Journal of Software: Evolution and Process

JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS
J. Softw. Evol. and Proc. 2012; **24**:625–642
Published online 5 October 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/smr.563

An evidence-based model of distributed software development project management: results from a systematic mapping study

Fabio Q. B. da Silva[1,*,†], Rafael Prikladnicki[2], A. César C. França[1], Cleviton V. F. Monteiro[1], Catarina Costa[1] and Rodrigo Rocha[1]

[1]*Centro de Informática (CIn) da UFPE, Recife, Pernambuco Brazil*
[2]*Faculdade de Informática, PUCRS, Porto Alegre, Rio Grande do Sul Brazil*

ABSTRACT

Distributed software development (DSD) has intensified over the past few years, and DSD project management is more complex than collocated project management. However, no systematic research effort has focused on aggregating evidence from the scientific literature to build models to support project management on DSD context. For these reasons, the goal of this paper is to build an evidence-based model of DSD project management from the research findings about challenges of DSD and the practices, models and tools proposed and used to overcome these challenges. We based the construction of this model on the evidence collected and synthesized by a comprehensive systematic mapping study of 70 research papers published between 1997 and 2009. We believe that our results can help practitioners and researchers to better understand the challenges and implement more effective solutions to improve project management within distributed project management teams. These results also provide a mapping of the research about DSD project management, identifying areas where further research is needed.

Received 2 August 2010; Revised 21 June 2011; Accepted 14 July 2011

KEY WORDS: Project Management; Distributed Software Development; Software Engineering; Systematic Literature Review; Systematic Mapping Studies

1. INTRODUCTION

Distributed software development (DSD) has intensified over the past few years. Software and IT industries are truly global nowadays, and so is software engineering (SE). Many reasons drove the rise of this development practice, where reducing development costs, improving the project quality and having access to skilled professionals worldwide can be highlighted. However, the growth of DSD increases the challenges associated to it. The diversity of culture and the dispersion over time and space require novel techniques, tools, and practices to overcome challenges and to take advantage of opportunities that DSD offers.

According to Betz & Mäkiö [3], about 40% of DSD projects fail to deliver the expected benefits because of the lack of theoretical basics and the fact that this kind of projects comprises complications that even experienced managers are not aware of. For this reason, DSD project management is more complex than collocated project management. Although there is increasing recognition among practitioners and academics that DSD imposes greater challenges and difficulties for project management, most organizations still manage distributed projects using the same methods, processes, and tools used in traditional projects [4,14,17].

*Correspondence to: Fabio Q. B. da Silva, Centro de Informática, Universidade Federal de Pernambuco (CIn – UFPE), Cidade Universitária, Recife 50740-540, PE, Brazil.
†E-mail: fabio@cin.ufpe.br

In addition, the knowledge about DSD project management is not structured. There are some studies that describe challenges of distributed project management, and other studies that describe solutions to improve the management of distributed projects [6,11,14–16,17]. But we are not aware of efforts that were done towards a more structured understanding of distributed project management, including the review of challenges and possible solutions in terms of practices, models, and tools. For this reason, we carried out a systematic mapping study (MS) of the DSD project management literature. Da Silva *et al.* [8] have published a summary of preliminary results of the MS, and the complete study has been submitted to publication [9].

The goal of this current work is to build an evidence-based model of DSD project management from the research findings identified in the MS about challenges of DSD and the practices, models, and tools proposed and used to overcome these challenges. We based the construction of this model on the evidence collected and synthesized from 70 research papers published between 1997 and 2009. We believe this model will be important for researchers that want to have a broad view of the factors that affect DSD project management and to look for the areas in which research is lacking. Moreover, the findings will be useful for practitioners that seek possible solutions for practical problems. Ultimately, this study will be useful for software organizations that undertake or are considering undertaking DSD projects.

This article is organized as follows. In Section 2, we present a brief overview of DSD. In Section 3, we briefly describe the method used in the systematic MS and how the evidences were collected and synthesized. In Section 4, we summarize the main findings of the systematic MS, used in the construction of the DSD project management model. In Section 5, we discuss the findings, present the model, and discuss some the implications for research and practice. Finally, in Section 6, we present some conclusions and areas for future work.

2. BACKGROUND—DISTRIBUTED SOFTWARE DEVELOPMENT

As part of the globalization efforts currently pervading society, software project team members have become geographically distributed. That is a characteristic of DSD. When the distance becomes global, with team members distributed around the world, this characterizes global software development (GSD). The many factors that contributed to DSD or GSD are well documented in literature. Engineers, managers, and entrepreneurs are facing many challenges on technical, social, political and cultural levels. This change is having a considerable impact on the way products are conceived, designed, tested, and delivered to customers. Thus, the organizational structure and development processes required to support DSD are different from those used in collocated environments. Herbsleb & Moitra [20] say that DSD has different effects on many levels: strategic issues (decision on developing a distributed project), cultural issues, technical issues (technological infrastructure and technical knowledge) and knowledge management issues.

In this context, DSD is a growing field within the SE domain. Many companies are distributing their software development facilities, looking for competitive advantages in terms of cost, quality, and skilled professionals. Carmel & Tjia [21] argue that the DSD phenomenon started in the early 1990s, but it was only recognized as a powerful competitive strategy in the last 10 years. Whether local (onshoring) or global (offshoring), within the same company (insourcing) or as a third-party relationship (outsourcing), organizations are facing several important challenges from a SE perspective.

After observing and documenting practices and challenges in the industry, it makes sense to try to understand how these practices are related, what are the challenges associated to each practice, and solutions that have been proposed to overcome some of the challenges. This is being done from several perspectives, with systematic literature reviews and mapping studies conducted in the last few years in order to have a more structured body of knowledge in the field.

More recently, our research team conducted a MS about the challenges, solutions and practices of DSD project management [8,9]. This MS analyzed over 4000 research papers and aggregated evidences from 70 studies that addressed challenges and solutions for DSD project management. In this article, we present a proposition of a model of DSD project management based on the findings reported in the MS. This model relates the challenges of DSD project management, building chains of causal relationships and state a set of hypothesis that are supported by the evidence found in the literature.

DOI: 10.1002/smr

3. REVIEW METHOD

In 2004, Kitchenham *et al*. [12] introduced the concept of evidence-based SE as a promising approach to integrate academic research and industrial practice in SE. Systematic literature review is the preferred method to collect, analyze, and synthesize results in evidence-based research [13]. The literature differentiates two main types of systematic literature reviews [18], including conventional Sistematic Literature Reviews (SLRs) [18] and mapping (or scoping) studies (MS) [1] that aim to identify all research related to a specific topic, i.e., to answer broader questions related to trends in research.

Da Silva *et al*. [9] presented a MS of the research about DSD project management. The goal was to analyze available studies and integrate the findings to produce guidelines for practitioners and academics. In their work, the guidelines of Kitchenham & Charters [13] were followed to plan and execute the review. In this section, we summarize the review method used by da Silva *et al*. [9] to collect and aggregate the evidence that is used in this paper to build the DSD project management model.

3.1. Inclusion and exclusion criteria

Studies were eligible for inclusion in the MS if they presented challenges and related solutions in terms of practices, tools, and models to improve the management of DSD projects. No restriction was imposed on the type of the studies; thus, articles reporting empirical studies (based on direct observation or experiments), theoretical studies (based on an understanding of the theme from experience or reference to other works), industrial experience reports, and literature reviews were included. Empirical studies that used either students or professional software developers as subjects were included. Only studies written in English were included. Studies showing opinion pieces, viewpoints, or purely anecdotal evidence, and those presenting in progress research or incomplete results were excluded.

3.2. Data sources and search strategy

A broad search process was performed looking for peer-reviewed articles published up to August 2010, combining automatic and manual search to increase coverage. Manual search was performed on relevant journals and conference proceedings (Table I) and on the references of selected studies. The researchers looked for title of all published articles in each source used in the manual search.

The automatic search was performed in five search engines and indexing systems: ACM Digital Library, IEEEXplore Digital Library, Science Direct, El Compendex, and Scopus. All automatic searches were performed on the entire paper, including title and abstract. The automatic search used search terms built based on the following research question:

RQ1. What are the challenges in DSD project management?

RQ2. What are the practices that may improve the management of DSD projects?

Table I. Manual search sources.

Software process improvement and practice, Oct/Dec 2003, Special Issue on Global Software Development
ACM Queue, Dec/Jan 2003–2004, special issue on distributed development
IEEE Software, Mar/Apr 2001, special issue on global software development
IEEE Software, Sept/Oct 2006, special issue on global software development
Information and Software Technology, Sept 2006, special issue on distributed software development
CACM, Oct 2006, special issue on flexible and distributed software processes
IT & People, 2008, special issue on global sourcing
Workshop on Global Software Development at ICSE, 2004 and 2003
International Workshop on Distributed Software Development, 2005
1st International Workshop on Global Software Development for the Practitioner, 2006
ICGSE 2006, 2007, 2008, and 2009—1st, 2nd, 3rd, and 4th International Conference on Global Software Engineering
SEAFOOD 2007, 2008, and 2009—Software Engineering Approaches for Offshore Software Development

DOI: 10.1002/smr

RQ3. What are the tools used to support the management of DSD projects?

RQ4. What models of DSD exist?

The search terms for each question were built in three steps. First, keywords were identified in the question. Second, synonyms for the keywords were defined by consulting specialists in DSD. Third, the search terms were built by joining the synonyms with operator OR and the set of synonyms of each keyword using the operator AND. Wildcards, like "*", were added. Additionally, other adjustments necessary to fit the syntax of the search engines were done in the original search term, and recorded in the protocol document. As an example, the search term for research question RQ3 is as follows:

> ("Distributed software development" OR "Global software development" OR "Collaborative software development" OR "Global software engineering" OR "Globally distributed work" OR "Collaborative software engineering" OR "Distributed development" OR "Distributed teams" OR "Global software teams" OR "Globally distributed development" OR "Geographically distributed software development" OR "Offshore software development" OR "Offshoring" OR "Offshore" OR "Offshore outsourcing" OR "Dispersed teams") AND (Model* OR Process* OR Framework* OR Method* OR Technique* OR Methodolog* OR Tool* OR Software* OR Program* OR System*) AND ("Project Management")

The coverage of the search terms was verified by checking whether the main authors and articles in the field were retrieved. Furthermore, the inclusion of manual search had the objective of complementing potential (an almost always unavoidable) limitations of the automatic search. Figure 1 summarizes the review process and the number of papers identified at each stage.

3.3. Study selection

Results from the automatic and manual search at Stage 1 ($n = 1992$) were evaluated in Stage 2 by one research looking at the title and excluding the studies that were clearly not relevant, resulting in 155 potentially relevant studies. In Stage 3, the same researcher read the abstract and conclusion, selecting the studies based on the inclusion and exclusion criteria described in Section 3.1. The 54 studies selected at Stage 3 were assessed for quality, and data were extracted from each paper. The findings from the studies selected at Stage 3 were synthesized, published, and presented at International Conference on Global Software Engineering (ICGSE) 2010 [8].

In Stage 4, automatic search was performed in the Scopus indexing system, and manual search was performed in the journal and proceedings listed on Table I and on the references of the 54 articles selected in Stage 3. This broader search strategy was deployed to increase coverage and therefore reduce bias in the selection of primary studies. The resulting articles from Stage 4 ($n = 2195$) were evaluated in Stage 5 by four researchers based on the title of the publication, resulting in 697 potentially relevant studies. In Stage 6, the same four researchers read the abstract and conclusion, selecting the studies based on the inclusion and exclusion criteria. The 16 studies resulting from Stage 6 were assessed for quality, and data were extracted. In Stage 7, the findings of all 70 studies resulting from Stage 3 ($n = 54$) and Stage 6 ($n = 16$) were integrated, and the aggregation of the results has been submitted for publication [9]. In Section 4, we summarize the results that are relevant in the construction of the DSD project management model. In Appendix A, we present the complete reference list of the papers selected in the MS.

4. RESULTS

The following sections present the description of challenges, practices, tools, and models related to DSD project management aggregated by the MS developed by da Silva *et al.* [9]. Challenges, practices, tools, and models are presented separately and are related in Section 5.1.

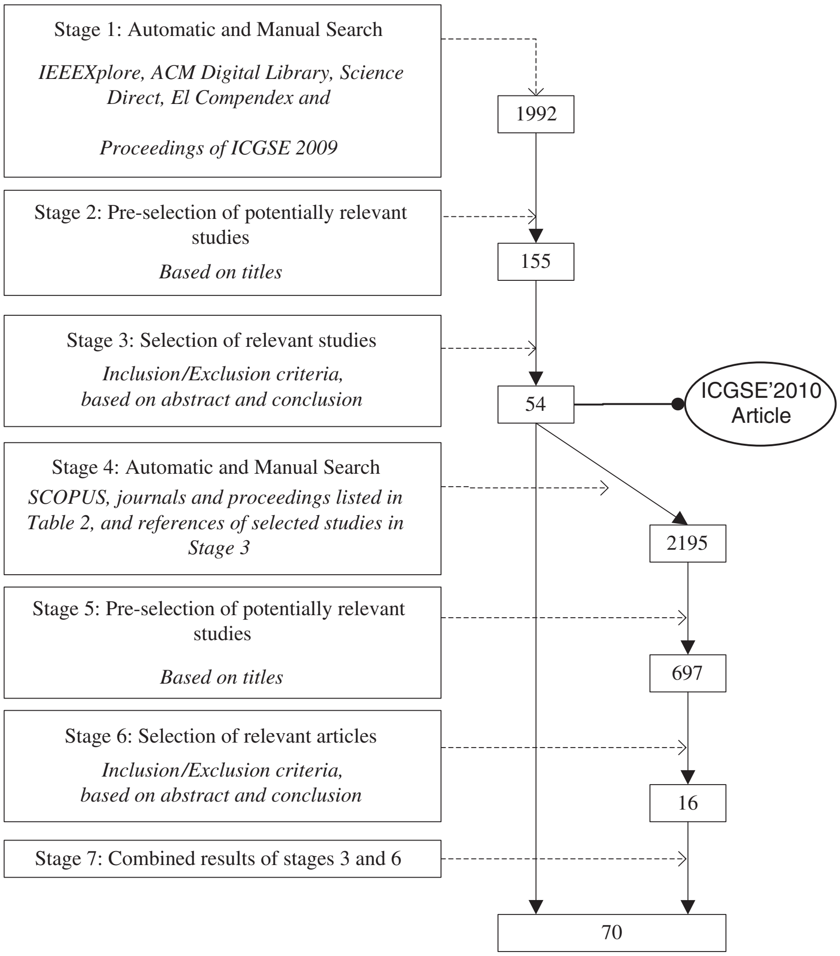


Figure 1. Study selection process.

4.1. Challenges of distributed software development project management

The 29 challenges found in the MS are listed in Table II organized by four categories created by a coding process described in details in da Silva *et al.* [9].

Challenges related to human factors and process and technology clearly dominates the findings from the studies, comprising 22 of the total of 29 challenges found. Achieving effective communication and effective coordination and synchronization of work, from a process perspective, are the most important and closely related challenges in DSD project management. Temporal, geographical, and socio-cultural distances inherent to DSD seem to exacerbate communication and coordination problems [PS13][PS28][PS47][PS63][PS70].

Cultural differences, issues of trust, differences in knowledge levels, and language barriers are cited as causes of poor or ineffective communication [PS02][PS13][PS26][PS52], as well as the lack of synchronous communication [PS21]. Too much information generated by email and frequent conference calls was also identified as source of low effectiveness [PS26].

On the other hand, the studies found that communication barriers have negative impact on cooperation among workers [PS40][PS47][PS53], increase the problems with conflict resolution [PS03][PS14][PS15][PS24][PS26][PS35][PS40][PS45][PS49], and reduce the sense of cohesion and team spirit [PS19][PS21][PS23][PS31][PS54][PS58].

Related to processes and technology; asymmetry in processes, policies, and standards used among sites [PS01][PS25][PS26][PS40][PS57]; differences in the use of technologies [PS02][PS10][PS11][PS46]; difficulties in the identification of roles and responsibilities [PS13][PS26][PS40][PS42]; and

DOI: 10.1002/smr

Table II. Challenges of distributed software development project management.

Human factors	Process and technology	Localization and infrastructure	Norms and rules
C1. Effective communication C2. Cultural differences C3. Trust C8. Cooperation C9. Language barriers C10. Different knowledge levels C17. Conflict resolution C28. Cohesion	C5. Effective coordination C7. Scope and change management C11. Identification of roles and responsibilities C12. Project planning C13. Asymmetry in processes, policies, and standards C14. Tracking and control C15. Task allocation C18. Schedule management C19. Differences in technologies C20. Knowledge management C22. Overall visibility C23. Quality C24. Risk management C26. Synchronization of work among sites	C4. Time zone differences C6. IT infrastructure C16. Physical distance C21. Need of office space	C25. Intellectual property, confidentiality and privacy C27. Different stakeholders C29. Different governments, laws, rules and regulations

DOI: 10.1002/smr

lack of overall visibility of the project [PS09][PS25][PS40][PS41] are identified as having negative impact on most of the project planning [PS13][PS25][PS56] and management areas: quality management [PS07][PS35], risk management [PS35][PS40], tracking and control and task allocation [PS04][PS13], scope and change management [PS13][PS26], schedule management [PS26][PS35], and knowledge management [PS34][PS58]. Altogether, these challenges were found to have strong negative impact on the effectiveness of coordination [PS13][PS28][PS32][PS40][PS42][PS47][PS60] and synchronization of the tasks and work in general [PS01][PS20][PS22].

Although only four challenges were found to be related to the distributed localization of teams and needs of infrastructure, their relevance to DSD project management is very high. Time zone differences impact the scheduling and proper running of meetings among sites [PS01][PS04] [PS25][PS26][PS52][PS53], as noted in one of the primary studies: 'working across a large number of time zones was an enormous issue. . .this makes it very difficult to schedule meetings, as every time is inconvenient for someone' [PS25]. Physical distance has obvious impacts in many aspects of DSD project management [PS01][PS16][PS39] and poses a particular problem for the project manager because, as noted in one of the studies, 'distance is a major barrier to project managers who normally exercise their leadership and influence using their personal traits, for example a strong presence, charisma, and the ability to speak articulately' [PS54]. IT infrastructure and dedicated office space for meetings are also important challenges because of the need for specific infrastructure to support the distributed work in addition the normal needs of each site [PS01][PS13][PS21][PS25][PS40][PS58][PS53].

Differences in norms and rules that apply to each site are also an important concern. The distinct approaches and regulations related to intellectual property management and the problems associated with information confidentiality and privacy because of the extensive need to use electronic communication were identified in several studies [PS05][PS20][PS22][PS35][PS58].

From the aforementioned discussion, we can observe that challenges form a complex cause–effect chain, with direct and indirect effects on DSD project management effectiveness.

4.2. Practices of distributed software development project management

Practices are organized in a similar way as the challenges in the previous section (Table III).

Consistently with the findings on challenges, most practices proposed to increase effectiveness in DSD project management are related to human factors and process and technology. Altogether, 31 practices have been identified, the majority in the process and technology category. Multiple channels to support synchronous communication [PS46][PS50][PS52][PS53] and the creation and deployment of communication protocols [PS26][PS47][PS32][PS46] have been used by the majority of the studies that addressed the challenges related to communication. Besides, the stimulation of informal interactions among distributed teams [PS21][PS23][PS29][PS31], the promotion of visits and exchanges among sites [PS52], and various techniques to improve collaboration, cooperation, and cultural awareness [PS26][PS27][PS52][PS53] are also proposed as practices that can reduce communication barriers in a DSD project.

The clear definition of team roles and responsibility have been proposed as strategies to people management in DSD context [PS35][PS40][PS42][PS43], complementing the practices related to communication. In the same context, two studies emphasize the use of teams with multiple and complementary skills [PS32][PS46].

Most of the practices in the category of process and technology are proposed to directly address identified challenges in a rather obvious way. For instance, to overcome challenge C13—asymmetry in processes, policies, and standards, the studies propose P12—to use and maintain common software process among sites, without deeper considerations about the effectiveness of the suggested practice. In fact, the quality of evidence of practices in process and technology category is lower than that of the practices in the human factors category, because most of the evidence comes from industrial reports instead of from empirical studies. This is a clear gap in research that presents as an opportunity for more empirical studies.

The practice P1—provision of training in collaboration and coordination tools is the most used [PS26][PS32][PS40[PS46][PS47][PS55], which is consistent with the identification of the challenges

DOI: 10.1002/smr

Table III. Practices of distributed software development project management.

Human factors	Process and technology	Localization and infrastructure	Rules and norms
P3. Multiple communication modes including support to face-to-face synchronous communication	P1. Provision of and training in collaboration and coordination tools	P09. To secure office space for local teams	P11. Effective policies for confidentiality, copyright protection, and intellectual property
P4. Training on different cultures/instill a sense of cultural awareness	P2. To apply agile practices (Scrum)	P15. To deploy and use a configuration management system	
P5. Creation of communication protocols	P8. To use a knowledge management systems	P27. Secure IT infrastructure: to ensure infrastructure compatibility among geographic locations	
P6. To stimulate cooperation and collaboration	P12. To use and maintain common software process among sites		
P7. To promote visits and exchanges among sites	P13. Detailed planning		
P10. People management	P14. To divide the work into well-defined modules and carry out progressive integration		
P17. Create teams with complementary skills and cultures	P16. To deploy knowledge transfer mechanisms		
P22.To have clearly defined roles and responsibilities	P18. Constant risk management		
P23. Maintain team involvement and cohesion	P19. To maintain common quality standards among sites		
P30. To promote informal interactions	P20. Synchronicity: to set up meetings at times reasonable for most teams		
P31. Outsourcing manager is part of two companies	P21. Visibility of work progress		
	P24. Schedule management		
	P25. Work synchronization among sites: to define synchronization points among teams		
	P26. Face-to-face kickoff: starting a new project with face-to-face meeting		
	P28. To have clear criteria for task allocation		
	P29. To implement follow up system		
	P32. Dependencies matrix		

to achieve effective collaboration among workers and coordination and synchronization of tasks in a DSD context. Some studies have considered the deployment of agile practices and, in particular, the use of Scrum in the DSD context mainly to motivate distributed communication and collaboration [PS20][PS29][PS53][PS63][PS64][PS66][PS70]. For instance, Scrum daily meetings have been used to increase visibility and coordination among sites [PS20]. We believe that more empirical investigations are needed to validate the use of agile practices in DSD, because one of the agile principles places great emphasis on the importance of face-to-face conversation for effective communication (‘The most efficient and effective method of conveying information to and within a development team is face-to-face conversation’ [2]).

4.3. Models and tools for distributed software development project management

In this section, the tools (Table IV) and models (Table V) reported by da Silva *et al*. [9] are presented. The vast majority of the studies that proposed the use of tools to increase the effectiveness of DSD project management focused on communication tools [PS01][PS07][PS25][PS26][PS52][PS53]. One issue that raises concern about the reliability of these propositions is that the majority of the studies report the use of traditional tools, i.e., tools used in collocated project management, without empirical evidence about their effectiveness in the DSD context. In fact, one of the studies identified effectiveness problems in the use of teleconference and email because of too much information generated [PS26].

Table IV. Tools of distributed software development project management.

Human factors	Process and technology	Specific
T1. Phone (including teleconference and audio-conference)	T4. Wiki	T14. TeamSpace
T2. Email	T8. Virtual whiteboard	T15. TAMRI
T3. Video-conference	T9. Monitoring and management systems	T16. CAMEL
T5. Online chatting	T11. Configuration management systems	T17. NEXTMOVE
T6. Electronic meeting systems (including NetMeeting)	T12. Group intranet	T18. MasePlanner
T7. Photo gallery	T13. Group calendar	
T10. Blog		

Table V. Models of distributed software development project management.

Models
M1. Project management model
M2. A reference model for global software development
M3. Approach to offshore collaboration
M4. Conceptual model for managing an international IS development project
M5. DRiMaP—a model of distributed risk management
M6. Solar system
M7. Dyadic model
M8. Project management framework
M9. Process maturity framework for managing distributed development
M10. Framework for supporting management in distributed information systems development
M11. TAPER
M12: NEXTMOVE
M13. GSD model
M14. Framework to enable coordination in distributed software development

DOI: 10.1002/smr

Five studies propose tools that have been specifically designed for DSD: TeamSpace, which supports geographically distributed teams by managing shared work processes and maintaining shared artifacts in a project; TAMRI, a tool for supporting task distribution in global software development projects; CAMEL, a tool for collaborative distributed software design; NEXTMOVE, a framework for distributed task coordination; the framework simulates the project manager's thought processes involved in prioritizing and allocating tasks; and MasePlanner, a card-based distributed planning tool for agile teams, which supports agile teams as they collaborate in a distributed environment during their planning meetings.

Fourteen models that support some or various aspects of DSD project management were identified. Similarly to the aforementioned five specific tools, all models and frameworks are only referenced in the original study in which it was proposed. This also indicates that the models have not been tried or evaluated by researchers or practitioners other than their authors, or if they have been, the evidence about their effect on DSD project management have not yet been published. This weakens the relevance of the evidence because of the potential bias that the authors of the tools and models may have introduced when reporting their results. An important conclusion is that empirical studies carried out by external research groups are necessary to construct more reliable evidence about these tools and models.

5. DISCUSSION

At this point, we relate the practices, tools, and models with the challenges they are supposed to address. To increase the relevance and reliability of the mappings between challenges and proposed solutions, only practices, tools, and models rated as having high quality of evidence are presented. Besides, to improve readability, only relationships that have been addressed in more than one study are presented. We present the relationships between challenges and practices (Table VI) and the relationships between challenges and tools and models (Table VII).

The 13 challenges shown in Table VI seem to be well covered by a set of 14 practices presented in studies with good quality of evidence. The shades of gray in the cells of the table emphasize that certain combinations of challenges and practices have been proposed more often than others. The challenges C01—effective communication and C02—cultural differences have been addressed mainly using five related practices: P01—provision of and training in collaboration and coordination tools; P4—training on different cultures; P5—creation of communication protocols; P11—effective policies for confidentiality, copyright protection, and intellectual property; and P3—multiple communication modes. Moreover, as mentioned before, the deployment of Scrum has been also used to improve communication in four different studies.

Consistent with the previous mapping, challenges C01 and C02 have been addressed in most studies by using various different tools. Not surprisingly, the most traditional communication tools, T01—phone and T02—email, are used in almost all contexts.

Table VI. Relating challenges and practices.

	P01	P04	P05	P11	P03	P10	P18	P15	P02	P14	P12	P13	P17	P19
C01	8	4	3	2	4	1	1	1	4	2	2	2	2	1
C02	6	4	2	2	2	1	1	1		1	1	2	2	2
C17	2	1	1	2		2	2	2		1	1			1
C05	3	2	3	1		1	1	1		1	1		1	
C24	1		1	2		2	2	2		1	1			1
C08	3	1	2	1	1	2	1	1	1	1	1			
C13	2	1	1	1		1	1	1		2	2	1		
C04	3	1		1	2				1			1		1
C03	2	2	1		1					1	1	1		
C19	2	2			1								1	
C06	2		1	1	1	1	1	1	1	1	1			
C11	2	1	1	1		1	1	1		1	1			
C14	2	1	1	1								1	1	1

DOI: 10.1002/smr

Table VII. Relating challenges and tools and models.

	T02	T01	T03	T06	T07	T04	T05	M01	M07	M08
C01	**9**	**8**	**7**	**3**	2	**3**	**3**	1	1	
C02	**7**	**6**	**5**	1	1	2	2	1		
C04	**4**	**4**	**3**	1	2	2	1	1		
C06	**3**	**3**	2	2	2	1				
C13	**4**	**4**	**3**	1	1					
C03	**4**	**4**	**3**		1		1			
C22	**3**	**3**	2	1	1		1			
C09	**3**	**3**	2			1	1			
C08	2	2	1	2	2	1				
C05	2	2	2	2			1			
C07	**3**	**3**	1		1					
C11	2	2	2	1						
C17	2	2	2	1						
C14	2	2	1	1	1		1	1		
C15	2	2	1				1	1		
C12										1
C16									1	

The challenges related to management of knowledge in DSD, C10—different knowledge levels and C20—knowledge management, have not been addressed in good quality studies by the proposition of tool or model support. Besides, challenges related to established project management themes like C12—project planning, C14—tracking and control, C15—task allocation, and C18—schedule management have also not been addressed with proposition of tool support. These gaps present good opportunity for research and tool development.

5.1. A conceptual model for distributed software development project management effectiveness

The analysis of the evidence related to challenges in DSD project management (Section 4.1) revealed that challenges could be interpreted as variables or factors. Then, a complex cause–effect relationship chain can be built to express the impact of certain challenges on the overall effectiveness of the project management in DSD context. In this section, we propose a conceptual model to explain these cause–effect relationships. This model was developed based on the method presented by Sjøberg *et al.* [19], following four steps. First, we identified the challenges as constructs in the model. In particular, we considered that certain challenges related to process of project management could be interpreted as indicators of project management process effectiveness. Second, we grouped challenges according to their effect on other challenges, as supported by the evidence reported by da Silva *et al.* [9]. In this step, eight categories were identified. Third, from the evidence reported by da Silva *et al.* [9], we built cause–effect relationships between categories of challenges. Finally, we developed propositions or hypothesis based on the evidence and from other theoretical work found in the literature.

The proposed model is presented in Figure 2.

The constructs related to physical and time distances affect certain challenges in a different way than the constructs related to social, political, and personal distances. This is expressed by the following hypothesis:

Hypothesis 1: *Physical and time distance, and social, political, and personal distances have direct effect on the antecedents of information sharing.*

Hypothesis 2: *Physical and time distances have an only indirect effect on the antecedents of teamwork through its effect on trust.*

Hypothesis 3: *Social, political, and personal distances have a direct effect on antecedents of teamwork, and this effect is increased by its indirect effect through trust.*

DOI: 10.1002/smr

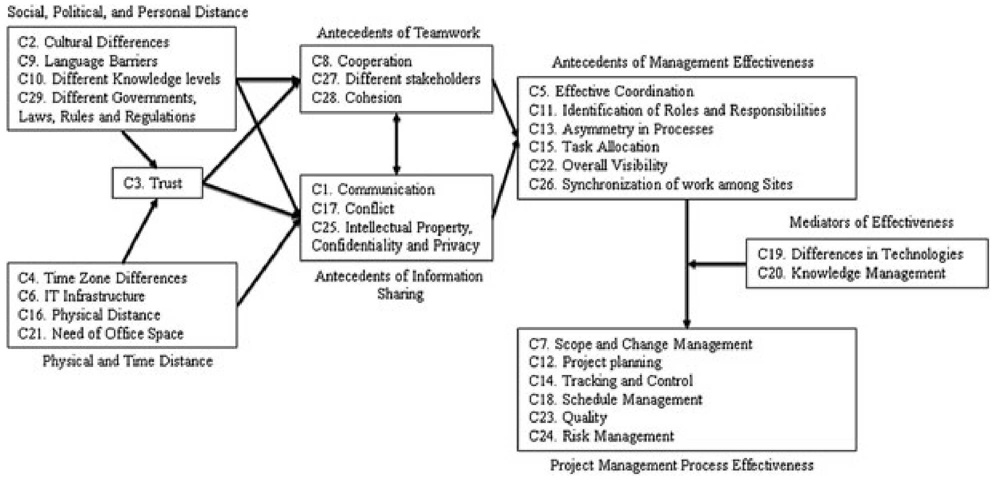


Figure 2. The model for distributed software development project management effectiveness.

In the proposed model, we separate two groups of challenges related to project management processes because the evidence shows that one group of challenges affects challenges in the other group. This is expressed by the following hypothesis:

Hypothesis 4: *The effect of the constructs related to the antecedents of management effectiveness on project management process effectiveness is mediated by the differences in technology and knowledge management.*

Moreover, consistently with theoretical work on team cohesion [10] and teamwork effectiveness [5,7], the challenges that relate to teamwork and information sharing have complex mutual relationships.

This model of DSD project management effectiveness is a conceptual proposition that has not been tested by direct empirical studies. Nevertheless, it is build from the empirical evidence found in the primary studies, which provides some level of confidence in its validity. In addition, the stated hypotheses are only examples of possible relationships between categories of constructs. Future research is necessary to refine and test these hypothesis, and others that can be derived from the model.

6. CONCLUSIONS

The results of this research contribute to a better understanding of the landscape of DSD project management and show important gaps that opens opportunities for future research. In addition, with information presented based on a rigorous process of extracting information from the primary studies using a protocol for conducting systematic mapping studies, this research can better support practitioners and researchers in the identification of relevant challenges and definition of mitigating solutions based on practices, tools, and models that have been proposed, with some of them tested in experimental and industrial settings.

One important conclusion drawn from this review is that human factors are of even greater importance in DSD project management than they already are in collocated project management. Temporal, geographical, and socio-cultural distances exacerbate problems of trust, cohesion, and conflicts with negative impact on communication, cooperation, coordination, and general management issues. Therefore, human factor studies, using approaches like, for instance, ethnography, present a good opportunity for researchers and are likely to have great impact in the practice of DSD.

Finally, the area of DSD project management still requires more research to become a mature field. In particular, more empirical studies using rigorous research protocols must be performed in industrial context to test the effectiveness of various proposed practices, tools, and models. These studies shall

DOI: 10.1002/smr

increase reliability and make explicit the validity and limitations of these propositions. This is relevant for the industrial practice as well as the academic research.

APPENDIX: SELECTED PRIMARY STUDIES

[PS1] A dimensional analysis of geographically distributed project teams: A case study. Journal of Engineering and Technology Management. Evaristo, J., Scudder, R., Desouza, K. & Sato, O. 2004. Science Direct

[PS2] A framework for supporting management in distributed information systems development. International Conference on Research Challenges in Information Science. Ralyte, J., Lamielle, X., Arni-Bloch, N. & Leonard, M. 2008. IEEE

[PS3] A framework to enable offshore outsourcing. International Conference on Global Software Engineering. Braun, A. 2007. IEEE

[PS4] A practical management and engineering approach to offshore collaboration. IEEE Software. Cusick, J. & Prasad, A. 2006. IEEE

[PS5] A risk profile of offshore-outsourced development projects. Communications of the ACM. Iacovou, C.L. & Nakatsu, R. 2008. ACM

[PS6] A software agent framework for the support of software project management. Conference of the South African institute of computer scientists and information technologists on Enablement through technology. Nienaber, R. & Cloete, E. 2003. ACM

[PS7] A survey on the state of the practice in distributed software development: Criteria for task allocation. International Conference on Global Software Engineering. Lamersdorf, A., Munch, J. & Rombach, D. 2009. IEEE/ICGSE 2009

[PS8] A team collaboration space supporting capture and access of virtual meetings. Conference on Supporting Group Work. Geyer, W., Richter, H., Fuchs, L., Frauenhofer, T., Daijavad, S. & Poltrock, S. 2001. ACM

[PS9] A virtual human resource organization model in dual-shore collaborative software development. International Conference on Wireless Communications, Networking and Mobile Computing. Liang, X., Ma, X., Yang, Q., Zhuo, Y., Xu, B. & Ma, A. 2008. IEEE

[PS10] Aligning it components to achieve-agility in globally distributed system development. Communications of the ACM. Lee, O.-K.D., Banerjee, P., Lim, K.H., Kumar, K., Hillegersberg, J. v. & Wei, K.K. 2006. ACM

[PS11] An adaptive tool integration framework to enable coordination in distributed software development. International Conference on Global Software Engineering. Sinha, V.S., Sengupta, B. & Ghosal, S. 2007. IEEE

[PS12] An empirical approach for the assessment of scheduling risk in a large globally distributed industrial software project. International Conference on Global Software Engineering. Avritzer, A. & Lima, A. 2009. IEEE/ICGSE2009

[PS13] An integration centric approach for the coordination of distributed software development projects. Information and Software Technology. Taxén, L. 2006. Science Direct

[PS14] Be successful, take a hostage or “outsourcing the outsourcing Manager”. International Conference on Global Software Engineering. Pichler, H. 2007. IEEE

[PS15] CAMEL: A tool for collaborative distributed software design. International Conference on Global Software Engineering. Cataldo, M., Shelton, C., Choi, Y., Huang, Y.-Y., Ramesh, V., Saini, D. & Wang, L.-Y. 2009. IEEE/ICGSE2009

[PS16] Communication tools for distributed software development teams. Conference on Computer personnel research: The global information technology workforce. Thissen, M.R., Page, J.M., Bharathi, M.C. & Austin, T.L. 2007. ACM

[PS17] Communication, knowledge and co-ordination management in globally distributed software development: Informed by a scientific software engineering case study. International Conference on Global Software Engineering. Taweel, A., Delaney, B., Arvanitis, T.N. & Zhao, L. 2009. IEEE/ICGSE2009

DOI: 10.1002/smr

[PS18] Coordinating distributed software development projects with integrated process modelling and enactment environments. International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises. Grundy, J., Hosking, J. & Mugridge, R. 1998. IEEE
[PS19] Coordinating mechanisms for agile global software development. International Conference on Global Software Engineering. Hossain, E. 2008. IEEE
[PS20] Distributed Scrum: Agile project management with outsourced development teams. Hawaii International Conference on System Sciences. Sutherland, J., Viktorov, A., Blount, J. & Puntikov, N. 2007. IEEE
[PS21] Evolving an infrastructure for student global software development projects: Lessons for industry. India software engineering conference. Gotel, O., Kulkarni, V., Phal, D., Say, M., Scharff, C. & Sunetnanta, T. 2009. ACM
[PS22] Evolving distributed project management. IEEE Software. Nidiffer, K.E. & Dolan, D. 2005. IEEE
[PS23] Experience developing software using a globally distributed workforce. International Conference on Global Software Engineering. Avritzer, A., Ostrand, T. & Weyuker, E.J. 2006. IEEE
[PS24] From experience: Leading dispersed teams. Journal of Product Innovation Management. Smith, P. & Blanck, E. 2002. El Compendex
[PS25] Global software development at Siemens: Experience from nine projects. International Conference on Software Engineering. Herbsleb, J.D., Paulish, D.J. & Bass, M. 2005. IEEE
[PS26] Global virtual teams for value creation and project success: A case study. International Journal of Project Management. Lee-Kelley, L. & Sankey, T. 2008. Science Direct
[PS27] Globally distributed product development using a new project management framework. International Journal of Project Management. Ghosh, P.P. & Varghese, J.C. 2004. Science Direct
[PS28] Globally distributed software development project performance: An empirical analysis. Symposium on The foundations of software engineering. Ramasubbu, N. & Balan, R.K. 2007. ACM
[PS29] How did we adapt agile processes to our distributed development?. Agile Conference. Young, C. & Terashima, H. 2008. IEEE
[PS30] Leveraging global resources: A process maturity framework for managing distributed development. IEEE Software. Ramasubbu, N., Krishnan, M.S. & Kompalli, P. 2005. IEEE
[PS31] Leveraging resources in global software development. IEEE Software. Battin, R.D., Crocker, R., Kreidler, J. & Subramanian, K. IEEE
[PS32] Management at the outsourcing destination—global software development in India. International Conference on Global Software Engineering. Deshpande, S. & Richardson, I. 2009. IEEE/ICGSE2009
[PS33] Managing international usability projects: Cooperative strategy. Conference on Human Factors in Computing Systems. Gorlenko, L. & Krause, S. 2006. ACM
[PS34] Managing knowledge in global software development efforts: Issues and practices. IEEE Software. Desouza, K.C., Awazu, Y. & Baloh, P. 2006. IEEE
[PS35] Managing risks in global software engineering: Principles and practices. International Conference on Global Software Engineering. Ebert, C., Murthy, B.K. & Jha, N.N. 2008. IEEE
[PS36] Managing software development in globally distributed teams. Communications of the ACM. Cusumano, M.A. 2008. ACM
[PS37] NextMove: A framework for distributed task coordination. Australian Software Engineering Conference. Mak, D.K.M. & Kruchten, P.B. 2007. IEEE
[PS38] Practical considerations for distributed agile projects. Agile Conference. Robarts, J.M. 2008.
[PS39] Project management model for a physically distributed software development environment. Hawaii International Conference on System Sciences. Zanoni, R. & Audy, J.L.N. 2003.

[PS40] Project management within virtual software teams. International Conference on Global Software Engineering. Casey, V. & Richardson, I. 2006.
[PS41] Siemens global studio project: Experiences adopting an integrated GSD infrastructure. International Conference on Global Software Engineering. Bass, M., El Houda, Z., Cataldo, M., Herbsleb, J.D. & Bass, L. 2006.
[PS42] Successful global development of a large-scale embedded telecommunications product. International Conference on Global Software Engineering. Leszak, M. & Meier, M. 2007. IEEE
[PS43] Successfully outsourcing embedded software development. IEEE Computer. Rottman, J.W. 2006.
[PS44] TAMRI: A tool for supporting task distribution in global software development projects. International Conference on Global Software Engineering. Lamersdorf, A. & Munch, J. 2009. IEEE/ICGSE2009
[PS45] TAPER: A generic framework for establishing an offshore development center. International conference on Global Software Engineering. Hofner, G. & Mani, V.S. 2007. El Compendex
[PS46] The global virtual manager: A prescription for success. European Management Journal. Kayworth, T. & Leidner, D. 2000. Science Direct
[PS47] The impact of fear on the operation of virtual teams. International Conference on Global Software Engineering. Casey, V. & Richardson, I. 2008 IEEE
[PS48] The impact of intercultural factors on global software development. Canadian Conference on Electrical and Computer Engineering. MacGregor, E., Hsieh, Y. & Kruchten, P. 2005. IEEE
[PS49] The impact of outsourcing on client project managers. IEEE Computer. Lacity, M.C. & Rottman, J.W. 2008. IEEE
[PS50] Using a real-time conferencing tool in distributed collaboration: An experience report from Siemens IT solutions and services. International Conference on Global Software Engineering. Damian, D., Marczak, S., Dascalu, M., Heiss, M. & Liche, A. 2009. IEEE/ICGSE2009
[PS51] Using methods and IT tools innovatively for the management of international IS development projects. Portland International Conference on Management of Engineering & Technology. Katainen, T. & Nahar, N. 2008. IEEE
[PS52] Using Scrum in distributed agile development: A multiple case study. International Conference on Global Software Engineering. Paasivaara, M., Durasiewicz, S. & Lassenius, C. 2009. IEEE/ICGSE2009
[PS53] Using Scrum in global software development: A systematic literature review. International Conference on Global Software Engineering. Hossain, E., Babar, M.A. & young Paik, H. 2009. IEEE/ICGSE2009
[PS54] Virtual teams: An exploratory study of key challenges and strategies. International Conference on Information Systems. Paré, G. & Dubé, L. 1999. ACM
[PS55] A case study of a successful collaboration in distributed software development. International Workshop on Distributed Software Development. Taweel, A., 2005.
[PS56] A reference model for global software development: Findings from a case study. International Conference on Global Software Engineering. Prikladnicki, R., Audy, J. L. N., Evaristo, R. 2006.
[PS57] An empirical study on global software development: Offshore insourcing of IT projects. Workshop on Global Software Development at ICSE. Prikladnicki, R., Audy, J. L. N., Evaristo, R., 2004.
[PS58] Challenges of virtual project management in developing countries. Proceedings. 2005 IEEE International Engineering Management Conference. Nauman S., Iqbal S., 2005.
[PS59] Collaboration in global software projects at Siemens: An experience report. International Workshop on Distributed Software Development. Bass, M., Herbsleb, J. D., Lescher, C., 2006.
[PS60] Continuous coordination: A new paradigm to support globally distributed software development projects. Wirtschaftsinformatik, V. 49, 2007, pp. 28–38. Redmiles, D., van der

Hoek, A., Al-Ani, B., Hildenbrand, T., Quirk, S., Sarma, A., Silva Filho, R. S., de Souza, C., Trainer, E. 2007.

[PS61] Continuous improvement through iterative development in a multi-geography. International Conference on Global Software Engineering. Laredo, A., Ranjan, R., 2008.

[PS62] Developing an inter-site coordination index for global software development. International Conference on Global Software Development. Sooraj, P., Mohapatra, P.K.J. 2008.

[PS63] Distributed agile development: Using Scrum in a large project. Software Process Improvement and Practice, Vol. 13, Issue 6, pp. 527–544. M. Paasivaara, S. Durasiewicz, C. Lassenius, 2008.

[PS64] Distributed Scrum for large-scale mission-critical projects. In Proceedings of the Conference on AMCIS 2007, paper 235. J. Cho, 2007.

[PS65] DRiMaP—a model of distributed risk management process. NCM 2009—5th International Joint Conference on INC. Kajko-Mattsson, M., Sjökvist, K., Söderström, J., Krogdahl, D., 2009.

[PS66] Implementing Scrum in a distributed software development organization. In proceedings of the Conference on AGILE 2007, pp.371-375. H. Smits, 2007.

[PS67] Improving global software development project performance using simulation. Portland International Conference on Management of Engineering and Technology. Setamanit, S.-O., Wakeland, W., Raffo, D., 2007.

[PS68] MasePlanner: A card-based distributed planning tool for agile teams. International Conference on Global Software Engineering. Morgan, R., Maurer, F., 2006.

[PS69] The virtual project: Managing tomorrow's team today. PM Network 11 (1), 37–42. Adams, J.R., Adams, L.L., 1997.

[PS70] Usage of Scrum practices within a global company. In Proceedings of ICGSE 2008, pp. 22–226. M. Cristal, D. Wildt, R. Prikladnicki, 2008.

ACKNOWLEDGMENTS

Professor Fabio Q. B. da Silva holds a research grant from the Brazilian National Research Council (CNPq), number 314523/2009-0. Professor Rafael Prikladnicki (from the MuNDDoS Research Group on Distributed Software Development) holds research grants from CNPq (483125/2010-5, and 560037/2010-4), Ci&T, and the PDTI program, financed by Dell Computers of Brazil Ltd. (Law 8.248/91). Cleviton V. F. Monteiro and César França are doctoral students at the Federal University of Pernambuco where they receive scholarships from CAPES and CNPq. Catarina Costa received scholarship from FACEPE during her studies at UFPE. The authors would like to thank the Samsung Institute for Development of Informatics (Samsung/SIDI) for partially supporting the work developed in this research. Finally, the authors would like to thank the reviewers of the manuscript for important comments that helped to improve the final version of the article.

REFERENCES

1. Arksey H, O'Malley L. Scoping studies: Towards a methodological framework. *International Journal of Social Research Methodology* 2005; **8**(1): 19–32.
2. Beck K, *et al*. Manifesto for agile software development (2001). (Available from: http://agilemanifesto.org) (last visited April 2011).
3. Betz S, Mäkiö J. Amplification of the COCOMO II regarding offshore software projects. In Proceedings of the Workshop on Offshoring of Software Development—Methods and Tools for Risk Management—at Second International Conference on Global Software Engineering, München. August 2007.
4. Binder JC. *Global Project Management: Communication, Collaboration and Management Across Borders*. Gower Publishing, Hampshire, England, 2007.
5. Campion MA, *et al*. Relations between work group characteristics and effectiveness: Implications for designing effective work groups. *Personnel Psychology* 1993; **46**(4): 823–847. DOI: 10.1111/j.1744-6570.1993.tb01571.x
6. Carmel E. *Global Software Teams—Collaborating Across Borders and Time-Zones*. Prentice Hall, NJ, 1999.
7. Cohen SG, Bailey DE. What makes team work: Group effectiveness research from the shop floor to the executive suite. *Journal of Management* 1997; **23**: 239–290.
8. da Silva FQB, Costa C, França ACC, Prikladnicki R. Challenges and solutions in distributed software development project management: A systematic literature review. *Proc. Fifth IEEE International Conference on Global Software Engineering, ICGSE* 2010.

DOI: 10.1002/smr

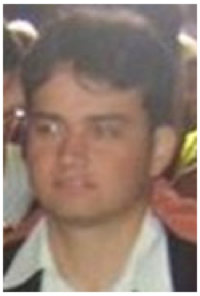
9. da Silva FQB, Prikladnicki R, França ACC, Monteiro CVF, Costa C, Rocha R. Research and practice of distributed software development project management: A systematic mapping study. Submitted to the Journal of Information and System Technology, 2011.
10. Jones MC, Harrison AW. IS project team performance: An empirical assessment. *Information Management* 1996; **31**: 57–65.
11. Jiménez M, *et al*. *Challenges and Improvements in Distributed Software Development: A Systematic Review*. Advances in Software Engineering, **2009**, 710971. DOI: 10.1155/2009/7109712009.
12. Kitchenham B, *et al*. Evidence-based software engineering. 26th International Conference on Software Engineering, (ICSE), IEEE, Washington DC, 2004; 273–281.
13. Kitchenham B, Charters S. Guidelines for performing systematic literature reviews in software engineering, Technical Report EBSE-2007-01, School of Computer Science and Mathematics, Keele University, 2007.
14. Komi-Sirvo S, Tihinen M. Lessons learned by participants of distributed software development. *Journal Knowledge and Process Management* 2005; **122**: 108–122.
15. Krishna S, *et al*. Managing cross-cultural issues in global software outsourcing. *Communications of the ACM*, 2004; **47**(24): 62–66.
16. Macgregor E, *et al*. The impact of intercultural factors on global software development. Proc. Canadian Conference on Electrical and Computer Engineering, 2005; 920–926.
17. Pichler H. Be successful, take a hostage or "outsourcing the outsourcing manager". Proc. Second IEEE International Conference on Global Software Engineering, ICGSE, 2007; 156–161.
18. Petticrew M, Roberts H. *Systematic Reviews in the Social Sciences*. Blackwell Publishing, Oxford, England 2006.
19. Sjøberg DIK, Dybå T, Anda BCD, Hannay J Building theories in software engineering. *Guide to Advanced Empirical Software Engineering, Chapter 12*. Shull F, *et al*. (ed.), Springer-Verlag: London, 2008.
20. Herbsleb JD, Moitra D. Guest editors'introduction. *IEEE Software* 2001; **18**(2): 16–20.
21. Carmel E, Tjia P. *Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce*. Cambridge University Press: Cambridge, 2005.

AUTHORS' BIOGRAPHIES

Fabio Q. B. da Silva has a degree of PhD in Computer Science from the Laboratory for Foundations of Computer Science, University of Edinburgh, Scotland (1992). He is one of the founders of the Recife Center for Advances Studies and Systems (CESAR) and of the Porto Digital Science Park, where he was the first President between 2000 and 2003. Currently, he is an Associate Professor in the Center of Informatics of the Federal University of Pernambuco, in Brazil, where he leads the Human Aspects in Software Engineering research group (H.A.S.E.; www.haseresearch.com). His research involves the study of human and social aspects of software engineering, management of innovation, and entrepreneurship and new enterprise creation.

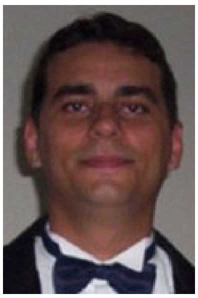
Rafael Prikladnicki is a Professor in the Computer Science School at the Pontifícia Universidade do Rio Grande do Sul (PUCRS), Brazil. Professor Prikladnicki's areas of expertise are distributed software development and agile methodologies. His 2007 book "Distributed Software Development—developing software with distributed teams" was the first Portuguese book on this topic. He also leads one of the main research groups in this area in Brazil (MuNDDoS research group) and is especially interested in how global software engineering (GSE) interplays with agile methodologies, how GSE impacts organizational decisions, and the role of Brazil in the global IT industry.

A. César C. França is a doctoral student at the Center of Informatics of the Federal University of Pernambuco where he receives a scholarship from the Brazilian National Research Council (CNPq), process #141156/2010-4. He is also a lecturer on Software Engineering-related areas at the Faculty of Philosophy, Science, and Letters of Caruaru (Brazil).



Cleviton V. F. Monteiro is a doctoral student at the Center of Informatics of the Federal University of Pernambuco where he receives a scholarship from Capes Agency. He is also the Director of FAST Applications and Solutions, a Brazilian software manufacturer.

Catarina Costa has an MSc degree in Computer Science at the Federal University of Pernambuco, Brazil (2010). She is currently a professor at the Federal University of Acre, working mainly in the following interest areas: Software Engineering, Project Management, and Distributed Software Development.



Rodrigo Rocha has an MSc degree in Computer Science at the Federal University of Pernambuco, Brazil (2010). He is currently a professor at the Federal Rural University of Pernambuco, working mainly with Software Engineering.

DOI: 10.1002/smr