

Requirements Management in Distributed Projects

Darja Šmite

(Riga Information Technology Institute, Latvia
darja.smite@riti.lv)

Abstract: Global software development expands with every year providing software developers with new opportunities. However, practitioners face global challenges and threats particular for distributed environment that require new methods and tools to be implemented. This paper provides a report on requirements management practices in globally distributed projects in one of the leading software development organizations in Latvia. The paper discusses how to form requirement analysis teams wisely, how to reduce diversity between the involved parties, what to be aware of during the development phase, and how to facilitate successful communication for entire project.

Keywords: requirements management, global software development, distributed software development projects, lessons learned, best practices

Categories: D.2.1, D.2.9, I.2.6

1 Introduction

Global software development nowadays is not a phenomenon. It expands with every year and turns from a trend into everyday type of doing business. GSD enable reaching mobility in resources, obtaining extra knowledge, speeding time-to-market and increasing operational efficiency [Smite, 05]. Nevertheless, globalization has also significantly changed the nature of software development projects (see Fig.1). Software development in distributed environment is facing changes by involving related partners which are distributed in time, space and culture, "...each with its own set of needs that require unique methods of organization and control" [Karolak, 98]. The complexity of collaboration grows with the number of parties involved in the supply chain of a global project.

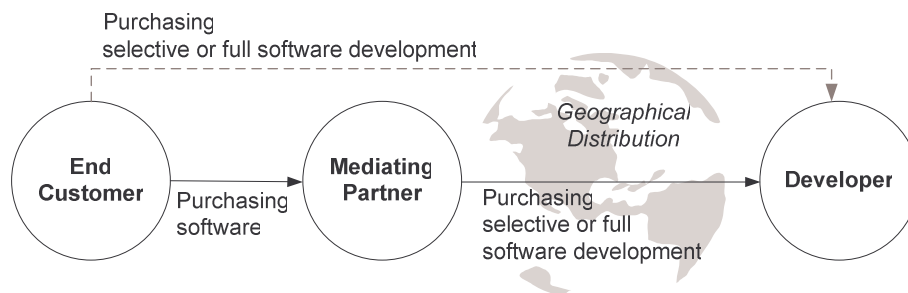


Figure 1: Globally distributed project scheme [Smite, 05]

While software systems complexity grows along with sophistication of software development processes, the quality of requirement analysis and management remains one of the most important sources of risk. Most of the problems encountered in software development are attributable to shortcomings in the processes and practices used for requirements engineering [Wieggers, 99]. This consideration remains the same for distributed projects. An investigation of 28 distributed projects has demonstrated the importance of requirements management and adoption of new practices for coping with global risks in distributed projects.

This paper is organized as follows. The next chapter provides research overview. Discovered practices are described in chapter 3, which is followed by discussion of the results in chapter 4. The paper ends with conclusions and insights in future work.

2 Research Overview

The author leads a research which aims to improve software development processes in globally distributed projects in one of the leading software development companies in Latvia. The research major objective is to build a framework that would serve as an Experience Factory providing best practices for global project improvement.

2.1 Research Approach

This paper describes an exploratory research based on the field-studies. The research is motivated by an industrial background. It takes place in one of the largest software houses in Latvia, which employs around 370 employees and is involved customer software development and maintenance in globally distributed projects with partners from Europe and Scandinavia.

Data on globally distributed software development projects was gathered through series of interviews with experienced project managers [Smite, 05], surveys [Smite, 04], and related literature overview. The data was analyzed according to principles prescribed by a grounded theory as described by Strauss and Corbin [Strauss and Corbin, 98]. Grounded theory is a research method that seeks to develop theory that is grounded by data about a certain phenomenon (here – global software development) that has been systematically gathered and analyzed.

Through applying open coding followed by axial coding, and then selective coding, a set of global factors and associated global threats has been derived.

2.2 Survey Overview

In order to validate the research results, a survey was conducted considering the following objectives:

- to investigate, which threats are faced by global projects;
- to evaluate the magnitude of consequences of the threats.

Web-based inquiry forms were filled by 28 distributed project managers from 3 software houses in Latvia, which are involved in software development projects with geographically distributed customers or prime contractors. The nature of the projects investigated is either custom or product software development, and software

maintenance including existing software improvement. The survey uncovered a list of most frequent threats and evaluation of their risk level.

Risk level (0-5) was evaluated as a combination of frequency of occurrence of the threat and the magnitude of its consequences for each of the following cost-related, time-related or morale-related project results: Unexpected management costs, Budget overrun, Time delays, Late product delivery, Customer dissatisfaction, Undermined morale, Disputes and litigations, Customer costs escalation.

The following scale was used for frequency evaluation: 5 (81-100%); 4 (61-80%); 3 (41-60%); 2 (21-40%); 1 (1-20%); and 0 (0%).

The following scale was used to evaluate the magnitude of consequences: 5 (Disastrous); 4 (Significant); 3 (Moderate); 2 (Minor); 1 (Negligible); 0 (None). Overall threat magnitude of consequences evaluation has been derived by calculating an average size of magnitude for each of the consequences caused by the threat, and then choosing maximum out of these values. Subsequently, if a threat significantly impacts cost-related results, but remains insignificant for e.g. morale-related results, it still is evaluated as significant.

3 Survey results

The survey uncovered the following TOP5 threats (see Table 1).

Threats	Risk Level	Frequency	Magnitude
Poorly defined or inconsistent software requirements specifications	3	4 (62%)	3
Faulty effort estimates	3	4 (62%)	3
Diversity in process maturity and/or inconsistency in work practices between the partners	3	3 (52%)	3
Increased level of unstructured poorly-defined tasks	3	3 (45%)	3
Poor or disadvantageous distribution of software development activities between the customer and supplier(s)	3	3 (41%)	3

Table 1: TOP 5 threats faced by distributed projects

The results show that issues connected with requirements management have been named as one of the most important sources of risk. Comparing the results of TOP 5, we can conclude that the mentioned threats are correlated. Software development lifecycle processes in global projects are often distributed between the parties involved in the project. This produces problems for accurate effort estimates. Requirement analysis and further management seen as a core activity is frequently not being outsourced but instead performed in-house. However, followed by diversity in process maturity or inconsistency of work practices, this can cause various problems.

Magnified by increased costs of logistics of holding face to face meetings, increased virtualness, lack of expertise in outsourcing projects, relatedness with other suppliers in the complex supply value chain, organizational culture mismatch, etc. requirements management needs new practices to be implemented in globally distributed projects.

The following additional threats can be named as reasons for poor requirements management:

- diversity in process maturity,
- inconsistency in work practices,
- lack of version control,
- relatedness with other suppliers,
- lack of language skills,
- terminology differences,
- customer employee unwillingness to collaborate.

One project manager reported „Customer claimed that they do not have sufficient human resources to validate all software requirements specifications“. Another reported that „it is the customer’s strategy for decreasing the expenses“.

Further requirements clarification can be also put under threat due to various risks that lead to troubled communication, such as the following:

- increased virtualness due to dominant use of asynchronous communication,
- increased cost of logistics of holding face to face meetings,
- increased complexity of spreading awareness and knowledge,
- customer’s belief that the work cannot be done from a far off location,
- lack of team spirit,
- lack of clarity about responsibility share,
- poor cultural fit,
- lack of experience and expertise with outsourcing projects.

In the next chapter the major practices for successful requirements management in distributed environment are described. These practices are derived from the field studies and address threats that are specific for global projects.

4 Practices

4.1 Form the requirement analysis team wisely

Requirements elaboration is produced without the distributed development supplier involvement in 80% of the investigation projects.

However, distribution in space, time and culture can cause significant problems in clarification of the requirements during the development phase. In order to mitigate the risk of poor transition from analysis to development phase, it is advisable to consider involving supplier representatives in the requirement analysis team. This can be achieved by sending one or more analysts from the supplier side to participate throughout requirements analysis on the partner’s premises.

4.2 Reduce diversity

A consistent challenge experienced in distributed work is maintaining coherence, commitment, and continuity across the multiple locations, priorities, and interests of the hundreds of people involved in the collaborative effort [Orlikowski, 02]. Diversity in process maturity and work practices can bring sudden risks to the project, such as time delays, unexpected management costs and low morale.

One project manager said: "Sometimes on-site project management treats Latvian partners in exactly the same way they treat Finnish developers sitting next to themselves - too little advance planning and clarification of expectations".

Awareness about diversity can be used to plan and mitigate the related threats. A common understanding of work practices shall be established and maintained through initial training and socialization workshops. Training in "soft skills" such as trust, cultural differences, communication, collaboration, context sharing, and knowledge management, is useful.

4.3 Agree upon requirement analysis template

Context differences, such as diversity in process maturity and practices, organizational and cultural differences, diversity in employee education, very often cause problems with the development team expectations for the requirements specifications.

One project manager said: "Poor requirements specification received from the customer was a great problem. We involved designers in developing detailed specifications before coding and then approved them with the customer. This work though was not paid by the customer".

It is therefore useful to discuss what kind of specification the supplier expects in the very beginning of the requirement analysis phase. A template based on these expectations can be developed, approved and used during requirement specification.

4.4 Develop a glossary

Linguistic and context differences can cause misunderstandings in achieving an understanding of requirements. Some project managers report that what is seen as obvious for the customer, sometimes is hard to understand for us. Most of the problems are caused by terminology in a certain business sphere.

Therefore, it is commendatory to come to a general agreement on terminology by developing a joint glossary.

4.5 Implement a version control tool

Though poor version control is faced only in 14% of the projects, lack of joint version control accompanied by lacking proximity can lead to misunderstandings, time delays and rework. One project manager reported „Sometimes we even do not know about the changes. To do something new we have to ask always for the latest version“.

Advanced tools shall be implemented to support multiple teams and provide online access for every partner.

4.6 Establish clear responsibilities and priorities

Lacking next door closeness and proximity between the teams involved in a joint project, there is a risk of miscommunication. One project manager reported: “We cannot allow everyone to communicate with anyone”. If this happens, requirements and changes with high priority comes from everywhere and can be hardly managed in order.

Another project manager reported: “Requirements are frequently not explicitly fixed in any reasonable document. Project management from the customer side is most often done by just forwarding e-mails (and forwarding them again, and again) until they reach Latvian suppliers. It is then difficult to figure out the chain of responsibility, management, and resourcing for each separate change request.”

Therefore, it is very important to define project member roles and establish proper communication liaisons. One project manager reported “We have defined not only the project members who should be added in “Cc” fields for emails, but we have also defined different titles and kinds of “Subjects” for major types of discussion”.

4.7 Maintain constant communication

The partners shall understand that not all the problems and questions can be solved by email or phone. One developer reported that when their systems analyst comes to the distributed development team on a visit, developers always discuss huge amount of insignificant problems that they can't solve through emails or phone calls.

Experienced project managers advise to organize face to face meetings once in a month or two for planning and requirement ambiguity clarification. One project manager explained: “Regular personal meetings are very necessary. When our travelling was restricted by the customer due to project budget economy, the relationship started to turn for the worse”.

Advanced communication tools as videoconferencing are very effective and should be used extensively whenever possible to discuss the requirements and changes.

Many project managers report that employees who lack fluent language skills are afraid to speak with the partner over phone. This causes dominant use of asynchronous communication tools and brings time delays in problem solution turnaround. A good thing to do is sending one supplier representative to the partner side for the entire project to relieve communication (particularly feasible in large projects).

4.8 Involve optimistic employees

When a company starts to outsource a part of software development life cycle activities, the in-house employees can be put under risk of possible human resource optimization. This can cause customer's employee unwillingness to collaborate with the sub-contractors throughout the project. Accompanied by a belief that the work cannot be done from a far off location it can significantly trouble cooperation between the distributed teams. In result, the developer, who has received requirements specifications for further development, is left without a helping hand.

A good piece of advice is to involve only optimistic and enthusiastic employees in global projects.

5 Discussion

Despite a wide variety of literature on whether and why to outsource, there is still a lack of research on how to achieve successful performance in distributed environment. Many companies fail in the execution of strategic outsourcing [Laplante et al., 04].

Most of the problems discussed in this paper address diversity, relatedness and poor communication issues that highlight particularities of software development involving geographically distributed teams. Managing requirements in a distributed environment can become a tough task if the process is not well defined and the teams are not experienced or prepared for this cooperation model.

To overcome the problem of distance in GSD, various managers are experimenting and quickly adjusting their tactical approaches [Carmel and Agarwal, 01]. However, it's difficult to muster the energy needed to overcome obstacles to change and to put new knowledge into action [Wieggers, 99]. Organizations are naturally resistant to changes. Supplier teams often report on the customer unwillingness to adopt mature processes because it requires more time and resources. One project manager reported "We can do nothing about it. The word of the customer is a rule".

Knowledge and information distribution among the involved parties is also a challenge. One project manager reported that poor requirement specifications actually "reflect poor process for knowledge distribution by the customer team". Another reported about problems with a mediating contractor company that prevented supplier and customer direct collaboration and introduced delays, passed on the information selectively, or added some redundant surplus information.

Dealing effectively with global factors requires much effort and a deep competence in what may be labelled "distributed organizing" – the capability of operating effectively across the temporal, geographical, political, and cultural boundaries routinely encountered in global operations [Orlikowski, 02]. Organizations that involve subcontractors should experiment and adjust the global product delivery models by decreasing the processes and interaction layers that do not add value and consider improvements that would enable effective cooperation of distributed teams. Diversity and lack of common goals make organizations consider new approaches as partnership to be implemented to achieve outsourcing benefits more effectively [Lee et al, 00].

6 Conclusions and Future Work

Virtual product development is considerably more complex than even the most complex project managed entirely in-house [Karolak, 98]. A set of global factors cause various threats that are particular for distributed environment. The survey results show that requirements management in global projects is one of the essential challenges that shall be paid adequate attention. Separation of the team that specifies requirements and the team that produces software is joined by diversity between processes, inconsistency in practices, linguistic and terminology differences, temporal distance etc. This makes practitioners to seek for new approaches and practices to be implemented in global projects for better performance.

Communication plays an important role in successful distance overcoming. Therefore training in “soft skills” such as trust, cultural differences, communication, collaboration, context sharing, and knowledge management, is essential.

While global factors preclude distributed environment transformation into a common way of producing software in-house, practices that mitigate global risks shall be implemented in order to overcome global challenges such as distribution in space, time and culture. The author’s future work is related to deriving facilitates, methods and practices for better performance in global project environment.

The practices described in this paper of course cannot be spread to any global project, as the nature of software development projects in distributed environment is very diverse. Nevertheless, the description of threats and possible outcomes may be useful, in order to make the right decision on project activity distribution and joint performance.

Acknowledgements

The author appreciates valuable research input received from the developers and project managers from the investigated software house.

This research is partly supported by European Social Fund and the Latvian Council of Science project Nr. 02.2002 “Latvian Informatics Production Unit Support Program in the Area of Engineering, Computer Networks and Signal Processing”.

References

- [Carmel and Agarwal, 01] E. Carmel, R. Agarwal, Tactical Approaches for Alleviating Distance in Global Software Development, *IEEE Software*, March/April, 2001, pp.22-29
- [Karolak, 98] D.W. Karolak, *Global Software Development: Managing Virtual Teams and Environments*, IEEE Computer Society, 1998
- [Laplante et al, 04] P.A. Laplante, T. Costello, P. Singh, S. Bindiganavile, M. Landon, The Who, What, Why, Where, and When of IT Outsourcing, *IT Pro*, January/February 2004, pp. 19-23
- [Lee et al, 00] J.N. Lee, M.Q. Huynh, K.R. Chi-wai, S.M. Pi, The Evolution of Outsourcing Research: What is the Next Issue? In *Proc. of the 33rd Hawaii Int. Conf. on System Sciences*, 2000
- [Orlikowski, 02] W.J. Orlikowski, Knowing in Practice: Enacting a Collective Capability in Distributed Organizing, *INFORMS*, May/June, 2002, pp. 249-273
- [Smite, 04] D. Smite, Global Software Development Project Management – Distance Overcoming, In *Proc. of Int. Conf. on European Software Process Improvement (EuroSPI)*, Springer, November 2004, pp. 23-33
- [Smite, 05] D. Smite, A Case Study: Coordination Practices in Global Software Development, In *Proc. of the 6th Int. Conf. on Product Focused Software Process Improvement (PROFES)*, Springer, June 2005, pp. 234-244
- [Strauss and Corbin, 98] A. Strauss, J. Corbin, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, Sage Publications, 1998
- [Wieggers, 99] K.E. Wieggers, *Software Requirements*, Microsoft Press, 1999