Information and Software Technology 72 (2016) 137–150

Contents lists available at ScienceDirect

Information and Software Technology

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

Towards pragmatic interoperability to support collaboration: A systematic review and mapping of the literature

Frâncila Weidt Neiva*, José Maria N. David, Regina Braga, Fernanda Campos

Federal University of Juiz de Fora, Minas Gerais, Brazil

ARTICLE INFO

Article history:
Received 16 July 2015
Revised 23 December 2015
Accepted 29 December 2015
Available online 8 January 2016

Keywords:
Pragmatic interoperability
Collaboration
Collaborative systems
Groupware
Interoperability

ABSTRACT

Context: Many researchers have argued that providing interoperability support only considering the format and meaning (i.e. syntax and semantic) of data exchange is not enough to achieve complete, effective and meaningful collaboration. Pragmatic interoperability has been highlighted as a key requirement to enhance collaboration. However, fulfilling this requirement is not a trivial task and there is a lack of works discussing solutions to achieve this level of interoperability.

Objectives: The aim of this study is to present a systematic review and mapping of the literature in order to identify, analyse and classify the published solutions to achieve pragmatic interoperability.

Method: To conduct a systematic review and mapping in accordance with the guidelines proposed in the evidence-based software engineering literature.

Results: Our study identified 13 papers reporting pragmatic interoperability computational solutions. The first paper in our set of selected papers was published in 2004; the main strategies used to address pragmatic interoperability issues were service discovery, composition and/or selection and ontologies. The application domain of the identified solutions was mainly e-business. In addition, most of the identified solutions were software architectures.

Conclusion: Mature proposals addressing pragmatic interoperability are still rare in the literature. Although many works have discussed the importance of pragmatic interoperability, it is necessary that researchers report solutions that implement and evaluate pragmatic interoperability in order to make progress in this area.

1. Introduction

Software plays an important role in society. The dissemination of software use for different purposes and our dependency on it to perform daily tasks have generated a need for solutions to increase productivity and quality and to reduce costs in development processes. In order to fulfill these needs, companies geographically distributed their activities. However, this scenario has brought new challenges because collaborative systems have complex requirements which are not easy to fulfill. This complexity is partially related to the diversity of interactions and processes types that these systems support, imposing the need for flexible rules and policies. For example, suppose that a discussion forum is used to support decision making in a meeting. In this scenario, at the moment when decisions are made, roles can be modified and new activities and policies need to be redefined. Besides these challenges, the dynamism of the components used to represent the interaction context also increases the complexity of the collaborative systems. When a decision is made, modifications in the context occur during the interaction, generating additional requirements and interest objects for the participants [1].

In order to support collaborative systems analysis, Ellis et al. [2] established the 3C collaboration model. This model defines collaboration as the combination of three dimensions: (i) Communication—related with the message exchanges generated from users interactions; (ii) Cooperation—management of people, their activities and resources and; (iii) Coordination—activities performed in a shared workspace.

A study conducted by Steinmacher [3] revealed that tools to support communication dimension during collaboration in a distributed software development (DSD) environment were very poorly explored. Furthermore, communication issues are usually identified as one of the main difficulties in a distributed

* Corresponding author. Tel.: +5532991398885.
E-mail addresses: francila.weidt.neiva@ice.ufjf.br, fran.weidt@gmail.com (F.W. Neiva), jose.david@ufjf.edu.br (J.M.N. David), regina.braga@ufjf.edu.br (R. Braga), fernanda.campos@ufjf.edu.br (F. Campos).

http://dx.doi.org/10.1016/j.infsof.2015.12.013
0950-5849/

environment, which can be considered a gap and a good opportunity for research.

The communication process is explored in semiotics, the study of meaning-making, the philosophical theory of signs and symbols in linguistics field. Semiotics is divided into three dimensions: syntactic, semantics and pragmatics. Syntax acts as a sign, semantics is related with which the sign refers to, and pragmatics is related to the effect of the sign on the interpreter. This effect can be noticed depending on the context where the sign is used [4]. These three aspects are important in the communication process.

Using semiotics, we can also explain a key requirement that arises with the DSD scenario, interoperability across collaborative systems. In DSD, the systems are distributed and they need to communicate accordingly. In other words, they need to interoperate in order to reach specified goals. Therefore, interoperability can be considered as a fundamental requirement to enable collaboration in this context.

In this research context, interoperability can be defined as the ability of different systems to use each other's services effectively [5]. These systems can share functionalities and information at different levels. Tolk and Muguira [6] proposed a formal definition of interoperability levels through a framework named LCIM (Levels of Conceptual Interoperability Model). This framework was designed to determine interoperability in the early phases of software engineering process, considering conceptual models. Based on LCIM definitions, in a syntactic level the data that are exchanged act as a sign and, to achieve this interoperability level, the sign syntax must be previously established as a standard. Semantic interoperability is concerned with ensuring that the meaning of the data, in other words, which the data refers to, is shared in an unambiguous way. Finally, pragmatic interoperability is concerned with ensuring that the message sender and receiver share the same expectations about the effect of the exchanged messages and the context where this exchange occurs plays an important role [6]. LCIM defines another three interoperability levels higher than pragmatic interoperability and conceptual level is the highest one. However, in this paper we focused on pragmatic interoperability.

Many researchers have argued that providing interoperability support only considering the format and meaning (i.e. syntax and semantic) in data exchange is not enough to achieve complete, effective and meaningful collaboration [7,8]. In this vein, pragmatic interoperability support has been seen as a key requirement to meet the desired effects during message exchange and different authors have discussed this, including [9,10]. The authors focused on pragmatic interoperability because, although the need to agree on the format and meaning, pragmatic aspects as the use and the context of this message are also important [5]. The authors discussed that the desired effects can be realized and then addressed by considering the pragmatic aspects that involve the message exchange.

Interoperability across collaborative systems is not a trivial requirement. Each system has its own particularities and needs. As a result, each system requires that different interoperability levels be achieved. For example, suppose a collaborative system designed to support experiments in biology. If a scientist which is working in this system wants to download an artifact 'A' from a collaborative system instance 'B' and he/she is situated in a collaborative system instance 'C', he/she needs to use only the syntactic interoperability services because this scenario only requires information related with data syntax/format to perform the interoperation process. However, sometimes scientists want to discover new services and compose them to solve a certain problem. In general, these services are geographically dispersed and developed by scientists from different institutions. The scientists do not have previous information about most of these services, once they did not develop them or use them before. Despite this, the services have to fulfill scientists' requirements in order to solve their problems properly. In this case, syntactic interoperability is not enough. In order to do this, it is important to consider the context where these services are inserted, the constraints about their use, the concepts that they represent, and the nonfunctional requirements that they have to fulfill, among others. In this scenario, providing pragmatic interoperability support can improve collaborative activities among the scientists taking into account tacit and non-formalized context factors in collaborative systems, such as components reliability, which scientist used which component and for what purpose. These context factors provide subsidies to realize the services that cause the desired effects and meet the scientist expectations.

Another example, suppose a scenario where services must interoperate. Syntactic level covers data exchange, if service A sends a string, service B has to expect a string. But, what this string is about? At the semantic level this question could be answered. If service A sends a string that represents a DNA sequence and service B expected a RNA sequence, there is no semantic interoperability. But, even if service A sent a DNA sequence while service B expected the same, is service B the best service? Do they meet the expectations? We answered these questions based on the pragmatic aspects as users context, service context, business rules, polices, restrictions, among others. In this way, imagine that service A sends DNA sequences to service B expecting that service B performs a specific task (e.g. DNA alignment task) using a specific method (e.g. local alignment method). However, service B uses a different method than expected by service A (e.g. global alignment method) to perform the specific task (e.g. DNA alignment task). In this case, we can say that the pragmatics aspects are misaligned and pragmatic interoperability is not possible among the services. Considering this scenario, probably service B will not cause the desired effects during the interoperation process. In other words, service B will not meet the expectations and it is not the best service to perform the desired task.

It is important to state that using pragmatic aspects to support service discovery and selection is just a mechanism that aims to ensure pragmatic interoperability. There are others, such as Base Object Models [11]. Imagine a scientist that developed a set of services, he/she probably know which services can interoperate in a pragmatic level, because he/she understands the context of these services and which of them are more suitable to be composed and perform a certain task. Now, imagine a scientist that wants to use services that he/she does not developed, services that he/she is not aware about the involved context. Considering this context, pragmatic interoperability is about to establish mechanisms to automate message exchange among services by considering the expectations about the effect of this exchange and the involved context.

The problem is that although pragmatic interoperability support has been mentioned as a key issue in literature, this field is still in its infancy as can be seen in [7]. This study found at least 44 unique pragmatic interoperability definitions. Besides revealing the lack of consensus, the authors found that each definition is associated with a particular research domain.

In this way, it is important to carry out research in order to investigate pragmatic interoperability as this is a fundamental requirement to enhance collaboration in the DSD domain. As previously state, a great number of studies pointed the importance of pragmatic interoperability achievement [9,10]. However, the knowledge about which and how pragmatic interoperability solutions were proposed is scarce and scattered in literature.

To the best of our knowledge, there is no systematic review and/or mapping which maps and analyses the existing proposed solutions in literature in order to provide support to pragmatic interoperability. In contrast to the usual process of literature review, a systematic review and mapping are designed to reduce bias and provide a reliable picture of the current state of the art of a specific