



A validated ontology for global software development



Aurora Vizcaíno ^{a,*}, Felix García ^a, Mario Piattini ^a, Sarah Beecham ^b

^a Alarcos Research Group, Institute of Information Technologies and Systems, Escuela Superior de Informática, University of Castilla-La Mancha, Spain

^b Lero, The Irish Software Research Centre, Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland

ARTICLE INFO

Article history:

Received 31 March 2015

Received in revised form 11 December 2015

Accepted 2 February 2016

Available online 10 February 2016

Keywords:

Global software development (GSD)

Global software engineering (GSE)

Distributed software development (DSD)

Ontology

Meta-model

ABSTRACT

The global software development (GSD) paradigm has, over the last 15 fifteen years, shifted from being novel and ground breaking to being widely adopted and mainstream. This wide adoption is partly owing to the many benefits provided by GSD, such as reduced labour costs, proximity to new markets and access to a diverse and experienced skills pool. Yet taking advantage of these benefits is far from straightforward, and research literature now includes a proliferation of guidelines, reviews and models to support the GSD industry. Although this active area of study is firmly established as a research area in its own right, the boundaries between general software engineering and GSD are somewhat confused and poorly defined. In an effort to consolidate our understanding of GSD, we have developed an ontology in order to capture the most relevant terms, concepts and relationships related to the goals, barriers and features of GSD projects. The study we present here builds on research conducted in a collaboration project between industry and academia, in which we developed an ontology in order to provide practitioners with a “common language and conceptualisation”. Its successful outcome encouraged us to create a broader ontology that captures the current trends in GSD literature. The key ontology, along with its three subontologies, are the result of a review of the relevant literature, together with several expert evaluations. This ontology can serve as a useful introduction to GSD for researchers who are new to the paradigm. Moreover, practitioners can take advantage of it in order to contextualise their projects and predict and detect possible barriers. What is more, using a common language will help both researchers and practitioners to avoid ambiguities and misunderstanding.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

There is currently a strong and growing trend that involves IT organisations employing multisite teams that span geographic boundaries to build their software. Organisations are adopting a Global Software Development (GSD) paradigm, either out of necessity or owing to the promise of competitive benefits. GSD can take various forms, and the proliferation of literature reviews on the subject reflects this active area of research and practise. Indeed, there are now even two tertiary research studies [1,2] that capture the groups of secondary literature reviews. Since GSD has now become a dominant paradigm, and thanks to the proliferation of research that the community can draw on, the research community now needs a common understanding of the terms and concepts related to GSD projects.

One way in which to achieve this goal is to create a GSD ontology. An ontology is defined by Gruber [3] as the formal specification of a shared conceptualisation. Ontologies provide a visual means to share a common understanding of the structure of information among people or software agents. Moreover, ontologies enable the reuse of domain knowledge, thus making domain assumptions explicit and helping us

to clarify any ambiguities [3,4]. What is more, and as is illustrated by the successful applications of ontologies in the Software Engineering field, an ontology can provide numerous benefits, from serving as a basis for discussion to achieving the necessary agreement and consensus and contributing to the harmonisation of standards in the field, as a vehicle for achieving the interoperability required between the ever-increasing number of groups and organisations working on languages and tools, or to support model driven engineering initiatives (see [5,6,7,8]).

Bearing in mind the aforementioned issues, our main research objective has therefore been to build an ontology with which to determine the advantages, challenges and concepts related to GSD projects. This ontology is based on the relevant related GSD ontologies that exist in the field and has been agreed on by experts in the field.

The remainder of this paper is organised as follows. The background to this research is presented in the following section, while the research method followed to build the GSD ontology is described in Section 3. In Section 4, we focus on describing the systematic mapping carried out to discover the ontologies and taxonomies exist in the literature related to GSD, and Section 5 provides a description of the process carried out to discover new terms and relations and the results obtained. An evaluation of the first version of the ontology by experts is presented in Section 6 and the refined result obtained (GSD ontology v2) is then described in

* Corresponding author at: Escuela Superior de Informática, Paseo de la Universidad 4, 13071 Ciudad Real, Spain.

Section 7. In Section 8, we present the validation of GSD ontology v2. Finally, the conclusions, limitations and guidelines for using the ontology, along with an outline of future work, are presented in Section 9.

2. Background

There is currently an important interest in developing ontologies and using them in the software engineering field. A proof of this fact is the increasing number of workshops focused on this, such as the workshop on ontology, conceptualisation and epistemology for software and systems engineering (ONTOSE), the workshop on ontologies and meta-modeling software and data engineering (WOMSDE) or the workshop on semantic web-enabled software engineering (SWESE). These and other efforts contribute towards ontologies attaining more maturity in the field as an engineering discipline.

Various works of interest exist in the field of Software Engineering. As a representative example of existing works, Wongthongtham et al. [9] described a software engineering ontology model based on the software engineering textbook [10] and SWEBOK with the aim of enabling communication between software engineers according to common software engineering knowledge.

With regard to GSD-related conceptualizations, Wongthongtham et al. [11] defined five ontologies to be used in a multisite software engineering environment: a business domain ontology to characterise the fundamental knowledge about a particular domain, since all software is designed to solve a business need such as accounting or a customer service etc.; a software engineering ontology in which software engineering principles and aspects are described; a project management ontology, created to enable all the stakeholders to have consistent knowledge when discussing project-related matters; an issues ontology, divided into ontological, technical and managerial issues; and the solution ontology, which is related to the knowledge of issues and solutions that drive the success of the project and product. These ontologies are developed to serve intelligent software agents, which support multisite software development. Wongthongtham et al. [12] proposed a software engineering sub-ontology to enable remote team members to browse, search for and share data in a distributed software engineering project environment. One of their main goals is therefore to enable communication between computer systems. The authors thus provide two subontologies, a generic ontology and an application-specific ontology. The generic ontology is a set of software engineering terms, including the vocabulary, the semantic interconnections and some simple rules of inference and logic for software development. It provides the vocabulary for the terms in software engineering, along with an application-specific ontology, which is an explicit specification of object-oriented development in software engineering. The approaches of both ontologies help to transform explicit knowledge into conceptual knowledge representation with the aim of using software agents to access data from this project ontology repository. In our previous research [13], we developed the O-GSD ontology with the objective of promoting a common understanding of GSD vocabulary based on a particular project called ORIGIN. This ontology achieved its purpose by facilitating communication between members of the ORIGIN project team and by helping members of the team to avoid misunderstandings when working in groups and to share a similar view of the project.

The analysis of relevant literature led to the discovery of certain limitations, particularly the lack of relevant terms that are specific to GSD projects or the need to reinforce the external validity of ontologies developed in the context of a single GSD project, such as in our previous research [13]. In addition, the benefits that the use of the O-GSD ontology [13] allowed us to obtain in this project encouraged us to consider the creation of a more general ontology for GSD projects during which the experts took will take part in the process of creating and evaluating it, since one important feature of an ontology is that of representing the consensus of the experts in a particular domain.

3. Research method for the creation of the GSD ontology

In this section, we present the research method followed to develop our validated GSD ontology. The general question used for this research was Is it possible to create an ontology with which to determine the advantages, challenges and concepts related to GSD projects? This research question was addressed by the following method shown below (Fig. 1).

As shown in Fig. 1, the GSD ontology was built by following three fundamental steps:

- 1) We augmented the initial version of the ontology (O-GSD) by means of the following:
 - o A rigorous analysis of GSD concepts and relationships found in the relevant literature. These concepts were found through the use of a systematic mapping of ontologies and/or taxonomies related to GSD and by searching for additional concepts in proceedings of the International Conference on Global Software Engineering (ICGSE) 2011, 2012 2013, 2014, as this is the most important international conference in this domain.
 - o Building on our previous O-GSD ontology. The concepts described in the O-GSD were also considered as a starting point for the GSD ontology, given that the building of this ontology was also based on the analysis of previous relevant literature and on a consensus between researchers and practitioners involved in the ORIGIN project. The aim was to discover the terms and relationships which were not specific to the project and that could be generalised to any GSD context, and focused mainly on the goals, barriers and features of a GSD project.
- 2) A survey was developed to serve as both a validation instrument and a means to review the ontology draught produced. As a result of this step, we gained very useful feedback which helped us to create the GSD ontology presented herein.
- 3) The new ontology was validated by experts by means of a survey in which Gruber's criteria for evaluating ontologies [3] were followed.

The first step is described in more detail in Sections 4 and 5. The evaluation of this ontology (draught) and the result GSD ontology are presented in Sections 6 and 7, respectively. Finally, the third step is described in Section 8.

4. GSD ontologies and taxonomies: a systematic mapping

In this section, we describe the protocol used to carry out the systematic mapping review. We first describe the design and then go on to present the results obtained.

4.1.1. Planning and design

One of the principal goals of ontologies is to formalise a domain in a way that reflects the consensus of the majority of the members of the given community. As this formalisation should promote a shared and common understanding of the domain, the first step was to explore what the other ontologies or taxonomies in GSD are proposing. This would then allow us to detect what topics are considered relevant by other authors and to determine whether these terms should be added to the ontology that we wished to extend. Furthermore, other researchers' previous work would help us discover the different views or focuses of GSD that we had not taken into account in our initial ontology (O-GSD). In summary, we analysed previous ontologies and taxonomies concerning GSD in order to discover new approaches or views.

- Detect new terms.
- Take advantage of other research efforts in which themes have been

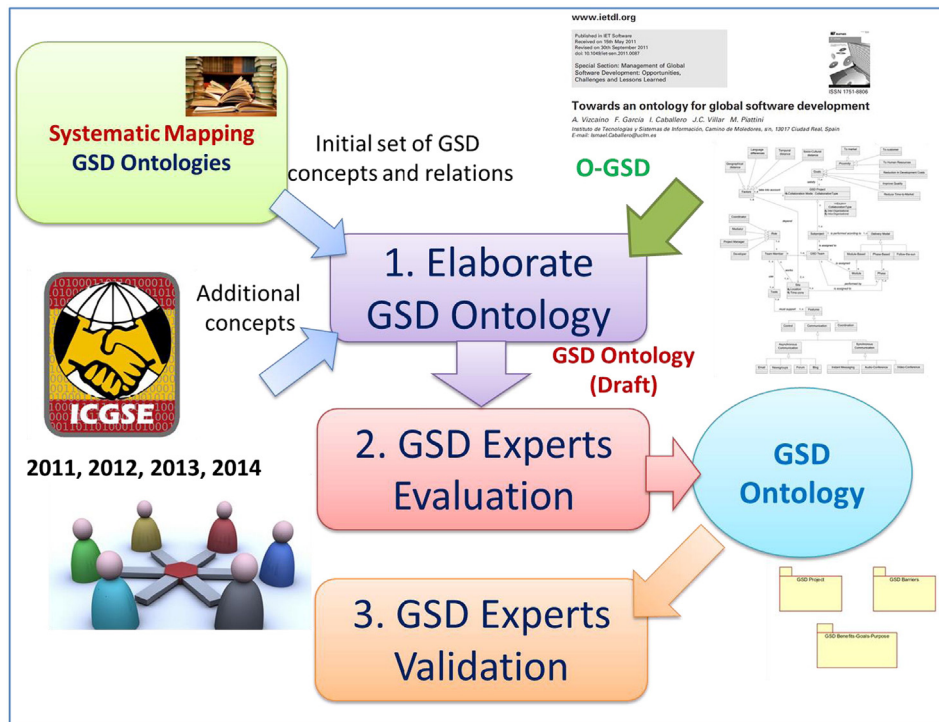


Fig. 1. Research method used to build a GSD ontology.

synthesised.

- Reach a consensus, since other ontologies reflect a corpus of work.

Our approach therefore consisted of an attempt to align previous ontologies or terms so as to create a more complete ontology.

In order to investigate which ontologies and taxonomies relate to GSD, we carried out a systematic mapping literature review based on Kitchenham and Charters [14], as the goal of a mapping study is to provide a general view of a research area which perfectly matches the objective.

The research question for the mapping study was What ontologies and taxonomies related to GSD exist. Our literature review was therefore guided by three research sub-questions:

1. RQ1: Which existing ontologies tackle GSD in general?
2. RQ2: Which existing ontologies describe one or more GSD processes?
3. RQ3: Which existing taxonomies describe terms related to GSD?

The following search string was built to fulfil our purpose:

((Ontology OR Metamodel OR "meta model" OR terminology OR taxonomy) AND ("Global Software Development" OR "Distributed Software Development" OR "Global Software Engineering"))

Our search string included the term "metamodel (or 'meta model')", as some authors use it as a synonym for ontology [15].

The study inclusion and exclusion criteria are shown in Table 1. As is shown in the table, the selection criteria are based mainly on papers which describe ontologies or related conceptualisations (taxonomies, etc.). Many papers in the related GSD literature make use of ontologies, such as those papers that describe GSD tools or methods [16–18]. However, if the scope of the supporting ontology focuses on explaining a GSD process or sub-process, the paper is not considered relevant and is therefore excluded. The entire contents of the papers were searched.

The databases chosen for the search were as follows. The number of papers we sourced in these databases before applying inclusion/exclusion criteria are shown in brackets:

- ACM Digital Library (2) (only selected journal articles).
- IEEE Digital Library (35) (only selected journal articles).
- Scopus (216) (everything selected—journals, conferences, books).

In the case of Scopus, papers from both conferences and books were also included, since this database contains the most important conferences on the topic, such as the International Conference on Global

Table 1
Inclusion and exclusion criteria of the systematic mapping.

Inclusion criteria	Exclusion criteria
1. The study addresses one or more of our research questions 1.1, 1.2 and 1.3	1. Those papers that are not focused on GSD or on a distributed software development (DSD) domain.
2. The study presents an ontology (as a primary study sourced through either our research string in the various electronic databases, or via snowballing [19]).	2. Those papers which describe the usage of ontology/ontologies, but whose scope is not to describe a GSD or DSD process itself.
3. The study is unique, rather than a duplicated study. When several papers are written by the same authors describing the same ontology but with some modifications, then the most complete and comprehensive will be considered.	3. Repeated papers. The first instance found will be considered.
4. Papers published since 2010, such as [13] in which the authors have already set out the ontologies related to GSD that existed up to and including 2010 in the "previous work" section.	
5. Papers written in English	

Table 2
Digital libraries chosen.

Digital library	No. papers extracted	Observations	No. papers accepted
ACM	2	Only selected journals	0
IEEE	35	Only selected journals	0
Scopus	216	Everything selected: journals, conferences and books.	6

Software Engineering (ICGSE) and the International Conference on Software Engineering (ICSE). Although 216 papers were found, only 6 were eventually included after applying our inclusion and exclusion criteria (see Table 2 and Appendix A). The papers accepted provided an overview of the domain and helped us to detect candidate terms for possible inclusion in the new ontology.

The second step that we followed replicated the process used when creating the previous O-GSD ontology [13], which involved a further search of the literature. In this case, we sourced all systematic literature reviews (SLRs) published in the area, as along with all papers published in the last four editions of ICGSE conferences (proceedings from 2011 to 2014). We chose to search for more terms in SLRs because SLRs can be considered to be a synthesis of what has recently taken place in a domain [14]. Reviewing the recent SLRs in GSD could therefore help us to discover current trends or concepts that might not have been included in other ontologies, including our own. For the same reason, we also searched in the last four ICGSE conferences, since this is the principle conference regarding the topic. We assumed that any finer grained terms not included in the SLRs would be picked up in our detailed searches of the ICGSE proceedings. Our objective when carrying out this exhaustive search and analysis was to ensure that we did not miss any important concept.

4.1.2. Results of the systematic mapping

As a result of the systematic mapping, six papers were found to meet our criteria. The main findings are summarised as follows (see the references of these papers in Appendix A):

- SL1: This study describes our O-GSD ontology. It was developed to provide a common language for all the members of a particular project, called ORIGIN. This ontology has been used as the starting point for the research presented in this paper.
- SL2: The authors provide an accurate terminology and definition for different global sourcing situations, thus defining a taxonomy for Global Software Engineering (GSE). The results of this study have been considered to a great extent during the development of the proposed ontology, as the terminology is the result of a consensus among several experts in the GSD domain.
- SL3: The authors created a meta-model in order to provide a common understanding of architectural knowledge management in a global software development context.
- SL4: A systematic mapping study on ontologies supporting distributed software development (DSD) is described. The authors considered a source to be any ontology which describes tools, techniques, models and best practises in DSD, in order to include open source development. They found 4 relevant papers: [20,21,22,23] and then used the snowballing technique [19] to analyse these papers as they might contribute new terms to a GSD ontology, namely:
 - o SL4.1. Mirbel proposes an ontology to support knowledge management in open source development communities. The author unites other published ontologies on open source to form a combined community of practise view. Mirbel's ontology emphasises the sharing dimensions in knowledge management services.
 - o SL4.2. The authors propose an ontology-based framework for knowledge sharing in distributed software development teams.

They focus on structural concepts in distributed environments, such as the reuse of components and semantic error handling.

- o SL4.3. Dillon and Simmons develop an ontology-based software development architecture for open source software development projects. The focus is on the architecture developed (not the ontology) with the objective of enabling a better categorisation of information, communication, coordination and development of sophisticated search agents.
- o SL4.4. The authors focus on modelling the semantic understanding in GSD communication in order to minimise ambiguity. Their derived ontology (OntoDiSEN v1) aims to support communication among geographically dispersed team members in a homogeneous way. They do this by integrating contextual information that notifies team members about actions occurring in their shared workspace and informs them about how that action might influence their work.
- SL5: The authors propose a meta-model that is broken down into six meta-models units: deployment, process, data, communication, tool and migration. Although the goal of the authors was not to develop a GSD ontology, the proposed meta-models include some relevant concepts and relations for GSD.
- SL6: The authors developed a taxonomy and associated meta-model with which to represent the primary components of a globally distributed requirements project. They identified three general types of entities: roles (types and relations between them), sites and artefacts. They also proposed a visual notation with which to model distributed requirements engineering processes.

In addition to the papers described above, which were extracted from the systematic mapping, another paper describing an ontology was discovered while we were attending ICGSE'2013: [24]. This was very interesting for us, as the authors presented an ontology for task allocation in distributed software development, because of this paper has been also included in Table 3, which shows the terms that we found in each of our accepted papers, along with the domain or GSE area to which they belong.

Once the terms from relevant ontologies, taxonomies and SLRs had been extracted, the next step was to prepare a complete table of terms by considering the union of terms extracted. This table was circulated to all four authors of this paper in order for them to reach an agreement. The idea was to choose those terms voted for by at least three of the four authors. Surprisingly, a considerable amount of consensus was reached; none of the terms was marked by only two authors. There was therefore no need for further discussions in order to consider or rule out any term.

5. Discovering new GSD terms and relations

In an effort to detect more representative concepts for GSD, 11 further SLRs were examined: [16,25–34]. All SLRs related to GSD published in the time spanning 2011 to 2014 were analysed, along with all papers from the proceedings of ICGSE 2011, 2012, 2013 and 2014.

A candidate term was considered to be relevant to GSD when it could be included in one of the categories defined by the authors when they were attempting to classify the terms found in the previous step. These are those terms that could be:

- Solutions and strategies. Despite the fact that both words are quite general, we believe that it is important to detect the different ways in which to resolve GSD challenges that have been found.
- A methodology. There is currently an important trend as regards agile methodologies. We believed that the addition of this category would make it possible to detect the methodologies most widely used in GSD.
- A process. Since in this new ontology we are also interested in including and describing GSD processes, we believed that this

Table 3
Main terms found in GSE SLRs.

Paper	Terms	Domain
SL1	Project, goals, factors, role, delivery model, team, site, tools, features	Global software development projects
SL2	Global software engineering, insourcing, nearshoring, offshore insourcing, offshore, outsourcing, offshoring, onshore insourcing, onshore outsourcing, onshoring, outsourcing, sourcing	Sourcing strategies
SL3	Architect, architectural style, components, coordination strategy, design decisions, distributed teams, interfaces, non-functional, organisation, software architecture, stakeholders	Architectural knowledge management
SL4.1	Community, activity, resources, decision making, actor, role, practise	Open source development communities
SL4.2	Development artefacts, problem, solution knowledge source, local metadata store, P2P infrastructure, context monitor, context interpreter, profiler, knowledge capturer, knowledge provider	Knowledge sharing in distributed software teams
SL4.3	Participant, role, activity, procedure, artefact, tool	Open source software development
SL4.4	Resource, user, workspace, process, project, actions, process phase, process activity, project phase, project activity, task, knowledge, place	Support global software development
SL5	Team, site, organisation, language, expertise area, social culture, work culture, coordination, process, function, data entity, language, data storage, digital format, actual format, communication, process, communication platform, function platform, tool, data storage, migration	Architecture design of GSD
SL6	Role, artefact, site	Globally distributed requirements engineering
SL7 (paper found in ICGSE 2013)	Artefacts, activities, competences, teams, organisations, project teams	Task allocation in distributed software development

category should be added.

- A skill/ability/competence (to detect the knowledge that a GDS engineer should have)
- A type of work/task allocation (to detect the different types of task distribution used in GSD)
- A risk for GSD
- A knowledge element (i.e., knowledge that should be considered in GSD)
- A resource (generic category, parent of the previous category; resource could help us to detect the different resources used in GSD).
- An activity/task (to detect important activities and tasks that should be carried out in GSD)
- A benefit or drawback (advantage or disadvantage of using GSD)
- A feature
- A goal
- A role involved in GSD
- Support for GSD

Appendix B and Appendix C show the accepted and rejected terms. Three or four reviewers accepted 49 terms, and the reviewers proposed that some of them were possibly synonyms; for instance, stakeholder and actor (team member is a specialisation of actor or stakeholder), place and site, language and language differences, social culture and socio-cultural distance, team and project team, and artefact and development artefact. The reason for rejecting a given term was mainly that it was considered to be too general, i.e., a term that covers a wider context than GSD, such as general software engineering.

As the main purpose of this paper is to focus on the description of the validated version of the ontology, the details of this draught ontology are shown in Appendix D. The following section provides a description of the expert evaluation of this draught, which in turn led to the production of the first version of the GSD ontology.

6. Evaluation of the version 1 of the GSD ontology: a survey

A survey was designed in order to obtain experts' consensus of opinion. This survey described the version 1 of the ontology (V1) and contained questions focusing on how suitable and comprehensive it was as regards representing GSD vocabulary (see Appendix E). In order to elaborate the survey, the guidelines by Kitchenham and Pfleeger [35] were followed. Namely, in drawing up the questions, the purpose and objectives of the survey were borne in mind, with special emphasis of the suitability of the chosen terms and relations included

in our draught ontology thus allowing us to ensure that the questions were directly related to the survey's objectives. In addition, some important factors were also taken into account when deciding what to ask (questions to be worded in such a way that those answering could do so easily and accurately; appropriate number of questions; using terminology that would be familiar to the participants). Moreover, as the questionnaires were non-supervised, it was important to take into account both the format of the questionnaire and the instructions that were provided to carry it out properly. The survey was sent to 15 experts in the GSD field who had considerable experience in GSD research and industrial projects. One of the experts declined the invitation to fill in the survey because she/he did not agree with the idea. Seven experts, whose profiles are shown in Table 4, answered the e-mail and returned the completed survey. One of the experts sent us some papers that might help us with our research but did not fill in the survey. Some of the other experts asked for more time, but although the deadline was extended, they did not answer, perhaps owing to the fact that they did not have sufficient time to fill in a survey. We are, nevertheless, of the opinion that the set of experts that did complete the survey is quite meaningful. As can be observed in Table 4, most of them have more than 5 years of experience as regards working in industry or in research in GSD. In addition, one of them had taken part in the creation of an ontology to support global software development, and has experience in this topic.

The feedback obtained from the surveys was extremely useful as regards improving the ontology since the experts proposed new terms and new relationships, and even included questions when a term was not sufficiently clear or they did not consider that something was intuitive. All of this helped us to reflect on different improvements and discuss solutions to all the challenges that had arisen, and GSD ontology version 2 (V2), which is the final one, consequently came into being.

7. GSD ontology V2

In this section, the final version of the GSD ontology, which was produced based on feedback from experts, is presented, while larger figures of the ontologies are shown in Appendix F. The GSD ontology was eventually structured into three main subontologies: goals, barriers and project, which are presented in the following subsection. The description provided for each ontology is divided into two main parts: the description of the sub-ontology itself, in which its concepts and relationships are explained, and the rationale behind its creation in which we answer the question "How did this ontology evolve?" In particular, in this second

Table 4
Expert profiles.

Researcher	Affiliation	Country	Years of experience with GSD research	Years of experience with GSD projects
Researcher 1	Panasas, Inc.	USA	10	8
Researcher 2	University of Applied Sciences, Emden	Germany	8	–
Researcher 3	Blekinge Institute of Technology	Sweden	11	–
Researcher 4	Automatic Data Processing (ADP)	USA	12	1.5
Researcher 5	PUCRS	Brazil	12	12
Researcher 6	University of Southern Denmark/ Technische Universität München	Denmark/Germany	5	5
Researcher 7	State University of Maringá	Brazil	10	–

part the reader can find the explanation regarding how the sub-ontology has evolved from version 1 to 2, following the experts' suggestions.

7.1. GSD goals sub-ontology

This ontology shows the goals/benefits/drives that a company attempts to obtain when using GSD (Fig. 2). It attempts to indicate the different benefits-goals-purpose that a company wishes to attain when using GSD.

In related literature, these three words are used interchangeably. This being the case, one of the aims of this work is to find a consensus as to what the most appropriate name is; in this respect, the extensive literature review and expert feedback led us to conclude that the goals of a company when using GSD are:

1. To access specialised and diverse resources from other locations, such as a global pool of skilled human resources, which enables companies to expand their activity.
2. Better use of limited resources since GSD can help in the reduction of the software development lifecycle duration, as several phases can be developed by following, for example, the “follow the sun” technique. This means that it is possible to extend working hours and obtain increased flexibility.
3. Reduced costs, which can be achieved through, for example, non-fixed licence fees or reduced labour costs.
4. Better competitiveness by accessing a global market. A global development can allow the company to experience “proximity to clients” and/or “get into new markets”. The proximity to clients may consequently lead to a “reduced time to market” or/and “quick adaptation to volatile business needs”.

7.1.1. Rationale: “How did this ontology evolve?”

As was explained in Section 2, the initial version of the ontology was sent to the experts and an explanation of the ontology was included in

the survey. In this section, we focus on explaining the modifications that the sub-ontology underwent after the experts' opinions were considered (see Table 5).

- The first step was to check which of the terms (benefit, goal, or purpose) was the most popular from the experts' point of view. Four of the experts thought that the term “goal” was that which most clearly indicated the objective of this sub-ontology. Two experts suggested a new term, “driver”, which also reflects the aim of this sub-ontology quite well. Nevertheless, this term is used less frequently in literature and the term “goal” was eventually chosen, as it was favoured by the greatest number of experts and authors of this paper.
- As regards the question “Would you remove any of the terms?” (see Appendix E): Five experts suggested removing the term increased quality, as it is quite a general concept and is rather important in all software projects, thus signifying that it should not be considered as being something specific to GSD projects. We agree with this clarification and it has therefore been deleted from the sub-ontology.
- Another term that some experts proposed removing was shared knowledge; they explained that it could mean a “loss of power”, mainly when working with different companies or when problems with “intellectual property” arise. We agree with this observation and this term was also removed from the sub-ontology.
- The third term that two experts proposed should be deleted was task allocation to low cost destinations, as it could be similar to reduced labour cost. The authors of this paper discussed this proposal and eventually decided that the sub-ontology would not lose meaning if the former term were to be removed. It was therefore also deleted.
- Regarding the question “Would you replace any of the terms used with another term?”, we obtained two comments on increased productivity, since this term characterises the effectiveness of productive effort and we therefore renamed it better use of limited resources, as suggested by another expert. This goal is thus clearer, and it also allows us to add a new relationship with a new term mentioned by

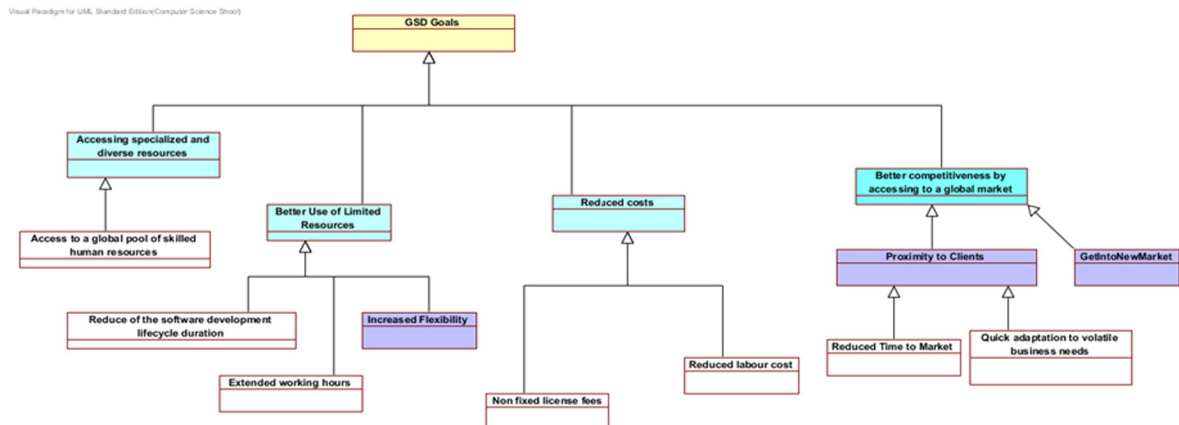


Fig. 2. GSD Goals sub-ontology V2.

Table 5
Expert recommendations for goal GSD sub-ontology.

Id	Comment	How addressed?	Effect
1	What should this ontology be called?	Four experts suggested calling it GSD Goals sub-ontology, while two experts suggested GSD driver sub-ontology as a name.	The name of the ontology was changed.
2	Remove the term “shared knowledge” because it is too general.	The reasons given were taken into account.	The term was removed.
3	Remove the term “increased quality” because that could mean a loss of power.	The reasons given were taken into account.	The term was removed.
4	Remove the term “task allocation to low cost destinations” because it is similar to “reduced labour costs”.	The proposal was checked and we noted that this change did not affect the sub-ontology	The term was removed.
5	Replace the term “increase productivity”.	The reasons given were taken into account.	The term was renamed “better use of limited resources”.
6	Add the term “increased flexibility”.	Two experts proposed adding flexibility, since when you have a large quantity of resources you have more flexibility.	The term “flexibility” was added.
7	The term “non-fixed licence fees” was easily misunderstood.		This term was explained properly.
8	Change the relationship between “gain access to global market” and “improving competitiveness”.	According to the experts this information could be redundant. After discussing this, the idea that finally matched both the experts' opinions and our own ideas concerned making two changes.	The term “gain access to global market” was removed. The second term was renamed “better competitiveness by accessing the global market”, and two more were added: “proximity to clients” and “get into new market”. Two sub-concepts were added to “proximity to clients”: “reduced time to market” and “quick adaptation to volatile business needs”.

two experts: increased flexibility. The experts explained that when you have a huge amount of resources you can be more flexible as regards choosing the most appropriate team to carry out a particular task. We thus also cover the question: Do you think any relevant term is missing? In this case, the term “flexibility” was proposed by two experts.

- Furthermore, two experts proposed that the term non-fixed licence fees should be better explained, and we would therefore like to state that this refers to the acquisition of tools when they are really needed. In global software development, it is common to use a different programme or tool in each phase, i.e., one for the requirements phase,

another for the design process and so on. Since the use of cloud computing provides tools for a specific period of time, it is more flexible for companies to acquire a specific set of tools at the precise point that it is needed [36].

- In relation to the question: Would you remove any of the relationships? Two experts agreed as regards changing the relationships between gain access to global market and improving competitiveness and their respective children. According to these experts, there is a close relationship between them, and this information could be redundant. We therefore debated how this information could be mentioned without repeating information. The idea that finally matched

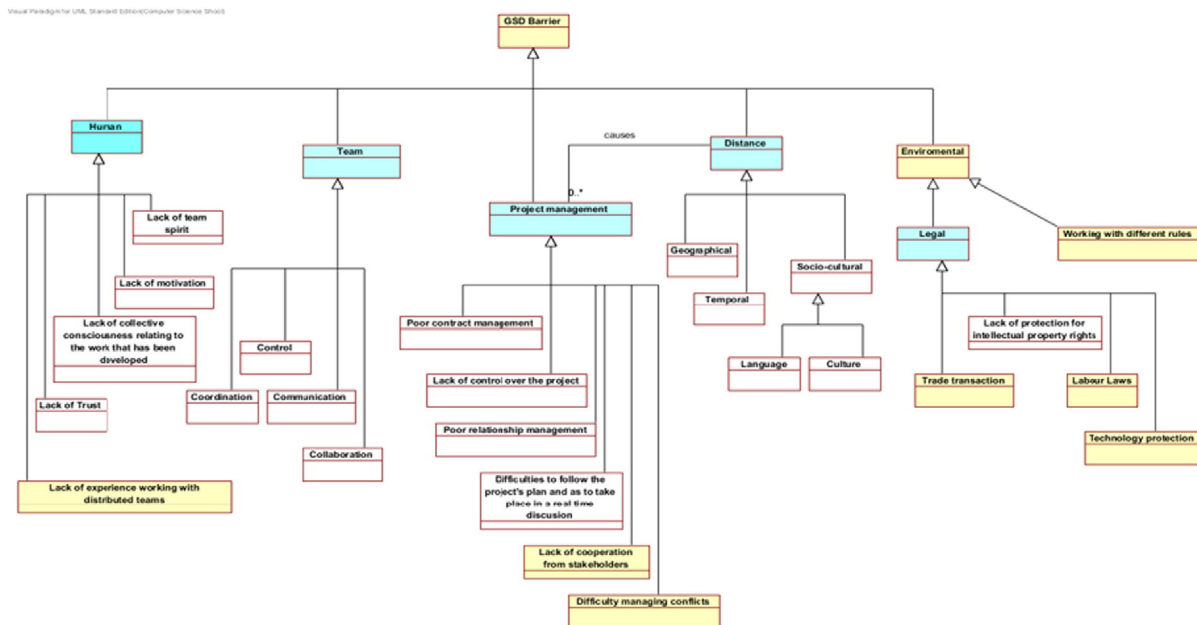


Fig. 3. GSD barriers sub-ontology V2.

both the experts' opinions and our own ideas concerned making two changes: one was to remove gain access to global market and the second was to rename Improving competitiveness, changing it to better competitiveness by accessing the global market; the terms proximity to clients and get into new market would be added to this term. Moreover, two children could be added to proximity to clients: reduced time to market and quick adaptation to volatile business needs.

7.2. GSD barriers sub-ontology

The GSD barrier sub-ontology (Fig. 3) is built around the principal concept of GSD Barrier, which can be specialised into the following:

1. Human factor, since distance can produce a lack of team spirit, motivation, collective consciousness relating to the work that has been developed, and trust. However, problems can also arise as the result of lack of experience in working with distributed teams.
2. Team factors: these aspects are frequently mentioned in literature, and they are collaboration, communication, coordination and control.
3. Project management factors such as poor contract management, a lack of control over the project, poor relationship management, difficulties in following the project plan and as regards the aspects: take place in a real-time discussion, lack of cooperation from stakeholders and difficulty in managing conflicts.
4. Literature also distinguishes between different kinds of distances, which increase the barriers; these can be in terms of locations, and there is a geographical distance. In terms of time, there is another distance, which is called temporal distance. Regarding socio-cultural factors, a new one, distance, has been established. This includes, on the one hand language, and on the other, it also embeds cultural factors which are called cultural differences in literature.
5. Environmental factors, since barriers can increase or decrease depending on the legal issues that each country has. For instance, they are low in topics related to trade transactions, technology protection or lack of protection for intellectual property rights because there is no international law that automatically protects anybody's work throughout the world; each particular country has its own legislation to protect an individual's work; the same occurs with the labour laws. Moreover, working with different countries could mean working with different rules.

7.2.1. Rationale: "How did this ontology evolve?"

First, it is important to state that when we started to analyse the answers regarding this sub-ontology, we realised that half of the experts

found it confusing and difficult to see the borderline between barrier and challenge. They proposed that the difference between both concepts should be clarified, or that the sub-ontology of GSD barriers should be joined to the GSD challenge sub-ontology. While searching for the meaning of *barrier* in the dictionary, it was found that it could be defined as "anything that prevents or obstructs passage, access, or progress, while a challenge is defined as a demanding or stimulating situation". This could therefore be seen as two sides of the same coin, one more positive than the other. We thus decided to merge both and call the new ontology (barrier/challenge).

In addition, after analysing the experts' suggestions, the main modifications were the addition of the following terms; the experts suggested these and we agree that they help to complete the ontology (see Table 6):

- The term lack of experience as regards working with distributed teams has been added as a child of human. The relevant literature also reports that this is a barrier, since it is described in [37] in which the authors performed a literature review to discover problems/challenges in GSD. They found that a lack of experience in working in a distributed team was frequently mentioned when dealing with human challenges.
- The terms lack of cooperation from stakeholders and difficulty in managing conflicts have been added as sub-term of project management. Both terms were also found in [37] as problems associated with management. The first was reported in 17 papers and the second in six papers, which supported our decision to include both of them (as suggested by the literature review described in [37]).
- A parent term environmental was added to indicate the barriers that appear as a result of the differences between sites. These could refer to legal differences such as: trade transactions, labour laws or technology protection (suggested by experts), or to the problems that may occur when working with different rules; all of these issues led to the addition of this term.
- Some experts mentioned the option of adding some ethical issues. However, we believe that this is a concept that is so generally relevant that it should be taken into account during any kind of development.

7.3. GSD project sub-ontology

The principal sub-ontology is the GSD project sub-ontology (Fig. 4), which is the result of major enhancements made to the v1 Project sub-ontology, carried out in accordance with expert opinion. The concepts

Table 6
Expert recommendations for improvements to the barrier GSD sub-ontology.

Id	Comment	How addressed?	Effect
1	There is no clear difference between barrier and challenge	The comment was taken into account and some research was carried out.	Calling the ontology barrier/challenge ontology.
2	Some barriers were missing in "human".	After reviewing the relevant literature, it was found that a lack of experience in working in a distributed team was frequently mentioned when considering dealing with human challenges.	The term "lack of experience as regards working with distributed teams" has been added as a child of "human".
3	Some barriers were missing in "project management".	The terms "lack of cooperation from stakeholders" and "difficulty in managing conflicts" found in [37] were reported in 17 and 6 papers, respectively.	These terms ("lack of cooperation from stakeholders" and "difficulty in managing conflicts") have been added as sub-term of "project management".
4	Some experts noticed that terms related to legal protection were missing.	These could refer to legal differences such as trade transactions, labour laws, or technology protection, or to the problems that may occur when working with different rules; those issues led to the addition of a new term.	A parent term "environmental" was added to indicate the barriers that appear as a result of the differences between sites.
5	Some experts mentioned the option of adding some ethical issues.	We believe that this is an ever-present general concept that should be taken into account during any kind of development.	Nothing was changed.

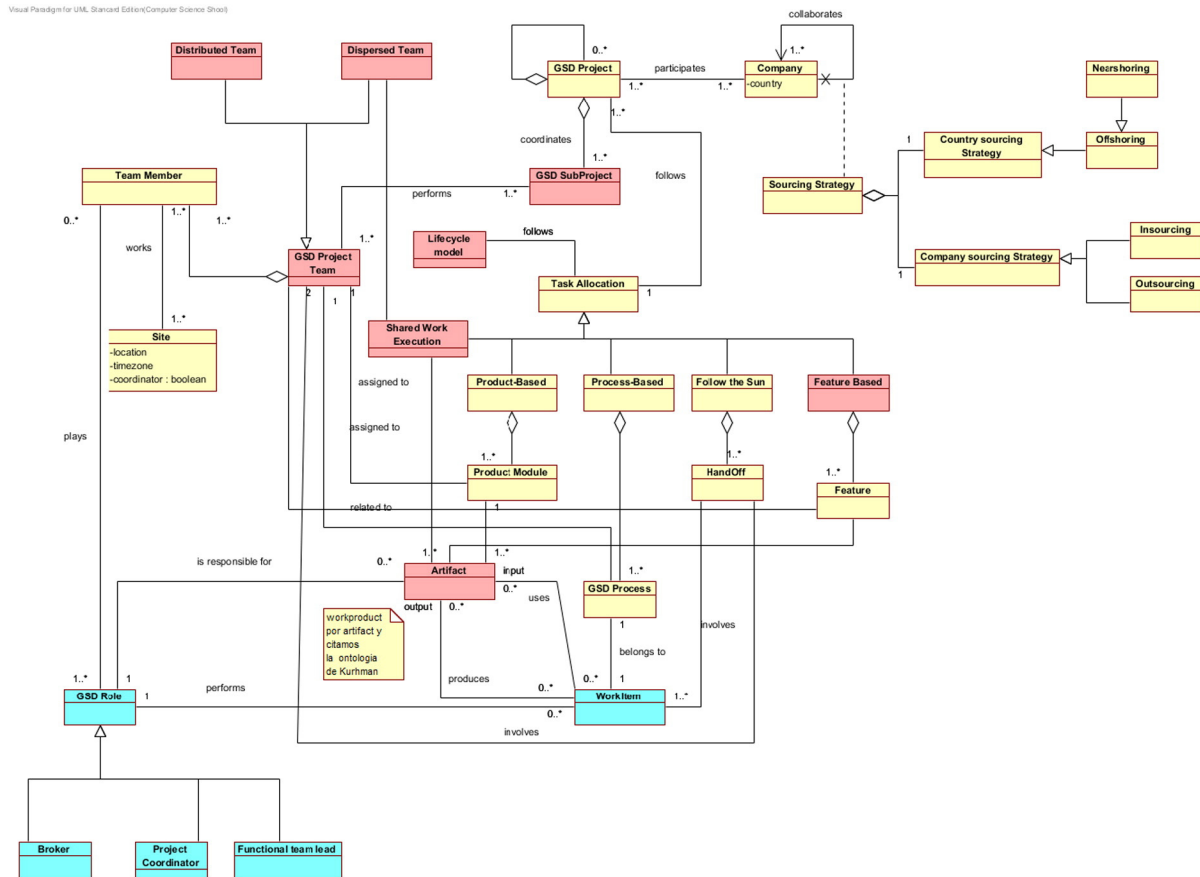


Fig. 4. GSD project sub-ontology V2.

and relationships around which this final version of the Project sub-ontology was built are as follows:

- A GSD project can span several companies. Each company coordinates at least one GSD sub-project. Moreover, a company can participate in the project according to a particular sourcing strategy, depending on the location of the country in which the company is located: country sourcing strategy; in this case we can distinguish between different types. Offshoring is when we leverage resources from a foreign country; a particular case of this is when the country is a neighbouring one, and in this case we use the term nearshoring. When the company collaborates with other companies, this is called outsourcing. In the case of leveraging company-internal human resources, we talk about insourcing.
- GSD projects follows a task allocation, which might be based on division of the product (product-based), process (process-based), time (follow the sun) or the features that the software has (feature based). To be more specific, a product-based approach means that work is divided up into product modules related to at least one artefact. However, task allocation could also be process-based, which means that work would be divided into GSD processes that belong to a work item. On the other hand, in the case of follow the sun, a handoff activity is required for task transition between teams. At least two GSD Project teams should be involved in the handoff.
- A GSD project team comprises a set of team members who work on a particular site with a location and time zone; the team often includes a project coordinator. Moreover, a team member plays a GSD role, which can be Broker (by this we mean mediator, cultural liaison, or cultural ambassador), project coordinator and functional team leader.
- A GSD role performs a work item, while a work item simultaneously belongs to a GSD Process. Moreover, there is an artefact that uses or produces a work item.

7.3.1. Rationale: “How did this ontology evolve?”

- As far as the question “Would you remove any of the terms?” is concerned, one expert realised that the term Onshoring (included in the previous versions) should be removed, as this means “Leveraging resources from the same country” [38], and including it in the sub-ontology was therefore a mistake. The term was deleted.
- Regarding the question “Would you exchange the term we have used for another term?”, two experts proposed renaming “GSD team” “GSD project team”, as a project might have several teams. Moreover, the experts clarified that a “GSD project team” could be distributed or dispersed. Distributed means that the teams are distributed in different locations, but the members of each team work in a co-located fashion. Dispersed means that even the members of the same team can be distributed in different locations [39]. These two terms were therefore added to the ontology.
- The term local project was confusing for some experts since there might not be any local projects. They therefore proposed that it should be removed or renamed. In view of these comments, we tried to find a more accurate term and decided to rename it GSD sub-project.
- With regard to the question “Do you think any relevant term is missing?”, two experts proposed indicating in the sub-ontology the development process approach used in the project (waterfall, agile, etc.). We agree that this is important information that should be included. This being so, a new term life cycle model was added, which is linked to task allocation. Furthermore, the experts stated that a GSD project has artefacts (code, resources, etc.) and that it was advisable to add this term to the sub-ontology. It was for this reason that the term artefact was added. Furthermore, in [40], an

Table 7
Expert recommendations for the project GSD sub-ontology.

Id	Comment	How addressed?	Effect
1	An expert realised that the term onshoring (included in the previous versions) should be removed, as this means "leveraging resources from the same country" [38].	We realised that it was a mistake to include it in the ontology.	The term was removed.
2	Two experts suggested renaming "GSD team" "GSD project team", since a project might have several teams. Moreover, the experts clarified that a "GSD project team" could be distributed or dispersed.	The comment was taken into account.	The term "GSD team" was therefore renamed "GSD project team"; "distributed team" and "dispersed team" were added as terms to the ontology.
3	The term "local project" was confusing for some experts, since there might not be any local projects.	The comment was taken into account and another more accurate term was found.	The term was renamed "GSD sub-project".
4	Two experts proposed indicating the development process approach used in the project in the sub-ontology.	The comment was taken into account and we agreed that this was important information that should be included.	A new term "life cycle model" was therefore added, linked to "task allocation".
5	Some experts stated that a GSD project has Artefacts and that it was advisable to add that term to the sub-ontology.	The comment was taken into account, and some research was carried out.	The term "artefact" was added.
6	Some experts noted the absence of terms related to the types of "task allocation"	The comment was taken into account and some research was carried out.	The term "feature based" was added
7	Some experts asked why "work item" was only connected to "product module".	We realised that the coordination of work by means of "handoffs" is required and the work to be done has to be linked to the different turns in which it is performed.	The term "work item" was linked to "artefact", "GSD process", "handoff" and "GSD role".
7	One expert proposed a new relationship in order to help to explain the "country sourcing strategy"	The comment was taken into account and some research was carried out.	The term nearshoring was added as a child of Offshoring.

ontology for artefacts is described. We recommend that readers look at this ontology if they need more information about the organisation of artefacts.

4) In addition, in order to complete the types of task allocation, the term feature based was added, thanks to the experts' suggestions.

5) In relation to the question "Would you add or remove any of the relationships?", the experts asked why work item was connected only to product module. We realised that work item should also be linked to artefact, GSD process, HandOff and GSD role. The rationale behind this is that Work Item as a process element is related to the other core elements of any process, such as the artefacts which are consumed or produced in the process, along with the responsibility roles. In addition, in the follow the sun allocation mode, the coordination of work by means of "handoffs" is required, and the work to be done has to be linked to the different turns in which it is performed.

6) Another new relationship was proposed, which helps to explain the country sourcing strategy. This is that nearshoring is a particular case of offshoring since the latter is defined as "leveraging resources from a different country", and the definition of nearshoring is "leveraging resources from a neighbouring country" [38]. This process is summarised in Table 7.

8. Validation of the GSD ontology V2

The new version of the ontology (V2) was subjected to a new evaluation by experts in order to validate it by means of a new survey. In the elaboration of this survey, the same framework by Kitchenham

and Pfleeger [35] was followed with special emphasis in including appropriate questions according to the purpose of the survey, which was focused on evaluating the GSD ontology according to the criteria by Gruber [3]. An ontology should be clear, coherent, extendable, focused on the knowledge rather the implementation language and should be generic (not tied to any specific domain). Furthermore, other important questions were added, such as those related to the experts' opinion about the ontologies or concerning whether they will use the ontology in their company or will recommend it (the complete survey is included in Appendix F). In addition, the answers were standardised using the following ordinal scale: 1: strongly disagree; 2: disagree; 3: neither agree nor disagree; 4: agree; 5: strongly agree.

In this case, fifteen experts who attended the 10th International Conference on Global Software Engineering (ICGSE' 2015) participated in the empirical study. They are from Brazil (2), Denmark (1), India (8), Ireland (1), Morocco (1), New Zealand (1) and Spain (1). Eight of the subjects had also served as experts in the first validation phase of this ontology.

The results of the surveys are summarised in Table 8 (those experts who have had more than 5 years of experience on GSD projects are highlighted in bold type). As can be observed in Table 8, most of the participating subjects proved to have broad experience in GSD. With regards to the results analysis, taking the Gruber criteria to evaluate GSD ontology V2, the results were as follows:

- Clarity: Does the ontology communicate its intended meaning effectively?

In the first evaluation the expert evaluation tackled issues concerning clarity of terms and logical relationships between concepts.

Table 8
ICGSE 2015 survey participant results.

Questions	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	Median
1	4	4	4	4	4	4	4	4	5	4	5	4	5	4	5	4
2	4	3	3	3	4	4	3	4	5	4	5	4	5	2	4	4
3	3	4	3	4	4	4	3	4	5	4	5	4	5	1	4	4
4	4	4	4	4	4	4	2	4	5	4	5	5	4	2	3	4
5	3	4	4	4	4	3	4	4	4	4	5	4	5	2	4	4
6	3	3	3	2	2	3	2	2	1	3	1	2	2	3	2	2
7	5	4	3	4	3	5	4	5	5	4	3	4	3	5	3	4
8	4	4	4	4	3	4	4	3	4	3	4	5	4	2	3	4
9	4	4	4	4	4	3	4	3	4	4	5	4	4	3	4	4

While not all the experts were initially convinced by the initial version they evaluated, we provided them with sufficient detail to allow them to express specific ways in which we could improve on the clarity of the presentation. Where possible, the first version of the ontology was enhanced by following the expert recommendations. In this second evaluation, the question “Do you think this ontology is clear?” was therefore added (Question 5 in Appendix F).

A median of 4 was obtained for this question (ranks varied from 1—strongly disagree to 5—strongly agree). After obtaining this score, we were more confident that the new version of the ontology communicates its intended meaning better. This confirmed that the former version had been improved thanks to the experts' suggestions. As a result, the O-GSD v2 now provides a better visualisation of the key concepts in GSD as regards goals, barriers and projects and how concepts relate to each other.

Questions Q8 and Q9 “Does this ontology resolve any questions regarding terminology?” and “Does this ontology resolve any questions as regards the relationships among concepts?” helped us to study the clarity of the ontology. In both cases, the median was 4. More specifically, in the case of Q8, ten experts indicated that they agreed or strongly agreed and in that of Q9 twelve answered agree or strongly agree. Therefore, and according to the data obtained, we are of the opinion that this ontology aims to provide a common language and understanding of the domain.

- Coherence: Is the ontology logically consistent?

This aspect was tackled by Q4, and the results show that twelve experts agree or strongly agree. Furthermore, we evaluated the consistency and coherence of the ontology as a whole and have ensured that, for example, the granularity of the terms is consistent across the abstraction levels. We have also ensured that there is no duplication or conflict in concepts. The breaking down of the top level ontology into several subontologies in a tree structure will ease navigation. Each subontology can stand alone or be combined with other subontologies at the same level.

- Extensibility: ontologies should be designed in a way that allows the definition of new terms for special uses without needing to redefine existing terms.

Q7 asked, “In your opinion, could this ontology be extended?” Five experts answered “neither agree nor disagree”, five agreed and five strongly agreed. Some experts proposed some ideas regarding how to extend it.

For instance, this ontology can be extended with the ontology related to artefacts described in [41] in order to specify the different artefacts that form part of GSD projects. Furthermore, the terms have been defined in a high level of granularity. In addition, we did not describe all the possible roles that can exist in an agile methodology, so this could be another possible extension, as going into this level of detail is outside the scope of this work. When we added the new terms, we did not need to change any of the others or the relationships between them.

- Minimum encoding bias: ontologies should be designed at the “knowledge level” rather than committing the ontology to a particular implementation language and its specific limitations.

Q6 asked, “Do you think this ontology is biased?” Nine experts disagreed or strongly disagreed and the rest neither agreed nor disagreed. This question was that with which more experts (6) neither explicitly agreed nor disagreed.

We used UML and natural language to describe this ontology. UML provides a reliable framework for describing knowledge and

relationships in a consistent way and does not require expert knowledge to interpret [42,43]. Furthermore, the description is easy for anybody to understand even if they do not have a lot of knowledge about ontologies. UML is used by the software engineering community and has been used previously by other authors to describe ontologies [43], and we have thus attempted to avoid any bias.

- Minimal ontological commitment: ontologies should make as few claims as possible about the domain being modelled, without sacrificing the usability of the ontology.

Our ontology was designed to be generic, and any confusing or ambiguous terms have been deleted after either discussion between the authors, or as the result of feedback from the expert evaluators. The objective was to ensure that this ontology is simple and easy to use.

Q2 and Q3 were focused on discovering whether the experts will use the ontology at their company or in their research and if they will recommend the ontology to others. In the survey, the mean was 4 for both questions, with ten experts stating that they agree or strongly agree as regards Q2 and eleven stating the same as regards Q3. We therefore believe that the experts accept this ontology as a common language and understanding of the domain. Moreover, it is important to note that all the experts considered the usage of ontologies to be beneficial.

The results obtained in this survey are very important for us as they were provided by experts with a lot of experience in GSD such as practitioners or researchers who have been involved with this topic for many years. This signifies that their feedback was really valuable for our research and the fact of obtaining positive results encouraged us to believe that the work and several years needed to create this ontology had been worthwhile.

9. Concluding remarks

In this paper, an ontology in which to capture the most relevant terms, concepts and relationships related to GSD projects has been presented. The main research objective was to create a common language and conceptualisation of the field through the development of a GSD ontology focused on the context of goals, barriers and GSD projects. In order to achieve this, a rigorous research method was followed to guarantee that the two main requirements of the ontology were satisfied (completeness based on relevant knowledge and agreed by researchers and practitioners). The conceptualization of GSD was therefore based on an in-depth analysis of the relevant literature, for which a systematic mapping was conducted and the ontology was enhanced and validated by experts in the field by means of two surveys. As a result, the proposed GSD ontology

- a) homogenises and considers the relevant terms regarding GSD in related literature,
- b) provides a harmonised conceptualisation which has been validated by experts in the field,
- c) could serve as a starting point for further application in academia and industry and is simultaneously sufficiently transparent to permit future refinement.

Some guidelines to assist in the use of this ontology before planning a GSD project are also suggested:

- Step 1: instantiate the goals sub-ontology in order to determine which set of goals you wish to attain.
- Step 2: instantiate the project sub-ontology, indicating the team members of each GSD project team, their sites, roles, the sourcing strategies, the type/s of task allocation, etc. This will help you to make decisions about the different options that a GSD project offers and to delimit the project. We recommend

that this instantiation be completed before starting the next step.

- Step 3: instantiate the barriers sub-ontology. Once the project is contextualised it is time to detect the barriers that could appear in the project. Practitioners should therefore analyse each item of this ontology (human, teams, project management, distances and environmental) and predict what problems could arise and how to avoid or minimise them. This step might be very important as regards saving costs since difficulties can be detected and solved before the project starts.

Furthermore, the ontology can provide both researchers and practitioners with some potential benefits such as the following:

- For researchers: When researchers start to investigate this topic they find a lot of information and even several systematic literature reviews regarding GSD. However, as this topic is not sufficiently mature there is no agreement as to the right terminology to use. The first attempt to remedy this situation was made by Šmite et al. [38] in which the authors describe a terminology concerning the different GSD strategies. However, a lot of papers have described the goals and barriers or challenges of GSD, but each paper focuses on a particular set of them, and we believed that it would be appropriate to join all of them together so that a researcher can know the goals and barriers that exist. We are therefore of the opinion that the goals and barriers subontologies could be very useful for this community, as this paper could allow researchers to obtain a preliminary idea about the advantages and disadvantages of GSD and the terminology to use when writing papers, as this is a handicap that we have to tackle when we start to publish in a new topic, and it takes time to become an expert in a specific domain terminology. With our contribution we therefore attempt to make this task easier for new researchers.
- For practitioners: The same contribution explained above can be applied to a practitioner who is starting a GSD project. However, practitioners have an extra difficulty: they do not only need to know the goals and barriers that they could encounter when using this paradigm, but they also need to know how to implement it. The aim of the GSD Project sub-ontology was to help practitioners to discover the different options that exist as regards driving a GSD project.

The main limitation of the current study could be the sample size of experts who participated in the study (fifteen). A larger sample size could help to reinforce the validity of the results, but we consider that this may be a good sample with which to build an agreed on GSD ontology as starting point to attaining maturity in this field with future empirical studies.

With regard to future work, this ontology is being used to build a software development governance framework, in which a solid characterisation of software development is fundamental. In addition, future work may be focused on other applications of the ontology in the field of GSD, such as supporting research that tackles the barriers discovered in the ontology.

Acknowledgments

We should first like to thank the reviewers for their useful suggestions. Moreover, our thanks to all the experts who have helped us to perform this work, concretely, Marco Kurhman, VS Mani, Manuel Arranz, Manal El Bajta, Ita Richardson, Uma Viswanath, Tony Clear, Madn Mohan, Rajeev Kumar, Gupta, Debi Prasad, Utpal Samanta, Nikhil Mokashi, Sildenir Alves, Elisa Huzita, Marcelo Cataldo, Juho Mäkiö, Darja Šmite, Ban Al-Ani and Sabrina Marczak.

This work was funded by the GINSENG project (Ministerio de Economía y Competitividad and Fondo Europeo de Desarrollo

Regional FEDER, (TIN2015-70259-C2-1-R). It is also supported by GLOBALIA (PEII-2014-038-P), Consejería de Educación y Ciencia, Junta de Comunidades de Castilla-La Mancha and SDGear (TSI-100104-2014-4), SDGear ITEA 2 (Call 7), co-funded by the “Ministerio de Industria, Energía y Turismo dentro del Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica” 2013-2016. This work was also partially supported by a Science Foundation Ireland grant 13/RC/2094 to Lero—the Irish Software Research Centre (www.lero.ie).

Appendix A. Supplementary data

Supplementary data to this article can be found online at <http://dx.doi.org/10.1016/j.csi.2016.02.004>.

References

- [1] J.M. Verner, O.P. Brereton, B.A. Kitchenham, M. Turner, M. Niazi, Systematic literature reviews in global software development: a tertiary study, 16th International Conference on Evaluation & Assessment in Software Engineering (EASE 2012), Ciudad Real, Spain, 2012.
- [2] A.B. Marques, R. Rodrigues, T. Conte, Systematic literature reviews in distributed software development: a tertiary study, IEEE International Conference on Global Software Engineering (ICGSE'12), Porte Alegre, Brasil, 2012.
- [3] T.R. Gruber, Toward Principles for the Design of Ontologies used for Knowledge Sharing, 431995 907–928.
- [4] N.F. Noy, D.L. McGuinness, Ontology development 101: a guide to creating your first ontology, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, 2001.
- [5] F. Ruiz, A. Vizcaíno, M. Piattini, F. García, An ontology for the management of software maintenance projects, Int. J. Softw. Eng. Knowl. Eng. 14 (2004) 323–349.
- [6] F. Ruiz, J.R. Hilera, Using ontologies in software engineering and technology, Ontologies for Software Engineering and Software Technology, Springer, Berlin 2006, pp. 49–102.
- [7] F. García, F. Ruiz, C. Calero, M.F. Bertoa, A. Vallecillo, B. Mora, M. Piattini, Effective use of ontologies in software measurement, Knowl. Eng. Rev. 24 (01) (2009) 23–40.
- [8] P. Fitsilis, V. Gerogiannis, L. Anthopoulos, Ontologies for software project management: a review, J. Softw. Eng. Appl. 7 (2014) 1096–1110.
- [9] P. Wongthongtham, E. Chang, T. Dillon, I. Sommerville, Development of a software engineering ontology for multisite software development, IEEE Trans. Knowl. Data Eng. 21 (8) (2009) 1205–1217.
- [10] I. Sommerville, Software Engineering, 2004.
- [11] P. Wongthongtham, E. Chang, T.S. Dillon, Towards ‘ontology’-based software engineering for multi-site software development, Industrial Informatics, 2005. INDIN '05. 2005 3rd IEEE International Conference on, 2005.
- [12] P. Wongthongtham, E. Chang, C. Cheah, T.S. Dillon, Software engineering sub-ontology for specific software development, 29th Annual IEEE/NASA Software Engineering Workshop 2005, pp. 27–33.
- [13] A. Vizcaíno, F. García, I. Caballero, J.C. Villar, M. Piattini, Towards an ontology for global software development, IET Softw. 6 (3) (2012) 214–225.
- [14] B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, Technical Report EBSE-2007-001, Keele University and Durham University Joint Report, 2007.
- [15] S. Goldman, J. Munch, H. Holz, A meta-model for distributed software development, IEEE 8th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE '99), IEEE, Stanford, CA 1999, pp. 48–53.
- [16] A. Borges, S. Soares, S. Meira, H. Tomaz, R. Rocha, C. Costa, Ontologies supporting the distributed software development: a systematic mapping study, Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, Porto de Galinhas, Brasil, 2013.
- [17] F. Salger, Knowledge transfer in global software development—leveraging ontologies, tools and assessments, in: Global Software Engineering (ICGSE) (Ed.), 5th IEEE International Conference on 2010, 2010 (Princeton, NJ, USA).
- [18] G. Aranda, A. Vizcaíno, M. Piattini, Analyzing ontology as a facilitator during global requirements elicitation, 2009 Fourth IEEE International Conference on Global Software Engineering, 2009 (Limerick, Ireland).
- [19] C. Wohlin, Guidelines for snowballing in systematic literature studies and a replication in software engineering, Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering (EASE'14), ACM, London, England, United Kingdom, 2014.
- [20] I. Mirbel, OFLOSSC, An Ontology for Supporting Open Source Development Communities, Proceedings of the 11th International Conference on Enterprise Information Systems, 2009 Milan, Italy.
- [21] W. Maalej, H.-J. Happel, A lightweight approach for knowledge sharing in distributed software teams, 7th International Conference PAKM 2008 2008, pp. 14–25 (Yokohama, Japan).
- [22] T.S. Dillon, G.L. Simmons, Semantic web support for open-source software development, 2008 IEEE International Conference on Signal Image Technology and Internet Based Systems, 2008.
- [23] A.P. Chaves, I. Steinmacher, G.C. Lapasini Leal, E. Hatsue Moriya Huzita, A.B. Biasão, OntoDiSEnv1: an ontology to support global software development, CLEI Electron. J. 14 (2) (2011).

- [24] A. Marques, J. Reginaldo, R. Rodrigues, R. Prikładnicki, S. Marczak, T. Conte, An ontology for task allocation to teams in distributed software development, International Conference on Global Software Engineering (ICGSE '13) 2013, pp. 21–30 (Bari (Italy)).
- [25] S. Ali, S.U. Khan, Critical success factors for software outsourcing partnership (SOP): a systematic literature review, International Conference on Global Software Engineering (ICGSE 2014) 2014, pp. 153–162 (Shanghai, China).
- [26] R. Britto, V. Freitas, E. Mendes, M. Usman, Effort estimation in global software development: a systematic literature review, International Conference on Global Software Engineering (ICGSE 2014) 2014, pp. 135–144.
- [27] C. Costa, L. Murta, Version control in distributed software development: a systematic mapping study, International Conference on Global Software Engineering (ICGSE 2013), 2013 (Bari, Italia).
- [28] R. Giuffrida, Y. Dittrich, Empirical Studies on the Use of Social Software in Global Software Development—A Systematic Mapping Study, Information and Software Technology, 2013 23.
- [29] A.A. Khan, S. Basri, F.E. Amin, Communication risks and best practices in global software development during requirements change management: a systematic literature review protocol, Res. J. Appl. Sci. Eng. Technol. 6 (19) (2013) 3514.
- [30] A.W. Khan, S.U. Khan, Critical challenges in execution of offshore software outsourcing contract from vendors' perspective: a systematic literature review, International Conference on Information and Communication Systems (ICICS 2014), 2014.
- [31] J. Kroll, S.J. Hashmi, I. Richardson, J.L.N. Audy, A systematic literature review of best practices and challenges in follow-the-sun software development, International Conference on Global Software Engineering Workshops (ICGSEW 2013) 2013, pp. 18–23 (Bari, Italia).
- [32] A. Nguyen-Duc, D.S. Cruzes, R. Conradi, The impact of global dispersion on coordination, team performance and software quality—a systematic literature review, Inf. Softw. Technol. (2014).
- [33] B. Raza, S.G. MacDonell, T. Clear, Topics and treatments in global software engineering research—a systematic snapshot, International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2013) 2013, pp. 85–96.
- [34] S. Schneider, R. Torkar, T. Gorschek, Solutions in global software engineering: a systematic literature review, Int. J. Inf. Manag. 33 (1) (2013) 119–132.
- [35] B.A. Kitchenham, S.L. Pflieger, Principles of survey research. Part 2: designing a survey, SIGSOFT Softw. Eng. Notes 27 (1) (2002) 18–20.
- [36] M.A. Chauhan, M. Ali Babar, Cloud infrastructure for providing tools as a service: quality attributes and potential solutions, Proceedings of the WICSA/ECSA 2012 2012, pp. 5–13.
- [37] V. Gomes, S. Marczak, Problems? We all know we have them. Do we have solutions too? A literature review on problems and their solutions in global software development, in: I.C. Society (Ed.), International Conference on Global Software Engineering (ICGSE 2012) 2012, pp. 154–158 (Helsinki, Finland).
- [38] D. Šmite, C. Wohlin, Z. Galvina, R. Prikładnicki, An empirically based terminology and taxonomy for global software engineering, Empir. Softw. Eng. (2012) 1–49.
- [39] J. Eckstein, Agile Software Development with Distributed Teams: Staying Agile in a Global World, Addison-Wesley, 2010.
- [40] H. Femmer, M. Kurhmann, J. Stimmer, J. Junge, Experiences from the design of an artifact model for distributed agile project management, International Conference on Global Software Engineering (ICGSE 2014), 2014 (Shanghai, China).
- [41] M. Kuhmann, S. Beecham, Artifact-based software process improvement & management: a method proposal, International Conference on Software and Systems Process (ICSSP) 2014, 2014 (Nanjing, Jiangsu, China).
- [42] C. Pardo, F. Pino, F. García, M. Piattini, M.T. Baldassarre, An ontology for the harmonization of multiple standards and models, Comput. Stand. Interfaces 34 (1) (2012) 48–59.
- [43] M. Genero, F. Ruiz, M. Piattini, C. Calero, Towards an ontology for software measurement, International Conference on Software Engineering and Knowledge Engineering (SEKE '03) 2003, pp. 78–84.