

Global Software Development and Quality Management: A Systematic Review

Deepti Mishra¹, Alok Mishra², Ricardo Colomo-Palacios³,
and Cristina Casado-Lumbreras⁴

¹ Department of Computer Engineering, Atilim University, Ankara, Turkey
deepti@atilim.edu.tr

² Department of Software Engineering, Atilim University, Ankara, Turkey
alok@atilim.edu.tr

³ Computer Science Department, Universidad Carlos III de Madrid, Madrid, Spain
ricardo.colomo@uc3m.es

⁴ Universidad Internacional de La Rioja, Madrid, Spain
cristina.casado@unir.net

Abstract. This paper presents a systematic literature review of global software development (GSD) and quality management aspects. The main focus is to highlight the current research and practice direction in these areas. The results have been limited to peer-reviewed conference papers and journal articles, published between 2000 and 2011. The analysis reports that major studies have been performed in quality and process management, while verification and validation issues of GSD can only get limited attention among researchers. This indicates the need for future research (quantitative and qualitative) in these areas.

Keywords: Global software development, Quality, Process, Verification, Validation.

1 Introduction

Globalization has produced a new way to develop software: Global Software Development (GSD). GSD is a particular kind of Distributed Software Development (DSD) in which the teams are distributed beyond the limits of a nation [1]. Thus, software development is evolving from a single site development to multiple localization team environments [2] and projects are being contracted out in whole or in part [3]. GSD provides several outstanding benefits. In these terms of benefits, the accounts about cheaper work and “follow the sun” approaches are fading, while factors like proximity to the markets, access to specific expertise, productive friction and innovation capability tend to take the lead in driving the trend toward global software development [4]. In spite of its newness, GSD has been analysed in depth in scientific literature [5, 6, 7, 8].

The importance of GSD management has led to a huge effort in the art and science of organizing and managing globally distributed software development, but there is

still a significant understanding to be achieved, methods and techniques to be developed, and practices to be evolved before it becomes a mature discipline [9]. Given that quality management is an important competitive advantage in organizations with geographically distributed software development centres [10], this paper aims at finding out what the main efforts and issues discovered are in the literature on the interaction between software quality management and GSD.

The paper is organized as follows. The next section presents research methodology. Section 3 reports the current status of research in software quality, software process, verification and validation related with GSD. Finally, section 4 concludes with results, discussions and limitations of the study.

2 Research Methodology

2.1 Motivation and Objectives

GSD has recently become an active research area and there is still a lack of quantitative studies in GSD. The effect of using best practices, models, and tools in DSD projects is still scarce in the literature [11]. Smite et al. [12] concluded in her systematic review that the amount of empirical studies in GSD/DSD areas are small, hence the field is still immature. So, this reflects that research in this theme is still in its early stages and requires maturation. On The other hand, quality is a significant component in software engineering and as a result of this for GSD. One of the challenges for GSD is quality and its management [13]. Although quality usually is not directly affected by its geographical location [14], some papers describe the indirect effects of distributed collaboration on quality [15, 16]. It was reported that regular quality problems exist in the products developed offshore [17] and “follow the sun” model is essentially a quick-and-dirty strategy that converts a schedule problem into a quality disaster [18]. Therefore, the aim of this paper is to perform a systematic review that includes software quality and GSD. This study will provide a comprehensive examination on the current status of research of quality, process, verification/ validation in GSD.

2.2 Research Method and Conduct

The research was initially designed to be a systematic literature review following the guidelines provided by Kitchenham and Charters [19]. This section presents all the steps taken in designing and performing the systematic review according to these authors. Regarding the need for conducting a systematic literature review in the area the following research questions are formed:

RQ1: What is the current status of quality practices (quality, process, verification & validation) in GSD?

RQ2: What are the software quality areas in which there is a gap regarding their application in GSD?

2.3 Data Retrieval and Data Sources/Resource Searched

Search strings were formulated by combining different quality practices and different types of distribution. It can be summarized as: (X1 OR X2.....OR Xn) AND (Y1 OR Y2.....OR Yn), where X covers quality practices (quality, process, testing, inspection, review, verification, validation ...) and Y includes different alternatives of GSD and DSD as following:

X: {quality, process, testing, inspection, review, verification, validation}

Y: {global software development, distributed software development, global software engineering, GSD}

Furthermore, some limitations were applied on the searches. 1) The search was only performed in the following databases: Science@Direct, IEEE Explore, ACM Digital Library, SpringerLink. 2) The search items were journal articles, workshop papers and conference papers. In this study only peer-reviewed publications were taken into consideration and grey literature (like Google Scholar) has not been explored. 3) The publication period was set to be between 2000 and 2011. 4) The written language was set to be English. 5) Search was applied to full text to avoid exclusion of the papers that do not include our keywords in titles or abstracts, but which are still relevant to the review.

2.4 Results of Literature Review

The outcome of a search generally resulted in a rather high proportion of papers, some of which were later considered 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. The criteria for including a specific paper in this systematic review was that the paper should have sufficient focus on GSD as well as one of the quality practices (quality, process, testing, inspection, review, verification, validation,.....). More formally, the authors read through all abstracts with the following exclusion criterion:

- Exclude if the focus of the paper is clearly not on GSD.
- Exclude if the focus of the paper is clearly not on software quality.
- Exclude if the method, tool or theory described is not tested.

This process reduced the number of articles to 144. Table 1 presents the results of this process with respect to its source. Table 2 shows major studies performed in various areas of GSD quality practices during the last decade according to the review performed.

Table 1. Results of Literature Review

Source →	Science Direct	IEEE	ACM	Springer	TOTAL
Issue ↓					
Quality	1	15	10	31	57
Process	3	19	7	19	48
Verification & Validation	0	18	5	16	39
Total					144

Table 2. Major Relevant papers in various areas of GSD Quality Practices

Quality Practices	Major Relevant Studies
Quality	[20], [21], [22], [23], [24], [25], [26], [27], [28], [29]
Process	[30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40]
Verification & Validation	[41], [42], [43], [44], [45], [46], [47]

3 **Quality Management in GSD**

This section reviews the main studies related with GSD and quality management.

3.1 **Quality**

Agarwal et al. [24] proposed a new model that modifies the core quality assurance facilitators’ structure by entrusting the quality facilitation activity within business groups. According to this model, each business group has a quality manager supported by a group of Virtual Quality Assurance Facilitators. Annous, Livadas, and Miles [25] presented OffshoreQA, a framework that can be deployed and used in an offshore software development outsourcing organization aiming to implement an ISO 9001:2008 compliant quality management system. Caprihan [26] proposed an experience-based methodology on how to manage the performance of an application that is developed under this radically new development paradigm. Kuni and Bhushan [27] described Wipro Offshore Outsourcing Methodology (WOOM) that focus to include quality metrics in the outsourcing process and provided guidelines to practitioners and decision makers to estimate the cost of IT Application Offshore Outsourcing. Laredo and Ranjan [28] identified challenges faced by global teams working in a very dynamic environment. According to them, challenges for such engagements are mainly two. The first challenge is to understand the team dynamics and project variables and the second one is to adjust the performance in an iterative feedback mechanism to evolve efficiency over time. They further suggested that more structure and innovation can be introduced in the process if a new suite of tools is designed specifically for GSD arrives in the industry. Cusick and Prasad [29] presented a model for offshore development and insights into their management and engineering techniques, which can be replicated in other environments. The proposed model provides a structural framework and the guidelines necessary to maintain the quality of offshore engagements. More specifically, they recommend specifying coding standards in detail and enforcing them.

3.2 **Process**

The software development process is considered one of the most important success factors for distributed projects [48]. Unclear requirements and new technologies make the waterfall model unsuitable for offshore developing strategic systems [30]. Although the use of a spiral model is uncommon in the development of business

information systems, the strategic importance of a system warrants detailed planning, experimentation, verification, validation and risk management provided by the spiral model [30]. As a result of this, the literature has produced a handful of remarkable contributions to GSD process.

Ramasubbu et al. [32] proposed a process-maturity model that features 24 new key process areas (KPA) mapped into four theoretical concepts for distributed work: mutual knowledge, technology readiness, collaboration readiness, and coupling in work. These KPAs address the wide-ranging capabilities needed for managing such development and arrange them in an evolutionary order similar to the CMM framework. Cusumano et al. [33] investigated offshore software development firms and observed that key CMM process areas can be used to create a platform for learning, thus making offshore development process improvement more effective.

Recently, Prikladnicki and Audy [40] reported systematic review of process models in the practice of distributed software development. Jalali, and Wohlin [34] presented a systematic review of agile practices in global software engineering (GSE) while Dullemond et al. [35] discussed advantages and challenges of combining GSE with agile development. Nisar and Hameed [38] and Xiaohu et al. [31] reported their experiences in using XP in offshore teams collaborating with onshore customers. Both papers discuss projects where the development work is done in offshore teams, whereas the onshore customer is tightly involved in project communication. They concluded that the reported projects have been very successful, and that the XP principles they have followed have proven to work. Karlsson et al. [39] found the XP practices useful but hard to implement in distributed projects. There is still scope towards defining the process framework and maturity level standards like CMMI, SPICE etc. for distributed software development towards quality.

3.3 Verification and Validation

Sangwan and LaPlante [41] reported that geographically distributed development teams in large projects can realize Test Driven Development's (TDD) considerable advantages. With good communication and judicious use of automated testing, they can overcome many problems. The transition from unit to system level testing is challenging for TDD, as in general TDD is not intended to address system and integration testing – certainly not for globally distributed development teams at any rate. Still, developers can realize the advantages of TDD through increased informal and formal communication, facilitated by appropriate change management and notification tools [41]. Recently, SoftFab tool infrastructure which enables projects to automate the building and test process and which manages all the tasks remotely by a control center was given by Spanjers et al. [43]. Tervonen and Mustonen [46] considered challenges of test automation in a company which has been involved in offshoring before. The challenges were studied with three test automation offshoring cases. Successful offshore subcontractors are willing to provide better know-how and quality all the time to keep their current customers, as competition is tightening. Salger, Engels and Hofmann [47] presented a systematic yet flexible assessment framework which addresses four challenges: Appropriateness of a software

requirements specification (SRS), viability of software architectures and SRS, wholeness of work packages, and compliance of results with predefined quality objectives. Mishra and Mishra [45] presented a software inspection process in the distributed software development towards quality assurance and management. Salger et al. [47] found that the issue of assessing the correctness and completeness of SRS (Software Requirements Specification) is compounded in GSD inspections due to impeded communication. Heinonen and Tanner [44] introduced a potential solution for selecting and utilizing the proper validation practices in distributed environments from the requirement engineers' point of view.

4 Results of the Study and Discussion

4.1 Summary

Out of total 144 studies, according to figure 1, a majority of studies are limited to quality (40%) and process (33%). Verification and validation (27%) in GSD is the least explored area among the three significant issues of software quality management.

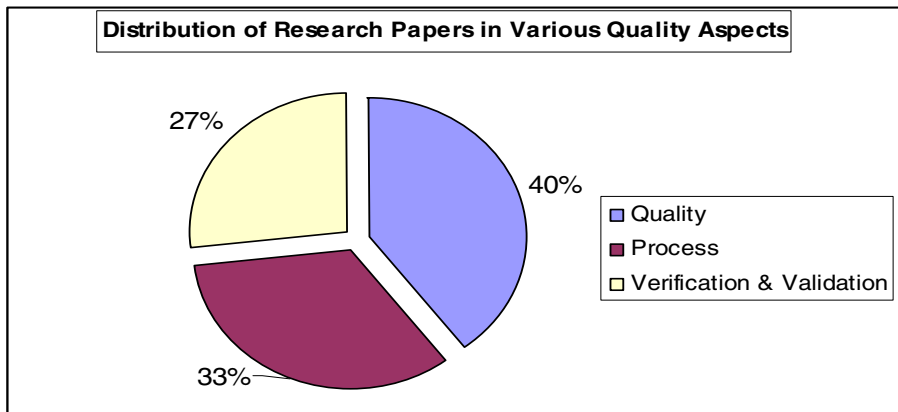


Fig. 1. Distribution of research papers in various quality issues

After an extensive systematic literature survey, it has been observed (According to figure 2) that global software development research started in early 2000 and most of the work related with quality, process and testing issues in GSD was published in 2006. Later on, the trend is downward in all these three with few papers published in these areas. It is also interesting to note that a good number of works got published related with process and quality while studies on various verification and validation issues associated with GSD are still limited. As globally distributed teams become prevalent, top management needs a framework to assess its performance and to initiate activities for continuous improvement in the management of such teams [32].

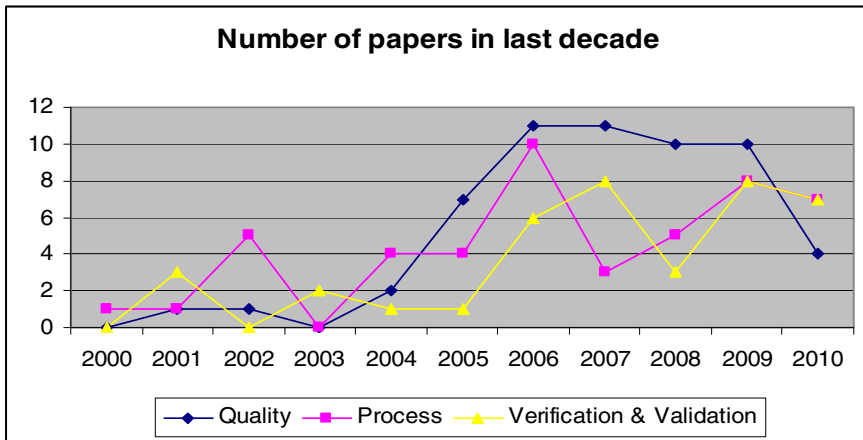


Fig. 2. Number of papers in last decade in various quality issues

4.2 Limitations of the Study

A major concern with any type of research is reliability. Therefore, four researchers were involved in this systematic review study, discussing the reliability threats early in the design phase. Moreover, the review procedure was discussed and agreed upon considering the activities to mitigate the effect of one researcher's bias.

In order to address the conclusion validity, we collected as many papers as possible from a variety of sources including, ACM, IEEE, Science Direct, SpringerLink. We included as many alternatives as possible for the keywords when formulating the search strings. Furthermore, the period was set to be from 2000 to 2011, which was wide enough to capture most of the relevant publications due to the fact that global/distributed software development is not older than one decade. It was possible to observe the trends in the area over the past decade.

Some papers may have been missed due to application of constraints on the search strings in order to reduce the number of irrelevant papers found in the searches. Further, there are some papers in the list which are related with teaching issues of GSD/DSD in an academic environment with teams of students and may not represent the real software industry environment and IT professionals working in this context.

Further due to time and budget constraints, the search did not consider some databases that are suggested by [19]: Wiley InterScience, InspectDirect, etc. This may represent a limitation and a threat to validity. However, main journals and conference outlets related to the area have been included in this study.

5 Conclusions

In this paper, we have reviewed significant quality issues in GSD like process, verification and validation and quality assurance. However, studies selected for this systematic review are from significantly different contexts and backgrounds. Therefore, results presented in these studies may not be applicable as a general

standard in all settings. Consequently, there is a need to build the body of knowledge on how to manage the quality in GSD projects which will classify experiences and practices in order to understand circumstances and contexts that will help to achieve positive results.

As a future work, we would like to extend this review on various other attributes, dimensions and comparisons. Further, we would like to include that area which could get only scant attention in GSD, for instance verification and validation. Various case/ industrial experience report may be included to enrich these issues related with GSD. As future work other databases can also be included and review rules can be modified towards inclusion of further studies. It would be interesting to interview (using online tools) the authors of significant studies to know their views about these assessments.

References

1. Herbsleb, J.D., Moitra, D.: Global software development. *IEEE Software* 18(2), 16–20 (2001)
2. Hernández-López, A., Colomo-Palacios, R., García-Crespo, A., Soto-Acosta, P.: Trust Building Process for Global Software Development Teams A review from the Literature. *International Journal of Knowledge Society Research* 1(1), 66–83 (2010)
3. Madachy, R.J.: Cost Modeling of Distributed Team Processes for Global Development and Software-Intensive Systems of Systems. *Software Process: Improvement and Practice* 13(1), 51–61 (2008)
4. Avram, G.: Of Deadlocks and Peopleware - Collaborative Work Practices in Global Software Development. In: *Proceedings of the International Conference on Global Software Engineering*, August 27–30, pp. 91–102. ICGSE (2007)
5. Yu, L., Mishra, A.: Risk Analysis of Global Software Development and Proposed Solutions. *Automatika* 51(1), 89–98 (2010)
6. Mishra, D., Mishra, A.: A Global Software Inspection Process for Distributed Software Development. *Journal of Universal Computer Science* 18(19), 2731–2746 (2012)
7. García-Crespo, A., Colomo-Palacios, R., Soto-Acosta, P., Ruano-Mayoral, M.: A Qualitative Study of Hard Decision Making in Managing Global Software Development Teams. *Information Systems Management* 27(3), 247–252 (2010)
8. Colomo-Palacios, R., Soto-Acosta, P., García-Peñalvo, F.J., García-Crespo, A.: A study of the impact of global software development in packaged software release planning. *Journal of Universal Computer Science* 18(19), 2646–2668 (2012)
9. Damian, D., Moitra, D.: Global software development: How far have we come? *IEEE Software* 23(5), 17–19 (2006)
10. Sa, L., Marczak, S., Antunes, D., Audy, J.L.N.: Quality Management as a Competitive Strategy in a Distributed Software Development Environment. In: *Proceedings of the Americas Conference on Information Systems Paper* 208 (2003)
11. da Silva, F.Q.B., Costa, C., Franca, A., Cesar, C., Prikladinicki, R.: Challenges and Solutions in Distributed Software Development Project Management: A Systematic Literature Review. In: *5th IEEE International Conference on Global Software Engineering* 2010, pp. 87–96 (2010)
12. Smite, D., Wohlin, C., Gorschek, T., Feldt, R.: Empirical Evidence in Global Software Engineering: A Systematic Review. *Journal of Empirical Software Engineering* 15(1), 91–118 (2010)
13. Jiménez, M., Piatini, M., Vizcaíno, A.: Challenges and Improvements in Distributed Software Development: A Systematic Review. *Advances in Software Engineering*, Article ID 710971 (2009)

14. Lamersdorf, A., Münch, J.: Studying the Impact of Global Software Development Characteristics on Project Goals: A Causal Model. *The Open Software Engineering Journal* 4, 2–13 (2010)
15. Sakthivel, S.: Managing Risks in Offshore Systems Development. *Commun. ACM* 50(4), 69–75 (2007)
16. DeLone, W., Espinosa, J.A., Lee, G., Carmel, E.: Bridging global boundaries for IS project success. In: *Proceedings of the 38th Hawaii International Conference on System Sciences* (2005)
17. Jaakkola, H.: Towards a Globalized Software Industry. *Acta Polytechnica Hungarica* 6(5), 69–84 (2009)
18. Seshagiri, G.: GSD: not a business necessity but a of folly. *IEEE Software* 23(5), 63–64 (2006)
19. 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)
20. Sangwan, R., Neill, C., Bass, M., El Houda, Z.: Integrating a software architecture-centric method into object-oriented analysis and design. *Journal of Systems and Software* 81(5), 727–746 (2008)
21. Cataldo, M., Nambiar, S.: On the relationship between process maturity and geographic distribution: an empirical analysis of their impact on software quality. In: *Proceedings of the the 7th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on The Foundations of Software Engineering (ESEC/FSE 2009)*, pp. 101–110. ACM, New York (2009)
22. Fukui, S.: Introduction of the Software Configuration Management Team and Defect Tracking System for Global Distributed Development. In: Kontio, J., Conradi, R. (eds.) *ECSQ 2002. LNCS*, vol. 2349, pp. 217–225. Springer, Heidelberg (2002)
23. Münch, J., Pfahl, D., Rus, I.: Virtual Software Engineering Laboratories in Support of Trade-off Analyses. *Software Quality Control* 13(4), 407–428 (2005)
24. Agarwal, R., Nayak, P., Malarvizhi, M., Suresh, P., Modi, N.: Virtual Quality Assurance Facilitation Model. In: *Proceedings of the International Conference on Global Software Engineering (ICGSE 2007)*, pp. 51–59. IEEE Computer Society, Washington, DC (2007)
25. Annous, H., Livadas, L., Miles, G.: OffshoreQA: A Framework for Helping Software Development Outsourcing Companies Comply with ISO 9001:2008. In: *5th IEEE International Conference on Global Software Engineering*, pp. 313–315 (2010)
26. Caprihan, G.: Managing Software Performance in the Globally Distributed Software Development Paradigm. In: *Proceedings of the IEEE International Conference on Global Software Engineering (ICGSE 2006)*, pp. 83–91 (2006)
27. Kuni, R., Bhushan, N.: IT Application Assessment Model for Global Software Development. In: *Proceedings of the IEEE International Conference on Global Software Engineering (ICGSE 2006)*, pp. 92–100. IEEE Computer Society, Washington, DC (2006)
28. Laredo, J.A., Ranjan, R.: Continuous Improvement through Iterative Development in a Multi-Geography. In: *Proceedings of the 2008 IEEE International Conference on Global Software Engineering (ICGSE 2008)*, pp. 232–236 (2008)
29. Cusick, J., Prasad, A.: A Practical Management and Engineering Approach to Offshore Collaboration. *IEEE Software* 23(5), 20–29 (2006)
30. Sakthivel, S.: Virtual workgroups in offshore systems development. *Information and Software Technology* 47(5), 305–318 (2005)
31. Xiaohu, Y., Bin, X., Zhijun, H., Maddineni, S.: Extreme programming in global software development. In: *Canadian Conference on Electrical and Computer Engineering 2004*, vol. 4, pp. 1845–1848 (2004)

32. Ramasubbu, N., Krishnan, M.S., Kompalli, P.: Leveraging Global Resources: A Process Maturity Framework for Managing Distributed Development. *IEEE Software* 22(3), 80–86 (2005)
33. Cusumano, M.A., MacCormack, A., Kemerer, C.F., Crandall, W.: Critical Decisions in Software Development: Updating the State of the Practice. *IEEE Software* 26(5), 84–87 (2009)
34. Jalali, S., Wohlin, C.: Agile Practices in Global Software Engineering-A Systematic Map. In: 5th IEEE International Conference on Global Software Engineering, pp. 45–54 (2010)
35. Dullemond, K., Gameren, B., Van Solingen, R.: How Technological Support Can Enable Advantages of Agile Software Development in a GSE Setting. In: ICGSE Fourth IEEE International Conference on Global Software Engineering, pp. 143–152 (2009)
36. Bendjenna, H., Zarour, N., Charrel, P.-J.: Enhancing elicitation technique selection process in a cooperative distributed environment. In: Rolland, C. (ed.) REFSQ 2008. LNCS, vol. 5025, pp. 23–36. Springer, Heidelberg (2008)
37. Oliveira, S.B., Valle, R., Mahler, C.F.: A comparative analysis of CMMI software project management by Brazilian Indian and Chinese companies. *Software Quality Control* 18(2), 177–194 (2010)
38. Nisar, M.F., Hameed, T.: Agile methods handling offshore software development issues. In: Proceedings of INMIC 2004 8th International Multitopic Conference, pp. 417–422 (2004)
39. Karlsson, E., Andersson, L., Leion, P.: Daily build and feature development in large distributed projects. In: Proceedings of the 22nd International Conference on Software Engineering, ICSE 2000, pp. 649–658 (2000)
40. Prikladnicki, R., Audy, J.L.N.: Process models in the practice of distributed software development: A systematic review of the literature. *Information and Software Technology* 52(8), 779–791 (2010)
41. Sangwan, R.S., LaPlante, P.A.: Test-Driven Development in Large Projects. *IT Professional* 8(5), 25–29 (2006)
42. Avram, G., Bannon, L., Bowers, J., Sheehan, A., Sullivan, D.K.: Bridging Patching and Keeping the Work Flowing: Defect Resolution in Distributed Software Development. *Computer Supported Cooperative Work* 18(5-6), 477–507 (2009)
43. Spanjers, H., Huurne, M.T., Graaf, B., Lormans, M., Bendas, D., Solingen, R.V.: Tool Support for Distributed Software Engineering. In: Proceedings of the IEEE International Conference on Global Software Engineering (ICGSE 2006), pp. 187–198 (2006)
44. Heinonen, S., Tanner, H.: Early Validation of Requirements in Distributed Product Development - An Industrial Case Study. In: Meersman, R., Dillon, T., Herrero, P. (eds.) OTM 2010. LNCS, vol. 6428, pp. 279–288. Springer, Heidelberg (2010)
45. Mishra, D., Mishra, A.: A software inspection process for globally distributed teams. In: Meersman, R., Dillon, T., Herrero, P. (eds.) OTM 2010 Workshops. LNCS, vol. 6428, pp. 289–296. Springer, Heidelberg (2010)
46. Tervonen, I., Mustonen, T.: Offshoring Test Automation: Observations and Lessons Learned. In: Proceedings of the 2009 Fourth IEEE International Conference on Global Software Engineering (ICGSE 2009), pp. 226–235 (2009)
47. Salger, F., Engels, G., Hofmann, A.: Assessments in global software development: a tailorable framework for industrial projects. In: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering, vol. 2, pp. 29–38 (2010)
48. Prikladnicki, R., Audy, J.L.N., Evaristo, R.: A Reference Model for Global Software Development: Findings from a Case Study. In: Proceedings of the IEEE International Conference on Global Software Engineering (ICGSE 2006), pp. 18–25 (2006)