigate a particular method, practice or aspect of software engineering. Only few papers address successful practices and share lessons learned, although it is possible to deduce some best practices as listed in Table 3.

Suggested practices address communication and collaboration challenges and tend to minimize geographical, temporal and cultural separation. At the same time, implementation of these practices often requires additional investments, thus conflicting with the cost saving strategies. This indicates that nearshoring is an alternative solution that may enable more successful collaboration, although it may not result in the cost-savings intended. Another notable trend is adoption of more human-centred agile approaches that have a potential of positively affecting the quality of collaboration across distance.

4.2 Strength of empirical evidence (RQ.2)

As a part of the systematic review we investigated the nature of empirical evidence presented in the GSE studies and analyzed the type of studies conducted. The main motivation being that different types of empirical studies provide different strengths of evidence. Practitioners should take the strength of evidence into consideration when deciding on GSE practices based on the existing literature. Thus, it is important to understand the state of research methods used when studying GSE.

4.2.1 Sources of evidence

Reading the methodological descriptions of the analyzed studies we captured sources of empirical evidence to understand the viability of the offered findings. We did not judge the claims of the researchers and made no corrections, thus the resulting map has repetitions and may be perceived as lacking the desired clarity (see Fig.11). The first column from the left represents the main method used by the researchers, and the following columns reflect additional sources of evidence. For example, we identified 34 case studies and in 23 of these the authors mention that they have performed interviews as part of the case study. In nine of these 23 studies the researchers also mentioned that they performed a document analysis, and so forth. However, it is important to note that this is how the authors presented the methodology. It may very well be the case that in the eleven case studies not reporting a sub-method, such as interviews, the researchers used a sub-method but did not report it. Once again it points to the need for clear presentations.

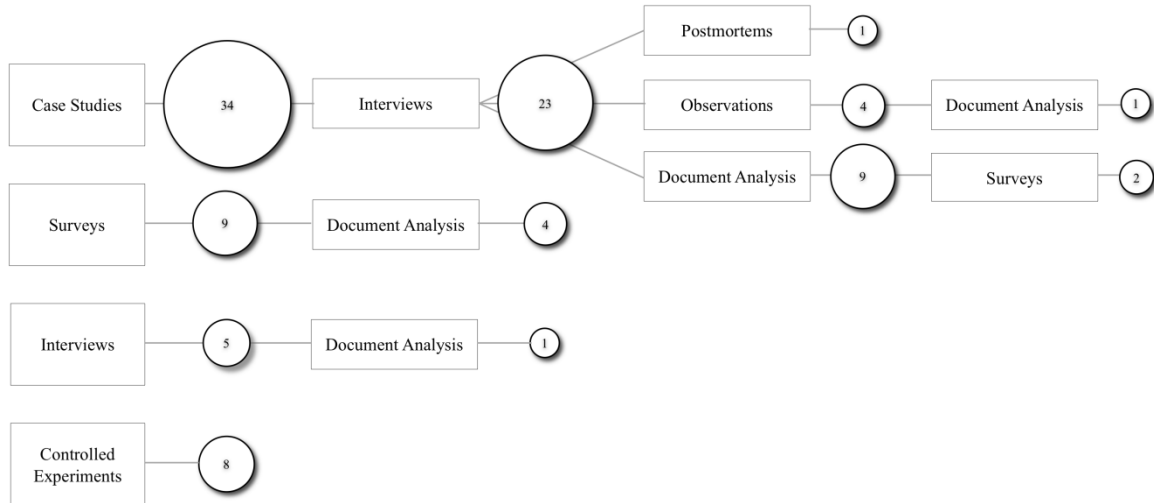


Fig. 11 Bubble Tree graph with cumulative analysis of sources of empirical evidence

The analysis of the studies shows that the majority of research is exploratory case studies. In many of these cases the researchers have collected data and report on challenges, problems and success factors for GSE.

4.2.2 Research approaches

Since research within immature disciplines tends to be more exploratory in nature than research in mature fields that focuses more on testing hypothesis, methods or tools, our review also addressed evaluation of the proportion between empirically-based versus empirically-evaluated research in the GSE field (see Fig.12). Here we refer to empirically-based as a study basing its conclusions on empirical data, but not performing any actual empirical evaluation. If doing the latter, be it a practice, a method, a framework or a tool, it is referred to as empirically-evaluated research. As previously noted, half-scores were rewarded to papers that had addressed several categories. As can be seen in Fig. 12 there was one research paper that included both students and practitioners in the reported study.

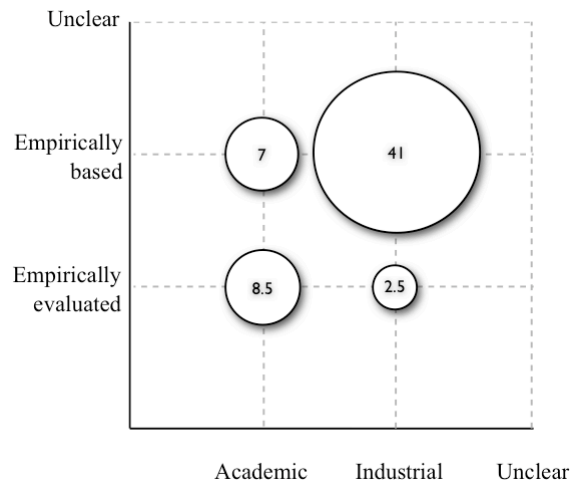


Fig. 12 Research approaches in academic and industrial studies

Fig. 12 shows that a majority of the studies are based on empirical data. Only 11 out of 59 studies were classified as empirically evaluated research, i.e. where the researchers actually evaluate a method, technique or tool for GSE. It is notable that most of studies performing an empirical evaluation are laboratory experiments using students as subjects, which leaves very few studies evaluating software engineering methods, techniques and tools in an industrial environment where GSE is practiced.

4.2.3 Relevant and irrelevant studies

One of our observations from conducting the systematic review addresses the ability for readers to search and find empirical studies in the GSE. Using the most popular terms related to the field, our search strategy revealed 387 studies, but only 59 papers were recognized as relevant after performing a thorough relevance evaluation. As stated above, although a paper is relevant for the goal of the systematic review it may for a specific investigation be irrelevant. This means that only 15% of the found studies qualified for the review. An overview of each step of the study selection/exclusion process is described above in Table 1 in Section 3.2.

Because we used our search criteria on full text, the search results provided a relatively high number of irrelevant studies. However, we do not advise the readers to use the same keywords on titles, since we found relevant papers that contained none of the searched keywords in their titles, or instead used very diverse keywords, such as distributed product management, globally distributed workforce, offshore outsourcing, global software teams, virtual teams or distributed teams.

The analysis shows that a majority of excluded studies was either not specifically focused on GSE or was GSE-related but not empirical studies. As stated above, it was observed that several studies were not studies conducted in a GSE context; instead they mentioned that the results could be relevant in a GSE context. Thus, the authors identified the potential relevance of their research for a GSE context, but given that it is mostly conjectures on behalf of the authors these papers have been excluded from the final data set. Several authors also report the same (or very similar) results in multiple publications, which we eliminated to avoid bias of the results. The list of the 59 included studies can be found in Appendix 2.

4.2.4 Summary

Summarizing the findings in relation to **RQ.2**, we report that the analyzed study material comes from different sources. More than half of the papers describe case studies, which are mainly based on interviews. All of the controlled experiments are conducted in laboratory setting using students as subjects, except for one that is an industrial pseudo-experiment.

It is also important to mention that deducing the methodological part of the studies was not an easy task. Many studies have vague descriptions of both empirical background and methods of investigation. Lacking evidence of empirical sources was also one of the major reasons for exclusion of the papers during the full-text review. The analysis indicates that the GSE field is still in an immature state with a lack of empirical evaluation of methods, techniques and tools in an industrial context.

4.3 Validity threats

One of the general ideas with a systematic review is to capture the current status of an area given that, for example, the search is made systematically and it is clearly shown how the review was conducted. This should be borne in mind when considering threats to the validity of the findings from a systematic review. Three types of validity should be considered: external validity that is the possibility to generalize the findings, construct validity that is concerned with obtaining the right measure and finally reliability (or conclusion validity) is related to the ability to replicate and to come to the same findings if the study is being done again.

In general, the external validity and construct validity are strong for a systematic review. The key idea is to capture as much as possible of the available literature to avoid all sorts of bias. The main challenge in addressing these threats in our systematic review was to define the scope of our investigation, since global software engineering refers to cross-disciplinary knowledge areas. We have chosen to focus on empirical software engineering research and not cover other related disciplines. This is the main limitation of our findings. It is worth noting that a systematic review is per definition limited by search date, sources used in the search and terminology used in the search. As part of the work we have observed the challenges in covering several different disciplines such as software engineering, information systems and management. In some cases these disciplines use different terminology, which makes cross-disciplinary searches much harder than just searching another database. This is a challenge that must be addressed to overcome some

of the limitations, and hence enabling possibilities of making progress in, for example, such an area as global software engineering (or if using a similar, but broader term global software development).

Another challenge with a systematic review is addressing the reliability threats. The reliability is addressed and mitigated as far as possible by involving several researchers, and having a scheme, a process and several steps where the scheme and process were piloted and hence evaluated. If the study is replicated by another set of researchers, it is highly likely that some papers that were removed in our study will be included and other papers would be excluded. However, it is highly unlikely that these random differences based on personal judgments would change the general findings. It may change the actual numbers somewhat, but it is not likely that it would change the overall findings. Thus, it is concluded that in general we believe that the validity of the study is high given the use of a very systematic procedure and the involvement and discussion among four researchers.

In summary, the openness in a systematic review when it comes to search strategy, criteria and so forth allows other researchers to judge the trustworthiness of the results more objectively.

5 Conclusions

The interest, in software development performed by geographically, temporally and/or culturally diverse teams, started with outsourcing in the last 10-20 years and continues to increase. The importance of the area means that it now deserves to be addressed as a specific area of research and practice called global software engineering (GSE). In this article we described the results of our systematic review of GSE-related empirical research aiming to evaluate the current state-of-the-art in the field.

5.1 Principal findings

Looking back, Loh and Venkatraman in 1995 emphasized that despite its popularity no research could determine the exact recipe for effective outsourcing performance (Loh and Venkatraman 1995). In 2006, Damian and Moitra in their introduction of the IEEE Software special issue on global software development declared: “although it’s true that 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” (Damian and Moitra 2006).

In response to RQ1, the systematic review confirms these views, as the amount of GSE-related empirical studies is still relatively small. However, we note that in order to provide an absolute judgement, more investigation into cross-disciplinary research work is necessary. The analysis of the empirical studies included in the systematic review thus shows that the GSE field is still in an immature state with a lack of empirical evaluation of methods, techniques and tools in an industrial context. With respect to RQ2, we discovered that a majority of empirical research represents problem-oriented reports focusing on different aspects of GSE management rather than in-depth analysis of particular practices or techniques.

We have also identified seven most frequently discussed practices that are highlighted as prerequisites for success. These practices, however, often conflict with the cost reduction strategies of the organizations that start collaboration with remote locations. While the mentioned practices have been widely discussed as a prerequisite of success, it also means that successful distributed work has limitations.

5.2 Implications for practitioners

The systematic review of empirical literature in GSE helps to observe the state-of-the-art as well as the state-of-practice in relation to the field of global software engineering. Analyzing the current status, we draw a parallel between previous observations in industry and empirical research today. In 2001, Carmel and Agarwal (2001) wrote that various managers are experimenting and quickly adjusting their tactical approaches for leveraging global software development risks. This is also reflected by the empirical literature, since the majority of the studies that we found are exploratory studies with the main emphasis on the difficulties with GSE and lessons learned from experience. While analyzing the used methodologies, the number of sites, their locations and other project characteristics, we could not determine any strong correlation with their success. On the other hand, the most frequently discussed practices aim at minimizing

the exposure of geographical, temporal and cultural separation by emphasizing regular visits and synchronous work. This ultimately requires additional investments and conflicts with the main driver for global software development, namely, the cost reduction strategy. Thus, we conclude that there is still no recipe for successful and efficient performance in globally distributed software engineering.

Nonetheless, the reported best practices and lessons learned may help practitioners to better understand the nature of GSE projects and analyze the problems associated with distributed collaboration.

5.3 Implications for future research

The analysis of the topics and the nature of the investigated studies uncovered several areas that require empirical results.

1. There is a clear lack of studies about inter-organizational collaboration and about highly distributed projects.
2. The coverage of topics in GSE-related literature also points out a gap regarding in-depth empirical investigations addressing particular aspects of software engineering. Thus, future research ought to evaluate different practices, methods and techniques rather than mainly focus on managerial problem-oriented lessons learned.
3. Since benefits of distributed software development are discussed and the benefits are sometimes controversial, future research shall focus on measuring and reporting evidence of success and failure of the studied cases, as well as capturing major benefits and the overall return on investment of GSE. Together with thorough descriptions of the context, this should enable a possibility to measure the correlation between different characteristics of the GSE projects and their success, and help to derive support for beneficial collaboration.

As already noted, obtaining the papers with empirical data related to the topic of our systematic review was one of the main challenges since globally distributed work is a cross-disciplinary research frontier. To obtain a comprehensive review across disciplines, the terminology used in the searches has to be adapted to cover the terminology of the different disciplines. Thus, we conclude that in the future attention should be paid to bridging the findings brought up by different disciplines and harmonizing the terminology.

Another difficulty that we faced was related to the process of deducing and collecting information about the empirical background of the analyzed studies. This underlines the necessity for thorough descriptions of contexts in which the reported studies are carried out. This is particularly important in GSE-related research, since lessons learned in one context may not directly apply in another context. The classification scheme can be used as a guide for structuring these descriptions (Smite et al. 2008).

Acknowledgements

This research is supported by the Knowledge Foundation in Sweden under a research grant for the project *BESQ*. We would like to thank the editor handling the paper and the anonymous reviewers for providing valuable feedback that helped improve the paper.

References

- Ågerfalk P J, Fitzgerald B, Holmström H, Lings B, Lundell B, Conchúir E Ó (2005) A Framework for Considering Opportunities and Threats in Distributed Software Development, In: Proceedings of the International Workshop on Distributed Software Development, Paris, 29, Austrian Computer Society, 47-61
- Abran A, Moore J W et al. (2004) Guide to the Software Engineering Body of Knowledge (SWEBOK®), IEEE Computer Society 2004 Guide
- Battin R D, Crocker R, Kreidler J, Subramanian K (2001) Leveraging resources in global software development. In: IEEE Software 18(2):70-77
- Beamish P W, Delios A (1997) Incidence and Propensity of Alliance Formation, in Cooperative Strategies: European Perspectives, New Lexington Press, San Francisco

- Boden A, Nett B and Wulf V (2007) Coordination Practices in Distributed Software Development of Small Enterprises, In: proceedings of the International Conference on Global Software Engineering ICGSE 2007, pp. 235-246
- Carmel E (1999) Global Software Teams: Collaborating Across Borders and Time Zones. Prentice-Hall
- Carmel E Abbott P (2007) Why 'Nearshore' Means That Distance Matters. In: Communications of the ACM, 50(10):40-46
- Carmel E, Agarwal R (2001) Tactical Approaches for Alleviating Distance in Global Software Development. In: IEEE Software, 18(2): 22-29
- Conchuir E O, Holmström H, Ågerfalk P J, Fitzgerald B (2006) Exploring the Assumed Benefits of Global Software Development, In: Proceedings of the 1st International Conference on Global Software Engineering, pp.159-168
- Creswell J W (2003) Research Design: Qualitative, Quantitative, and Mixed Methods Approaches, 2nd edition. In: Sage Publications Inc.
- Damian D, Moitra D (2006) Global Software Development: How Far Have We Come? IEEE Software, 23(5):17-19.
- Ebert C, De Neve P (2001) Surviving global software development. In: IEEE Software 18(2): 62-69
- Ebert C, Parro C H, Suttels R, Kolarczyk H (2001) Improving validation activities in a global software development, In: proceedings of the International Conference on Software Engineering ICSE 2001, pp. 545-554
- Friedman T L (2005) The World is Flat: Brief History of the 21st Century. Farrar, Straus and Giroux, New York
- Holmström H, Fitzgerald B, Ågerfalk P J, Conchuir E Ó (2006) Agile practices reduce distance in global software development, In: Information Systems Management 23 (3), pp. 7-18
- Karolak D W (1998) Global Software Development: Managing Virtual Teams and Environments. IEEE Computer Society
- Kitchenham B, Charters S (2007) Guidelines for performing Systematic Literature Reviews in Software Engineering, version 2.3. EBSE Technical Report
- Lee J N, Huynh M Q, Kwok C W, Pi S M (2000) The Evolution of Outsourcing Research: What is the Next Issue? In: Proceedings of the 33rd Hawaii International Conference on Systems Sciences, Hawaii, pp.1-10
- Loh L, Venkatraman N (1995) An Empirical Study of Information Technology Outsourcing: Benefits, Risks, and Performance Implications. In: Proceedings of the 16th International Conference on Information Systems, pp.277-288
- Petersen K, Feldt R, Mujtaba S, Mattsson M (2008) Systematic Mapping Studies in Software Engineering. In: proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering, pp. 71-80
- Poikolainen T Paananen J (2007) Performance Criteria in Inter-Organizational Global Software Development Projects. In: proceedings of International Conference on Global Software Engineering ICGSE, pp. 60-70
- Ramesh B, Cao L, Mohan K, Xu P (2006) Flexible and distributed software processes: old petunias in new bowls? Can distributed software development be agile?, In: Communications of the ACM, 49(10), pp. 41-46
- Sahay S, Nicholson B, Krishna S (2003) Global IT Outsourcing: Software Development across Borders. Cambridge University Press
- Smite D, Wohlin C, Feldt R, Gorschek T (2008) Reporting Empirical Research in Global Software Engineering: a Classification Scheme. In: proceedings of International Conference on Global Software Engineering ICGSE 2008, pp. 173-181

Appendix 1: Data Extraction Form

General Information	
Identifier	
Title	
Author(s)	
Source	
Abstract	
Relevance	
Is this paper relevant to SE field?	<input type="radio"/> Highly relevant <input type="radio"/> Relevant <input type="radio"/> Irrelevant
Is this paper relevant to GSE field?	<input type="radio"/> Highly relevant <input type="radio"/> Relevant <input type="radio"/> Irrelevant
Is this an empirical study?	<input type="radio"/> Yes <input type="radio"/> No
Does this paper repeat already reviewed paper(s)?	<input type="radio"/> Yes (Overlap with Nr.: ... / Supplement/Extension to: ...) <input type="radio"/> No
Empirical Background	
Main Method	<input type="checkbox"/> Survey <input type="checkbox"/> Case Study <input type="checkbox"/> Interviews <input type="checkbox"/> Controlled experiment <input type="checkbox"/> Other:...
Sub-Methods	<input type="checkbox"/> Survey <input type="checkbox"/> Case Study <input type="checkbox"/> Interviews <input type="checkbox"/> Archive analysis <input type="checkbox"/> Controlled experiment <input type="checkbox"/> Other:...
Background	<input type="checkbox"/> Laboratory <input type="checkbox"/> Industry/Real world
Subjects of investigation	<input type="checkbox"/> Students <input type="checkbox"/> Industry/Real world
Empirical focus	<input type="checkbox"/> Empirically based <input type="checkbox"/> Empirically evaluated
GSE Background	
Collaboration mode	<input type="checkbox"/> Inter-organizational <input type="checkbox"/> Intra-organizational <input type="checkbox"/> Unclear <input type="checkbox"/> Irrelevant
Number of sites	... <input type="radio"/> Unclear <input type="radio"/> Irrelevant
Location of the originator	... <input type="radio"/> Unclear <input type="radio"/> Irrelevant
Location of the suppliers	... <input type="radio"/> Unclear <input type="radio"/> Irrelevant
Perspective	<input type="radio"/> Supplier <input type="radio"/> Originator <input type="radio"/> Collaboration in general <input type="radio"/> Unclear <input type="radio"/> Irrelevant
Reason for outsourcing	<input type="checkbox"/> Costs <input type="checkbox"/> Extra knowledge <input type="checkbox"/> Extra people <input type="checkbox"/> Unclear <input type="checkbox"/> Irrelevant <input type="checkbox"/> Other ...
Study	
Dev. methodology	... <input type="radio"/> Unclear
Focus of the study	<input type="radio"/> Collaboration in general: ... <input type="radio"/> Single practice(s): ... <input type="radio"/> Development phase(s): ... <input type="radio"/> Other: ...
Success or failure?	<input type="checkbox"/> Clear success story <input type="checkbox"/> Success of the practices described <input type="checkbox"/> Clear failure story <input type="checkbox"/> Failure of the practices described <input type="checkbox"/> Evidence of the GSE-related problems <input type="checkbox"/> Unclear <input type="checkbox"/> Other: ...
Application domain	<input type="checkbox"/> Telecom <input type="checkbox"/> Automotive <input type="checkbox"/> Web <input type="checkbox"/> Finance <input type="checkbox"/> Automation <input type="checkbox"/> Other: ... <input type="checkbox"/> Unclear
Definitions in the introduction-like sections?	<input type="checkbox"/> No <input type="checkbox"/> GSE, pp.: ... <input type="checkbox"/> GSD pp.: ... <input type="checkbox"/> Outsourcing pp.: ... <input type="checkbox"/> Offshoring pp.: ... <input type="checkbox"/> Nearshoring pp.:... <input type="checkbox"/> Global team pp.:... <input type="checkbox"/> Virtual team pp.: ... <input type="checkbox"/> Other related definition ... pp.: ...
Qualitative Evaluation	
Claims	<i>narrative</i>
Personal reflection	<i>narrative</i>
Recommendations	<i>narrative</i>

Appendix 2: Included studies

Paper	Ref
Andaloussi B. S.; Braun A. "A Test Specification Method for Software Interoperability Tests in Offshore Scenarios: A Case Study", In the Proceedings of the 1st International Conference on Global Software Engineering, Oct. 2006, pp.169-178	1
Babar M. Ali, B. Kitchenham and R. Jeffery „Comparing distributed and face-to-face meetings for software architecture evaluation: A controlled experiment“, Empirical Software Engineering, Vol. 13, Issue 1, 2008, pp. 39-62 (In SpringerLink since 4 Oct 2007)	2
Babar M. Ali, B. Kitchenham, R. Jeffery „Distributed versus face-to-face meetings for architecture evaluation: a controlled experiment“, In the Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering, 2006, pp. 252 - 261	3
Babar M. Ali, J. M. Verner, and P.T. Nguyen „Establishing and maintaining trust in software outsourcing relationships: An empirical investigation“, Journal of Systems and Software, Vol. 80, No. 9, September, 2007, pp. 1438-1449	4
Battin R. D., R. Crocker, J. Kreidler and K. Subramanian „Leveraging resources in global software development“, IEEE Software, Vol. 18, No. 2, March/April, 2001, pp. 70-77	5
Bhat J. M., M. Gupta, and S. N. Murthy „Overcoming Requirements Engineering Challenges: Lessons from Offshore Outsourcing“, IEEE Software, Vol.23, No.5, Sept.-Oct. 2006, pp. 38-44	6
Boden A., B. Nett and V. Wulf „Coordination Practices in Distributed Software Development of Small Enterprises“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.235-246	7
Calefato F., D. Damian and F. Lanubile „An Empirical Investigation on Text-Based Communication“ in Distributed Requirements Workshops of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.3-11	8
Calefato, F., Lanubile, F., Mallardo, T. "A Controlled Experiment on the Effects of Synchronicity in Remote Inspection Meetings", In the 1 st International symposium on Empirical Software Engineering and Measurement, Sept. 2007, pp.473-475	9
Casey V. and I. Richardson „Uncovering the reality within virtual software teams“, In the Proceedings of the international workshop on Global software development for the practitioner, 2006, pp. 66-72	10
Cataldo M., M. Bass, J. D. Herbsleb, and L. Bass „On Coordination Mechanisms in Global Software Development“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.71-80	11
Conchúir Ó., H. Holmström, P. J. Ågerfalk and B. Fitzgerald „Exploring the Assumed Benefits of Global Software Development“, In the Proceedings of the 1st International Conference on Global Software Engineering, Oct. 2006, pp.159-168	12
Dalberg V., E. Angelvik, D. R. Elvekrok, A. K. Fossberg „Cross-cultural collaboration in ICT procurement“ In the International Conference on Software Engineering Proceedings of the 2006 international workshop on Global software development for the practitioner, 2006, pp. 51-57	13
Damian D., F. Lanubile and T. Mallardo „An empirical study of the impact of asynchronous discussions on remote synchronous requirements meetings“, published by Lecture Notes in Computer Science, Vol. 3922, In the Proceedings of the 9th international conference on Fundamental Approaches to Software Engineering, 2006, pp. 155-169	14
Damian D., L. Izquierdo, J. Singer and I. Kwan „Awareness in the Wild: Why Communication Breakdowns Occur“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.81-90	15
Damian D. and D. Zowghi „The impact of stakeholders' geographical distribution on managing requirements in a multi-site organization“ In the Proceedings of the IEEE Joint International Conference on Requirements Engineering, 2002, pp. 319-328	16
De Souza C. R. B., T. Hildenbrand and D. Redmiles „Toward Visualization and Analysis of Traceability Relationships in Distributed and Offshore Software Development Projects“, published by LNCS, Vol. 4716, Software Engineering Approaches for Offshore and Outsourced Development, 2007, pp. 182-199	17
Dutoit A. H., J. Johnstone and B. Bruegge „Knowledge scouts: Reducing communication barriers in a distributed software development project“, In the Proceedings of the Asia-Pacific Software Engineering Conference and International Computer Science Conference, 2001, pp. 427-430	18
Ebert C. and P. De Neve, „Surviving global software development“, IEEE Software, Vol. 18, No. 2, March/April, 2001, pp. 62-69	19
Ebert C., C. H. Parro, R. Suttels, and H. Kolarczyk „Improving validation activities in a global software	20

development“, In the Proceedings of the International Conference on Software Engineering, 2001, pp. 545-554	
Favela J., H. Natsu, C. Pérez, O. Robles, A. L. Morán, R. Romero, A. M. Martínez-Enríquez and D. Decouchant „Empirical Evaluation of Collaborative Support for Distributed Pair Programming“, In Lecture Notes in Computer Science: Groupware: Design, Implementation and Use, Vol. 3198, 2004, pp. 215–222	21
Gumm D.-C. „Mutual Dependency of Distribution, Benefits and Causes: An Empirical Study“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.113-124	22
Harjumaa L. „Distributed software inspections - An experiment with Adobe Acrobat“, In the Proceedings of the IASTED International Conference on Computer Science and Technology, 2003, pp. 26-31	23
Herbsleb J. D., D. J. Paulish, and M. Bass „Global software development at Siemens: Experience from nine project“, In the proceedings of the 27th International Conference on Software Engineering, 2005, pp. 524-533	24
Herbsleb J. D. and A. Mockus „An empirical study of speed and communication in globally distributed software development“, IEEE Transactions on Software Engineering, Vol. 29, No. 6, June, 2003, pp. 481-494	25
Holmström 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 the Proceedings of the 1st International Conference on Global Software Engineering, Oct. 2006, pp. 3-11	26
Holmström H., B. Fitzgerald, P. J. Ågerfalk and E. Ó. Conchúir „Agile practices reduce distance in global software development“, Information Systems Management 23 (3), 2006, pp. 7-18	27
Huang H. and R. Ocker „Preliminary insights into the in-group/out-group effect in partially distributed teams: an analysis of participant reflections“, In the Proceedings of the ACM SIGMIS conference on computer personnel research: Forty four years of computer personnel research: achievements, challenges & the future, pp. 264 - 272	28
Jacobs J., J. van Moll, P. Krause, R. Kusters, J. Trienekens and A. Brombacher „Exploring defect causes in products developed by virtual teams“, Information and software technology, Vol. 47, No. 6, 2005, pp. 399-410	29
Kommeren R. and P. Parviainen „Philips experiences in global distributed software development“, Empirical Software Engineering, Vol. 12, No. 6, December 2007, pp. 647-660	30
Korkala M. and P. Abrahamsson „Communication in Distributed Agile Development: A Case Study“ In the Proceedings of the 33rd EUROMICRO Conference on Software Engineering and Advanced Applications, 28-31 Aug. 2007, pp.203-210	31
Kotlarsky J., I. Oshri, K. Kumar and J. van Hillegersberg „Globally distributed component-based software development: an exploratory study of knowledge management and work division“, Journal of Information Technology No. 22 (2), June 2007, pp. 161-173	32
Lanubile F., T. Mallardo, and F. Calefato „Tool support for geographically dispersed inspection teams“, Software Process Improvement and Practice, Vol. 8, No. 4, October/December, 2003, pp. 217-231	33
Lasser S. and M. Heiss, „Collaboration maturity and the offshoring cost barrier: The trade-off between flexibility in team composition and cross-site communication effort in geographically distributed development projects“, In the Proceedings of the IEEE International Professional Communication Conference, 2005, pp. 718-728	34
Layman L., L. Williams, D. Damian, and H. Bures, „Essential communication practices for Extreme Programming in a global software development team“, Information and Software Technology, Vol. 48, No. 9, September, 2006, Distributed Software Development, pp. 781-794	35
Lings B., B. Lundell, P. J. Ågerfalk and B. Fitzgerald „A reference model for successful Distributed Development of Software Systems“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.130-139	36
Ma J., J. Li, W. Chen, R. Conradi, J. Ji and C. Liu „An Industrial Survey of Software Outsourcing in China“, published by LNCS, Vol. 4589, In the Proceedings of the International PROFES Conference, 2007, pp. 5-19	37
McChesney I. R. and S. Gallagher „Communication and co-ordination practices in software engineering projects“, Information and Software Technology, Vol. 46, Issue 7, June 2004, pp. 473-489	38
Moe N. B., and D. Smite „Understanding lacking trust in global software teams: A multi-case study“, published by Lecture Notes in Computer Science, Vol. 4589, In the proceedings of the 8th international PROFES conference, 2007, pp. 20-34	39
Munkvold B. E. and I. Zigurs „Process and technology challenges in swift-starting virtual teams“, Information & Management, Vol. 44, Issue 3, April 2007, pp. 287-299	40
Nicholson B. and S. Sahay „Some political and cultural issues in the globalisation of software development: case experience from Britain and India“, Information and Organization, Vol. 11, Issue 1, January 2001, pp. 25-43	41

Oshri I., J. Kotlarsky and L. P. Willcocks „Global software development: Exploring socialization and face-to-face meetings in distributed strategic projects“, The Journal of Strategic Information Systems, Volume 16, Issue 1, March 2007, pp. 25-49	42
Ovaska P., M. Rossi, P. Marttiin „Architecture as a coordination tool in multi-site software development“, Software Process Improvement and Practice, Vol. 8, Issue 4, October/December 2003, pp. 233-247	43
Prikladnicki R., J. L. N. Audy, D. Damian and T. C. de Oliveira „Distributed Software Development: Practices and challenges in different business strategies of offshoring and onshoring“, In the proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp. 262-274	44
Prikladnicki R.; J. L. N. Audy and R. Evaristo „Reference Model for Global Software Development: Findings from a Case Study“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.18-28	45
Prikladnicki R., J. L. N. Audy and R. Evaristo „Global software development in practice lessons learned“, Software Process Improvement and Practice, Vol. 8, No. 4, October/December, 2003, pp. 267-281	46
Ramasubbu N. and R. K. Balan „Globally distributed software development project performance: an empirical analysis“, In the Proceedings of the the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on the foundations of software engineering, pp. 125-134	47
Ramesh B., L. Cao, K. Mohan and P. Xu „Flexible and distributed software processes: old petunias in new bowls?: Can distributed software development be agile?“, Communications of the ACM, Vol. 49, Issue 10, October 2006 pp.: 41 - 46	48
Smite D. „Project Outcome Predictions: Risk Barometer Based on Historical“, In the Proceedings of the 2nd International Conference on Global Software Engineering, 27-30 Aug. 2007, pp.-112	49
Smite D. „Global software development projects in one of the biggest companies in Latvia: Is geographical distribution a problem?“, Software Process Improvement and Practice, Vol. 11, No. 1, January/February, 2006, pp. 61-76	50
Smite D. „A case study: Coordination practices in global software development“, published by Lecture Notes in Computer Science, Vol. 3547, In the proceedings of the 6th International PROFES Conference, 2005, pp. 234-244	51
Spinellis D. „Case studies: Global software development in the freeBSD project“, In the International Conference on Software Engineering Proceedings of the 2006 international workshop on Global software development for the practitioner, pp. 73-79	52
Stewart K. J., G. Sanjay „The moderating role of development stage in free/open source software project performance“, Software Process Improvement and Practice, V. 11, Issue 2, March/April 2006, pp. 177-191	53
Swigger K., F. Alpaslan, R. Brazile and M. Monticino „Effects of culture on computer-supported international collaborations“, International Journal of Human-Computer Studies, Vol. 60, Issue 3, March 2004, pp. 365-380	54
Taweel A. and P. Brereton „Developing software across time zones: An exploratory empirical study“, Informatica (Ljubljana), Vol. 26, No. 3, November, 2002, pp. 333-344	55
Taxen L. „An integration centric approach for the coordination of distributed software development projects“, Information and Software Technology, Vol. 48, No. 9, September, 2006, Distributed Software Development, pp. 767-780	56
Treinen J. J. and S.L. Miller-Frost „Following the sun: Case studies in global software development“, IBM Systems Journal, 45 (4), October-December 2006, pp. 773-783	57
Välimäki A. and J. Kääräinen „Requirements Management Practices as Patterns for Distributed Product Management“, published by LNCS Vol. 4589, In the Proceedings of the International PROFES Conference, 2007, pp.188-200	58
Van de Walle B., C. Campbell and F. P. Deek „The Impact of Task Structure and Negotiation Sequence on Distributed Requirements Negotiation Activity, Conflict, and Satisfaction“, published by LNCS, Vol. 4495, Advanced Information Systems Engineering, 2007, pp. 381-394	59



DARJA SMITE is an assistant professor of software engineering at Blekinge Institute of Technology and a docent at University of Latvia. Her research interests include global software engineering with the emphasis on improving distributed team efficiency, requirements engineering, team coordination and software process improvement. Smite received a PhD in computer science from University of Latvia. Contact her at Blekinge Inst. of Technology, PO Box 520, SE-372 25 Ronneby, Sweden; Darja.Smite@bth.se or University of Latvia, Computing Department; Raina bulv. 19, LV-1869, Riga, Latvia; Darja.Smite@lu.lv



CLAES WOHLIN is a professor of software engineering and the pro vice chancellor of the Blekinge Institute of Technology. His research interests include empirical methods in software engineering, software metrics, software quality, and requirements engineering. Wohlin received a PhD in communication systems from Lund University. He is the Editor-in-Chief of Information and Software Technology published by Elsevier. Furthermore, he is a professorial visiting fellow of University of New South Wales in Sydney, Australia. He is a senior member of the IEEE and a member of the ACM. Contact him at Blekinge Inst. of Technology, PO Box 520, SE-372 25 Ronneby, Sweden; Claes.Wohlin@bth.se.



TONY GORSCHKEK is an associate professor of software engineering at the Blekinge Institute of Technology, Sweden. He also works as a consultant and manages his own company. His research interests include requirements engineering, technical product management, process assessment and improvement, and quality assurance. He conducts most of his research in close collaboration with industry. Gorschek received a PhD in software engineering from the Blekinge Institute of Technology. He is a member of the IEEE. Contact him at Blekinge Inst. of Technology, PO Box 520, SE-372 25 Ronneby, Sweden; Tony.Gorschek@bth.se



ROBERT FELDT is an assistant professor in software engineering at Blekinge Institute of Technology. His research interests include automated verification and validation, search-based software engineering, global software development and artificial intelligence for software engineering support. He received his PhD in Software Engineering from Chalmers University of Technology in Gothenburg. He is a member of the IEEE. Contact him at Blekinge Inst. of Technology, PO Box 520, SE-372 25 Ronneby, Sweden; Robert.Feldt@bth.se