

Communications in Distributed Software Project Management

Lj. Kazi*, B. Radulović*, D. Ivin*, M. Bhatt** and S. Gheeya***

* University of Novi Sad, Technical faculty "Mihajlo Pupin", Zrenjanin, Serbia

** University of Mumbai, R.D. National College, Mumbai, India

*** SUPINFO Institute of Information Technology, Grenoble, France

ljubicakazi@ptt.rs, bradulov@ptt.rs, ivin_bd@hotmail.com, mmbhatt@gmail.com, suraj19@hotmail.fr

Abstract - Software engineering and project management today are using advantages of modern communication technologies, with processes in distributed working environment. This paper addresses research and practical issues in the domain of distributed software development and management and particularly in communications area. We propose an integrated and layered model of all relevant aspects of distributed software project management communications, suitable for performance measurement and teaching alignment purpose.

I. INTRODUCTION

We are facing the fact that today software is developed by teams that are, for large projects, usually physically dislocated. Terms that are used for this type of software development organization are global software development, distributed software development, teamwork software development etc. Distributed software engineering and project management are facing many issues, categorized as general project management issues, specific to software development and particularly addressed to teamwork and distributed development.

Many years of practical work, issues, solutions and research resulted in sublimation in the form of standards as general accepted approaches to certain working areas. Project management (PM) standard framework include PMBOK guidelines [1], while more precise it is defined in worldwide accepted PRINCE2 body of knowledge. At the other hand, software engineering body of knowledge gives directions in SWEBOK [3].

Some of software project management best practices include [4]: accurate project metrics and measurements, accurate time/cost estimation, project management and software development methodology, project status reporting, automated tool usage, good communication with client, user involvement, adequate understanding of customer problems and requirements, team leadership, participation in decision making, good communication among project staff, morale boosting rewards and performance appraisals.

This paper aims to present most important issues regarding communication in multi-site development working environment. It will present layered model of all structures needed for communication and specially it will present data flow structures regarding PRINCE2 document templates that could be used as basis for

standard data format in information exchange during software project management.

II. RELATED WORK

Research in the field of distributed project management and distributed software engineering (i.e. global software development [5]) is recently very productive with general research trends and directions [6] and presenting experiences from diversity of global projects [7]. Results are related to measurement and monitoring performance [8], comparing performances with face-to-face communication [9] and improving performance of distributed teamwork.

Distributed teamwork software project engineering and management are facing issues such as: technical problems of communication [10], coordination [11], communication speed [12, 13], knowledge sharing [14, 15], group awareness [16] and adjustment of heterogeneous teams [17], creativity and quality [18]. Special issues are examined regarding communication practices in different software development methodologies, such as Extreme Programming [19].

As one of solutions to these issues, specialized software ("groupware") is developed to be used for organizing, communication and integration of teams and their work in distributed working environments. They support activities such as meetings [20], organizational learning [21], self and peer assessment [22], integration of process and project management [23] for collaborative multi-site development [24].

III. RESULTS

A. Layered communication model for distributed software project management

As one of results of this paper, we present layered communication model, that integrates different aspects of communications in distributed software development and project management.

This model consist of several layers that enable communications, form a communication structure and enable functionality of communications. It is based on

general structure of information system (hardware, software, lifeware and orgware).

TABLE I. LAYERED COMMUNICATION MODEL FOR DISTRIBUTED SOFTWARE PROJECT MANAGEMENT

Layer	Sub-Layers
Orgware	Communication protocols and standards
	Business rules
	Project management methodology
	Development frameworks
Lifeware	Roles
	Responsibilities
	Constraints
Software	Communication software
	Groupware
	Project management software
	Development environment
Hardware	Telecommunication equipment
	LAN equipment
	Workstation hardware equipment

In this model, hardware presents technical equipment needed for personal and teamwork software development and communication between teams, software enable development, project management and communications,

lifeware present set of working roles of individuals within teamwork and project (like developer, project manager, team leader etc.) and orgware present procedures and methodology for project management.

Proposed model presents a basis for teaching process in this field, with systematic presentation of knowledge areas. It also presents a framework for performance evaluation of relevant aspects of communications in distributed software development process and software project management.

B. Orgware and Lifeware

Regarding data that are used in project management, more precise prescription regarding document templates and procedures gives PRINCE 2 methodology. Based on this methodology and document templates, business process model (BPM) has been presented (Fig. 1).

This BPM presents workflow, data flow, data stores and basic work roles (lifeware) of the model that are responsible for each activity, data flow and data store. These work roles are: User, Client, Portfolio Manager, Team Leader, Project Manager and Team Member.

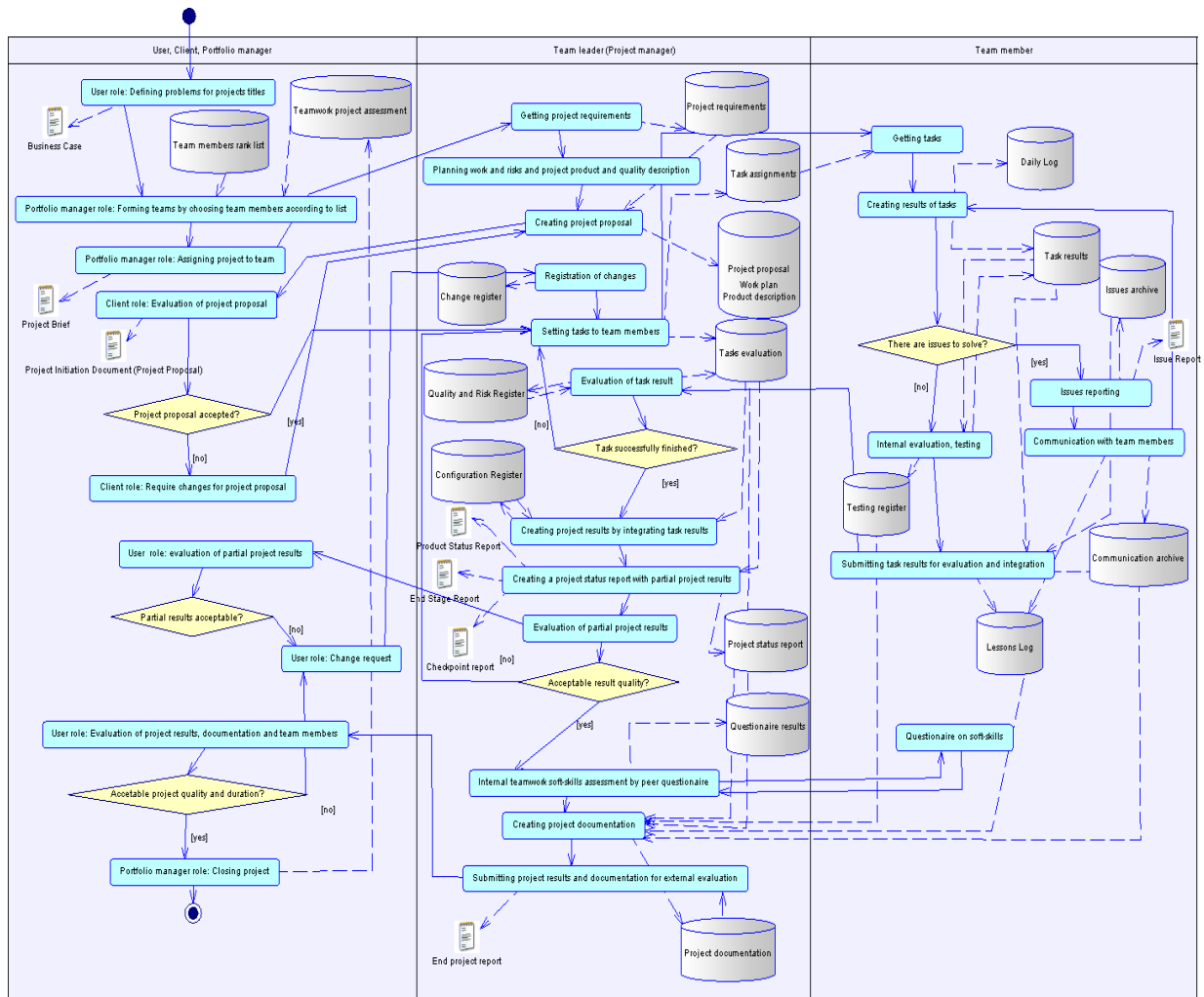


Figure 1. BPM according to PRINCE 2 methodology [25]

C. Software

Software support consist of basic communication software (network system software, applicative software such as e-mail, messangers etc.), project management software, groupware tools and software development environment.

Basic knowledge areas in project management as well as basic PM activities should be covered by project management software support. Knowledge areas define important data structures and group of processes require appropriate software functionality.

TABLE II. PROJECT MANAGEMENT KNOWLEDGE AREAS AND PROCESSES

Basic knowledge areas [1]	Additional knowledge areas[26]
Scope	Resources
Time (duration)	Conflicts
Costs	Changes
Quality	Groups of processes [27]
Human resources	<i>Initiation</i>
Communications	<i>Planning</i>
Risk	<i>Execution</i>
Procurement	<i>Control</i>
Integration	<i>Closing</i>

Groupware software as a tool need to support:

- coordination of activities, meetings and setting/getting assignments
- evaluation of results
- integration of work
- knowledge sharing and organizational learning
- social networking and enhance teamwork cohesion
- quality assurance, self and peer assessment
- enhancing creativity.

Recently improved software development environments (like Microsoft Visual Studio 2010) have enhancements in functionality with modules that enable groupware and project management with distributed teamwork. This way software development environments are integrated with previously developed tools - groupware and project management software.

Some of these integrated systems are using web sites or file servers for sharing code and other artifacts during distributed software development and project management.

D. Communication

Figure 1. shows communication between all working roles during project activities, from initiation to finalization in BPM according to PRINCE 2 methodology.

In this model it has been shown that communication between working roles are of several types:

1. Initiation of activity - after completion of one activity, another activity (assigned to another role) can be started.

2. Reading data from data store - data stores are assigned to specific roles and some activities of other roles need to read data from data store that is assigned to another role. Permissions to access data are needed to be defined. Obviously, writing to data stores is restricted only to data store owners and other roles are not allowed to access these data stores for writing new/updating data.

Technology solutions support this communication activities by:

1. Business process management systems - enable orchestration of activities that are processed parallelly at distributed locations. Starting one activity depending on another (at some other location) is managed by these systems.

2. XML and web services - Data that are to be exchanged can be in informal, semistructural or structural form. It is obvious that there is no direct access to data stores of other working roles, but only using services (like web services) that enable access to previously prepared and deployed data in XML form. Widely accepted standard format for data exchange is XML. It is recognized from all software development environment vendors and database management system developers, so it is most appropriate to structure and exchange data in this format.

IV. CONCLUSION

This paper gives a brief overview of knowledge areas and issues in the field of communication within distributed software project management.

Recent research show most communication experiences, global trends and issues that are addressed and solutions that improve these aspects. Special software development methodologies bring specific challenges in communication, such as Extreme Programming, where time-to-deliver is crucial and all unnecessary administration and time consuming is avoided.

We presented an integrated communication layered model for different aspects of communications within distributed software project management. These layers are organized generally as a structure of an information system - orgware, lifeware, software and hardware. Particularly, software needs are addressed from functional aspect.

Special concern has been given to business process model based on PRINCE 2 methodology for project management, where data flow present communication between working roles in project management. It has been shown that communication types are initiating of activities and reading data stores. Technology such as business process management systems, XML and web services are described as support to these communication types.

Future work in this field is planned to be in detailed presentation of data structures of data flows and data stores, projecting databases and XML structures, in aim to enable development of specific software that could integrate all software modules for the needs of distributed software development and project management process.

REFERENCES

- [1] A guide to the Project Management Body of Knowledge, "PMBOK", Project Management Institute, www.pmi.org, 1996
- [2] "SWEBOOK", IEEE Computer Society, 2004.
- [3] Department for Business innovation and skills: PRINCE 2 templates, <http://www.berr.gov.uk/aboutus/corporate/projectcentre/pm-templates/page12526.html>
- [4] C. Manzil-e-Maqsood and T. Javed: "Practicum in Software Project Management – An Endeavor to Effective and Pragmatic Software Project Management Education", ESEC/FSE'07, September 3–7, 2007, Cavtat, Croatia
- [5] J. Herbsleb and D. Moitra, "Global Software Development", Software, IEEE, Mar/Apr 2001., Vol 18, Issue 2, pp. 16-20
- [6] B. Sengupta, S. Chandra and V. Sinha, "A Research Agenda for Distributed Software Development", ICSE'06, May 20–28, 2006, Shanghai, China.
- [7] Herbsleb J, Paulish D and M. Bass, "Global Software Development at Siemens: Experience from Nine Projects", ICSE'05, May 15–21, 2005, St. Louis, Missouri, USA
- [8] N. Ramasubbu and R. K. Balan, "Globally Distributed Software Development Project Performance: An Empirical Analysis", ESEC/FSE'07, September 3–7, 2007, Cavtat, Croatia
- [9] H. Andres, "A comparison of face-to-face and virtual software development teams", Team Performance Management, 2002, Vol. 8 Iss: 1/2, pp.39 - 48
- [10] M. Sosa, S. Eppinger, M Pich, D. McKendrick and S. Stout, "Factors that influence technical communication in distributed product development: an empirical study in the telecommunications industry", IEEE Transactions on Engineering Management, Feb 2002. Vol 49, Issue 1
- [11] A. Oberweis, T. Wendel and W. Stucky, "Teamwork Coordination In A Distributed Software Development Environment", Proceedings of the IFIP '94 Workshop FG 9: Communication and Coordination in Distributed Corporate Application Systems, Hamburg, August 28 - September 2, 1994
- [12] J. D. Herbsleb, A. Mockus, T. A. Finholt and R. E. Grinter, "An Empirical Study of Global Software Development: Distance and Speed", ICSE '01 Proceedings of the 23rd International Conference on Software Engineering, IEEE Computer Society Washington, DC, USA, 2001
- [13] J. Herbsleb and A. Mockus, "An Empirical Study of Speed and Communication in Globally Distributed Software Development," IEEE Transactions on Software Engineering, vol. 29, no. 6, pp. 481-494, June 2003
- [14] S. Faraj and L. Sproull, "Coordinating Expertise in Software Development Teams", Management Science, Vol. 46, No. 12 (Dec., 2000), pp. 1554-1568, Published by: Institute for Operations Research and Management Sciences
- [15] J. Espinosa, S. Slaughter, R. Kraut and J. Herbsleb, "Team Knowledge and Coordination in Geographically Distributed Software Development", Journal of Management Information Systems, Volume 24, Number 1 / Summer 2007, pp. 135 - 169
- [16] C. Gutwin, R. Penner, and K. Schneider, "Group Awareness in Distributed Software Development", CSCW'04, November 6–10, 2004, Chicago, Illinois, USA.
- [17] W. A. Hall, B. Long, N. Bermbach, S. Jordan and K. Patterson, "Qualitative Teamwork Issues and Strategies: Coordination Through Mutual Adjustment", Qualitative Health Research, Vol. 15 No. 3, March 2005 394-410
- [18] R. Ocker, S. Hiltz, M. Turrof and J. Fjermestad, "The effects of distributed group support and process structuring on software requirements development teams: results on creativity and quality", Journal of Management Information Systems - Special section: Information technology and its organizational impact, Volume 12 Issue 3, December 1995 M. E. Sharpe, Inc. Armonk, NY, USA
- [19] L. Layman, L. Williams, D. Damian and H. Bures, "Essential communication practices for Extreme Programming in a global software development team", Information and Software Technology, Volume 48, Issue 9, September 2006, Pages 781-794 Special Issue Section: Distributed Software Development
- [20] A.H. Dutoit, B. Bruegge and R.F. Coyne: "Using an issue-based model in a team-based software engineering course", International Conference on Software Engineering: Education and Practice, Proceedings, New Zealand, 1996.
- [21] R. M. Vick, "Perspectives On And Problems With Computer-Mediated Teamwork: Current Groupware Issues And Assumptions", Journal of Computer Documentation May 1998, Vol. 22, No. 2
- [22] M. Freeman and J. McKenzie, "SPARK, a confidential web-based template for self and peer assessment of student teamwork: benefits of evaluating across different subjects", British Journal of Educational Technology, Volume 33, Issue 5, pages 551–569, British Educational Communications and Technology Agency, November 2002
- [23] M. Buffa, P. Sander and J.-C. Grattarola, "Distant Cooperative Software Development For Research And Education: Three Years Of Experience", International Conference on Computer Aided Learning in Engineering education, February 16-18, 2004, Grenoble, France
- [24] K.C.C. Chan and L.M.L. Chung, "Integrating Process and Project Management for Multi-Site Software Development", Annals of Software Engineering 14, 115–143, 2002, Kluwer Academic Publishers
- [25] Lj. Kazi, D Radosav, M. Nikolic and N. Chotaliya, "Balanced Scorecard Framework In Software Project Monitoring", International Symposium Engineering Management And Competitiveness 2011 (Emc 2011), June 24-25, 2011, Zrenjanin, Serbia
- [26] D. Letić D and V. Jevtić, "Upravljanje projektima-metode i softver", Tehnički fakultet "Mihajlo Pupin", Zrenjanin, 2007.
- [27] K. Schwalbe, "Managing information technology projects", Course Technology, Cengage Learning, 2011