Model-Based Task Allocation in Distributed Software Development

Ansgar Lamersdorf\(^1\) and Jürgen Münch\(^{1,2}\)

\(^1\) University of Kaiserslautern
\(^2\) Fraunhofer IESE

a_lamers@informatik.uni-kl.de, Juergen.Muench@iese.fraunhofer.de

Abstract. Task allocation is one central aspect in planning and managing global software development projects. To date, several models that support task allocation have been proposed, including cost models and risk-based approaches. However, systematic integration of such models and a guiding process for task allocation activities is widely missing. In this article, we integrate existing models that reflect different viewpoints and abstraction levels of task allocation decisions. Based on the integrated approach, we sketch a process for systematic evaluation and selection of task assignments that defines the model interfaces and the sequential order of their use. In detail, the approach presented here integrates a risk model that is able to identify the possible risks for each assignment individually, an optimization model that uses Bayesian networks to suggest assignment alternatives with respect to multiple criteria, and an effort overhead model that is able to estimate the project effort for each assignment alternative. All three models are significantly grounded in empirical studies. Besides the introduction of all three models and the description of the process, the article provides an example application, sketches related work, and presents an overview of future work.

1 Introduction

Every software development project – be it in offshoring or outsourcing – that involves work from different, distributed sites requires a decision about task allocation: How can the individual tasks that together constitute the development project be distributed across the available sites?

There are many different ways to allocate tasks to different sites, especially with respect to the division of work into different tasks and with respect to criteria for deciding where tasks will be enacted: The work can be divided by process steps, along architectural components, by functionality, or using combinations of these models [1] [2]. In an extensive empirical study on the practice of distributed development, we identified three main criteria for task allocation: labor cost rates, availability of people, and expertise [3]. The cost rate, in particular, is often reported as a major criterion for task allocation: In global development, work is mainly assigned to those sites that have the lowest labor cost rates [4] [5].
However, despite the differences, we observed that, in general, the assignment of work is done rather unsystematically in practice, without any specific process or defined decision procedure [3]. The criteria for work allocation are not defined systematically and are often just applied by the responsible project manager based on personal experiences. This highly increases the risks of distributed development projects, as there are many problems in global software development that are influenced by the task allocation decision: Language, cultural, and time zone differences are just some examples of barriers between sites that may cause problems such as decreased productivity [6] and increased lack of trust [7]. Assigning work to sites with low differences reduces these barriers and the resulting problems (often referred to as “nearshoring” [8]). Other problems such as high turnover rates [9] and little knowledge of the customer or application domain are also influenced by assigning work to different sites. Together with the already mentioned criteria for work allocation (labor cost rates, availability, expertise) this shows that the task allocation decision is highly complex and should take into account multiple criteria and influencing factors in order to reduce or avoid the problems immanent in distributed and global software development (GSD).

There are different perspectives under which a task allocation decision could be regarded: From a risk management perspective, there are various risks specific to GSD that are influenced by the work assignment (e.g., assigning work to sites with large cultural differences increases the risk of mistrust between sites). Thus, any given task allocation alternative can be evaluated under the perspective of the potential risks it creates. From the perspective of a project planner, one specific task allocation has to be selected from the (potentially large) pool of alternatives, taking into account multiple, sometimes conflicting, goals and influencing factors. Finally, the cost perspective requires special attention in GSD, as low labor costs are one driving factor for global development. However, since the distribution of work has an impact on both labor cost rates and productivity, any task allocation alternative should also be evaluated under the perspective of effort and cost estimation.

While all three perspectives are clearly different, they all regard the same decision and phenomena. Thus, they underlying influencing factors and causal relationships are common to all perspectives. In this article, we present an approach for systematic task allocation that includes models for all three of these perspectives and integrates them into one process for evaluating and selecting task allocation alternatives. The individual models have already been published separately [10], [11], [12], [13] but have not yet been integrated into one decision process as presented here.

The remainder of this article is structured as follows: Section 2 gives an overview of related work in task allocation decision support from different perspectives. Section 3 explains the task allocation process by briefly introducing the three sub-models and then presenting the process that integrates all models into one task allocation decision. Section 4 gives an example of how the process can be applied to an industrial context. Finally, Section 5 concludes the article and gives an overview of future work.

2 Related Work

Task allocation in global software development has been the focus of several research approaches and has already been analyzed in detail in another publication