Distances between Requirements Engineering and Later Software Development Activities: A Systematic Map

Bjarnason, Elizabeth

Published in:
Lectur Notes in Computer Science/ Requirements Engineering: Foundation for Software Quality

DOI:
10.1007/978-3-642-37422-7_21

2013

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

Published in Proceedings of REFSQ 2013, pp. 292-307.

Distances between Requirements Engineering and Later Software Development Activities: A Systematic Map

Elizabeth Bjarnason

Department of Computer Science, Lund University
SE-221 00 Lund, Sweden
elizabeth.bjarnason@cs.lth.se

Abstract. **[Context and Motivation]** The main role of requirements engineering (RE) is to guide development projects towards implementing products that will appeal to customers. To effectively achieve this RE needs to be coordinated with and clearly communicated to the later software development activities. **[Question/Problem]** Communication gaps between RE and other development activities reduce coordination and alignment, and can lead to project delays and failure to meet customer needs. **[Principle ideas/results]** The main hypothesis is that coordination is enhanced by proximity to RE roles and artefacts, and that distances to later activities increase the effort needed to align requirements with other development work. Thirteen RE-related distances have been identified through a systematic map of existing research. **[Contribution]** Reported distances are mapped according to research type, RE activity and later software development activities. The results provide an overview of RE distances and can be used a basis for defining a theoretical framework.

Keywords: systematic map, distance, requirements, software development

1 Introduction

Effective requirements engineering (RE) greatly depends upon successful coordination [12, 16] and communication of requirements with the downstream development activities [6, 24], e.g. design, implementation, and testing. Merely producing a perfect requirements specification is not sufficient. Rather it is vital to ensure that the requirements are clearly understood and agreed with implementation-near roles, and that sufficient requirements information is available for later development activities [14, 24]. Communication gaps between people may contribute to project delays, software quality issues and even failure to meet customer expectations [6].

Within global software development (GSD), project teams and members are globally distributed. These geographical distances between people have been found to negatively affect the communication and thereby also the coordination and success of the distributed development. In addition to geographical distance, socio-cultural and temporal distances have been found to be in play within GSD [1]. Agerfalk et al. have defined a theoretical framework of these different types of distances and how they affect communication, coordination and control [1]. However, coordination and

adfa, p. 1, 2011.

communication is also a challenge within non-distributed development, in particular for large development organizations and projects [6, 12].

Our main hypothesis is that distance plays an important role in development, whether distributed or not. In particular, the distances between RE and later software development activities may impact project effectiveness and efficiency. The systematic mapping study reported in this paper provides an overview of existing knowledge of RE-related distances within software engineering research.

Work related to the targeted area is described in Section 2. Section 3 outlines the research method while Section 4 presents the results, which are then discussed in Section 5. Finally, the paper is concluded in Section 6.

2 Software Development and RE

'Requirements are the basic building blocks gluing together [the] different ... activities needed to define, develop, implement, build, operate, service, and phase out a product and its related variants.' [16] However, in general most people focus mainly on one area of expertise: RE, project management, architecture, implementation, testing etc. Both in practice and in research, there is generally weak insight and knowledge into how to leverage software development by improving on the interaction and coordination of RE with later activities within software development.

In contrast, concurrent engineering [22] is an approach to product development where several engineering activities are carried out concurrently (at the same time by the same project team) with extensive feedback and iteration. The developers are to consider all aspects of the development cycle from requirements to cost and quality. Reported gains for this approach include increased efficiency, productivity and quality, and reduced waste and shortened lead times [22]. A concurrent approach is applied within agile software development by integrating the activities for requirements, architecture, implementation and testing, and the claimed gains are similar to those for concurrent engineering, including increased responsiveness to change.

Damian et al. found that improved RE practices within a more traditional plan-based development project may have an effect also on later software development activities. Effective RE can thereby support increased development effectiveness and augment the efficiency and productivity of the other development activities, and lead to improvements for a wide range of software development aspects, e.g. project planning, managing feature creep, testing, defects, rework, and product quality [14]. This indicates that RE can play a vital role for the total development effort, if RE is effective and well-coordinated with later development activities.

Requirements and design are interdependent activities. While design (either by architecture or directly during implementation) aims to realize the requirements, architectural and technical limitations, and new technical possibilities may affect the requirements and, thus, require requirements changes. For these reasons, it has been suggested that RE should be intertwined and performed in parallel with design [25, 29]. Nuseibeh et al. have designed a method that does this while still separating between problem and solution structure. The method is receptive to handling change in an efficient way, allows early exploration of the problem space, and enables engineers to identify requirements and match them to available components and products [25]. Similarly, Fricker et al. found that aligning requirements and architecture through a negotiation process between product management and architecture led to identifying missed requirements, and to a shared requirements understanding that mitigated problems related to missed requirements and requirements dependencies [17].

Coordination and alignment of RE and testing. We have previously reported on the situation of alignment between RE and testing in industry [5]. Two of the main challenges was found to be RE quality and the softer aspects of development, i.e. communication and collaboration [5]. Furthermore, a number of industrial practices for supporting alignment have been reported both by Bjarnason et al and by Uusitalo et al. These practices include traceability between requirements and test cases, and increased communication between roles [5, 31], e.g. by involving testers early in the project and in requirement reviews, and by establishing communication between testers and requirement owners [31]. Similarly, Marczak et al. found that in requirements-driven collaboration, close communication between requirements and testing depends on key roles which when absent cause disruptions within the development team [24].

3 Research Method

The systematic map reported in this paper was performed based on guidelines for systematic mapping [26] and insights for systematic literature reviews [7]. The steps taken in designing and performing the study are described below. The study protocol and full list of papers included in the study can be found on-line [4].

3.1 Research Questions

With the aim of locating research into RE distances within/between RE and later software development activities, the following research questions were formulated:

RQ1: Which RE-related distances are reported in peer-reviewed literature?
RQ2: To which extent is ‘distance’ used in GSD versus non-GSD papers?
RQ3: For which activities within RE has the concept of distance been researched?
RQ4: Towards which later development activities are RE distances investigated?

3.2 Search Strategy

The defined scope covers RE research and its intersection with later development activities. Papers focusing on non-RE topics were excluded, while general software development papers were included. Based on scope and research questions, search keywords were defined. The initial keywords were searched in well-known databases, e.g. IEEE Xplore, SciVerse. Based on search results, the keyword, scope and research questions were refined and search strings reformulated. The set of databases was expanded and re-searched for relevant papers.

3.3 Data Sources

Searches into the following databases are included in this mapping study:

1. IEEE Xplore (http://ieeexplore.ieee.org) covers computer science, electrical engineering, and electronic subject areas. Full-text and bibliographic access to almost 3 million of IEEE's publication including transactions, journals, magazines and conference proceedings published are provided.
2. Elsevier's SciVerse (http://sciencedirect.com) covers papers from more than 2,500 computer science and engineering journal.
3. ACM Digital Library (http://dl.acm.org) provides access to ACM journals, proceedings and transaction including ACM computing literature.
4. Inspec and Compendex provide access to huge amounts of scientific literature in many subjects including information technology, and are accessible via Engineering village's unified search interface (http://www.engineeringvillage2.org).

3.4 Data Retrieval

Search strings were constructed by combining the defined scope (software engineering OR software development OR requirements engineering) with the term 'distance'. The searches were limited to peer-reviewed material written in English. Material on 'distance learning' was excluded in the search to avoid a large number of irrelevant hits. The searches were limited to title, abstract and keywords.

3.5 Screening Process

The final searches yielded 2,427 papers (see Table 1). A title scan resulted in 161 relevant papers. The full references, abstract and search source of these papers were then stored in MS Excel (available on-line [4]). Duplicates were removed; 148 unique papers. These papers were then included or excluded based on the abstracts. The inclusion/exclusion decisions for both title and abstract were cautious, i.e. when in doubt the paper was included. When an abstract contained insufficient information, the introduction was reviewed. In total 53 papers were included in the final set.

3.6 Data Extraction, Classification and Synthesis

During data extraction and mapping, a classification scheme was developed according to guidelines provided by [26]. A set of keyword were identified through exploratory coding of the abstracts, and then clustered into the categories of the map. In a few cases, the abstract was insufficient and parts of the full text were reviewed to ensure a correct understanding. Two sets of categories were identified. One related to context and focus of the research (main development activity, specific RE activity, and organisational distribution) and the other related to distance type.

The initial set of keywords for distance types was refined through analysing parts of the full paper text. In some cases, *forwards snowballing* was applied to locate addi-

Table 1. Number of papers in each step of the screening process.

Source	Initial selection	Title review	Abstract review
SciVerse	51	7	2
IEEE Xplore	79	4	1
ACM Digital Library	1,951	52	33
Inspec	346	11	0
Compendex		74	17
TOTAL	**2,427**	**148**	**53**

tional papers, which were consulted to ensure a correct understanding of the used terms. The coding of all included papers was then revised to match the final set of codes. The final coding of the included papers is available on-line [4].

Finally, a synthesis was performed on the included papers for each distance type to identify how the term is defined and applied, and if any causal relationships are reported for that term. In some cases, additional papers were located through forwards snowballing. For example, in GSD papers distances would typically be mentioned with a reference to previous work. In addition, for distances with only a few located papers supplementary searches on the specific distance type names were performed to identify additional papers. Parts of the full text was analysed for the synthesis, in particular introduction and conclusions sections, and all mentions of the term 'distance'.

4 Results

4.1 Demographics of Retrieved Literature (and RQ2)

The search and selection resulted in 53 individual peer-reviewed papers. The majority of these (42) were within GSD. The distribution of papers over time, split into GSD / non-GSD context, is shown in Figure 1. The maximum was in 2009 with 11 papers. It is worth noting that within GSD a framework for categorizing GSD challenges based on three types of distances was published in 2005 [1] and that the following 4 years (2006-2009) have the largest number of papers found in this study.

The research type for each paper was classified according to the scheme suggested by Wieringa et al. The following categories were considered in this study [33]:

1. *Evaluation research* investigates a problem or technique in practice and provides new knowledge of causal or logical relationships.
2. *Solution proposals* present a solution without a full-blown validation.
3. *Validation research* presents a solution proposal validated outside of industrial practice, e.g. experiments, prototyping, theoretical proof etc.
4. *Philosophical papers* sketch new theories or frameworks.
5. *Experience papers* describe the author's personal experience and may contain anecdotal evidence.

The distribution of the included papers according to research type and distribution context (GSD or non-GSD) is shown in Figure 2. The numbers indicate that, for the GSD context, more empirical evaluations and theoretical frameworks on the concept

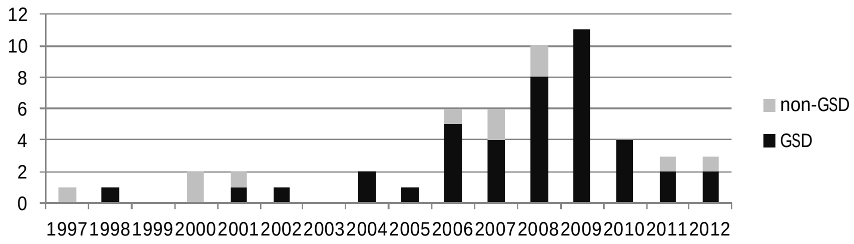


Fig. 1. Number of papers per year, categorised according to GSD or non-GSD context.

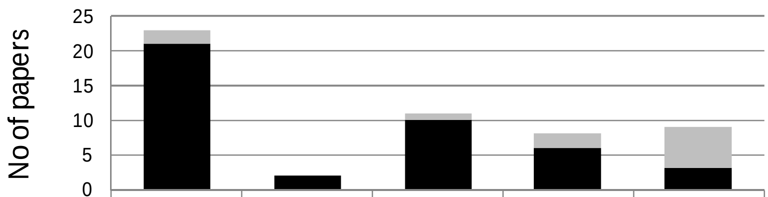
of distance have been researched than for the non-GSD context. For general development (non-GSD), the majority of included papers are in the form of validation research, indicating that more evaluation research is required into distances in the general software development context to establish foundations for more mature knowledge and for establishing theories based on empirical evidence.

4.2 Type of Distances (RQ1)

This study identifies thirteen distances. Eight of these, are distances between people, e.g. between roles, teams and organizations, while four address distances between artefacts. One distance concerns distance between an artefact (e.g. formal model) and reality. Unsurprisingly (since the majority of included papers address GSD), the most commonly referred distances are the ones defined within GSD, i.e. geographical, socio-cultural and temporal distances. Table 2 shows an overview of the number of papers for each distance. (The distances are described in Section 4.4.)

4.3 RE Activities (RQ3) and Later Software Development Activities (RQ4)

Distances were found in papers related to RE, project management, design, implementation, tools and processes. More than half of the papers (29 of 53) cover software development in general, while a third of the papers (17 of 53) cover RE, and a fourth (8 of 53) cover implementation. The numbers indicate that RE is acknowledged as an important activity for which distances are relevant to investigate. However, more research is needed to fully explore the field. In particular, research is needed on how RE distances relate to testing for which no papers were found, which is surprising



	Evaluation	Experience	Philosophical	Solution Proposal	Validation
non-GSD (11)	2		1	2	6
GSD (42)	21	2	10	6	3

Fig. 2. The number of papers per research type and GSD vs. non-GSD context.

Table 2. The number of papers per distance type and software development activity. The bar indicates relative amount. Papers covering several categories are counted for each category.

			General	RE	Impl	Tools	Prj mgmt	Design	Process
	TOTAL		*29*	*17*	*8*	*7*	*6*	*1*	*1*
PEOPLE	Geographical	*41*	27	7	6	6	3		
	Socio-cultural	*25*	17	3	4	1	2		
	Temporal	*15*	11	1	2	1	2		
	Power	*3*	3						
	Opinions	*2*		2		1			
	Psychological	*2*			2				
	Organisational	*1*		1					
	Cognitive	*1*		1					
ARTEFACTS	Similarity	*3*		3			3		
	Semantic	*2*	1	2					
	Syntactic	*1*		1					
	Impact	*1*		1				1	
OTHER	Adherence	*2*	1	1	1				1

considering that testing verifies that the requirements are fulfilled in the final product. A map of the number of papers per distance type and software development activity for which they were mentioned is shown in Table 2.

Of the 17 RE-specific papers, 7 address negotiation and 4 cover RE in general, while for handling changes, elicitation, specification, validation and traceability only the odd papers was found for each RE activity. 7 of the RE-specific papers purely address RE, while the others also cover software development in general (3), project management (3), tools (3) and implementation (1). Table 3 shows a map of RE-specific papers per development activity and RE activity.

4.4 RE Distances in Context

The systematic map identifies 13 RE distances between people, artefacts, and other entities. This section describes each distance based on included papers.

Distance between people

Geographical distance denotes ‘a directional measure of the effort required for one actor to visit another at the latter’s home site [or home work place]’ [1]. Even a geographical distance of 25 metres, i.e. within the same office building, has been found to reduce communication between engineers [2]. For off-shored projects where RE is geographically separated from other software development activities Dibbern et al. found that this distance can be a significant cost driver [15]. In particular, in cases where client-specific knowledge was crucial face-to-face collaboration was required for adequate knowledge transfer of domain knowledge and for requirements analysis and specification [15]. Tools for enhancing distributed group communication have been suggested for collaborative RE activities such as requirement negotiation and requirements traceability towards goals and design artefacts [18]. Calefato et al. found that computer-based communication provided better support for elicitation than for negotiation, and suggest that the general preference for face-to-face communication

Table 3. The number of RE-specific papers per RE activity and later activities. The bar indicates relative amount. Papers covering several categories are counted for each category.

		Negotiation	General RE	Specificaton	Analysis	Validation	Changes	Elicitation	Traceability
TOTAL		*7*	*4*	*3*	*3*	*2*	*1*	*1*	*1*
Pure RE	*7*	**4**	**1**	**1**	**1**	**1**		**1**	**1**
General	*3*			**2**	**2**	**1**			
Project managmt	*3*		**3**						
Tools	*3*	**3**							
Design	*1*						**1**		
Implementation	*0*								
Process	*0*								

might be explained by this weakness of computer-based negotiations [9]. In contrast, Damian found that when using technology for negotiating requirements the group's overall performance was not decreased compared to when negotiating face-to-face, and could even be more effective in integrating multiple stakeholders' needs [13]. Similarly, Wolf et al. found no significant delays for geographical distance in a case study. This was believed to be due to practices applied to bridge these distances (collaborative tools, and processes and practices adapted to distributed software teams), but may also be explained by the fact that the delays were quantified as opposed to qualitatively measured as for most other studies [35].

Temporal distance denotes 'a directional measure of the dislocation in time experienced by two actors wishing to interact' [1] due to different time zone, work shifts etc. In general, short temporal distances allow for timely synchronization between team members, while long temporal distances reduced the opportunities for synchronous communication and introduce delayed feedback [1]. Time zones and work shift schedules may work together to decrease temporal distance by adjusted office hours or utilized for working around the clock by passing on tasks between teams in different time zones [1]. Yousuf et al. suggest that when temporal distance is present certain requirements validation techniques which do not rely on synchronous communication are more suitable than others [36].

Socio-cultural distance denotes 'a directional measure of an actor's understanding of another actor's values and normative practices' [1] and includes organisational and national culture, language, individual motivations, work ethics, and politics. In general, communication is improved by low socio-cultural distance thereby reducing risk, while long socio-cultural distances increase the risk of misunderstandings and may make coordination harder [1]. However, long distances also have a potential for increased learning and access to a richer skill set, and be stimulating for innovation [1].

In the context of RE for GSD, Dibbern et al. found that cultural distance can be a significant cost driver for a company with off-shored projects. Increased costs may be incurred for transfer of knowledge of domain, requirements etc., and additional specification effort to ensure accurate requirements [15]. Yousuf et al. mention socio-cultural distance as potentially influencing requirements validation though without specifically analysing how [36]. Real-time machine translation has been proposed for requirements negotiation among stakeholders separated by language barriers, and found to not disrupt real-time interaction in text-based chat [9].

Opinion distance denotes a measure of the difference of opinion on a certain aspect of an item between two actors. This distance has been investigated between decision makers and stakeholders in requirements negotiations with the aim of supporting group decision by measuring the differences in linguistic opinions of alternatives based on multiple criteria [10]. Chakraborty and Chakraborty propose using a fuzzy distance measure to measure the distance between fuzzy clusters of the opinions in order to improve 'accuracy' of the decision by identifying dissimilar opinions [10]. Similarly, Zhu and Hipel propose a method for dealing with multi-stage information, i.e. when information about alternatives evolves over time [38].

Organisational distance denotes a measure of one organisational unit's understanding of another unit's goals and perspectives, e.g. concerning priority of customer requirements relative cost of code design and quality. The organisational distance between people involved in RE was categorised in a study on pairing on RE tasks as *internal* or *external* depending on if they are part of the development team or not [37]. The study suggests that sharing RE tasks is more effective when there is a shorter organisational distance due to less delay in the (shorter) communication paths [37].

Psychological distance denotes a measure of the perceived psychological (subjective) effort of an actor to communicate with another actor [27]. This distance has been researched for software development in general, though not specifically for RE. Prikladnicki has defined a measurement for the perceived distance between people. This measurement relates to the social dimension of psychological distance that addresses the distance of a stimulus (social object or event) from the perceiver's self, e.g. my best friend or a person from another culture [23]. The measurement was evaluated in a project with development distributed between Brazil and India. The study shows that the psychological distance does not necessarily correspond to the geographical distance, but to a high degree depends upon trust and communication though the impact of these factors varied per country and per role [27]. For example, a project engineer in Brazil perceived the lowest distance while a project manager (also in Brazil) perceived the highest psychological distance [27].

Power distance denotes a measure of the degree to which unequal distribution of power is accepted within a society [19]. This distance has been researched for software development in general, though not specifically for RE**.** This distance is one of the dimensions of socio-cultural distance and has been found to affect relationships within distributed development and thereby also the success of distribution [34]. Winkler et al. found that difference in power distances may negatively affect communication. For example, in a culture with a large power distance saying no or voicing criticism is avoided, detailed specifications are preferred and instructions are preferred from superiors rather than from peers. All of these factors pose a risk of complicating collaboration with team members used to shorter power distances and more open communication [34]. Wende and Philip found communication via instant messaging improved communication and, thus, enabled bridging power distances [32].

Cognitive distance denotes a measure of the difference between two actors' cognition, e.g. what they each know and are aware of. Yu and Sharp observed this distance in a case study on pairing on RE tasks and identified that when one person fills many roles communication is immediate since the cognitive distance between the roles is zero, which is beneficial for communication and coordination [37].

Distance between artefacts

Similarity distance denotes a measure of the similarity between an entity and another entity of the same type, e.g. project. This distance has been suggested as supporting the coordination between RE and project management, in particular for cost estimation of requirements. In analogy-based software effort estimation, the concept of similarity distance is used to identify completed projects with similar characteristics by measuring the Euclidian distance between project features [28], e.g. number of requirements, number of interfaces, project model etc. This approach has been validated using industrial data sets and the results confirm that this approach outperforms the usage of algorithmic models for effort estimation [28].

Several different approaches and variations have been proposed for measuring similarity distance. Chiu and Huang propose adjusting the estimations to take into account the re-use effect of the project identified as the most similar [11]. Azzeh et al. propose an approach that supports handling uncertainties and imprecision in project attributes by the use of fuzzy C-means clustering and fuzzy logic. With this approach, each attribute is represented with several fuzzy sets instead of by a single value. Furthermore, this approach clusters together the most similar projects and their values are represented in the same fuzzy set. The similarity between two projects is then measured by the similarity distance between the two sets to which they mostly belong [3].

Impact distance denotes a measure of the number of steps with which a change in one entity impacts another entity, e.g. through dependencies. This distance has been proposed by Briand et al. for addressing the issue of impact analysis, e.g. for requirements changes, in a UML modelling context. A measurement of the distance between a changed element and an impacted element is defined as the number of impact analysis rules, or steps, required to identify that the impacted element is affected by the change [8]. Initial empirical evaluations indicate that impacted elements at distance one lead to code changes, while those with a greater distance, in most cases, do not. However, further evaluations are required to determine at which maximum distance code changes for impacted elements should be considered [8].

Semantic distance denotes a directional measure of the amount of functionality of a specification that distinguishes it from another related specification. Semantic distance between requirements specifications and other artefacts may be used for supporting software re-use, e.g. to identify library components with a short semantic distance to the requirements. Jilani et al. pose a theoretical case that the use of semantic distance is applicable for decisions on black-box re-use and define a number of metrics for semantic distances. These include metrics for functional deficit that reflect how much functionality needs to be added to one specification in order to satisfy another, and metrics for functional excess that measure the amount of functional features of one specification that are irrelevant to another one [21].

Syntactic distance denotes a measure of dissimilarity of the design structure of two artefacts [21]. Syntactic distance between specifications has been suggested by Jilani et al. for supporting decisions on white-box reuse (where a component is modified). While providing theoretical arguments for applicability of this type of distance Jilani et al. also argues that it is unrealistic to define a measure for syntactic distances since this requires a uniform representation of specifications irrespective of abstraction level and a canonical scheme that supports the definition of a unique representation of specifications. Instead, semantic distances (for which measurements are defined) are suggested to be used as an approximation of syntactic distances [21].

Distance between other entities

Adherence distance denotes the size of the difference between a formal or theoretical model of a process or a phenomena and the actual enactment of it. Within software development this distance has been suggested for gauging the degree of adherence for models. For example, Huo et al. consider the distance between a formal process model and the actual work practices observed in a project [20], though no measurement of this distance is defined. Furthermore, a measure of the distance between a theoretical distribution and actual estimates is defined and evaluated by Thelin and Runeson in the context of assessing the accuracy of remaining faults in an inspected software artefact [30], which could be applied to validation of requirements specifications.

4.5 Limitations

Reliability of the results due to the risk of researcher bias in the inclusion process and the classification process remains an open issue since only one researcher was involved. However, for inclusions/exclusion a generous policy was used, and independent validation of both inclusion and classification is possible since the full set of papers, including the ones excluded through abstract review, is available on-line. Furthermore, there is a risk of incorrect classification when only performed on an abstract. This was addressed by reviewing the full text when the abstract was unclear. However, replication of the study may result in a slightly different set of papers, both in the initial search and in the inclusion/exclusion step.

Conclusion validity concerning the completeness of the results (e.g. number of distances) is one of the main limitations of this study. The search string was limited to 'distance' and did not include synonyms such as gap, proximity etc. This risk of missing relevant papers was partly addressed by broad searches for other aspects. For example, papers were collected from multiple sources incl. IEE and ACM, and wide search terms (software development, software engineering) were used for the scope aspect of the search. Furthermore, no limitation was set on publication year or type of publication (journal, conference etc.). These measures resulted in the study starting with a large set of papers (more than 2,000). However, extending the search to include synonyms would produce an even larger set of papers, and may uncover additional types of distances and applications of these. The main intention of this study was to act as a starting point and further research is planned to further explore the area.

5 Discussion

RE is a communication intense activity and the identified distances between people (see overview in Figure 3) may have an impact on the efficiency and effectiveness of communication and collaboration [1, 2, 13, 34, 35, 37] and can be a significant cost driver [15]. Within GSD, cases where communication is equally strong, or even improved, compared to co-located development have been reported [13, 35]. For example, computer-based group meetings were found to be more effective for requirements negotiation than face-to-face meetings [13]. Similarly, development environments with computer-based support for collaborative work in combination with best practices were found to contribute to reducing communication delays [35].

These contradicting results might be explained by the effect the applied practices have on the division between formal and informal communication. When (previously) informal information is re-routed to more formal communication channels the communication flow may be improved, resulting in reaching a wider audience. This correlates well with findings by Agerfeldt et al. Distance tends to affect informal communication in particular and leads to reduced trust, difficulty in conveying vision and strategy and lack of awareness [1]. Cases where formal communication including documentation is weak and the informal channels are important (e.g. for agile development) are likely to be very vulnerable to distances between people.

Some of the distances are objective (e.g. geographical) while others are subjective and based on people's perception [27], values and normative practices. The perceived (subjective) distance can vary over team members and over time [27], and research has shown that quantifying this distance can support management and be beneficial for GSD practices [27]. All the subjective people distances, i.e. organisational, power, opinions, cognitive and psychological, seem to be covered by the socio-cultural distance (see Figure 3). More research into these distances specifically for RE and for collocated development could potentially explain issues reported for RE communication and collaboration [6, 12, 24]. For example, several distances may be at play in co-located cross-functional teams with a product owner from a different organisational unit and with an RE background; short geographical, but long organisational and cognitive distances between the product owner and other team members. Awareness

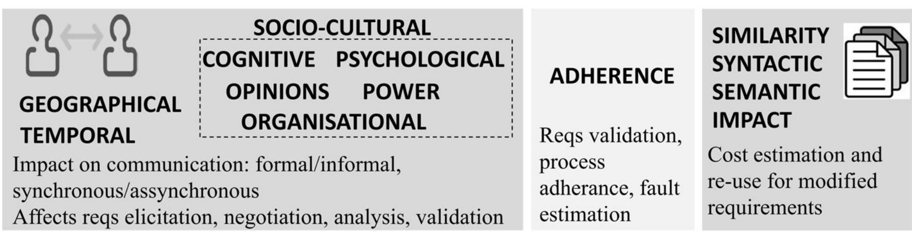


Fig. 3. Overview of interpretation of identified RE distances including relevant RE areas.

of distance and their impact could support management in optimising organisations [37], training efforts, and selected methods [36] and tools [9, 13, 18, 32, 35].

Temporal distance affects the possibly of synchronous communication and within GSD asynchronous communication is common [1, 36]. In addition, subjective distances caused by differences in culture, language etc. may make people reluctant to communicate directly, thus resulting in preferring to communicate via e-mail or through issue management systems. In general, the asynchronous communication that these distances may incur induce delays and increase lead times of RE and the entire development effort [36]. This may affect communication intense activities such as RE, in general, and elicitation and negotiation in particular.

Artefacts play an important role in communicating requirements to stakeholders and within a development project. The identified distances between artefacts have primarily been researched for cost estimation and re-use based on changes to, or different versions of, RE artefacts [8, 21, 28]. These distances may be used to characterise coverage and consistency between artefacts of different activities, e.g. as a measure of the alignment between RE and later development activities. For this reason, RE distances to artefacts of later development activities are an important area to research.

Adherence distance between an artefact and the actual enactment of it has been suggested for process improvement [20] and for estimating remaining fault content [30]. Additional interesting applications could be adherence between a requirement specification and the final product, as well as, the actual customer needs. Both of which are key factors for successful RE.

Finally, most of the identified distances are reported to be better the shorter they are, but there are some interesting exceptions. Within GSD, long socio-cultural distance may potentially increase learning by providing access to a richer skill set, and be stimulating for innovation [1]. Furthermore, organisational distance between testers and developers has been reported to improve alignment between testing and requirements by avoiding testing against developers' interpretation of the requirements [6]. Identifying and understanding additional cases where long distances result in positive effects can support defining a comprehensive theory of the impact of RE distances on software development.

6 Conclusions

Coordination and alignment of requirements with later activities is vital for enabling continuous development of successful products. Within global software engineering distances are reported as increasing risk and cost. Distances between RE and other development activities, e.g. in decision making and requirements communication, may hinder effective and efficient development of customer requirements.

In this systematic mapping study 13 RE-related distances were identified. Distances were mainly found between people (roles, teams etc.) and between artefacts (requirements and design specifications etc). Distance between people has primarily been researched within the context of GSD (geographic, socio-cultural and temporal), while distance between artefacts was found exclusively in non-GSD research.

GSD research on *distance between people* is fairly mature, though more empirical research is needed to understand the impact of these distances for non-distributed development, e.g. for large-scale development. Furthermore, no theory was found in the reviewed papers that could explain the contradicting findings of several studies concerning geographical distance. Further investigations are required to gain a deeper insight into relationship between different distances and the impact they have on division between formal and informal communication. Findings from other fields like psychology and cognitive science are relevant to consider when investigating these people-related distances in relation to RE activities.

Distance between artefacts has been suggested in the context of requirements change and traceability and is an interesting area for future RE research. Distance between RE artefacts and artefacts of later development activities, e.g. design and testing, could potentially be used to measure coverage and consistency between RE and other artefacts such as design and test specifications, and source code.

The systematic map reveals that RE distances in relationship to later development activities (e.g. design, implementation and testing) is largely un-researched. If distance is indeed an important factor in the coordination and communication of RE, research is much needed to address this gap. Examples of RE activities where distance may play an important role include elicitation, negotiation, specification, managing requirements changes and requirements traceability.

This study is a first step towards exploring and defining a theory for the role of RE distances in software development. Future work includes constructing a theoretical framework for RE distances in relationship to testing based on previous research and on empirical data.

Further empirical research into how RE distances, and combinations of these, affect later development activities may support constructing a theory that explains what mechanisms are at play in development projects, between people, artefacts and activities. Increased knowledge of such factors might enable optimization of RE methods and practices for eliciting, negotiating and communicating requirements. Furthermore, through researching new methods and practices for bridging or decreasing distances the effectiveness of RE in software development may be improved, ultimately resulting in more efficient development of better products.

Acknowledgement. This research was funded by EASE Industrial Excellence Centre for Embedded Applications Software Engineering (http://ease.cs.lth.se).

7 References

1. Agerfalk, P. J., Fitzgerald, B., Holmstrom Olsson, H., Lings, B., Lundell, B., Ó Conchúir, E.: A Framework for Considering Opportunities and Threats in Distributed Software Development. In: Proc. Of Int. Works. on Distr. Softw. Eng., DiSD 2005, pp. 47-61. (2005)
2. Allen, T.: Managing the flow of technology. Cambridge, MA, MIT Press (1977)
3. Azzeh, M. Neagu, D., Cowling, P.: Software project similarity measurement based on fuzzy C-means. In: Proc. of Int. Conf. on the Softw. process, ICSP'08, pp. 123-134. (2008)
4. Bjarnason, E.: Study material for RE distance study incl. list of all papers, http://serg.cs.lth.se/research/experiment_packages/distmap/, latest access 2013-01-28.
5. Bjarnason, E. Runeson, P., Borg, M. et al.: Challenges and Practices in Aligning Requirements with Verification and Validation: A Case Study of Six Companies. Submitted to Empirical Software Engineering Journal, 2013.
6. Bjarnason, E., Wnuk, K., Regnell, B.: Requirements are Slipping Through the Gaps – A Case Study on Cause & Effects of Communication Gaps in Large-Scale Software Development. In: Proc. of 19th IEEE Int Requirements Engineering Conf., pp. 37-46.(2011)
7. Brereton, P. Kitchenham, BA. Budgen, D. et al.: Lessons from applying the systematic literature review process within the software engineering domain. In: Journal of Systems and Softw., 80(4), Pages 571-583. (2007)
8. Briand, L. C., Labiche, Y., O'Sullivan, L., Sówka, M. M.: Automated impact analysis of UML models. In: Journal of Systems and Softw. 79(3), pp. 339-352. (2006)
9. Calefato, F., Damian, D., Lanubile, F.: An Empirical Investigation on Text-Based Communication in Distributed Requirements Workshops. In: Proc. Of Int. Conf. on Global Softw. Engineering (ICGSE '07), pp. 3-11. (2007)
10. Chakraborty, C. Chakraborty, D.: A fuzzy clustering methodology for linguistic opinions in group decision making. In: J. of Applied Soft Computing, 7(3), pp. 858-869. (2007)
11. Chiu, N.H., Huang, S.J.: The adjusted analogy-based software effort estimation based on similarity distances. In: Journal of Systems and Softw., 80(4), pp. 628-640. (2007)
12. Curtis, B., Krasner, H., Iscoe, N.: A Field Study of the Software Design Process for Large Systems. In: Commun. ACM, vol. Nov. 1988, pp. 1268-1287. (1988)
13. Damian, D.: An empirical study of requirements engineering in distributed software projects: Is distance negotiation more effective? In: Proc. Of 8th Asia Pacific Softw. Engineering Conf. APSEC'2001, pp. 149-152. (2001)
14. Damian, D, Chisan: An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management. In: IEEE Trans on Softw. Eng. 43(7), pp 433-453. (2006)
15. Dibbern, J., Winkler, J., Heinz, A.: Explaining variations in client extra costs between software projects offshored to India. In: MIS Quarterly: Management Information Systems, vol. 32, n:o 2, pp. 333-366. (2008)
16. Ebert, C.: Understanding the product life cycle: four key requirements engineering techniques. In: IEEE Softw., vol.23, no.3, pp.19-25. (2006)
17. Fricker, S., Glinz, M.: Comparison of Requirements Hand-Off, Analysis, and Negotiation: Case Study. In: Proc of 18th Int Requirements Engineering Conf., pp. 167-176. (2010)
18. Herlea, D., Greenberg, S.: Using a groupware space for distributed requirements engineering. In: Proc. of Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE, pp. 57-62. (1998)
19. Hofstede, G.: Cultural constraints in management theories. In: Academy of Management Executive, 7(1), pp. 81 – 94. (1993)

20. Huo, M, Zhang, H. Jeffery, R.: Detection of consistent patterns from process enactment data. In: Proc. of Int. Conf. on Softw. Process, ICSP'08, pp. 173-185. (2008)
21. Jilani, LL, Desharnais, J, Mili, A: Defining and Applying Measures of Distance Between Specifications. In: Journal IEEE Transactions on Softw. Eng, 27(8), pp. 673-703. (2001)
22. Lawson, M, Karandikar, H. M.: A Survey of Concurrent Engineering. In: Concurrent Engineering 1994 2:1. (1994)
23. Liberman, N., Trope, Y., Stephan, E.: Psychological Distance, book chapter in "Social psychology: handbook of basic principles (2nd ed)", Kruglanski, Higgins (Eds), pp. 353-381, Guilford Press. (2007)
24. Marczak, S, Damian, D.: How Interaction between Roles Shapes the Communication Structure in Requirements-Driven Collaboration. In. 19th IEEE Int Req. Eng. Conf., pp. 47 – 56. (2011)
25. Nuseibeh, B.: Weaving together requirements and architectures. In: Computer , vol.34, no.3, pp.115-119. (2001)
26. Petersen, K., Feldt, R., Mujtaba, S., et al.: Systematic Mapping Studies in Software Engineering. In: 12th Int. Conf. on Evaluation and Assessm. in Softw. Eng., pp.71-80 (2008)
27. Prikladnicki, R.: Propinquity in global software engineering: Examining perceived distance in globally distributed project teams. In: Journal of Softw. Evolution and Process, 24(2), pp. 119-137. (2012)
28. Shepperd, M. Schofield, C.: Estimating Software Project Effort Using Analogies. In: IEEE Trans. on Softw Eng, Vol. 23 Issue 11, pp. 736-743. (1997)
29. Swartout, W., Balzer, R.: On the Inevitable Intertwining of Specification and Implementation. In: Comm. ACM, vol. 25, no. 7, pp. 438-440. (1982)
30. Thelin, T., Runeson, P.: Robust estimations of fault content with capture–recapture and detection profile estimators. In: Journal of Systems and Softw., 52(2–3), pp. 139-148, (2000)
31. Uusitalo, E.J., Komssi, M., Kauppinen, M. et al.: Linking Requirements and Testing in Practice. In: 16th IEEE Int Requirements Engineering Conf, NJ, USA, pp. 265-270. (2008)
32. Wende, E, Philip, T.: Instant messenger in offshore outsourced software development projects: Experiences from a case study. In: Proc. of 44th Annual Hawaii Int. Conf. on System Sciences. (2010)
33. Wieringa, R., Maiden, N., Mead, N., Rolland, C: Requirements Engineering Paper Classification and Evaluation Criteria: a Proposal and a Discussion. In: Journal of Requir. Eng. 11(1), pp. 102-107. (2006)
34. Winkler, JK, Dibbern, J, Heinzl, A: The impact of cultural differences in offshore outsourcing-Case study results from German-Indian application development projects. In: Inf. Systems Frontiers, v 10, n 2, pp. 243-258. (2008)
35. Wolf, T., Nguyen, T., Damian, D.: Does distance still matter? In: Journal of Impr. and Practice of Softw. Process, 13(6), pp. 493-510. (2008)
36. Yousuf, F., Zaman, Z., Ikram, N.: Requirements validation techniques in GSD: A survey. In: Proc. 12th IEEE Int. Multitopic Conf. INMIC'08, pp. 553-557 (2008)
37. Yu, Y., Sharp, H.: Analyzing requirements in a case study of pairing. In: Proc. of 1st Int. Workshop on Agile RE, Lancaster, UK. (2011)
38. Zhu, J., Hipel, K.W.: Multiple stages grey target decision making method with incomplete weight based on multi-granularity linguistic label. In: Journal of Information Sciences, vol. 212, pp. 15-32. (2012)