Information and Software Technology 52 (2010) 779–791

Contents lists available at ScienceDirect

Information and Software Technology

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

Process models in the practice of distributed software development: A systematic review of the literature

Rafael Prikladnicki *, Jorge Luis Nicolas Audy

Pontifícia Universidade Católica do Rio Grande do Sul – PUCRS, Computer Science School, 6681 Ipiranga Avenue, Building 32, 90619-900 Porto Alegre, Brazil

ARTICLE INFO

Article history:
Received 23 March 2009
Received in revised form 1 February 2010
Accepted 15 March 2010
Available online 28 March 2010

Keywords:
Distributed software development
Global software engineering
Offshoring
Process models
Process improvement

ABSTRACT

Context: Distributed Software Development (DSD) has recently become an active research area. Although considerable research effort has been made in this area, as yet, no agreement has been reached as to an appropriate process model for DSD.
Purpose: This paper is intended to identify and synthesize papers that describe process models for distributed software development in the context of overseas outsourcing, i.e. "offshoring".
Method: We used a systematic review methodology to search seven digital libraries and one topic-specific conference.
Results: We found 27 primary studies describing stage-related DSD process models. Only five of such studies looked into outsourcing to a subsidiary company (i.e. "internal offshoring"). Nineteen primary studies addressed the need for DSD process models. Eight primary studies and three literature surveys described stage-based DSD process models, but only three of such models were empirically evaluated.
Conclusion: We need more research aimed at internal offshoring. Furthermore, proposed models need to be empirically validated.

Contents

1. Introduction . . . 780
2. Process models . . . 780
3. Taxonomy used in this systematic review . . . 780
   3.1. Scope of the systematic review (business and technical perspectives) . . . 781
4. Method . . . 781
5. Results . . . 782
   5.1. Quantitative analysis . . . 783
      5.1.1. Research-related information . . . 785
      5.1.2. Content-related information . . . 785
   5.2. Qualitative analysis . . . 786
      5.2.1. DSD process models from a business perspective . . . 786
      5.2.2. DSD process models from a technical perspective . . . 787
      5.2.3. Studies describing the need for process models from a business perspective . . . 787
      5.2.4. Studies describing the need for process models from a technical perspective . . . 787
      5.2.5. Studies describing the need for process models from both perspectives . . . 788
      5.2.6. Other studies describing the need for DSD process models . . . 788
   5.3. Summarizing the studies found . . . 788
   5.4. Limitations . . . 789
6. Discussion and conclusions . . . 789
Acknowledgments . . . 790
References . . . 790

* Corresponding author.
E-mail addresses: rafaelp@pucrs.br (R. Prikladnicki), audy@pucrs.br (J.L.N. Audy).

0950-5849/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2010.03.009

1. Introduction

As part of the globalization efforts currently pervading society, software project team members have become geographically distributed [3,8]. That is a characteristic of Distributed Software Development (DSD). When the distance becomes global, with team members distributed around the world, this characterizes Global Software Development (GSD). The many factors that contributed to DSD or GSD are well documented in the literature [11,13,35,57]. Engineers, managers, and entrepreneurs are facing many challenges on technical, social, political and cultural levels. This change is having a considerable impact on the way products are conceived, designed, tested, and delivered to customers [32]. Thus, the organizational structure and development processes required to support DSD are different from those used in collocated environments [18]. According to Herbsleb and Moitra [32], DSD has different effects on many levels: strategic issues (decision on developing a distributed project); cultural issues; technical issues (technological infrastructure and technical knowledge); and knowledge management issues.

In this context, DSD is a growing field within the Software Engineering (SE) domain. Many companies are distributing their software development facilities, looking for competitive advantages in terms of cost, quality, and skilled professionals [53]. According to Carmel and Tjia [11], the DSD phenomenon started in the early 90s, but it was only recognized as a powerful competitive strategy in the last ten years. Whether local (onshoring) or global (offshoring), within the same company (insourcing) or as a third-party relationship (outsourcing), organizations are facing several important challenges from a SE perspective [43]. After observing practices in the industry, it makes sense to try to understand how these practices have evolved over time, and whether there are process models based on these practices which can be used by organizations that are starting to adopt DSD [48].

Most of the existent literature on DSD stage-based process models tackles strategic aspects of the phenomenon, such as establishing distributed software development centers [12,34], project allocation decisions [25], and client–vendor relationship [45], e.g., from a business perspective. At the same time, however, there are several reasons to consider DSD process models from a technical perspective [43,52,59]. However no agreement has been reached concerning proper process models for DSD.

For this reason, the goal of this systematic review is to identify papers that describe either process models or the need for process models. Our contribution is to identify and categorize studies addressing or developing DSD process models, differentiating between models based on external vendor organizations and those based on wholly-owned subsidiaries.

This paper is organized as follows: in Section 2, we present the concepts involved in the identification of process models (also called stage, capability and maturity models). In Section 3, we set forth the taxonomy used in this systematic review, while in Section 4 we present the method. In Section 5, we set out the results, while in Section 6 we discuss the findings and future directions in this area.

2. Process models

Process models encompass a set of practices or a set of standard steps (or stages) that were successfully followed in the past by individuals, project teams, or organizations, and were documented as a successful practices capable of adoption by other peers. Carmel [10] defines stage models as powerful frameworks in understanding a phenomenon, given that they capture evolution and growth, and also reflect learning curves and diffusion. Carmel [10] argues that such models are useful for both research and practice: practitioners can use such models to understand where they are, where the competition is, and what they can do to evolve. On the other hand, researchers can not only identify and propose the models, but also use them to better understand the behaviors behind a given phenomenon. Such process models can also be defined as maturity and capabilities models. Chrissis et al. [15] define capability as the predictability of the process and its outcomes, or the range of expected results that can be achieved by following a process. On the other hand, the authors define maturity as the growth in the process capability, a well-defined evolutionary path toward achieving a mature process, where each maturity level provides a layer in the foundation for continuous process improvement. Achieving each level of a maturity framework means an increase in the process capability.

But despite the usefulness of such models, they have always been an easy target for criticism, as stated by Carmel [10]. Some criticism includes: the models are heuristically developed, they are usually not validated, they are incomplete, and they assume a linear evolution through each stage. While such criticism is valid, the author also states that, in the end, the collective understanding of a phenomenon would be poorer if these models were not identified. In addition, the author also argues that these models are more useful at the early stages of the phenomenon. Once the phenomenon is mature, the interest in such models is not so evident.

The use of process models or stage models is not something new in Computer Science. They are also very common and can be found in the Social Sciences, where Tuckman proposed a well-known model [64]. The author developed a model to describe the stages (or sequences) of group development. In Computer Science, within the Information Systems domain, one of the first stage models was proposed by Nolan [46], with the purpose of analyzing the evolution of managing the computer resource. In SE, it is possible to find the influence of Nolan's thoughts on the development of models such as the SW-CMM and CMMI [15], among many others. During the development of his work, Nolan [46] also said that stage theories have proved to be useful to develop knowledge in several fields during their formative periods, which is exactly the case of Distributed Software Development.

In DSD, after a couple of years understanding specific problems faced by organizations [11,18,19], both academia and industry realized that it might also be useful to understand which models, by documenting successful practices and processes, can be derived from this past experience [31,52,57,59]. That is exactly what we aim to accomplish with this systematic review.

3. Taxonomy used in this systematic review

The terminology used for Distributed Software Development is not standardized. In this paper we are concerned solely with situations where software development is moved to another country, which is sometimes called Global Software Development. We used a taxonomy based on previous studies [11,39,49,55,56,63] to define the way how distributed software development was organized:

- Offshore outsourcing is used when software development is moved to an external third party in another country.
- Internal offshoring is used when software development is moved to a division of a specific company established in another country.
- Offshoring is used as a generic term when the relationship of the overseas company with the client's company is unknown.

The company that requires the software is referred to as "client". In the context of internal offshoring the client's company

may also be referred to as "headquarters". The company that undertakes the software development is referred to as "vendor". In the context of internal offshoring the vendor company can be referred to as "subsidiary".

Some relevant papers could not be fitted into the above categories. Those are referred to as "Others" in the classification system.

3.1. Scope of the systematic review (business and technical perspectives)

Developing software in a distributed customer–vendor relationship involves a number of business and technical decisions. When embarking on DSD, the outsourcing companies (referred to as *client* or *headquarters*) have to make very important *business* decisions, such as the number of distributed sites (referred to as *vendors* or *subsidiaries*), geographical location of the distributed sites, and organizational structure. Once established, other equally important decisions of *technical* nature relate to the operational environment at the distributed sites, such as project structure, development process, project management, architectural strategies for each project or portfolio, and project modularity.

In an offshore outsourcing model, the vendor might have more autonomy regarding technical decisions, while in internal offshoring approaches technical decisions are often made jointly with the headquarters. In our review, we classified each paper regarding the scope, that is, either business or technical. Those papers exploring both business and technical levels were classified accordingly.

4. Method

The research method used is a systematic literature review [37,38]. The main purpose on the systematic review was to find evidence regarding process models in the practice of DSD. We followed the recommendations provided by Kitchenham [37], and other experiences documented in both the SE and IS literature [7,9,17,21–24]. Our review protocol was based on the one used in Dias Neto et al. [20], and the research question that guided the systematic review was:

> *What DSD process models (also capability, maturity, and stage models), or descriptions of the need for such models, have been published, and what are the details of each paper?*

The keywords were defined based on two main categories of terms: those related to DSD, and those related to process improvement of DSD practice. Table 1 outlines the keywords used in the search.

The search was a combination of A and B. Category A has more keywords and reflects the fact that the area is still maturing, and there are many variations of the same term. By identifying many keywords, we have adopted a high sensitivity strategy, as defined by Dieste and Padua [22], and we understand that despite the availability of a number of papers, only a few of them are relevant to answer our research question (low precision). The search included digital libraries available and papers published in journals, conference and workshop proceedings. We searched seven digital libraries and one conference proceedings, as following:

- IEEEXplore (http://ieeexplore.ieee.org/).
- ACM Digital Library (http://www.sciencedirect.com/).
- Compendex EI (http://www.engineeringvillage2.org).
- INSPEC (http://www.engineeringvillage2.org).
- Wiley InterScience (http://www3.interscience.wiley.com/cgi-bin/home?SRETRY=0).
- Elsevier ScienceDirect (http://www.sciencedirect.com/).

Table 1
Keywords used in the review process.

Reference	Category	Keywords
A	Distributed software development	Distributed software development Global software development Collaborative software development Global software engineering Globally distributed work Collaborative software engineering Distributed development Distributed teams Global software teams Globally distributed development Geographically distributed software development Offshore software development Offshoring Offshore Offshore outsourcing Dispersed teams
B	Process model of DSD practice	Process maturity Process capability Process evolution Evolution sequence Capability model Maturity model

- AIS eLibrary (http://aisel.aisnet.org/).
- Proc. of the ECIS - European Conf on IS (http://csrc.lse.ac.uk/asp/aspecis/default5.asp).

We searched for industrial experience reports, theoretical and empirical papers. To include a paper in the analysis, the paper must have been available online, must have been written in English, and must have described (1) DSD process models or (2) a need for the development of such models. The papers were classified following a two-step approach. First, based on the reading of the papers' titles and abstracts, the papers were classified into two categories:

- [Incl], indicating the papers collected and possibly related to DSD process improvement.
- [Excl], indicating the papers collected but not related to DSD process improvement.

All the papers in category [Excl] were excluded, while the papers in category [Incl] were analyzed more carefully based on the reading of the introduction, conclusion, and specific parts related to the main contribution. Then, a subset of papers in [Incl] was selected, keeping only those addressing DSD process improvement. A pilot of this protocol showed that in some cases the reading of titles and abstracts was not enough to classify each paper properly.

One researcher applied the search strategy to identify the primary papers, and screened the identified papers, by reading the abstract, in order to produce an initial categorization. This was followed by a reading of the full text, and a second classification step was taken, checking whether the inclusion/exclusion criteria were met. In case of any conflict, a second researcher made the verification. After this process, both researchers reached an agreement about what papers should be selected. The papers were classified according to three general categories of information:

- General information: digital library, title, authors, source (e.g. journal or conference proceedings) type of source (i.e. journal, conference, workshop, technical report), and category ([Incl] or [Excl]).
- Research-related information: type of paper (i.e. theoretical, industrial experience report, or empirical study), research empirical strategy (i.e. case study, survey, experiment, ethnog-

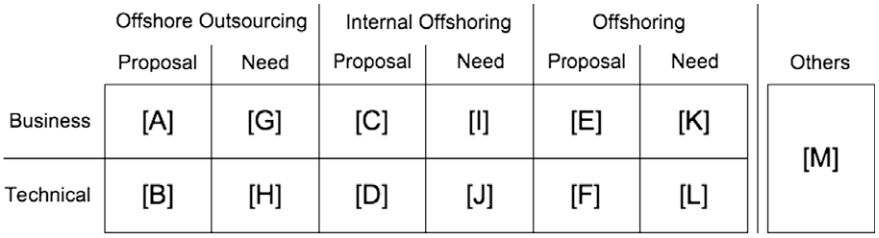

Fig. 1. Paper categorization.

raphy, action research, combination), data collection methods (i.e. interview, observation, questionnaire, document inspection, or multiple data collection methods), type of data analysis (i.e. qualitative, quantitative or both), and data analysis method (i.e. statistics, grounded theory, content analysis). For papers reporting empirical work, the type of study was classified according to the proposal in Dias Neto et al. [20]. Research strategy, data collection, type and method of data analysis were classified according to the terminology used by Oates [47].

– Content-related information: business model (i.e. offshore outsourcing, internal offshoring, offshoring, others), scope of the study (i.e. business, technical), outcome (i.e. model proposal, need for a model), process model type (i.e. maturity, capability, stage, others), focus of the study (i.e. people, project, organization), which site (i.e. client/headquarters, vendor/subsidiary), attributes, and general comments. Attributes are related to themes explored in each study, and general comments are a brief summary of each selected paper to guide the qualitative analysis.

After the information was extracted, the papers were classified in one of the categories illustrated in Fig. 1.

Both quantitative and qualitative analyses were employed for each of the categories outlined in Fig. 1. While our quantitative analysis includes descriptive statistics, the qualitative analysis brings information about characteristics of each study, strengths, and weaknesses.

5. Results

The systematic review was conducted from October to December, 2007. A total of 227 papers were found, as shown in Table 2.

After the initial screening, 63 papers were selected for the second screening, where 26 were selected for an in-depth analysis (Table 3). As we can see, the lack of standard terminology in DSD resulted in a large number of papers to start with, but only a few were selected, confirming the high sensitivity and low precision of our search, as described in Section 4 and suggested by Dieste and Padua [22].

Table 2
Search execution, first results.

Digital library (DL)	Number of papers	First classification	
		[Incl]	[Excl]
IEEEXplore	10	7	3
ACM Digital Library	20	10	10
Wiley InterScience	10	5	5
Elsevier ScienceDirect	78	10	68
Compendex EI	14	1	13
Inspec	11	3	8
ECIS	18	10	8
AIS eLibrary	66	17	49
Total	227	63	164
Percentage (%)	100	28	72

Table 3
Final set of papers to be reviewed.

Digital library (DL)	Number of papers [I]	Second classification [SC]			Total selected ([I]-[SC])
		Own paper	Repeated	Not relevant	
IEEEXplore	7	1	–	1	5
ACM Digital Library	10	–	–	6	4
Wiley InterScience	5	–	–	2	3
Elsevier ScienceDirect	10	–	–	7	3
Compendex EI	1	–	1	–	0
Inspec	3	–	2	1	0
ECIS	10	–	–	6	4
AIS eLibrary	17	–	1	9	7
Total	63	1	4	32	26
Percentage (%)	100	2	6	51	41

One paper spotted in IEEEXplore was previously published in a conference found in the AIS eLibrary, and was classified as "repeated". Two other papers proposed maturity models related to pure outsourcing. Since that does not necessarily characterize DSD, they were not selected for further analysis. Moreover, four more papers (P27, P28, P29, and P30) were included in the list to be reviewed. One is a journal paper (P27) referred to in some of the papers selected [12]. The other three (P30, P28, and P29) are related to the research question and were selected based on our knowledge of the area [34,43,45]. In total, 30 papers were selected for analysis. Table 4 provides a summary of each paper and the detailed results are presented in Tables 5 and 6.

Out of the thirty papers found, 11 describe *DSD process models* (P1, P2, P5, P11, P14, P16, P20, P22, P27, P29, and P30), while 19 address *the need for such models* (P3, P4, P6, P7, P8, P9, P10, P12, P13, P15, P17, P18, P19, P21, P23, P24, P25, P26, and P28). Among the 11 papers describing *DSD process models*, eight were primary studies, and three were based on literature surveys as the research method (P5, P14, and P22). In all the three papers, the authors searched the literature in order to identify significant contributions for their proposals.

Balaji and Brown (P5) looked at the dynamic capabilities perspective as the foundation to derive the client-side capabilities for sourcing [6]. Their literature review identified four models that can be used to understand a company's sourcing decision. The authors also identified the following as the three most important capabilities: vendor management (the ability of the client's company to control, coordinate and maintain the vendor relationship), project management (the way outsourced projects are managed by the client's company), and process management (the way key processes in an organization are performed, maintained and managed for a sustained period of time).

Gannon and Wilson (P14) looked at the literature about outsourcing, offshoring, and IS maturity models to propose a maturity

Table 4
Papers selected for analysis.

Study	DL	Title	Authors	Source	Year
P1	IEEE	Leveraging global resources: a process maturity framework for managing distributed development	N. Ramasubbu et al.	IEEE Software	2005
P2	IEEE	Collaboration maturity and the offshoring cost barrier: the tradeoff between flexibility in team composition and cross-site communication effort in geographically distributed development projects	S. Lasser, M. Heiss	International Professional Communication Conference (IPCC)	2005
P3	IEEE	Optimizing Supplier Management in Global Software Engineering	C. Ebert	ICGSE	2007
P4	IEEE	Offshoring: what can go wrong?	N. Matloff	IT Professional	2005
P5	IEEE	Strategic IS sourcing and dynamic capabilities: bridging the gap	S. Balaji, S. A. Brown	HICSS	2005
P6	ACM	A research agenda for distributed software development	B. Sengupta et al.	ICSE	2006
P7	ACM	Globally distributed software development project performance: an empirical analysis	N. Ramasubbu, R. Krishna	FSE	2007
P8	ACM	Agile software process and its experience	M. Aoyama	ICSE	1998
P9	ACM	Uncovering the reality within virtual software teams	V. Casey, I. Richardson	International Workshop on GSD for the practitioner	2006
P10	Wiley	The moderating role of development stage in free/open-source software project performance	K.J. Stewart, S. Gosain	SPIP	2006
P11	Wiley	Software outsourcing quality achieved by global virtual collaboration	K.V. Siakas, B. Balstrup	SPIP	2006
P12	Wiley	Where do capabilities come from and how do they matter? A study in the software services industry	S.K. Ethiraj et al.	Strategic Management Journal	2005
P13	ECIS	Requirements engineering during global software development: some impediments to the requirements engineering process: a case study	J. Hanisch, B.J. Corbitt	ECIS	2004
P14	ECIS	IS offshoring: a proposed maturity model of offshore is suppliers	B. Gannon, D.W. Wilson	ECIS	2007
P15	ECIS	Offshore information systems outsourcing: strategies and scenarios	N. Khan et al.	ECIS	2003
P16	ECIS	IT outsourcing maturity model	O. Adelakun	ECIS	2004
P17	AIS	Evolution of trust in distributed software development teams: a punctuated equilibrium model	A.L. McNab, S. Sarker	AMCIS	2006
P18	AIS	A qualitative investigation of IS offshore sourcing	L. Sayeed	AMCIS	2006
P19	AIS	Crafting and executing an offshore IT sourcing strategy: globshop's experience	C. Ranganathan, P. Krishnan	ICIS	2006
P20	AIS	Exploring the key capabilities for offshore IS sourcing	S. Balaji, C. Ranganathan	ICIS	2006
P21	AIS	Knowledge transfer in offshore insourcing	A.L. Chua, S. Pan	ICIS	2006
P22	AIS	Offshore subsidiary engagement effectiveness: the role of subsidiary capabilities and parent – subsidiary interdependence	M. Ramamani	Conference of Midwest US Association for IS	2006
P23	AIS	Developing a model for offshore outsourcing	N. Khan et al.	AMCIS	2003
P24	Elsevier	Procedural coordination and offshored software tasks: lessons from two case studies	R. Mirani	Information & Management	2007
P25	Elsevier	A dimensional analysis of geographically distributed project teams: a case study	J.R. Evaristo et al.	Journal of Engineering and Technology Management	2004
P26	Elsevier	The Indian software services industry	A. Arora et al.	Research Policy	2001
P27	Other	The Maturation of Offshore Sourcing of IT Work	E. Carmel, R. Agarwal	MISQ Executive	2002
P28	Other	The unspoken revolution in software engineering	B. Meyer	Computer	2006
P29	Other	Client–vendor relationships in offshore applications development: an evolutionary framework	R. Mirani	Information Resources Management Journal	2006
P30	Other	Taper: a generic framework for establishing an offshore development center	G. Hofner, V.S. Mani	ICGSE	2007

model for offshore IS suppliers [28]. While the literature review on IS outsourcing and IS offshoring was related to the basics concepts associated with each topic, the literature review of IS Maturity Models identified well-known models such as the model proposed by Nolan [46] and the CMMI [15]discussed in Section 2. In addition, the authors cited the SITO stage model [12], which was selected as a primary study to be analyzed in our systematic review.

Ramamani (P22) looked at the existing literature on the resource-based view of the company to present a capability-based argument for predicting subsidiary-based sourcing effectiveness. The author presented a brief literature review on subsidiary capability and interdependence, but did not search the literature on process models. The author proposed organizing the supplier's offerings into supplementative capabilities (those that help firms to increase their efficiency and respond to increased demand), complementative capabilities (those where the vendor provides complementary competencies to that of the client), or differentiative capabilities (those that are recognized as important for the company as a whole to disseminate to other parts of the company).

In general, the authors did not undertake a structured literature review, and only one of them (P14) searched for other DSD process models. In the other two studies the authors proposed process models for specific topics (i.e. subsidiary engagement effectiveness). They did not search all the existing literature about process models as we did in this review. Thus, this paper is the first study that has systematically tried to understand how process models have been proposed for use in the DSD domain.

5.1. Quantitative analysis

The quantitative analysis was divided into content-related information and research-related information. Tables 5 and 6 outline the detailed categorization of content-related and research-related information. The three papers that were based solely on literature survey were not included in our analysis. Three other papers that included literature surveys are included because they also used other methods of study.

Most of the papers report empirical studies (Table 7). We also found more papers from a business perspective.

Table 5
Research-related information.

Study	Type	Research strategy	Data collection	Data analysis	Analysis method	Outcome
P1	Empirical	Literature review Focus group Survey	Not defined (ND) Interviews Questionnaire	ND Quantitative	ND Statistics	Proposal
P2	Industrial experience	Not applicable (NA)	NA	NA	NA	Proposal
P3	Industrial experience	NA	NA	NA	NA	Need for process model
P4	Industrial experience	NA	NA	NA	NA	Need for process model
P6	Empirical	Case study	Interviews	Qualitative	ND	Need for process model
P7	Empirical	Case study	Documentation Observation Interviews	Quantitative	Statistics	Need for process model
P8	ND	ND	ND	ND	ND	Need for process model
P9	Empirical	Case study Action research	Documentation Observation Interviews	Qualitative	Content analysis	Need for process model
P10	Empirical	Survey	Questionnaire	Quantitative	Statistics	Need for process model
P11	Empirical	Case study	Questionnaire Interview	Qualitative	Content analysis	Proposal
P12	Empirical	Case study	Documentation	Quantitative	Statistics	Need for process model
P13	Empirical	Case study	Interviews	Qualitative	Content analysis	Need for process model
P15	Empirical	Case study	Interviews	Qualitative	Grounded theory	Need for process model
P16	Empirical	Literature review	Informal interviews	Qualitative	ND	Proposal
P17	Empirical	Case study	Questionnaire	Qualitative Quantitative	Content analysis and statistics	Need for process model
P18	Empirical	Case study	Interview	Qualitative	Content analysis	Need for process model
P19	Industrial experience	NA	NA	NA	NA	Need for process model
P20	Empirical	Literature review Focus group	Interview	Qualitative	Content analysis	Proposal
P21	Empirical	Case study	Interview	Qualitative	Content analysis	Need for process model
P23	Empirical	Literature review Case study	Interview	Qualitative	Content analysis	Need for process model
P24	Empirical	Case study	Interview Observation	Qualitative	Content analysis	Need for process model
P25	Empirical	Case study	Interview	Qualitative	Content analysis	Need for process model
P26	Industrial experience	ND	Interviews Observations	Qualitative	Content analysis	Need for process model
P27	Empirical	Case study	Interview	Qualitative	Content analysis	Proposal
P28	Industrial experience	NA	NA	NA	NA	Need for process model
P29	Empirical	Case study	ND	Qualitative	ND	Proposal
P30	Industrial experience	NA	NA	NA	NA	Proposal

Table 6
Content-related information.

Study	Scope	Business model	Process model type	Focus	Outcome	Site
P1	Technical	Offshoring	Maturity	Projects	Proposal	Vendor/subsidiary
P2	Business	Internal offshoring	Maturity	Projects	Proposal	Vendor/subsidiary
P3	Business	Offshore outsourcing	Maturity	Organization	Need for process model	Vendor/subsidiary
P4	Both	Offshore outsourcing	Maturity	People	Need for process model	Vendor/subsidiary
P6	Technical	Internal offshoring	Maturity	Projects	Need for process model	Vendor/subsidiary
P7	Technical	Offshoring	Capability	Projects	Need for process model	Vendor/subsidiary
P8	Technical	Other	Capability	Projects	Need for process model	ND
P9	Technical	Offshore outsourcing	Capability	Projects	Need for process model	Both
P10	Technical	Other	Stages	Projects	Need for process model	ND
P11	Both	Offshore outsourcing	Capability	Organization	Proposal	Vendor/subsidiary
P12	Both	Offshore outsourcing	Capability	Projects	Need for process model	Vendor/subsidiary
P13	Technical	Offshoring	Capability	Projects	Need for process model	Both
P15	Both	Offshore outsourcing	Capability	Organization	Need for process model	Vendor/subsidiary
P16	Business	Offshore outsourcing	Maturity	Organization	Proposal	Client/headquarters
P17	Technical	Other	Capability	People	Need for process model	ND
P18	Business	Offshoring	Capability	Organization	Need for process model	Client/headquarters
P19	Business	Offshore outsourcing	Capability	Organization	Need for process model	Client/Headquarters
P20	Business	Offshore outsourcing	Capability	Organization	Proposal	Client/headquarters
P21	Both	Internal offshoring	Capability	Organization	Need for process model	Client/headquarters
P23	Business	Offshore outsourcing	Capability	Organization	Need for process model	Vendor/subsidiary
P24	Both	Offshoring	Capability	Organization	Need for process model	Both
P25	Technical	Other	Capability	Projects	Need for process model	Both
P26	Business	Offshore outsourcing	Capability	Organization	Need for process model	Vendor/subsidiary
P27	Business	Offshoring	Stages	Organization	Proposal	Client/headquarters
P28	Technical	Offshoring	Capability	Projects	Need for process model	Vendor/subsidiary
P29	Business	Offshoring	Capability	Organization	Proposal	Both
P30	Business	Internal offshoring	Maturity	Organization	Proposal	Vendor/subsidiary

In general, most of the papers found were related to offshore outsourcing or offshoring (Table 8).

An important remark is that 70% of the papers address aspects of offshore outsourcing and offshoring business models. Another pattern is that some of the research currently being done in DSD does not explicitly explain the distribution. In our review, 30% of the papers claim to study globally distributed development (offshoring), but there is no evidence related to any of the business models. As stated by Herbsleb and Moitra [32], the processes employed in offshore outsourcing might be different than those employed in internal offshoring, and the characterization in this case could make a difference for the practice of DSD. Moreover, research conducted in one type of distribution is not necessarily valid for all types of DSD. Another note is that the number of papers classified as "offshoring" and "others" indicate that almost half of the papers do not define the relationship between the companies. In this case, the findings show a good indication that a better contextualization is needed for all papers, in order to understand the practices that apply to each type of DSD (considering the relationship between organizations and geographic location). This is also corroborated by Smite et al. [61].

5.1.1. Research-related information

All papers were classified based on the research methods employed (Table 9), as well as data collection (Table 10) and data analysis (Table 11).

We did not find any information explaining the research methodology in the paper where the type was not identified (P8). When analyzing the experience reports, only one paper (P26) employed and explained some research methodology [2]. The authors planned a case study with observations and interviews with some Indian vendors, having qualitative data to be analyzed using content analysis. From the analysis of the 19 empirical papers (Tables 9–11), we can conclude that most of the studies use case studies as the research method, with data collection using interviews and performing qualitative data analysis.

5.1.2. Content-related information

We present the main results based on the content of each paper. First, the papers were classified regarding the type of process model (maturity or capability model, or a stage model not explicitly defined as maturity or capability). In Table 12, "C" stands for capability, "M" for maturity, and "S" for stage.

According to the table, two papers (7%) explored the concept of stage models, seven papers (26%) explored the concept of maturity models, and 18 papers (67%) explored the concept of capability models.

As mentioned previously, eight primary studies describe DSD process models. Table 13 shows information about the level of analysis in these models (i.e. people level, project level, or organization) and the site (client or vendor for offshore outsourcing, headquarters or subsidiary for internal offshoring, or any of these combinations for offshoring).

One paper (P29) is considered a model developed for both client/headquarters and vendor/subsidiary sides, and for this reason the table shows nine and not eight models. Most of the models – seven (77%) – are focused on organization. Table 14 brings the same information based on the other 19 papers exploring the need for DSD process models.

Based on Table 13, we can see that more than half of the models are related to the vendor/subsidiary side. As outlined in Table 14,

Table 7
Type of paper.

	Empirical		Experience report		Not classified
	Proposal	Need	Proposal	Need	
Business	4	2	2	3	1
Both	1	4	–	1	
Technical	1	7	–	1	
Total	6	13	2	5	1

Table 8
Scope of the study.

	Offshore outsourcing		Internal offshoring		Offshoring		Others	
	Proposal	Need	Proposal	Need	Proposal	Need	Proposal	Need
Business	2	4	2	0	2	1	0	0
Both	1	3	0	1	0	1	0	0
Technical	0	1	0	1	1	3	0	4
Total	3	8	2	2	3	5	0	4

Table 9
Research method.

Research method	Number of papers	Paper ID
Case study	15	P6, P7, P11, P12, P13, P15, P16, P17, P18, P21, P23, P24, P25, P27, P29
Survey	1	P10
Focus group	1	P20
Multimethod (focus group and survey)	1	P1
Multimethod (case study and action research)	1	P9

Table 10
Data collection method.

Data collection method	Number of papers	Paper ID
Interviews	10	P6, P13, P15, P16, P18, P20, P21, P23, P25, P27
Questionnaires	2	P10, P17
Documentations	1	P12
Interviews and questionnaire	2	P1, P11
Interviews, observations	1	P24
Documentation, interviews, observations	1	P7, P9
Could not be identified	1	P29

Table 11
Data analysis.

Data analysis method	Number of papers	Paper ID
Qualitative	14	P6, P9, P11, P13, P15, P16, P18, P20, P21, P23, P24, P25, P27, P29
Quantitative	4	P1, P7, P10, P12
Both qualitative and quantitative	1	P17

Table 12
Analysis of the type of process model.

	Offshore Outsourcing						Internal offshoring						Offshoring						Others					
	Proposal			Need			Proposal			Need			Proposal			Need			Proposal			Need		
	C	M	S	C	M	S	C	M	S	C	M	S	C	M	S	C	M	S	C	M	S	C	M	S
Business	1	1	–	3	1	–	–	2	–	–	–	–	1	–	1	1	–	–	–	–	–	–	–	–
Both	1	–	–	2	1	–	–	–	–	1	–	–	–	–	–	1	–	–	–	–	–	–	–	–
Technical	–	–	–	1	–	–	–	–	–	–	1	–	–	1	–	3	–	–	–	–	–	3	–	1
Total	2	1	–	6	2	–	–	2	–	1	1	–	1	1	1	5	–	–	–	–	–	3	–	1

Table 13
Focus of the models proposed and the site involved.

	Offshore outsourcing		Internal offshoring		Offshoring	
	Client	Vendor	Headquarters	Subsidiary	Client/headquarters	Vendor/subsidiary
People	–	–	–	–	–	–
Projects	–	–	–	1	–	1
Organization	2	1	–	1	2	1
Total	2	1	0	2	2	2

we found twice as many papers claiming the need for models in the vendor/subsidiary side. Regarding the focus, although the majority of papers (10) describing models focus on the organization level, the papers describing the need for such models are balanced between both organizations and projects – in two studies the focus is on people.

5.2. Qualitative analysis

Since our main interest is on the study of DSD process models from a Software Engineering perspective, our analysis was focused on papers describing technical contribution. However, given the results found, papers centered on a business perspective are also described.

5.2.1. DSD process models from a business perspective

We have identified six models (75%) from a business perspective, and one from both business and technical perspectives. Two models (one for offshoring and one for offshore outsourcing) were based on a stage model published in 2002 at MISQ Executive [12], and revisited by one of the authors in Carmel and Tjia [11]. This stage model was firstly named SITO (Sourcing of IT Work Offshore), and later updated to OSM (Offshore Stage Model).

5.2.1.1. Models related to offshoring. Carmel and Agarwal [12]: the SITO model was proposed by Carmel and Agarwal [12], and revisited by Carmel and Tjia [11]. In addition to being an input for other studies [1,28,45], this model has been adapted by IT consultancies (i.e. Forrester and Meta Group) to evaluate offshoring evolution [10]. The four dimensions of the stage model are: offshore bystander, offshore experimenter, proactive cost focus, and proactive strategic focus. Each stage is characterized by a set of strategic imperatives and internal firm dynamics. At stage one, the company does not do offshore yet, but rather observes many other companies that are offshoring. At stage two, the company tests the offshoring model. At stage three, the company has offshore projects and processes, where cost savings is the main goal. Finally, at stage four companies achieve strategic advantages that cannot be achieved by other means. The authors suggest that technology companies at stage four have different organizational structures and mechanisms. These companies have accumulated considerably more experience in offshore IT sourcing, but they would usually prefer to have their own IT units, sourcing from within their organizations (internal offshoring or wholly-owned subsidiary).

Mirani [45]: the authors have proposed an evolutionary framework for client–vendor relationships in offshore applications development. The authors suggest that "such a relationship typically begins as a cost-reduction exercise, with the client contracting out simple, structured applications to one or more offshore vendors. Over time, the client assigns increasingly complex applications to selected vendors and cultivates the relationships with them. As offshore applications continue to evolve and become business-critical, the client may seek to regain control by establishing a command-based hierarchy. This may be achieved through partial or full ownership of a vendor's organization or by starting a captive offshore subsidiary. Thus, the client's initial cost reduction goal is ultimately displaced by one of risk control." The proposal is very similar to the SITO model, but focused on the relationship between client and vendor and how this relationship evolves over time. In the same way, the authors also suggest that internal offshoring is the last stage in the evolution.

5.2.1.2. Models related to offshore outsourcing. Adelakun [1]: the author has proposed an IT outsourcing maturity model based on

Table 14
Focus of the need for models and the site involved.

	Offshore outsourcing		Internal offshoring		Offshoring		Other		
	Client	Vendor	Headquarters	Subsidiary	Client/headquarters	Vendor/subsidiary	Client/headquarters	Vendor/subsidiary	Not defined
People	–	1	–	–	–				1
Projects	1	2	–	1	1	3	1	1	2
Organization	1	4	1	–	2	1			
Total	2	7	1	1	3	4	1	1	3

the Tuckman's stage model [64]. He argues that this proposal extends the SITO model to domestic outsourcing, nearshore outsourcing and offshore outsourcing, and for this reason it is significantly different, since it focuses primarily on domestic outsourcing, while the SITO stage model focuses primarily on offshore outsourcing. The initial model was developed based on literature review, then discussed with five practitioners and tested through a case study. The problem with this proposal is that the case study description is very high level, and the feasibility of this model in practice is not clear.

Balaji and Ranganathan [5]: in this paper, the authors discussed critical capabilities for effective IS offshoring of application development, from the clients' point of view. This paper was later improved and published at MISQE [53]. Based on focus group discussions with senior IS managers and case studies of IS offshoring, four capabilities were proposed: IS systemic thinking (the ability to clearly set goals, map expectations and choose appropriate sourcing strategy for the offshore sourcing arrangement), IS vendor management (the appropriate selection of vendor, structuring the contract and managing the vendor relationship), global IS resource development (the client's ability to manage both the client and vendor resources applied to the offshoring arrangement), and IS change management (managing the several changes caused by offshoring arrangements).

5.2.1.3. Models related to internal offshoring. Lasser and Heiss [40]: in this proposal the authors argued that there is a relationship between the maturity of the collaboration and the cost associated with offshoring activities. Based on a real experience with software development centers within Siemens, they have identified fifteen stages of collaboration, relating the location of high-cost and low-cost sites, type of activities, and responsibilities. The argument in this study is that the cost should be balanced based on the maturity of collaboration.

Hofner and Mani [34]: in this paper, the authors have proposed the TAPER framework, to help companies with the establishment of offshore software development centers. The framework is organized into five phases (Trust, Assess, Prove, Enhance, and Reengineer), and suggests that the creation of an offshore center should follow some standard steps in order to minimize possible risks and increase the chance of success. The proposal was based on their experience with the creation of an offshore software development center in India owned by Siemens in Germany.

5.2.2. DSD process models from a technical perspective

From a technical perspective, we have identified two models (one for offshoring and one for offshore outsourcing). The one related to offshore outsourcing was classified on both technical and business levels, since it involves not only client management relationship, but also capabilities improvement in the development of products and services.

Ramasubbu et al. [52]: in this paper, the authors have proposed a maturity framework for globally distributed development. They identified 24 key process areas for managing global projects, organized into four concepts: collaboration readiness (the ability of an overall software development governance model to set business goals translated into tasks across geographically distributed teams), common ground (the shared knowledge of distributed development participants), coupling in work (the mechanisms for dividing labor into distributed product development), and technology readiness (development infrastructure and personnel capability levels for using collaborative technologies). The framework was empirically evaluated and then tested in distributed projects in a multinational company.

Siakas and Balstrup [60]: in this paper, the authors argued that the quality of software outsourcing in global virtual collaboration could be achieved by using two models: the eSCM-SP (eSourcing Capability Model for Service Providers), and the SQM – CODE (Software Quality Management – Cultural and Organisational Diversity Evaluation). The eSCM-SP was originally proposed by Hyder et al. [33], because, according to the authors, the existing frameworks do not address all of the critical issues in eSourcing (IT-enabled Sourcing). It is important to note that sourcing is related to the relationship between the organizations, and does not necessarily mean globally distributed development. For this reason, the capabilities and practices in this model are related to a successful sourcing relationship. The model also contains practices that could be useful in a distributed environment, since global sourcing of IT work is related to distributed development. The other proposal in the paper (SQM – CODE) is a tool for assessing the fit between organizational and national culture, in order to identify cultural factors that require proper action. For the purpose of our systematic review, only the eSCM-SP was taken into consideration. The eSCM-SP is a "best practices" capability model with three purposes: (1) giving service providers the guidance that will help them improve their capability across the sourcing life cycle; (2) providing clients with a straight-forward means of evaluating the capability of service providers; and (3) offering service providers a standard to use when setting themselves apart from competitors. The Sourcing life cycle proposed in the model is divided into Ongoing, Initiation, Delivery, and Competition. The Capability Areas provide logical grouping of practices and are divided into Knowledge Management, People Management, Performance Management, Relationship Management, Technology Management, Threat Management, Contracting, Service Design and Deployment, Service Delivery, and Service Transfer. The Capability Levels are divided into Providing Services (Level 1), Consistently Meeting Requirements (Level 2), Managing Organizational Performance (Level 3), Proactively Enhancing Value (Level 4), and Sustaining Excellence (Level 5).

5.2.3. Studies describing the need for process models from a business perspective

We have identified five papers describing the need for DSD process models from a business perspective: one related to offshoring, and four related to offshore outsourcing. Sayeed [58], for example, conducted a study using the SITO stage model, with the purpose of understanding how 15 companies make their decisions on offshoring of software development based on the stage they are classified in. The author has classified each company into one of the stages, identifying the companies' main characteristics, and confirming the need for the identification of process models in this context.

The other four papers explore similar concepts and motivations for process models: the type of work to be offshore outsourced over time, including risk analysis and a systematic process for defining type of projects and activities to be developed by a third-party company [2,25,36,54].

5.2.4. Studies describing the need for process models from a technical perspective

We found five studies describing the need for process models based on a technical perspective (three related to offshoring, one related to offshore outsourcing, and one related to internal offshoring). In their study, Hanisch and Corbitt [30] address the importance of communication for the Requirements Engineering process within globally distributed teams. They claimed that the use of communication technologies, such as e-mail, phone, and video conference evolve over time as the team becomes more mature, and suggested that there could be a sequence of steps in this evolution. In addition, Meyer [43], Sengupta et al. [59], and Ramasubbu and Krishna [51] addressed the need for updating existing maturity and capability models for software development

in order to include specific distributed development practices, and made specific recommendations. While Ramasubbu and Krishna [51] and Meyer [43] discuss offshoring in general, Sengupta et al. [59] have identified such recommendations based on a case study conducted in the internal offshoring of software development. Finally, the study carried out by Casey and Richardson [14] covered the need for better preparation of distributed teams, suggesting that distributed team experience evolves over time and this learning curve should not be underestimated.

5.2.5. Studies describing the need for process models from both perspectives

We found five studies describing the need for process models based on both technical and business levels (three related to offshore outsourcing, one related to offshoring and one related to internal offshoring). Mirani [44], for example, addressed the concept of procedural coordination techniques to be applied within distributed teams and organizations in an offshoring environment. The author argues that the evolution and success of an offshore software task is critically dependent on managing an inherent interdependence between onshore and offshore teams, as well as the strategy employed (for this reason, the scope involves both perspectives). Regarding offshore outsourcing, Khan, Currie and Guah [36] identified some fundamentals that usually drive offshore outsourcing evolution, related to both business and technical perspectives. They include contract, trade policy, quality, project management, expertise, trust, security, infrastructure, and culture. On the other hand, Matloff [41] reports not only the need for better planning of the offshoring strategy, but also the preparation and evolution of individuals to deal with offshoring (though the paper is focused on offshore outsourcing). In another study of offshore outsourcing (specifically the Indian software services industry), Ethiraj et al. [26] highlight the importance of capabilities, saying that they are context-specific, and fruitful research related to capabilities identification might emanate from in-depth study of the capabilities specific to a context, including strategic and technical capabilities, client-specific capabilities and project-management capabilities. Finally, Chua and Pan [16] investigated knowledge management in internal offshoring, identifying key activities that have to be developed over time in order to foster a knowledge management environment in a global context (including, for example, to determine whether or not to offshore a team, knowledge transfer, knowledge integration), and suggest that this should also evolve over time.

5.2.6. Other studies describing the need for DSD process models

Four studies could not be classified into one of the three types defined in Section 3. All of them discuss distributed development from a technical perspective, as well as suggesting the need for process models in this context, but the studies do not explicitly deal with global environments. The first one suggests that a software development process should evolve over time to meet distributed software development needs, but the author does not present any specific suggestions [4]. In another study, conducted by Evaristo et al. [27], the authors identified several factors to characterize what "distributed" means when discussing the management of distributed projects, suggesting some dimensions that might evolve over time. In addition, Stewart and Gosain [62] conducted a study of the open-source community, trying to identify the virtual teams' evolution within a project. The authors found that trust and shared understanding should come in the early stages of a project, and cited People-CMM as a model that could guide personal development, together with specific practices that have to be identified for distributed projects. Finally, McNab and Sarker [42] discussed the importance and evolution of trust in distributed projects, also suggesting that trust development practices should be developed from the beginning of a distributed project.

5.3. Summarizing the studies found

As part of our research question, we wanted to identify the main characteristics of each study and possible gaps and opportunities for future research. Part of this question was answered with the quantitative data. In the qualitative analysis, additional data was used to support a complementary evaluation of each study. Table 15 sets forth the studies characterizing DSD process models.

Table 16 sets forth the studies characterizing the need for DSD process models.

In summary, regarding the models, most of the studies did not explicitly define practices for each level proposed, having only a general description of what is expected on a certain level. Moreover, not many proposals have been empirically evaluated or tested [12,45,52].

Regarding the studies describing the needs for DSD process models, most of them were based on qualitative data, which is a characteristic of this type of study. The need for helpful DSD process models was mentioned by a majority of authors, and we found four main recurring themes. Authors seemed to agree on the need for DSD organizations to focus on:

- A *strategy selection process* that highlights the DSD-related decisions that have to be made and the reasons for selecting one strategy over another, the types of projects that will be distributed, and the tasks to be performed by distributed teams. (Mentioned in 8 studies.)
- Assessing *practices and capabilities of individuals* on DSD teams; understanding whether team members have a sufficient understanding of software development practices for deployment in a global perspective (5 studies).
- Extending *existing maturity models such as CMMI* to include specific DSD practices (3 studies).
- Documenting and following general software engineering *best practices for projects*, such as project management structure, requirements engineering, project life cycle, etc. (3 studies).

Table 15
Studies characterizing DSD process models.

Study	Levels	Practices	Scope	Empirically evaluated?
P1	3 Levels	24 Practices in 4 areas	Maturity of collaboration	Yes
P2	15 Levels	Not defined	Maturity of collaboration	No
P11	5 Levels	84 Practices in 10 areas	Capabilities for suppliers	No
P16	5 Levels	Not defined	Capabilities for an offshoring relationship	No
P20	Not defined	Not defined	Capabilities for clients	No
P27	4 Levels	Not defined	Capabilities for an offshoring relationship	Yes
P29	3 Levels	Not defined	Capabilities for an offshoring relationship	Yes
P30	5 Levels	Not defined	Capabilities for establishing an offshore subsidiary	No

Table 16
Studies characterizing the need for DSD process models.

Study	Scope	Main concepts	Empirical?
P3	Evolution of supplier selection process, and the need to reach high maturity in a global process	Supplier selection – strategy and DSD decisions	Industrial report
P4	Better preparation of individuals working on distributed projects	Capabilities for individuals	Industrial report
P6	Research agenda for distributed software development	Extending existing maturity models such as CMMI	Qualitative
P7	Investigation of effects of dispersion on productivity and quality in high quality organizations	Extending existing maturity models such as CMMI	Quantitative
P8	One of the first papers to discuss distributed development, at ICSE 98	Capabilities for individuals	Not defined
P9	Challenges for global teams	Capabilities for individuals	Qualitative
P10	Study of virtual teams evolution (Open source)	Capabilities for individuals	Quantitative
P12	Importance of capabilities	Best practices for projects	Quantitative
P13	Requirements engineering challenges in DSD	Best practices for projects	Qualitative
P15	Reasons for selection of offshore outsourcing from a strategic point of view	Strategy	Qualitative
P17	Discuss trust based on the desired evolution within distributed teams	Capabilities for individuals	Qualitative Quantitative
P18	Interview with executives on many levels of the SITO model to understand decisions for each company	Strategy and DSD decisions	Qualitative
P19	Description of companies experience and evolution	Strategy and DSD decisions	Industrial report
P21	Understand how a company moved from onshore to offshore and the importance of knowledge management in this process	Strategy and DSD decisions	Qualitative
P23	Offshore outsourcing challenges identified based on empirical data and literature review	Strategy and DSD decisions	Qualitative
P24	Evolution of offshore strategies in two companies	Strategy and DSD decisions	Qualitative
P25	Understand what "distributed" means when managing distributed projects	Best practices for projects	Qualitative
P26	India experience and potential on DSD	Strategy and DSD decisions	Qualitative
P28	Importance of improvement on existent maturity and capability models to support DSD	Extending existing maturity models such as CMMI	Industrial report

Overall, the vast majority (18 out of 19) of these studies were based either on empirical (usually qualitative) data or industrial experience.

5.4. Limitations

Systematic review is a useful method which, based on a research question and a detailed planning, searches for primary papers within a specific domain. But as any other method, there are some limitations. We address three main limitations: the first is related to the number and the sources (libraries) selected; the second refers to the reliability of the paper classification method; and the third is related to the quality of the search engines.

First of all, we did not look into every possible source. Eight digital libraries were selected based on experiences shared by other groups [9,20,23,24] and on the subject under review. First, by selecting the libraries in our review, we increased our range of search within the SE domain. Since DSD literature is documented in both SE and IS domains, we added two libraries from the IS domain (AIS eLibrary and the proceedings of ECIS) to cover another significant number of primary papers, and important IS conferences such as ECIS, ICIS, and AMCIS, as referred to in Gonzalez et al. [29]. Other IS papers were covered by looking into the HICSS proceedings (using IEEEXplore DL), Compendex, INSPEC, and Elsevier ScienceDirect. However, we did not search for books, neither other sources of IS papers that could focus on studies from a business perspective. However, we believe that the results presented provide a good indication of the "state of the art" and the "state of the practice" of DSD process models in global scenarios.

Second, the classification process based on some criteria could be subjective. To minimize this limitation, a two-step approach was planned for paper selection, as explained in Section 4, and another two-step approach was planned for paper categorization. All papers were reviewed at least three times by the same researcher. To define the criteria, as well as the concepts for paper categorization/classification, we engaged into many other interactions with at least two other researchers outside the systematic review. The second step was the review of the categorization with at least one other researcher.

Third, with regard to the quality of search engines, we could not use the same search string in all digital libraries. We found two of the search engines to be excessively simplistic (AIS eLibrary and the website with ECIS papers), offering no support to logic operators and no clear instructions on how to do the search. For this reason, we had to search each keyword individually. Another search engine that we used (ACM DL) did not support complex search strings and thus we combined a subset of keywords and split the search into several searches. The result was positive, although we would have used less effort if some of the selected search engines provided better support to users.

6. Discussion and conclusions

Both quantitative and qualitative analyses present interesting findings related to DSD process models. Based on the data collected, it was possible to identify the following conclusions.

#1: There is a need for more studies addressing the technical aspects of process models in DSD.

Most of the models proposed are related to a business perspective. This creates the opportunity for SE researchers to explore and understand DSD process models from a technical perspective. There are already several papers published in the SE literature that take this viewpoint (i.e. [51,59]).

#2: There is a need for more studies focusing on project-level analysis, as opposed to organizations only.

Since most of the models are focused upon a business perspective, not surprisingly most of the papers are focused on organization-level analysis, instead of on people or projects. But there is also a need to further our understanding of DSD process models within a project, or a set of projects, and not just from a strategic perspective. This is also a topic that would benefit from further research that could include analysis of existing maturity or capability models, which have projects as part of the scope (e.g., CMMI), interpreting how they can adapt to a DSD environment, where

we can have several stakeholders participating (e.g., including more than one subsidiary, several teams, and many locations). Some research included in this review has taken this approach [52], and some papers have already shared these ideas [43,59].

#3: There is an opportunity for studies to employ quantitative data analysis methods.

Most of the capability, maturity or stage models that have been proposed so far are largely based on qualitative data analysis. This is the case of CMMI [15], or the eSCM-SP [33], for example. This was also found in our systematic review, where a significant number of papers (half of them) conducted the research based on qualitative methods. One of the reasons is that mostly the nature of the phenomenon is not known beforehand so an exploratory strategy is followed, using case studies for example, utilizing interviews or other qualitative data collection methods. Quantitative data analysis, however, offers the opportunity to statistically evaluate the findings identified through qualitative methods. As an example, in the study developed by Ramasubbu et al. [52] the authors have used quantitative analysis methods to propose and evaluate a process maturity framework for managing distributed development.

#4: There is a need for more studies related to the internal offshoring model.

The internal offshoring business model, also known as offshore insourcing, captive subsidiaries, or wholly-owned subsidiaries, is the least studied model. This is surprising, given the large number of companies involved with this strategy. According to Ramamani [50], from over 900 companies associated with NASSCOM (National Association of Software Companies), an Indian organization that represents all the companies in the Information Technology industry, more than 300 are wholly-owned subsidiaries. Consequently, the challenges and practices should also be understood for this type of DSD. And this is another opportunity for DSD researchers.

#5: There is a need for more studies on DSD on the vendor's side.

In a literature review of information systems outsourcing, Gonzalez et al. [29] identified 131 papers published in IS journals, where they found that only 16% of the papers explored outsourcing from the perspective of the service provider (or vendor). In our systematic review, most of the models proposed (Table 7) were focused on the phenomenon at the vendor's side (58%). We also found twice as many papers arguing for the need for such models on the vendor's side (Table 8). In total, 67% of the papers we found were exploring the vendor's side (33% of them exploring both sides). This is clearly a difference between the two reviews, and they also had different purposes. While Gonzalez et al. [29] searched for any type of papers exploring IS outsourcing, looking into IS journals only, we focused on globally distributed development, and searched both SE and IE domains, including conferences and also workshops. Thus, our review was based on distributed software development, and the many outsourcing arrangements that can ultimately create a distributed environment (locally or globally). In contrast, Gonzalez et al. [29] have analyzed outsourcing from an IS perspective, focusing on business drivers and decisions. An interesting conclusion is that the study of outsourcing, in the IS domain, is not only more concentrated on strategic decisions, and outsourcing relationship, but is also more client-oriented. Based on our results, there appears to be a need to better understand the vendor's viewpoint from a technical perspective. This conclusion is corroborated by a recent study in the area of distributed software development [61].

#6: Distributed software development should be better contextualized.

Almost half of the studies found in this systematic review fail to explain the type of DSD environment (see Table 6). But with the development of this area, it is becoming necessary to better contextualize the type of distribution under study [31]. A successful practice carried out in a locally distributed environment might not work well in a globally distributed scenario. As DSD becomes more mature, it is also necessary to differentiate the many types of distribution, as well as their implications.

#7: DSD process models could be more detailed.

In our systematic review, most of the models have described levels, but only two of them have identified specific practices for each level. We understand that such a model is not necessarily need to have detailed descriptions. In some cases, the main goal is to identify standard behaviors over time, or evolution steps. But what was found in most of the studies is a general description of the levels, in a way that makes it hard to understand or to put it into practice. For this reason, we recommend researchers provide a better description of the models being proposed and report all the necessary details.

#8: DSD process models should be empirically evaluated.

Most of the models found were not evaluated. A certain model may be part of the outcome of a certain study, but it should not be relied on if it is not properly evaluated. One possible argument for this lack of empirical evaluation is how new this area actually is, and that most of the models proposed are from 1, 2, or 3 years ago. Notwithstanding, researchers should also put effort into evaluation activities, following a scientific approach. This is also corroborated by a recent study where the authors suggest that we should have more studies focused on the evaluation of methods, practices and solutions for distributed and global software development [61].

Acknowledgments

This study was developed in collaboration between the MuNDDoS research group in Distributed Software Development, at PUCRS, Brazil, and the SEGAL Lab, at UVIC, Canada. The study was also partially supported by the Research Group on Distributed Software Development of the PDTI program, financed by Dell Computers of Brazil Ltd. (Law 8248/91), and partially supported by CAPES (Brazilian Ministry of Education), financed by the CAPES Ph.D. Internship Program, process number 426006-6.

References

[1] O. Adelakun, IT outsourcing maturity model, in: European Conference on Information Systems, vol. 2, 2004.
[2] A. Arora, V.S. Arunachalam, J. Asundi, R. Fernandes, The indian software industry, Research Policy 30 (8) (2001) 1267–1287.
[3] W. Aspray, F. Mayadas, M.Y. Vardi (Eds.), Globalization and Offshoring of Software, A Report of the ACM Job Migration Task Force, Association for Computing Machinery, 2006.
[4] M. Aoyama, Agile software process and its experience, ICSE (1998).
[5] S. Balaji, C. Ranganathan, Exploring the key capabilities for offshore IS sourcing, in: International Conference on Information Systems, 2006, pp. 543–552.
[6] S. Balaji, S.A. Brown, Strategic IS sourcing and dynamic capabilities: bridging the gap, HICSS (2005).
[7] J. Biolchini, P.G. Mian, A.C.C. Natali, G.H. Travassos, Systematic Review in Software Engineering, Technical Report TR – ES 679/05, COPPE/UFRJ, 2005.
[8] B. Boehm, A view of 20th and 21st century software engineering, in: Proceedings of the 28th International Conference on Software Engineering, Shanghai, 2006, pp. 12–29.

[9] P. Brereton, B.A. Kitchenham, D. Budgen, M. Turner, M. Khalil, Lessons from applying the systematic literature review process within the software engineering domain, Journal of Systems and Software 80 (2007) 517–583.
[10] E. Carmel, The Offshoring Stage Model: an epilogue, April 2005, <auapps.american.edu/~carmel/papers/epilogue.pdf> (accessed on November 2007).
[11] E. Carmel, P. Tjia, Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce, Cambridge University Press, Cambridge, 2005.
[12] E. Carmel, R. Agarwal, The maturation of offshore sourcing of information technology work, MIS Quarterly Executive 1 (2) (2002) 65–77.
[13] E. Carmel, Global Software Teams – Collaborating Across Borders and Time-Zones, Upper Saddle River, NJ, Prentice Hall, 1999.
[14] V. Casey, I. Richardson, Uncovering the Reality within Virtual Software Teams, International Workshop on Global Software Development for the Practitioner, 2006, pp. 66–72.
[15] M.B. Chrissis, M. Konrad, S. Shrum, CMMI: Guidelines for Process Integration and Product Improvement, 2nd ed., Addison-Wesley, SEI Series on Software Engineering, EUA, 2006.
[16] A.L. Chua, S. Pan, Knowledge transfer in offshore insourcing, in: International Conference on Information Systems, 2006.
[17] D. Cruzes, V. Basili, F. Shull, M. Jino, Automated information extraction from empirical software engineering literature: is that possible? in: First International Symposium on Empirical Software Engineering and Measurement (ESEM), 2007, pp. 491–493.
[18] D. Damian, D. Moitra, Guest Editors' Introduction: Global Software Development: How far Have We Come?, IEEE Software 23 (5) (2006) 17–19
[19] D. Damian, D. Zowghi, The impact of stakeholders' geographical distribution on requirements engineering in a multi-site development organization, in: 10th IEEE International Conference on Requirements Engineering (RE'02), Essen, Germany, 2002, pp. 319–328.
[20] A.C. Dias Neto, R. Subramanyan, M. Vieira, G.H. Travassos, Characterization of Model-based Software Testing Approaches, Technical Report TR – ES 713/07, COPPE/UFRJ, 2007.
[21] J. Dibbern, T. Gole, R. Hirschheim, B. Jayalaka, Information systems outsourcing: a survey and analysis of the literature, ACM SIGMIS Database 35 (4) (2004) 6–102.
[22] O. Dieste, A.G Padua, Developing search strategies for detecting relevant experiments for systematic reviews, in: First International Symposium on Empirical Software Engineering and Measurement (ESEM), 2007, pp. 215–224.
[23] T. Dyba, T. Dingsoyr, G.K. Hanssen, Applying systematic reviews to diverse study types: an experience report, in: First International Symposium on Empirical Software Engineering and Measurement (ESEM), 2007, pp. 225–234.
[24] T. Dyba, V.B. Kampenes, D.I.K. Sjoberg, A systematic review of statistical power in software engineering experiments, Information and Software Technology xx (2005) 1–11.
[25] C. Ebert, Optimizing supplier management in global software engineering, in: International Conference on Global Software Engineering, Munich, Germany, 2007, pp. 177–185.
[26] S.K. Ethiraj, P. Kale, M.S. Krishnan, J.V. Singh, Where do capabilities come from and how do they matter? A study in the software services industry, Strategic Management Journal 26 (1) (2005) 25–45.
[27] J.R. Evaristo, R. Scudder, K.C. Desouza, O. Sato, A dimensional analysis of geographically distributed project teams: a case study, Journal of Engineering and Technology Management 21 (3) (2004) 175–189.
[28] B. Gannon, D.W. Wilson, IS offshoring: a proposed maturity model of offshore IS suppliers, in: European Conference on Information Systems, 2007, pp. 950–960.
[29] R. Gonzalez, J. Gasco, J. Llopis, Information systems outsourcing: a literature analysis, Information & Management 43 (2006) 821–834.
[30] J. Hanisch, B.J. Corbitt, Requirements engineering during global software development: some impediments to the requirements engineering process: a case study, in: European Conference on Information Systems, 2004.
[31] J.D. Herbsleb, Global software engineering: the future of socio-technical coordination, in: 29th International Conference on Software Engineering, Minneapolis, USA, 2007, pp. 188–198.
[32] J.D. Herbsleb, D. Moitra, Guest editors' introduction: global software development, IEEE Software 18 (2) (2001) 16–20.
[33] E.B. Hyder, K.M. Heston, M.C. Paulk, The sCM-SP v2.01: The eSourcing Capability Model for Service Providers (eSCM-SP) v2.01 – Practice Details, Carnegie Mellon University, Pittsburgh, 2006. <http://itsqc.cs.cmu.edu/>.
[34] G. Höfner, V.S. Mani, TAPER: a generic framework for establishing an offshore development center, in: International Conference on Global Software Engineering, Munich, Germany, 2007, pp. 162–172.
[35] D. Karolak, Global Software Development – Managing Virtual Teams and Environments, IEEE Computer Society Press, 1998.
[36] N. Khan, W. Currie, M. Guah, Developing a Model for Offshore Outsourcing, in: Americas Conference on Information Systems, 2003, pp. 996–1003.
[37] B. Kitchenham, Procedures for Performing Systematic Reviews, Joint Technical Report SE0401 and NICTA Technical Report 0400011T.1, Software Engineering Group, Department of Computer Science, Keele University, 2004.
[38] B.A. Kitchenham, T. Dyba, M. Jorgensen evidence-based software engineering, in: 26th International Conference on Software Engineering (ICSE), Scotland, 2004, pp. 273–281.
[39] K. Kumar, L. Willcocks, Offshore Outsourcing: A Country to Far? in: European Conference on Information Systems, Lisbon, Portugal, 1996, pp. 1309–1325.
[40] S. Lasser, M. Heiss, Collaboration maturity and the offshoring cost barrier: the tradeoff between flexibility in team composition and cross-site communication effort in geographically distributed development projects, in: International professional Communication Conference, 2005, pp. 718–728.
[41] N. Matloff, Offshoring: what can go wrong, IT Professional (2005) 39–45.
[42] A.L. McNab, S. Sarker, Evolution of trust in distributed software development teams: a punctuated equilibrium model, in: Americas Conference on Information Systems, 2006.
[43] B. Meyer, The unspoken revolution in software engineering, IEEE Computer 39 (1) (2006) 121–123. 124.
[44] R. Mirani, Procedural coordination and offshored software tasks: lessons from two case studies, Information & Management 44 (2) (2007) 216–230.
[45] R. Mirani, Client–vendor relationship in offshore applications development: an evolutionary framework, Information Resources Management Journal 19 (4) (2006) 71–86.
[46] R. Nolan, Managing the computer resource: a stage hypothesis, Communications of the ACM 16 (7) (1973) 399–405.
[47] B.J. Oates, Researching Information Systems and Computing, Sage Publications, CA, 2006.
[48] R. Prikladnicki, D. Damian, J.L.N. Audy, Patterns of Evolution in the Practice of Distributed Software Development: Quantitative Results from a Systematic Review," Evaluation and Assessment in Software Engineering, Bari, Italy, 2008, pp. 1–10.
[49] R. Prikladnicki, J.L.N. Audy, D. Damian, T.C. Oliveira, Distributed software development: practices and challenges in different business strategies of offshoring and onshoring, in: International Conference on Global Software Engineering, Germany, 2007, pp. 262–274.
[50] M. Ramamani, Offshore subsidiary engagement effectiveness: the role of subsidiary capabilities and parent – subsidiary interdependence, in: Conference of Midwest United States Association for IS, 2006, pp. 75–80.
[51] N. Ramasubbu, R.K. Balan, Globally Distributed Software Development Project Performance: an Empirical Analysis, ACM SIGSOFT Symposium on the Foundations of Software Engineering, 2007, pp. 125–134.
[52] N. Ramasubbu, M.S. Krishnan, P. Kompali, Leveraging global resources: a process maturity framework for managing distributed development, IEEE Software 22 (3) (2005) 80–86.
[53] C. Ranganathan, S. Balaji, Critical capabilities for offshore outsourcing of information systems, MIS Quarterly Executive 6 (30) (2007) 147–164.
[54] C. Ranganathan, P. Krishnan, Crafting and executing an offshore IT sourcing strategy: globShop's experience, in: International Conference on Information Systems, 2006.
[55] M. Robinson, R. Kalakota, Offshore Outsourcing: Business Models, ROI and Best Practices, Mivar Press, GA, USA, 2004.
[56] S. Sakthivel, Managing risk in offshore systems development, Communications of the ACM 50 (4) (2007) 69–75.
[57] R. Sangwan, M. Bass, N. Mullick, D.J. Paulish, J. Kazmeier, Global Software Development Handbook, Auerbach Publications, Boca Raton, NY, 2007.
[58] L. Sayeed, A qualitative investigation of IS offshore sourcing, in: Americas Conference on Information Systems, 2006.
[59] B. Sengupta, S. Chandra, V. Sinha, A research agenda for distributed software development, in: 28th International Conference on Software Engineering, Shanghai, 2006, pp. 731–740.
[60] K.V. Siakas, B. Balstrup, Software outsourcing quality achieved by global virtual collaboration, Software Process Improvement and Practice 11 (3) (2006) 319–328.
[61] D. Smite, C. Wohlin, T. Gorschek, R. Feldt, Empirical evidence in global software engineering: a systematic review, Empirical Software Engineering Journal (2009).
[62] K.J. Stewart, S. Gosain, The moderating role of development stage in free/open source software project performance, Software Process Improvement and Practice 11 (2) (2006) 177–191.
[63] C.H. Szymanski, R. Prikladnicki, The evolution of the internal offshore software development model at Dell Inc, in: International Conference on Global Software Engineering, Germany, 2007, pp. 40–50.
[64] B. Tuckman, Develomental sequence in small groups, Psychological Bulletin 23 (1965) 384–399.