Information Sciences 264 (2014) 230–259

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Using agents to manage Socio-Technical Congruence in a Global Software Engineering project

Javier Portillo-Rodríguez [a,*], Aurora Vizcaíno [a], Mario Piattini [a], Sarah Beecham [b]

[a] Alarcos Research Group (University of Castilla-La Mancha), Paseo de la Universidad, 4 13071 Ciudad Real, Spain
[b] Lero-The Irish Software Engineering Research Centre, University of Limerick, Ireland

ARTICLE INFO

Article history:
Received 17 June 2012
Received in revised form 13 October 2013
Accepted 3 January 2014
Available online 11 January 2014

Keywords:
Global Software Engineering
Socio-Technical Congruence
Software agents

ABSTRACT

In Global Software Engineering (GSE) there are a number of communication and coordination challenges which are brought about by the factor of distance. Measuring Socio-Technical Congruence (STC) is, however, presented as a suitable technique for helping to resolve those issues. This leads us to believe that applying STC measurements to GSE might be beneficial, improving communication and coordination. However, after studying existing tools that use STC measurements, we detected some gaps, both in the way they measure STC and in the features offered by the tools for the GSE environment. That is why we have designed an Agent Architecture for coordination and communication that aims to fill the gaps found in the current tools and includes features adapted to GSE characteristics. This is achieved by taking advantage of the special features that agents offer. Moreover, this proposal has been validated in a case study performed at Indra Software Labs, a global software development company. Results show that, by using our proposal, it is possible to improve coordination and communication in a distributed environment.

1. Introduction

Modern software engineering, and more specifically Global Software Engineering (GSE), which takes place in globally dispersed teams, presents specific advantages and challenges to the software industry [21]. Working in a globally-distributed team is fast becoming the norm for many software engineering organizations, as they seek to attract new customers, strive to be nearer to their existing customers, broaden their skills base and reduce their labor costs. Realizing these benefits, however, goes hand in hand with additional challenges such as increased complexity and communication overheads [12,28].

In Global Software Engineering, coordination becomes more difficult; this is due to the problems derived from distance (i.e. different working times, different understandings or difficulties in communicating efficiently). A key challenge for GSE companies is to coordinate and allocate tasks, and ensure that important information is successfully communicated to all partners [6,10,5,24]. Despite this need, there is little help available for GSE managers to assess their organization structure and to decide which coordination and communication processes need to be strengthened. If organizations cannot assess how well their communication and coordination strategies are working, it will be extremely difficult to pinpoint where changes are needed. An independent measure of how people coordinate their efforts could help to solve this problem, allowing GSE organizations to leverage the benefits of GSE and increase productivity and quality [33].

* Corresponding author. Tel.: +34 645921751.
E-mail addresses: javier.portillo@uclm.es (J. Portillo-Rodríguez), aurora.vizcaino@uclm.es (A. Vizcaíno), mario.piattini@uclm.es (M. Piattini), sarah.beecham@lero.ie (S. Beecham).

0020-0255/$ - see front matter © 2014 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2014.01.009

One technique designed to measure coordination is that of Socio-Technical Congruence [8,23]. The main idea of Socio-Technical Congruence is based on Conway's Law [9], which says that the structure of a software product reflects the physical layout of the development organization. Based on this law, Socio-Technical Congruence (STC) can be defined as a measure or a technique that assesses the "fit" between the structure of a software system and the structure of the organization that develops it [30] or "an intuitive way to compare required coordination effort within a software development project with the actual ongoing coordination" [23].

There is now a growing effort to demonstrate empirically the benefits of gaining a good level of congruence between coordination requirements and the coordination activities actually performed, where, in theory, a high level of congruence should relate to improvements in productivity [7] and quality [22]. However, a high level of congruence can also be associated with increased risks and costs, where too many interactions can overload and overwhelm team members [4]. This means that there is clearly a need to balance the number of interactions and time spent on actual development tasks.

On the other hand, there are tools available to help to measure STC and STC-related aspects and achieve a suitable socio-technical balance. Tools such as Ariadne [29] or Tesseract [26] have been implemented, but the majority of STC tools have been designed to visualize coordination relationships and coordination gaps (lack of coordination interactions where coordination is needed) but they do not help the organization to maintain a good level of STC. Moreover, they use STC measurements that are based on collocated development, and because of this, they are not suited to a Global Software Project.

In this paper we present an Agent Architecture that aims to manage (measure, control and maintain) the Socio-Technical Congruence, in order to improve coordination and communication in a Global Software Project. We introduce a new approach with which to measure STC, by combining current STC measurements and agent properties, such as autonomy, reactivity, pro-activity and social ability. These properties can be used to perform tasks like monitoring, reporting and measuring Socio-Technical Congruence levels. In summary, agents could provide a much-needed systematic approach that can be incorporated and used to improve current STC tools. Focusing on agent architecture and STC, therefore, we aim to answer the following research question:

RQ1: Can software agents be used to measure Socio-Technical Congruence in Global Projects?
RQ2: How can agents improve current Socio-Technical Congruence tools to make them more useful for a Global Development Project?

To answer these questions we present how, SCT can be measured, by using the agents' properties. We also demonstrate how, by using the proposed Agent Architecture, agents are able to perform tasks to calculate, control and maintain STC, improving the current tools and extending their usage to global settings. This proposal has been validated in a case study performed at the Indra Software Labs, showing how it is possible to improve coordination and communication by using our architecture, which combines agent technology with STC measurement.

This paper is structured as follows: In the next section, Section 2, central terms such as Congruence, Coordination, Communication (important aspects in GSE), as well as the relationships between them, are introduced. In Section 3, the main benefits and risks of measuring and maintaining a high level of STC are presented. Section 4 presents a review of current approaches to measuring Socio-Technical Congruence, with a look at some of the existing tools that measure STC or STC-related aspects. In Section 5 we present our proposal (an Agent Architecture) for managing STC and improving the existing tools in a global setting. We also explain, in Section 5, why agents might be an appropriate technology with which to measure STC in GSE. Section 6 describes the results obtained in the case study performed. In Section 7, the main limitations found and assumptions made in our proposal are discussed. Finally, in Section 8, we outline the conclusions obtained and lines for future research.

2. Congruence, coordination and communication

To clarify why measuring Socio-Technical Congruence is related to the improvement of coordination and communication, in this section we define the terms *congruence*, *coordination* and *communication* and how they inter-relate.

Congruence has several definitions in literature, but they all refer to "the match between a particular organization design and the organization's ability to carry out a task" [8]. We have taken the definition given in [23], where it is defined as "an intuitive way to compare required coordination effort within a software development project with the actual ongoing coordination" or "the fit between an organization's coordination requirements and an organization's social interactions".

From this definition, we can already draw out the main relationship between coordination and Socio-Technical Congruence (STC). Basically, what STC measures is the alignment between **coordination needs** (extracted from technical dependencies) and the **real coordination** performed (drawn from socio-technical aspects). This means that to measure STC it is necessary, first of all, to obtain the coordination needs and then check the coordination that the team members are actually performing.

Coordination needs (represented in Fig. 1) can be obtained in different ways, but they are usually found by combining technical dependencies between Technical Entities and dependencies between Technical Entities and stakeholders of the project.