Towards a Framework for Work Package Allocation for GSD

Marcos Ruano-Mayoral[1], Ricardo Colomo-Palacios[2],
Joaquín M. Fernández-González[1], and Ángel García-Crespo[2]

[1] EgeoIT
Av. Brasil 17, 28020 Madrid, Spain
{marcos.ruano,joaquin.fernandez}@egeoit.com
[2] Computer Science Science Department, Universidad Carlos III de Madrid
Av. Universidad 30, Leganés, 28911, Madrid, Spain
{ricardo.colomo,angel.garcia}@uc3m.es

Abstract. Global software development is an inexorable trend in the software industry. The impact of the trend in conventional software development can be found in many of its aspects. One of them is task or work package allocation. Task allocation was traditionally driven by resource competency and availability but GSD introduces new complexities to this process including time-zones differences, costs and cultural differences. In this work a report on the construction of a framework for work-package allocation within GSD projects is presented. This framework lies on three main pillars: individual and organizational competency, organizational customization and sound assessment methods.

Keywords: Global Software Development, task allocation, competency, cultural differences.

1 Introduction

Globalization in business is a trend that is here to stay. One of the results of globalization is the transfer of several production processes from one country to other. In software industry, this transfer is known as Distributed Software Development (DSD). The nature of the software production process enables its globalization and as a result of this, today, it is common for software development to take place in multiple or even distributed groups working together on a common system [1]. DSD has been facilitated by the revolution in communication which the Internet has made possible, developing tools that made feasible communication among development sites [2].

Global Software Development (GSD) can be seen as a particularization of DSD. Not in vain, according to [3], GSD is a particular kind of DSD in which teams are distributed beyond the limits of a given nation. More precisely, when the distance becomes global in DSD, with team members distributed around the world, this characterizes GSD [4]. Approaches have ranged from subcontracting portions of software development projects to third-party companies or subsidiaries, to establishing truly virtual development teams [5].

1

GSD gained momentum as it promised spectacular benefits but also attracted attention due to the complexity and challenges related to GSD [6]. Compared to localized software development, global software development has various risks, such as those induced by the spatial and temporal distance between development teams [7]. Literature presents many works devoted to underline GSD benefits (e.g. [8]) and caveats (e.g. [9]). In spite of the challenges, an increasing amount of software projects are run under GSD and this practice is becoming a norm in the software industry [10].

Because of its popularity, GSD must be studied carefully. However, the art and science of global software development is still evolving [10] and thus, immature [6].

Given that work allocation is an important issue in GSD [11], the aim of this paper is to define a framework to help managers to distribute work packages among project partners in GSD projects.

The remainder of the paper is organized as follows. Section 2 surveys the relevant literature about task allocation in GSD. Section 3 describes the framework to allocate work packages among project partners. Section 4 brings the main conclusions and future works.

2   Task Allocation in GSD

Traditionally, the allocation process of personnel within software development projects is mainly based on the technical skills of the teams and the task requirements [12]. In GSD scenario, task allocation is quite different, involving aspects like costs, time-zone differences and cultural imparities among others. As a result of this diversity, the allocation of tasks and responsibilities to distributed teams can have a significant impact on GSD project success [13]. While, back in 2008, [11] stated that there is a lack of attention to the problem of allocating projects in distributed teams, recent and relevant efforts have been devoted to this field of study.

Maybe the most relevant efforts come from the works of Lamersdorf, Munch et al. [14], [15], [16], [17], [18]. In these works, authors overview the different kinds of GSD projects and propose a set of criteria that is used when an allocation of tasks in a GSD project is faced. Using this, these authors propose an approach to perform a systematic task allocation evaluation.

Other works have been devoted to support GSD task allocation using multi-criteria decision analysis [11], allocating resources for software product line projects [19], using work breakdown techniques [20], system dynamics approaches [21] or just basic module allocation [22].

In this work a different approach is adopted. By means of qualitative research, authors propose a framework for work package allocation. This approach is novel, due to three intrinsic characteristics. Firstly, its novelty roots on the organizational and personal competency orientation of the framework. In the second term, its originality is based on the customization that this approach permits and needs to be fully operative. In [14], adaptability was one of the requirements of the tool developed. But apart from this adaptability, it is needed to describe the approach to

achieve this adaptability. This is the second big contribution of the work. Finally, the framework is based on the use of state of the art assessment methods. Thus, instead of a set of standard assessment metrics (using, for instance Likert-like values), the approach is to provide a set of sound and different techniques to evaluate all variables present in GSD task and work-package allocation.

3   A Framework for Work Package Allocation in GSD

This section describes the framework designed to assist work package allocation among GSD partners and locations.

3.1   Framework Construction Process

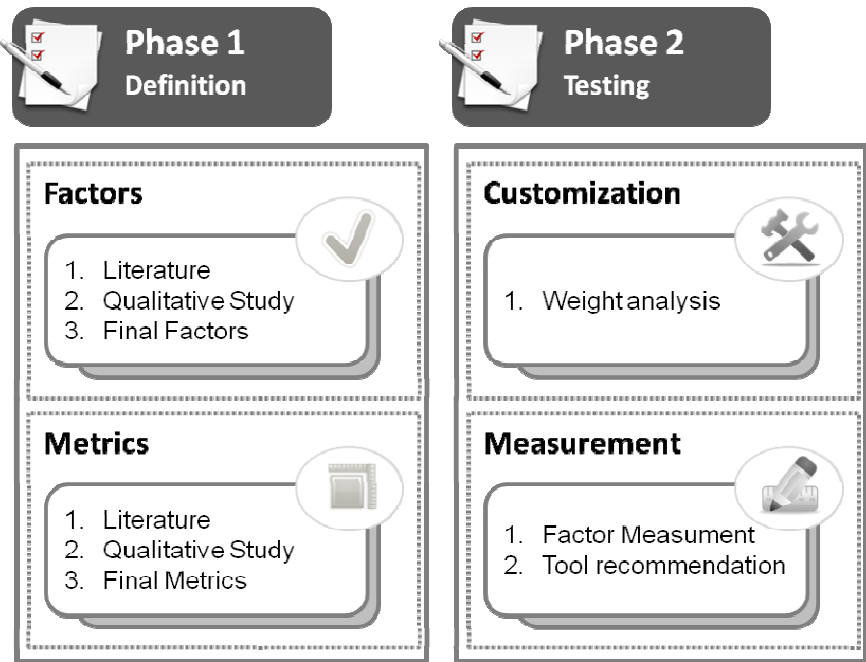The framework development consists in two main phases, as depicted in Figure 1.

Fig. 1. Definition and testing phases of the framework

Phase 1 comprehends the definition of the framework. This definition presents two main steps. The first one is the definition of the factors that are present in the allocation of resources in GSD projects. This definition is performed using a three point approach. Firstly, a study of the literature is performed. The output of this study is the set of initial factors that must be taken into account to allocate work-packages in GSD scenarios. The second point is a qualitative study in which several experts are

3

gathered using the focus group technique. In this case, the output of this study is a refined list of factors. The Focus Group method involves assembling small groups of peers to discuss particular topics [23]. The discussion within these groups, although directed by a researcher, is largely free-flowing [24]. Finally, authors compile a final list of factors is developed as the final output of Phase 1 Step 1. The second step is aimed to specify a set of valid metrics for all elements defined in Step 1. To do so, in the first term, an analysis of the literature is performed. Later, a qualitative study is performed in order to set a single metric for each factor. In this case, Delphi Study is the tool to achieve this objective. Delphi method was designed by Dalkey and Helmer [25] in the 1950s for military purposes and, from the 1960s onwards it was also used in organizational spheres thanks to the further developments of these authors [26]. This method uses a panel of experts as the source of information and questionnaires as the medium of interaction [27]. The Delphi method presents three main features [28]: anonymous response; iteration and controlled feedback; and statistical group response. Once the metrics are selected, this step ends with the definition and documentation of such metrics.

Phase 2 represents the customization of results obtained in Phase 1 to a particular GSD environment. This phase presents two steps. In Step 1, given a particular GSD project, all factors are weighted in order to adapt the decision to the expectatives and aims of the participants. Phase 2 Step 1 must be completed using techniques like interviews and qualitative approaches to get a common and agreed response to the weights of the project. In Step 2 all factors are measured using the metrics defined in Phase 1 Step 2. Once all measures have been obtained, by means of an automated tool, a set of recommendations for work-package allocation are issued.

In complex projects where many partners and work-packages are involved, different configurations cannot be handled without the help of an automated tool. WP-Locate is the proof-of-the-concept tool to support this decision making process. This tool stores three different kinds of information: project requisites (e.g. minimum assignations, round the clock need), work-package descriptions mainly in terms of competency needs and interactions and partners' information (competency levels and other variables). WP-Locate, taking into account the restrictions of the given project, provides a set of possible assignations ranked according to the assessment of every factor included in it.

3.2 Factors: Initial Steps

According to a preliminary study, the work of Lamersdorf, Münch and Rombach [17] identifies almost all factors that should appear in task or work-package allocation within GSD projects. These factors are:

- Cost
- Time
- Temporal Distance
- Geographic Distance
- Cultural Distance
- Size
- Economic situation

- Customer Proximity
- Collaboration history
- Strategic Considerations
- Software development maturity
- Human Resource Management maturity
- Trust
- Competency.

Focusing on competency in the individual and organizational model, the aim of this framework is to integrate competency measurement in these two levels in order to integrate in these two broad dimensions many of the aspects found. Based in previous works of the authors ([29], [30], [31], [32], [33]), the aim here is to integrate competence definition and assessment with the final objective to get a matching between the required competences and skills and those available in a given GSD site. This must be done comparing competency descriptions in the individual level and competence needs of each work-package.

With regard to cultural differences, authors believe that, although these factors are widely present in the software development related literature (e.g. [34], [35], [36], [37], [38], [39]), this are not properly included in decision making processes. Thus, the aim is to integrate findings present in the works of Hofstede (e.g. [40], [41 [42]) in the assessment of cultural differences by means of the comparison of the five cultural dimensions reported by this author, namely Power distance, Individualism / collectivism, Masculinity / femininity, Uncertainty avoidance and Long-term / short-term time orientation.

3.3 Testing Environment

The aim of PROPS-Tour project is to design, build and prototype an ICT platform for open and integrated promotion of different tourism services available in a certain geographic area, ensuring an efficient and unbiased provision of services information. PROPS-Tour is conceived as a catalyst to allow an uneven and heterogenic tourist services offering to reach its natural targets: tourists already in their destination. The promotion mechanisms will therefore allow the elaboration of updated and consistent information about the plethora of tourism services to feed recommendation systems that tourist may already use. The main technological and functional innovations provided by the PROPS-Tour project can be summed up as:

- Construction of a platform capable of thoroughly storing all the tourist services offered by the different providers in a certain destination regardless their size, presence or technological capacity.
- Recording the complete descriptions about services as well as the required semantic annotation for subsequent processing.
- Creation of a social interface so that services providers are able to refresh and update those contents considered as highly valuable for its immediacy.
- Provision of the required interfaces to allow the communication between the repository and other recommendation systems by means of open standards.
- Development of the necessary mechanisms so that the platform is able to promote equally all the recorded services generating no biases of any kind.

- Elaboration of metrics focused on the platform impact and relevance in the small tourism companies' ecosystem, identifying their influence in productivity and sustainability.

The project has been endorsed by the Eureka Network with the E! Label (E! 6244) and its development will involve a consortium comprehending four different partners from two European countries. In order to achieve the objectives of the project and to fulfill the requirements of the Eureka Programme, especially those regarding the effective cooperation and collaboration among the partners, the project plan was structured into tasks, following the PSS-05-0 standard for software development projects (provided by the European Space Agency), and those tasks where grouped into work packages with different functional or methodological scope. Additionally, the consortium is currently considering the possibility of subcontracting some of the tasks in the project. Thus, the project can be considered as a GSD project and is suitable to be impacted by the framework introduced in this paper in terms of the support to the decision making regarding the allocation of the work packages and tasks of the project to the partners and subcontractors.

4   Conclusions and Future Work

This paper describes the ongoing research performed towards the definition of a framework for the allocation of work-packages among project partners in a GSD project set. The aim of the framework is to integrate sound assessment methods to common allocation factors in order to provide a set of ranked allocation recommendations aimed to fulfill partners' needs and maximize resources and effort.

This work is heading towards a three-pronged approach in terms of future work. The first is the full definition and testing of this framework. The second is the expansion of this framework to include improvement and organizational learning issues. Finally, it is aimed to integrate semantics in the solution by means of the adoption of competency ontologies and other formalisms to provide web 3.0 like machine processable content.

References

1. Ovaska, P., Rossi, M., Marttiin, P.: Architecture as a Coordination Tool in Multi-site Software Development. Software Process: Improvement and Practice 8(4), 233–247 (2003)
2. Casey, V.: Developing Trust In Virtual Software Development Teams. Journal of Theoretical and Applied Electronic Commerce Research 5(2), 41–58 (2010)
3. Herbsleb, J.D., Moitra, D.: Global software development. IEEE Software 18(2), 16–20 (2001)
4. 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)

5. Kobitzsch, W., Rombach, D., Feldmann, R.L.: Outsourcing in India. IEEE Software 18(2), 78–86 (2001)

6. Smite, D., Wohlin, C., Gorschek, T., Feldt, R.: Empirical evidence in global software engineering: a systematic review. Empirical Software Engineering 15(1), 91–118 (2010)

7. Yu, L., Mishra, A.: Risk Analysis of Global Software Development and Proposed Solutions. Automatika 51(1), 89–98 (2010)

8. Conchuir, E.O., Holmstrom-Olson, H., Agerfalk, P.J., Fitzgerald, B.: Benefits of Global Software Development: Exploring the Unexplored. Software Process Improvement and Practice 14(4), 201–212 (2009)

9. Jiménez, M., Piattini, M., Vizcaíno, A.: Challenges and Improvements in Distributed Software Development: A Systematic Review. Advances in Software Engineering, Article ID 710971 (2009)

10. Damian, D., Moitra, D.: Global software development: How far have we come? IEEE Software 23(5), 17–19 (2006)

11. Barcus, A., Montibeller, G.: Supporting the allocation of software developmentwork in distributed teams with multi-criteria decision analysis. Omega 36, 464–475 (2008)

12. Duggan, J., Byrne, J., Lyons, G.J.: A Task Allocation Optimizer for Software Construction. IEEE Software 21(3), 76–82 (2004)

13. Noll, J., Beecham, S., Richardson, I.: Global software development and collaboration: barriers and solutions. ACM Inroads 1(3), 66–78 (2010)

14. Lamersdorf, A., Münch, J.: A multi-criteria distribution model for global software development projects. Journal of the Brazilian Computer Society 16(2), 1–19 (2010)

15. Lamersdorf, A., Münch, J., Rombach, D.: Towards a multi-criteria development distribution model: An analysis of existing task distribution approaches. In: ICGSE 2008: Proceedings of the 2008 IEEE International Conference on Global Software Engineering, pp. 109–118. IEEE Computer Society, Washington, DC (2008)

16. Lamersdorf, A., Münch, J., Rombach, D.: A Decision Model for Supporting Task Allocation Processes in Global Software Development. In: Bomarius, F., Oivo, M., Jaring, P., Abrahamsson, P. (eds.) PROFES 2009. LNBIP, vol. 32, pp. 332–346. Springer, Heidelberg (2009)

17. Lamersdorf, A., Munch, J., Rombach, D.: A survey on the state of the practice in distributed software development: Criteria for task allocation. In: Fourth IEEE International Conference on Global Software Engineering, ICGSE 2009, pp. 41–50 (2009)

18. Lamersdorf, A., Münch, J.: Model-based Task Allocation in Distributed Software Development. In: Nordio, M., Joseph, M., Meyer, B., Terekhov, A. (eds.) SEAFOOD 2010. LNBIP, vol. 54, pp. 37–53. Springer, Heidelberg (2010)

19. Pereira, T.A.B., dos Santos, V.S., Ribeiro, B.L., Elias, G.: A Recommendation Framework for Allocating Global Software Teams in Software Product Line Projects. In: Proceedings of the 2nd International Workshop on Recommendation Systems for Software Engineering (2010)

20. Mohan, S., Fernandez, J.: New Opportunities presented by novel Work Breakdown techniques for Distributed Software Development. In: Proceedings of the International Conference on Global Software Engineering, pp. 305–307 (2010)

21. Setamanit, S.-o., Wakeland, W., Raffo, D.: Improving global software development project performance using simulation. In: Portland International Center for Management of Engineering and Technology, pp. 2458–2466 (2007)

22. Battin, R., Crocker, R., Kreidler, J., Subramanian, K.: Leveraging resources in global software development. IEEE Software 18(2), 70–77 (2001)

23. Baddoo, N., Hall, T.: Motivators of Software Process Improvement: an analysis of practitioners' views. The Journal of Systems and Software 62(2), 85–96 (2002)

24. Hall, T., Beecham, S., Rainer, A.: Requirements problems in twelve software companies: an empirical analysis. IEE Proceedings 149(5), 153–160 (2002)
25. Dalkey, N., Helmer, O.: The use of experts for the estimation of bombing requirements. A project Delphi experiment. The Rand Corporation, RM-727-PR (1951)
26. Dalkey, N., Helmer, O.: An experimental application of the Delphi method to the use of experts. Management Science 9(3), 458–467 (1963)
27. Blind, K., Cuhls, K., Grupp, H.: Personal attitudes in the assessment of the future of science and technology: a factor analysis approach. Technological Forecasting and Social Change 68(2), 131–149 (2001)
28. Landeta, J.: Current validity of the Delphi method in social sciences. Technological Forecasting & Social Change 73(5), 467–482 (2006)
29. Colomo-Palacios, R., García-Crespo, Á., Soto-Acosta, P., Ruano-Mayoral, M., Jiménez-López, D.: A case analysis of semantic technologies for R&D intermediation information management. International Journal of Information Management 30(5), 465–469 (2010)
30. Colomo-Palacios, R., Ruano-Mayoral, M., Soto-Acosta, P., García-Crespo, Á.: The War for Talent: Identifying competences in IT Professionals through semantics. International Journal of Sociotechnology and Knowledge Development 2(3), 26–36 (2010)
31. Colomo Palacios, R., García Crespo, A., Gómez Berbís, J.M., Casado-Lumbreras, C., Soto-Acosta, P.: SemCASS: technical competence assessment within software development teams enabled by semantics. International Journal of Social and Humanistic Computing 1(3), 232–245 (2010)
32. Colomo Palacios, R., Tovar Caro, E., García Crespo, A., Gómez Berbís, J.M.: Identifying Technical Competences of IT Professionals: The Case of Software Engineers. International Journal of Human Capital and Information Technology Professionals 1(1), 31–43 (2010)
33. Colomo Palacios, R., Ruano Mayoral, M., Gómez Berbís, J.M., García Crespo, A.: Semantic Competence Pull: A Semantics-Based Architecture for Filling Competency Gaps in Organizations. In: García, R. (ed.) Semantic Web for Business: Cases and Applications. IGI Global, Hershey (2008)
34. Casey, V.: Imparting the importance of culture to global software development. ACM Inroads 1(3), 51–57 (2010)
35. Shih, C.C., Huang, S.J.: Exploring the relationship between organizational culture and software process improvement deployment. Information & Management 47(5-6), 271–281 (2010)
36. Jaakola, H., Heimburger, A., Linna, P.: Knowledge-oriented software engineering process in a multi-cultural context. Software Quality Journal 18(2), 299–319 (2010)
37. Hahn, E.D., Bunyaratavej, K.: Services cultural alignment in offshoring: The impact of cultural dimensions on offshoring location choices. Journal of Operations Management 28(3), 186 (2010)
38. Aramo-Immonen, H., Jaakkola, H., Keto, H.: Multicultural Software Development: The Productivity Perspective. International Journal of Information Technology Project Management 2(1), 19–36 (2011)
39. Casado-Lumbreras, C., Colomo-Palacios, R., Soto-Acosta, P., Misra, S.: Culture dimensions in software development industry: The effects of mentoring. Scientific Research and Essays 6(11), 2403–2412 (2011)
40. Hofstede, G.H.: Culture's Consequences: International Differences in Work-Related Values. Sage Publications, Newbury Park (1984)
41. Hofstede, G.H.: Culture's Consequences: Comparing Values, Behaviours, Institutions, and Organisations, 2nd edn. Sage Publications, Thousand Oaks (2001)
42. Hofstede, G.H.: Cultures and Organizations: Software of the Mind. McGraw-Hill (2010)