GRCM: A Model for Global Requirements Change Management

Waqar Hussain, Tony Clear

Auckland University of Technology
{waqar.hussain,tclear}@aut.ac.nz
http://www.aut.ac.nz

Abstract. [**Context and motivation**] In the delivery driven context of contract software production, efficient and effective requirements change management (RCM) remains a challenge for global software development (GSD). [**Question/problem**] New RCM models need to be devised for GSD settings, to reduce confusion and improve the efficiency of managing requirements change and the resulting impacts. [**Principal ideas/results**] We present a model drawn from a case study which evaluated RCM practices in a GSD organization, with sites based in USA and Pakistan. [**Contribution**] We extend the observed practices by developing a theoretically informed process model to improve RCM efficiency and effectiveness by using a baseline requirements artifact and tool supported collaboration process.

Keywords: Global Software Development, Multi Site Requirements Change Management Model, Global Requirements Change Management Model, Requirements Engineering

1 Introduction

For software companies working in a global context, producing against tightly constrained software delivery contracts, requirements change management (RCM) is a critical task. Poorly handled change leads to reduced product and service quality, and unsatisfactory resourcing, technical and commercial outcomes. Recently there have been calls [1] for global software development (GSD) researchers to engage in practical partnerships, adapting existing methods and tools, rather than developing elegant theoretical models in isolation from practitioners.

This work investigates the RCM process as practiced in a GSD field setting and compares it with available RCM models (primarily suitable for single site development) from the literature [2–4, 11, 13]. We propose a global requirements change management (GRCM) model accommodating multi-site development extended from the activities, roles and artifacts identified in existing models for requirements change management [5].

2 Background

2.1 GSD and Requirements Management

GSD poses challenges for managing requirements change because distance (cultural, geographical, temporal and language) aggravates coordination and control problems, through its negative effects on communication [7]. Requirements management, one of the most collaboration-intensive activities in software development, presents significant difficulties when stakeholders are distributed [6].

Many partial solutions have been offered for the implementation of Requirements Engineering (RE) in a global environment but they lack process level detail [8]. GSD demands robust models, methods and processes that can efficiently and effectively execute GSD work [10]. This research responds to that need.

2.2 RCM Process Models

The RCM models found in the literature [2–4, 11, 13], are not designed for the GSD environment. Mapping these models to multi site development is difficult as they do not describe how the collaborative activity for managing change will be handled in a globally distributed project, and process level detail is missing. Yet practitioners are wrestling with these challenges on a daily basis.

A survey [5] was conducted that compared the various activities, roles and artifacts (ARA) in the existing process models of RCM. It was concluded that [12] gives the highest level of ARA coverage by a single model, (13 out of the total 34 elements found in the literature). It was further concluded that there were no standard models of RCM and lack of detail of the ARA involved reduced the value of these models for industrial practice.

Our proposed model is developed specifically for the GSD environment and is more comprehensive than the RCM models proposed in the literature (covering 24 of the 34 elements). It also prescribes the use of collaborative technology to more efficiently manage RCM activities across distributed sites. We believe this gives our model strength in reducing requirements management challenges arising from development projects conducted at a distance.

3 Research Process

We profile here the outcomes of an exploratory case study [14] aiming to enhance existing RCM models to better support GSD. The characteristics and context of the setting for this study are mapped below, followed by an elaboration of the data collected for the study.

3.1 The Case Study Settings

GSD Inc, the selected company for our case study, is a CMMI Level-II certified small to medium sized company with almost 100 employees. Two projects, *SDE*

(Project 1) and *DataDive 2.0* (Project 2) were observed during the case study. *SDE* is a web application development project for a leading publishing client organization in the USA. *DataDive 2.0* is a centralized web based application which provides a suite of tools for query and analysis. The GSD Inc Pakistan office undertakes development projects on a contract basis, to a client supplied specification, to meet the company's need for low cost solutions and additional expertise. The software development life-cycle is thus driven by up-front requirements, and negotiated pricing. In practice this results in a pragmatic version of *waterfall by feature* development, wherein changes with significant resourcing impacts result in renegotiation of pricing.

3.2 Data Collection and Analysis Methods

Data was collected for the two projects over a period of 8 months from August 2009 to April 2010 at the development site situated in Pakistan. A total of 36 change request forms were collected, 24 for project 1 and 12 for project 2. Our data analysis process, adapted from [14], investigated the change management process, related issues and the rationale for requirements change. Critical artifacts such as Change Request Forms (CRF), Software Requirements Specifications (SRS), email messages, status reports etc. were included for qualitative analysis of data. Semi structured interviews were conducted to support and validate this analysis. Key project members with at least three years experience in GSD, (the Change Moderator - *CM*, Quality Assurance Manager, Team Lead and Analyst) were interviewed.

4 The Proposed Global RCM Model

The company operated with a variable degree of adherence to CMMI prescribed RCM procedures. Issues identified with the existing RCM process in the study site were: insufficient impact analysis; limited sharing of information relating to rationale for changes; and poor recording of requirements change information. To address the inefficiencies introduced by these practices we propose a Global Requirements Change Management (GRCM) model for the GSD environment. The model draws upon frameworks from the literature, incorporating the typical change activities (namely *request, verify, implement, validate* and *update* [11]) of the normative RCM models [2–4] and extending the model presented by [13].

4.1 Description of The GRCM Model

The process model presented in Figure 1 uses the terms *Role* and *Site* to show the distribution of the work environment with multiple team members at multiple sites. In the inset at the top left corner the model shows (*Role1-Site1*) which means any *Role* (such as tester, developer, project manager) at any particular *Site* (Pakistan, US, India etc.) played by a stakeholder who can initiate change. Similarly (*Role2-Site2*) means any other key stakeholder role at a designated
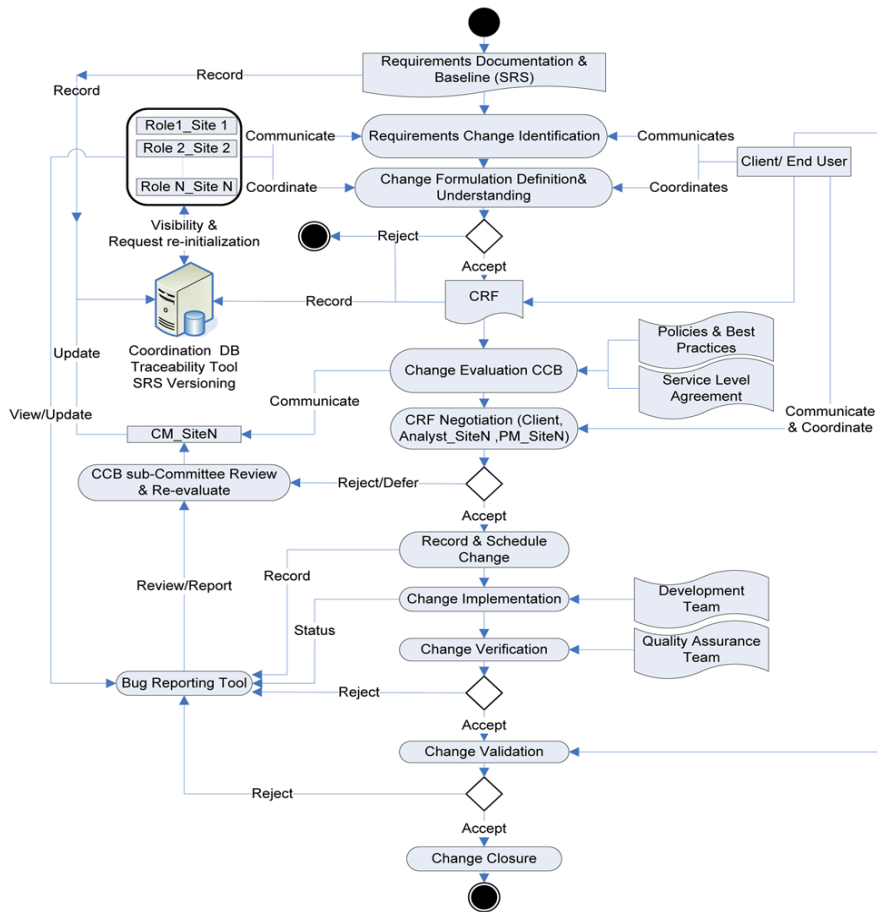
Fig. 1. Proposed Global Requirements Change Management Model (GRCM)

location (e.g. *Site2*). The model is extendable to include any number of teams, sites and stakeholders (*RoleN-SiteN*). In the proposed model only one client is shown for simplicity. However the model can equally reflect a number of clients at multiple locations, for example (*ClientN-LocationN*), and so on.

4.2 Operation of the GRCM Model

The model takes a baseline requirements document (in this case an *SRS*, but could equally include agile artifacts such as user stories) as an initial input into the process model. The baseline requirements artifact is linked with the coordination database to record and trace changes to the requirements. The *SRS* remains visible to all stakeholders across sites, once linked with this collaboration database, whereas specific design artifacts are visible to the local teams

only. When a change is identified and communicated by any stakeholder from a given site, it undergoes a process of *change formulation, understanding and definition.* This is a technology supported collaborative activity among distributed stakeholders. Upon its acceptance it moves to the formalisation stage, when a change request form (*CRF*) is filled out by the change initiator and submitted for formal review and evaluation by the change control board. The *CRF* is the key artifact circulated among the parties when considering a change. The requested change (whether accepted or rejected), is recorded in the online repository for future reference. The formally approved change request then enters the negotiation process. If the change is accepted for implementation it is recorded and scheduled using a tool which makes change data visible to all the stakeholders. After implementation by the development team it is verified and validated and then closed. If the change is rejected it goes to a subcommittee of the change control board for a review and re-evaluation process. The report is sent to the Change Moderator who then updates the coordination database and makes the status of the change available to all the stakeholders.

4.3 Application of the Proposed Model

The proposed GRCM model *Figure 1* may represent a variety of GSD contexts, and could be adapted to accommodate new roles identified in specific settings. The process model, with its support for collaboration through technology and shared artifacts, contributes to cross-site negotiations, awareness and visibility of changes. It provides a pragmatic balance between software production and control, thereby improving the efficiency of the RE process. While devised in a web application context, it is not limited to any organization or type of software project. Thus we believe it could be applied in a range of GSD settings.

4.4 Limitations of the Proposed Model

This GRCM model has been synthesized from theory and practice and has had some initial use within the case study site, to validate its effectiveness. The scope for testing and optimization of the process model still remains. The model applies primarily in support of RE activities and contract modification decisions, and thus has potential limitations in its applicability to the detail of later development phases. Yet within this study's constrained scope of pragmatic *waterfall by feature* development it provides a practicable approach. The model also lacks any prescription of the mediating technology that may be employed. Since many kinds of collaborative technologies (e.g. repositories, bug reporting tools etc.) can be used for GSD projects, we believe most organizations will tailor a technology set to suit their needs.

5 Conclusion and Future Work

Existing requirements change management models have not been specifically developed for the GSD environment. We report the findings from a case study that

investigated the change management process employed by a GSD organization. We identified several problems with their existing RCM process. We propose a resulting global requirements change management (GRCM) model, informed by our insights from theory and practice. The model incorporates the commonly adopted change activities (namely *request, verify, implement, validate* and *update* [11, 13]) of the normative RCM models [2–4]. The GRCM model augments these with a collection of activities, roles, and artifacts [13] from the literature. Currently the observations from its initial use at the case study site are encouraging and show signs of its efficiency and effectiveness in this industrial setting. The model now needs wider application in a variety of GSD project settings for a full assessment of its workability and scope of application.

References

1. Damian, D.: Requirements Engineering in Global Software Engineering: How far have we come? [Panel Session]. 6th IEEE International Conference on Global Software Engineering, IEEE, Helsinki, Finland (2011)
2. Olsen, N.C.: The software rush hour [software engineering]. Software, IEEE 10 (1993) 29-37
3. Mkrinen, M.: Application management requirements for embedded software. Technical Research Centre of Finland VTT Publications, Espoo (1996) 286
4. Ince, D.: An introduction to software quality assurance and its implementation. McGraw-Hill (1994)
5. Ramzan, S., Ikram, N.: Requirement Change Management Process Models: Activities, Artifacts and Roles. Multitopic Conference. IEEE Islamabad, Pakistan (2006) 219-223
6. Sengupta, B., Chandra, S., Sinha, V.: A research agenda for distributed software development. Proceedings of the 28th international conference on Software engineering. ACM, Shanghai, China (2006) 731-740
7. Carmel, E.,Agarwal, R.: Tactical Approaches for Alleviating Distance in Global Software Development. Software, IEEE 18 (2001), pp. 2229.
8. Lopez, A., Nicolas, J., Toval, A.: Risks and Safeguards for the Requirements Engineering Process in Global Software Development. 4th IEEE International Conference on Global Software Engineering, IEEE, Limerick, Ireland (2009)
9. Sangwan, R., Bass, M., Mullick, N., Paulish, D.: Global Software Development Handbook. Auerbach Publishers, New York, NY (2007)
10. Damian, D., Moitra, D.: Guest Editors' Introduction: Global Software Development: How Far Have We Come? Software, IEEE 23 (2006) 17-19
11. Niazi, M., Hickman, C., Ahmad, R., Ali Babar, M.: A Model for Requirements Change Management: Implementation of CMMI Level 2 Specific Practice. In: Jedlitschka, A., Salo, O. (eds.), Vol. 5089. Springer Berlin / Heidelberg (2008) 143-157
12. Leffingwell, D., Widrig, D.: Managing software requirements, A unified approach. Addison-Wesley, Boston (2000)
13. Imtiaz, S., Ikram, N., Imtiaz, S.: A process model for managing requirement change. Proceedings of the Fourth IASTED International Conference on Advances in Computer Science and Technology. ACTA Press, Langkawi, Malaysia (2008)
14. Briand, L.C., Basili, V.R., Kim, R.: A Change Analysis Process to Characterise Software Maintenance Projects. Presented at International Conference on Software Maintenance, Victoria, Canada (1994).