What makes Service Oriented Requirements Engineering challenging? A qualitative study

Muneera Bano [1], Didar Zowghi [1], Naveed Ikram [2], Mahmood Niazi [2, 3]

[1] Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia
[2] Faculty of Computing, Riphah International University Islamabad, Pakistan
[3] Department of Information and Computer Science, King Fahd University of Petroleum and Minerals, KSA

Muneera.Bano@uts.edu.au, Didar.Zowghi@uts.edu.au, naveed.ikram@riu.edu.pk, mkniazi@kfupm.edu.sa

Abstract

The focus of Service Oriented Software Development (SOSD) is to develop software by integrating reusable services to lower the required cost, time and effort of development, and increase reusability, agility, quality and customer satisfaction. It has been recognized in the literature that SOSD faces various challenges especially in Requirements Engineering (RE). The objective of this paper is to investigate these challenges of Service Oriented Requirements Engineering (SORE) from practitioners' perspectives in order to gain a deeper understanding of the related issues and to reveal potential gaps between research and practice in SORE. We present a qualitative study of the challenges and issues in SORE. The data was collected by conducting interviews with practitioners working in IT companies in Sydney, who have had substantial experience with service oriented software projects. Our findings reveal that most of the challenges of SORE are similar to those that are faced during RE in traditional or component based software development. According to the practitioners, the research and practice has made some advances in the technical direction but the human related issues in SORE have not been addressed adequately.

Keywords: Service Oriented Software Engineering, Requirements Engineering, Interviews

1. Introduction

Service Oriented paradigm was introduced for providing good quality software solutions according to the business requirements in competitive market conditions in order to reduce the cost and time of development. Service Oriented Software Development (SOSD) [1], is an evolutionary form of Component based Software Development (CBSD) and Object Oriented software development with the difference that it uses web services instead of objects or packaged components, which can be accessed via their interfaces in distributed environment [2]. The use of services requires a contractual agreement between service provider and service consumer. Reusing existing components in form of services in a collaborative and distributed environment is helpful for saving time and cost of development. A great deal of research efforts were put into the directions of achieving the true spirit of SOSD from academia and industry alike [3, 4].

In spite of all the improvements in tools and technologies, literature suggests that the SOSD is still facing various challenges especially in Requirements Engineering (RE) [5]. Service Oriented Requirements Engineering (SORE) [6, 7], is considered different from traditional RE due to the separation between service development (concerns for service providers) and developing systems that use those services (concerns for service consumers). Service Oriented solutions have to consider both service provider and service consumer. The service provider needs to understand the functional and non-functional aspects of the service being offered, which has to compete in the marketplace with other services. This requires understanding the needs of potential consumers of that service. For service consumers, the challenge is to find the correct and cost effective solutions to the business needs of the organization.

SOSD had been the focus of research for more than a decade. Many methods, techniques and tools have been proposed by different mega projects and research teams [3-5, 8]. The real benefits of any research efforts can only be verified when it is applied in a real setting. With the exception of few (e.g. IBM), most of the methods proposed to date are not adapted and verified by industry [3] [4]. The software engineering research community has a tendency of building their solutions based on the problems reported in the earlier literature. It is not particularly useful to conduct research in providing technological solutions to the problems and challenges that may not be considered accurate or even real by the practitioners. The results of the academic research are produced in a meticulous and methodical way, but often lack the considerations for real industrial settings and hence may not be applicable there. According to the Davis and Hickey [9], the RE researchers have to practice what they preach and analyze the reality behind the problems for which they propose the solutions. They argue that many RE researchers fail to understand the current industrial practices, therefore, their proposed solutions are not applicable to industry. There is a lack of empirical work that investigate the challenges of SORE [10], which necessitates further work in this area with real life projects. This would provide feedback for improvement in current methods and practices, to enrich the knowledge in service oriented domain, and open further research directions [5]. This motivates the need to validate the identified challenges of SORE before any solution can be proposed which can be applicable in industrial settings.

We have carried out a three-step exploratory research plan to investigate and understand the challenges of SORE. Our study was guided by the following main research question:

"What are the issues and challenges of Requirements Engineering in Service Oriented Software Development?"

Our first step was to perform a comprehensive literature review. Utilizing the results of literature review in the second step, we conducted an online survey. The objective of this quantitative study was to confirm the results of the literature review from practitioners working in the industry. Based on the results of the survey, as a final step, we performed a qualitative study and conducted 14 industrial interviews to gain a deeper understanding of the identified challenges.

In this paper, we present the findings of the interviews with practitioners from different companies in Sydney, who have experience in SOSD. The objective of the interviews was twofold;

1. *Investigate the reality of the issues and challenges of SORE as reported in the literature*
2. *Identify potential gaps in research and practice of SORE*

This paper is the extended version of the preliminary results of this study that was published in [11].

The rest of the paper is organized as follows; Section 2 describes the Research Design and explains the overall research strategy. Section 3 the details of the interview study design, execution and results. Section 4 discusses the findings from interviews. Section 5 gives conclusion and future directions of the research.

2. Research Strategy Design

2.1 Literature Review

The objective of the literature review was to conduct an exhaustive search for publications that were available on online electronic databases to identify the challenges faced during SORE. The procedure for the search and the results of the analysis of the challenges from literature was published in [5]. The analysis of our findings revealed the following list of top ten challenges of SORE:

1	Integration of Knowledge Management strategy to SOSD life cycle
2	Alignment of business requirements and services
3	Iterative service discovery Process
4	Semantic gaps in specifications
5	Automated and Dynamic Service Discovery
6	High Level Language Support
7	Service Testing
8	Requirement Change Management
9	Non functional Requirements gathering and assessment
10	Lack of standard RE Process

TABLE I. LIST OF ISSUES AND CHALLENGES IDENTIFIED FROM LITERATURE REVIEW [5]

2.2 Online Survey

The second step of our research inquiry was to conduct an online survey. Surveys are mostly suitable for approaching a large sample from a population for surface level feedback and agreement. Using the results of the first step of our research, we designed a questionnaire to validate the identified challenges from practitioners working on service oriented projects. Our target population comprised of the practitioners having experiences of working on service oriented projects either as technical team member or as a researcher. We selected convenient and non probabilistic sampling due to the fact that it was an online web based survey. The survey instrument was a questionnaire based on the identified factors (Table I) from the challenges of SORE. A Likert scale of five levels was used to measure agreement to the challenges of SORE. We administered the survey on the web and sent the web link through emails to invite the practitioners around the world using online special interest groups. The survey link was active for one month. During that period a total of 117 responses were received worldwide. The survey provided us with the verification and ranking of the challenges. We were able to refine our list of challenges based on the results from the survey. The details of the survey design, execution and results were published in [12]. Table II and Fig I summarize the analysis of the results of survey.

Sr#	Measurement Factors	Percentages on Lickert Scale of five for 117 responses				
		SA	A	N	D	SD
1	Alignment of business requirements and services	32	44	16	5	3
2	Integration of Knowledge Management strategy to SOSD life cycle	29	47	17	6	1
3	Iterative service discovery Process	28	47	16	7	2
4	Requirement Change Management	20	41	21	15	3
5	Non Functional Requirements gathering and assessment	19	41	21	15	4
6	Semantic gaps in specifications	15	37	28	17	3
7	Service Testing	23	32	29	13	3
8	Automated and Dynamic Service Discovery	19	38	27	12	4
9	Lack of standard RE process for SOSD	20	34	35	9	2
10	High Level Language Support	8	28	29	27	8

TABLE II. PERCENTAGES OF AGREEMENT FOR MEASUREMENT FACTORS (SA=STRONGLY AGREE, A=AGREE, N=NEUTRAL, D=DISAGREE, SD=STRONGLY DISAGREE)

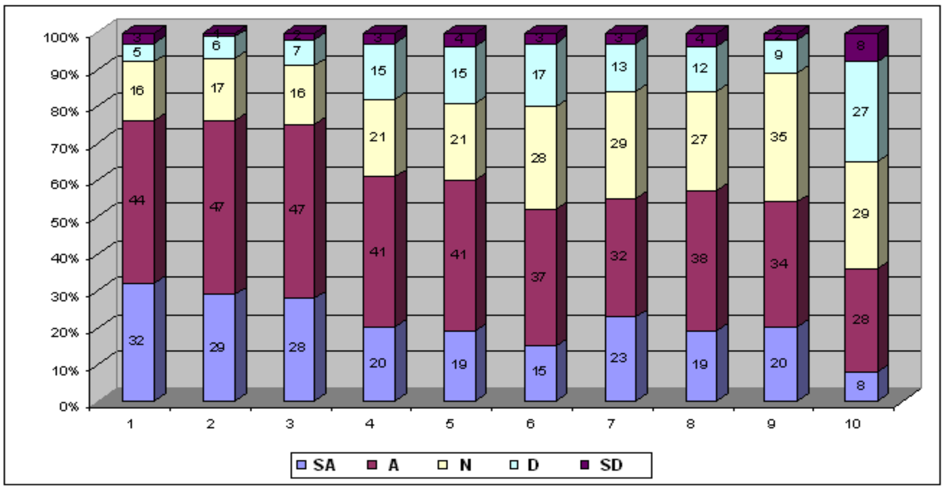

FIG I. PERCENTAGES OF THE RESULTS FOR THE ISSUES PRESENTED IN TABLE II

Before conducting actual interviews, we conducted a pilot study with four practitioners in Sydney and discussed the results of the survey. Based on the results from the survey and discussions with these practitioners, we selected the top five challenges for our qualitative study. Challenge 6 to 10 in table II were not considered as important as the others and hence dropped from the list. Table III represents the challenges that remained for further exploration in the next phase.

1	Alignment of business requirements and services
2	Integration of Knowledge Management strategy to SOSD life cycle
3	Iterative service discovery Process
4	Requirements Change Management
5	Non Functional Requirements gathering and assessment

TABLE III. LIST OF ISSUES AND CHALLENGES IDENTIFIED FROM ONLINE SURVEY

Before proceeding to describing the conduct and results of our interview study we give a brief review of literature for each challenge to explain their nature in order to explain exactly why they are considered challenging.

2.2.1 Alignment of business requirements and services

An important task for business analysts in service-oriented requirements engineering is to align service descriptions with business requirements. This task is considered the most challenging, as the services are not usually described at the right level of granularity. Service granularity (sometimes referred to as abstraction level), is about the degree of modularity of service [13], and refers to the service size and the scope of functionality a service exposes [14, 15]. The level of granularity is considered appropriate from a technical point of view if a service can accomplish a business task in a single invocation. Services are considered either fine-grained or coarse-grained [13]. Fine granularity would offer more flexibility in customizing the system but also results in more effort on integrating small modules (granules). A coarse-grain service would typically be expected to carry out more functions but due to increase in the exchange of data and messages, this may present a more complex interface [16]. From the consumers' point of view, the choice of an acceptable granularity of services for alignment also presents an economic problem. Providing a large number of fine-grained services would reduce development and maintenance cost and would potentially lead to higher level of reusability of services. However, it will increase the costs and complexity for service composition [15, 17]. On the other hand, very coarse grained services would lower its reuse as the functionality might be too overloaded for the actual need of most consumers and therefore might be suitable only for small customer segment [17]. To achieve the acceptable level of alignment, trade-offs need to be made between cost and required functionality [14]. The process of alignment in SORE has the following two aspects [18];

- *Requirement to service alignment: finding services that closely match the requirements, and*
- *Service to requirement alignment: modifying and adjusting requirements to make better use of existing services*

It is necessary to fully understand the level of granularity of a service for accurate alignment to business requirements. This necessitates performing a trade-off analysis between cost and various facets of flexibility, reusability and performance of the service to name but a few. For aligning business requirements to services, there can be three main scenarios (Figure II) [18]; fully aligned, not aligned, or partially aligned. The challenge arises when the decision has to be made for partial alignment. Partial alignment has three cases;

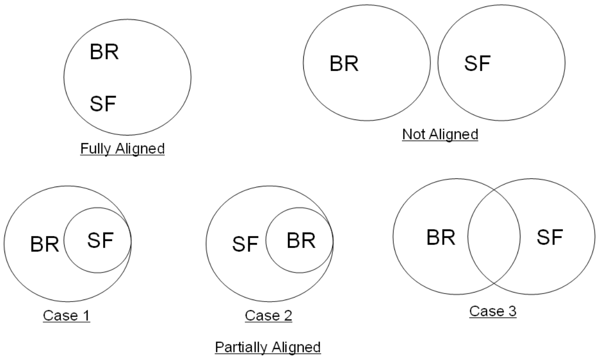

FIG II. FIVE SCENARIOS OF ALIGNMENT OF SERVICE FUNCTIONALITY (SF) AND BUSINESS REQUIREMENTS (BR)

1. *Service functionality fulfils only a part of the business requirement (service is too fine grained causing performance related issues e.g. integration problem, delays in interaction among multiple services)*
2. *Service functionality offers more than business requirement specifics (service is too coarse grained increasing cost)*
3. *Service functionality and business requirements overlap (causing increase in cost and integration issues)*

Dealing with partial alignment requires a decision making process for the selection of a service, based on trade-offs between functional/non functional requirements and cost. This process requires a systematic method that could combine both technical and non-technical aspects of alignment between services and requirements. This method must satisfy the various needs of customers while being able to find an acceptable level of granularity, right functional range and abstraction level for successful service selection [15]. Many solutions in the past have been proposed to tackle alignment from a technical perspective e.g. [19, 20], but they all focus only on the technical aspect of this challenge.

2.2.2 Integration of Knowledge Management (KM) strategy to SOSD life cycle

Software engineering knowledge is dynamic in nature and there are many factors which contribute to its evolution like; technology, organizational culture, changes in needs of organization and change in software development practices etc. Software processes are considered to be knowledge processes. Software development can be improved by recognising related knowledge content and structure, as well as appropriate knowledge and engaging in planning activities, which would improve the performance of organization [21-23]. KM is reported to support these core activities of software engineering: Document management, Competence management and expert identification, Software reuse (making developers aware of existing contents).

According to Pilat and Kaindl [24], RE can be viewed as a knowledge process. It involves the conversion of tacit knowledge related to requirements and domain into explicit knowledge. The RE phase of the project can greatly benefit from the knowledge stored from the previous projects from similar domains and contexts. Integrating KM techniques and tools in SOSD lifecycle is considered helpful for the analysts in future projects during RE phase when the evaluation of reusable services takes place either from within organization or from third party service provider. They can make more informed decisions for service selection during the early phase of software development. Therefore adopting KM perspective is considered to support handling the issues and challenges of SORE [12].

But implementation of KM strategy is considered challenging in the literature due to the following reasons:

- *A lot of time, efforts and resources are required before the actual benefits become visible [21]*
- *Most of the knowledge in SE is not in explicit form [21]*
- *SE activities are more towards technical knowledge management and it would take the behavioural or human aspect of knowledge in the background [25].*

2.2.3 Iterative service discovery process

In the literature, it has been suggested that the SORE should follow an iterative process [26-28]. This is considered beneficial in the initial phases of development process while requirements are being elicited from the stakeholders. The initial requirements are usually incomplete and to align business requirements to a service that fulfils the business needs accurately is problematic. With an iterative process, the development team has the opportunity to present closely related services to the customers for their feedback. The services work as prototypes and help in completing the requirements.

2.2.4 Requirements Change Management (RCM)

RCM in traditional software development is an essential part to make the software up to date and acceptable to the customers' requirements and dynamic market needs. Requirement change is considered inevitable during software development life cycle. To continuously meet with the users' changing needs and demands, and addition of new features the system has to undergo the changes. These changes to the existing system are not easy to implement and it requires good management efforts to avoid system failure. The system has to work reliably even though unexpected changes are inevitable to the requirements. Therefore when a change request happens, it is necessary to understand the type of change, its impact of the scope and level of effect on the entire system and what strategies are required to deal with this change request. This impact analysis leads to the decision making for that change request [29]. According to the literature, the distributed and highly dynamic nature of the service oriented paradigm makes RCM a challenging issue in SOSD [30-32].

2.2.5 Non functional requirements (NFR) gathering and assessment

NFR, which also includes the quality attributes for a software are tremendously challenging to elicit [33]. Their importance in influencing the design and development process has been acknowledged both in traditional as well as SOSD [34]. The main difference about NFR gathering and assessment in traditional and service oriented SE is that due to the highly distributed nature of service-oriented applications, it is more difficult to handle NFR [35]. The reusable, distributed, and loosely coupled services require new approaches for dealing with NFR in SOSD [34, 35].

3. Interview study

3.1 Interview design and execution

We conducted face-to-face interviews as a final step of our research. Interviews provide with rich qualitative data when a deeper insight of a phenomenon is required. Our survey provided us with a surface level understanding of the RE challenges and issues in SOSD. Interviews were our choice for a deeper and vertical understanding of the challenges in a real life context. It was the best choice to get the opinion and reflections of practitioners by talking to them face to face and giving them freedom to express their thoughts on the subject.

The interviewees were selected from software companies based in Sydney. The criterion for their selection was that the interviewee should be a practitioner in the industry and should have the experience of working as a development team member in a service oriented project. The practitioners were initially approached using social networking and later followed by snow-ball or referral sampling technique. The interviews were semi structured and open ended. Based on the refined list of challenges from the online survey, we designed the semi-structured interview questions. The interviewees were asked to provide their views on the issues with reference to the examples of the service oriented projects they had experienced. The interview questions were focused on the five issues identified in online survey but for further understanding we took advantage of face-to-face interview style to ask follow up questions wherever it was required. All 14 interviews took place during June-August 2012 in Sydney and each lasted between 30 minutes to one hour. They were audio recorded and later transcribed. Out of 14, six had the experience of being Business Analyst, 8 as Project Managers. Six out of 14 had additional experience as Developers in SOSD Projects. The names of the interviewees, the companies and the projects will not be disclosed in this paper as per confidentiality requirement of Human Research Ethics Committee (HREC) at UTS for the approval of this research. The transcripts of the interviews were analyzed for the thoughts and concepts presenting for each of the issues, partial preliminary results were published in [11]. In this paper we present the details of the findings from the interviews.

3.2 Findings from interviews

In the following, we present the aggregated results which emerged after analysis with respect to the challenges of RE in SOSD.

3.2.1 Alignment of business requirements and services

The main difference between SORE and traditional RE is the additional task for a business analyst to align service descriptions with business requirements. The interviewees confirmed that the alignment of business requirements and services was considered to be the most challenging activity in SORE. For the correct alignment of requirements and business processes to services it is necessary to fully understand the level of abstraction and granularity of services. The respondents pointed out that this issue is not new to service oriented domain only (as claimed in most of the literature on SORE) and is also faced during component based software development but does get exacerbated in service oriented domain due to the fact that the services are developed free of context so they can be used in multiple projects.

A deeper analysis of the responses for this challenge showed that most of the reasons behind this activity being challenging are those that are also present in traditional RE i.e. incomplete and ambiguous requirements, communication problems with stakeholders etc. Lack of human centred approaches and ignoring the real users in SOSD makes this task more challenging.

3.2.2 Integration of Knowledge Management strategy to SOSD life cycle

SOSD requires organizations to take initiatives to help their business needs. KM requires organizational strategy for identifying, acquiring, storing, and sharing the knowledge to improve the business and benefit to the organization. The interviewees pointed out that it is critical for the service oriented organizations to take initiatives for implementing a well defined KM strategy to meet demands of dynamic market conditions where there is inevitable change in technology and human resource. In the last 20 years many methods, tools and techniques for KM have been proposed in the research literature. All the interviewees confirmed using knowledge management tools to assist them in various activities of software development managing and sharing requirements' documentations and artefacts with the team members. Different tools are being used for this purpose e.g. JIRA, SharePoint, Google docs and wikis. Only two respondents mentioned having a clear organization level strategy for knowledge management. Others considered it as an overhead for the development team. The respondents described the benefits of using knowledge management strategy for SORE. KM strategies are considered to be helpful in:

- *alignment of requirements and services and vice versa*
- *re-usability of services and business processes*
- *requirement change management (keeping track of versions, impact analysis)*
- *gap-analysis*
- *document management*
- *sharing experiences and lessons learned from previous projects*
- *project management and scheduling*
- *better understanding of people, process and product related to the project*

All these benefits come with the challenges of implementing a KM practice along with software development. According to the respondents, for small organizations KM brings overhead for the development team in terms of knowledge codification efforts. The respondents referred to it as: *"tedious hard work consuming a whole day for filling up various data forms"*. For the development team, there seems to be a need for great deal of visible effort for invisible benefits. The organizations have to keep track of the Knowledge Base (KB), and specialized personnel are needed for that purpose, whose task is to make sure that the KB is up to date. The identified challenges by the respondents in this category are;

- *time, cost and effort is required for implementation of a knowledge management strategy*
- *the benefits are visible after some times and for the technical members of the team it is overhead*
- *managing the knowledge base to keep it tidy and maintained*

According to the overall analysis of our results, all the above stated benefits are those that are achieved by integrating knowledge management activity in any software development methodology and are not specific to SOSD.

3.2.3 Iterative service discovery process

According to the practitioners they do follow the iterative service discovery process for carrying out the alignment of requirements and services. The process is mostly carried out with the help of agile methodology. It helps the development team to increase their understanding of clients' business requirements (both functional and non functional), and of available services. This understanding is necessary for a better alignment of services to the requirements.

The respondents pointed out some concerns that arise with the iterative process. If the development team is using services within their organization, then they already know about the possible solutions. In the iterative process, service providers try to encourage and coerce the customers into accepting the available services by presenting them as prototypes. By this strategy they enforce the alignment of services and requirements.

3.2.4 Non functional requirements

According to the interviewees, NFR play a crucial role during alignment of business requirements and services for trade-off analysis and service selection. The respondents have considered elicitation and implementations of NFR more challenging in SOSD due to various reasons, e.g. Context independent development of services, lack of trust on third party software.

3.2.5 Requirements change management

One of the promised benefits of service orientation is to have loosely coupled services integrated in such a way that it would make it easy to implement a change in the system. According to the respondents the technological changes are easier to implement in service oriented software projects. Replacing one service with another is technically easy. The challenging part is the analysis of the impacts of change on the system. The practitioners are using the same techniques and methods in service oriented projects as used for change management in traditional software development. The real challenge in implementing a change in service oriented systems is the modification of the functionality within a service especially when a service is being used by multiple users in multiple contexts.

4. Discussion

The analysis of the results of our survey and interviews show some insights about the challenges of SORE from the industry's perspective. First, though most of the challenges identified in the research literature also exist in the practice of SORE but they are not explicitly caused by the service oriented development as claimed. These challenges are all the same as RE for traditional and component based software development.

Second, the research community has provided different solutions to these challenges but most of them have no applicability in industry. In our previous study[12], we have presented an analysis of the shortcomings of those solutions to address the challenges of SORE. Like other fields of software engineering, SORE also suffers from a significant gap between the research and practice. Any research that cannot be applied and be aligned to the objectives, goals and strategies of the current industrial practices would be of limited use. Table IV is the result of our comparison of the findings form Literature review, survey and the interviews to highlight the existing gap in the research and practice of SORE. This will help the research community of SORE to identify and understand the 'real' industry problems and provide solutions for them, rather than working on invalidated designs and continue producing research artefacts without real industrial evaluation and feedback.

Sr#	Literature [5]	Survey [12]	Interviews
1	Alignment of business requirements and services	Yes	Challenging but same as faced during RE in CBSD
2	Integration of KM strategy to SOSD life cycle	Yes	Same as using KM tools in any SE development methodology
3	Iterative Discovery Process	Yes	Process used for alignment (as in 2)
4	Requirement Change and Evolution	Yes	The process is same as traditional RCM
5	NFR gathering and Assessment	Yes	Part of alignment process (as in 2)
6	Semantic gaps in specification	Not critically challenging	-
7	Service Testing	Not critically challenging	-
8	Automated and Dynamic Service Discovery	Not challenging	-
9	Lack of standard RE process for SOSD	Not challenging	-
10	High Level Language/Tool Support	Not challenging	

11	Not mentioned	-	Lack of human centred approaches
12	Not mentioned	-	Lack of common understanding of service oriented concepts by researchers and practitioners

TABLE IV. COMPARISON OF RESULTS IN LITERATURE REVIEW, SURVEY AND INTERVIEWS

Third, the respondents expressed that the real problem is with gathering requirements from clients, customers and users regardless of the fact that whether or not the project would be service oriented. Over the last ten years improvements have been made as a result of research in technologies and standard protocols for network infrastructure to support distributed development and deployment of services. But the research community has largely ignored the socially oriented aspects of SORE. The respondent informed us that in practice the actual users of the service are not considered during the development process. In most cases, a customer representative or a subject matter expert is carefully selected to join the development team to provide feedback. This results in end user dissatisfaction and sometimes in project failure. Most of the problems in RE (whether service oriented or not) are caused due to inadequate requirements and the dissatisfaction of customers with the end product.

According to Ian Mitroff and Harold Lindstone [36], every problem has three perspectives which are interlinked: T (Technical), O (Organizational or Social) and P (Personal or individual). While trying to solve a problem, all three perspectives have to be considered otherwise the solution will be of limited use. The technical perspective of any issue is just a single view of a multifaceted problem. The technology is important, but how to make the best use of that technology by understanding the real objective of the system to build and the satisfaction of those who will ultimately use it, is far more crucial. It is important that the decision and acceptance of selection should consider the value of the service to its users. Without users' satisfaction and approval the results would be considered a failure no matter how advanced the technology would become. There is a need for further exploration on the challenges of SORE from a human centred perspective.

The issues regarding requirements elicitation (lack of communication with stakeholders, ambiguities in requirements etc.) are essential [37], and therefore inherent in the nature of any RE process (whether service oriented or not) and efforts can only be made to reduce their impact. One approach that has been advocated and axiomatically accepted over four decades by the by both academia and industry alike is the effective end users' involvement in the software development to achieve success of the project. Various techniques and methods have been proposed in different domains of IT, IS and software engineering that proposes users' involvement and participation during various phases of software development. According to the reported empirical literature of more than 30 years, there is a positive relationship between users' involvement in software development and system success [38] [39]. There are a few initiatives taken recently to analyze and utilize end users' feedback collected from web repositories to better understand their needs and requirements, e.g. in online mobile applications [40], distributed collaborative application development environment [41], and software requirements evolution [42]. Similar concepts can be adopted in the field of service orientation to reduce the impact of the socially oriented problems in software development. Further efforts are required to explore the existing techniques empirically that address the social side of software, in service oriented domain to validate their applicability and to find out the room for improvements.

5. Conclusion and Future Work

After the analysis of interviews, our three step exploratory research has culminated in the following points:

1. *Both understanding and implementation of SOSD methodologies seems to be lacking in research and practice.*
2. *Many of the issues and challenges reported in the literature are not considered important by the practitioners working in the industry.*
3. *The challenges of service oriented RE are not significantly different in their nature from those present in RE for traditional software development and component-based development.*
4. *The most focused aspect of the SOSD by researcher is the technical dimension, whereas the practitioners appeared to be more concerned about the challenges caused by the social dimension.*
5. *The alignment of business requirements and services is the most challenging part and the success of the resultant system depends on the correctness of decision for service selection.*

This research is part of our ongoing project for developing a human centred method for SORE using the situational method engineering approach. We are currently investigating the role of users' involvement in service oriented SDLC and its impact on the success of RE phase [38]. Our objective is to incorporate the feedback of actual users of the service during SORE by social computing methods to increase the customer satisfaction.

References

[1] W.T. Tsai, "Service-oriented system engineering: a new paradigm," in *IEEE International Workshop on Service-Oriented System Engineering, SOSE 2005*, pp. 3-6.

[2] Z. Stojanović and A. Dahanayake, "*Service-oriented software system engineering: challenges and practices*" Idea Group Publishing, 2005.

[3] A. Kontogogos and P. Avgeriou, "An overview of software engineering approaches to service oriented architectures in various fields," in *Enabling Technologies: 18th IEEE International Workshops on Infrastructures for Collaborative Enterprises, WETICE*, 2009, pp. 254-259.

[4] E. Ramollari, *et al.*, "A survey of service oriented development methodologies," in *The 2nd European Young Researchers Workshop on Service Oriented Computing*, 2007, p. 75.

[5] M. Bano and N. Ikram, "Issues and challenges of requirement engineering in service oriented software development," in *Fifth International Conference on Software Engineering Advances (ICSEA)* 2010, pp. 64-69.

[6] P. van Eck and R. Wieringa *"Requirements engineering for service-oriented computing: A position paper"* In J. Gordijn and M. Janssen, editors, Proceedings of First International Workshop on e-Services, ICEC'03, Pittsburg, PA, USA , pages 23–28, 2003.

[7] W.-T. Tsai, *et al.*, "Requirement engineering in service-oriented system engineering," in *IEEE International Conference on e-Business Engineering ICEBE 2007.* 2007, pp. 661-668.

[8] M. P. Papazoglou, *et al.*, "Service-oriented computing: State of the art and research challenges," *Computer,* vol. 40, pp. 38-45, 2007.

[9] A. M. Davis and A. M. Hickey, "Requirements researchers: Do we practice what we preach?," *Requirements Engineering,* vol. 7, pp. 107-111, 2002.

[10] Q. Gu and P. Lago, "Exploring service-oriented system engineering challenges: a systematic literature review," *Service Oriented Computing and Applications,* vol. 3, pp. 171-188, 2009.

[11] M. B. Sahibzada and D. Zowghi, "Service Oriented Requirements Engineering: Practitioner's Perspective," in *Service-Oriented Computing-ICSOC 2012 Workshops*, 2013, pp. 380-392.

[12] M. Bano and N. Ikram, "KM-SORE: Knowledge Management for Service Oriented Requirements Engineering," in *ICSEA 2011, The Sixth International Conference on Software Engineering Advances*, 2011, pp. 494-499.

[13] M. P. Papazoglou and W.-J. Van Den Heuvel, "Service-oriented design and development methodology," *International Journal of Web Engineering and Technology,* vol. 2, pp. 412-442, 2006.

[14] N. Kulkarni and V. Dwivedi, "The role of service granularity in a successful SOA realization a case study," in *IEEE Congress on Services-Part I,* 2008, pp. 423-430.

[15] S. Adam, *et al.*, "Deriving Software Services from Business Processes of Representative Customer Organizations," in *Service-Oriented Computing: Consequences for Engineering Requirements, 2008. SOCCER'08. International Workshop on*, 2008, pp. 38-45.

[16] C. Steghuis, "Service granularity in SOA projects: a trade-off Analysis", Master's thesis, University of Twente, 2006.

[17] B. Heinrich, *et al.*, "Granularity of Services–an Economic Analysis," *Business & Information Systems Engineering,* vol. 3, pp. 345-358, 2011.

[18] A. Gehlert, *et al.*, "Goal-driven alignment of services and business requirements," in *International Workshop on Service-Oriented Computing: Consequences for Engineering Requirements, SOCCER'08.*, 2008, pp. 1-7.

[19] M. Galster and E. Bucherer, "A business-goal-service-capability graph for the alignment of requirements and services," in *IEEE Congress on Services-Part I,* 2008, pp. 399-406.

[20] B. Verlaine, *et al.*, "Towards conceptual foundations for service-oriented requirements engineering: bridging requirements and services ontologies," *Software, IET,* vol. 6, pp. 85-102, 2012.

[21] I. Rus, *et al.*, "Knowledge management in software engineering," *IEEE software,* vol. 19, pp. 26-38, 2002.

[22] F. O. Bjørnson and T. Dingsøyr, "Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used," *Information and Software Technology,* vol. 50, pp. 1055-1068, 2008.

[23] A. Aurum, *et al.*, "Investigating Knowledge Management practices in software development organisations–An Australian experience," *Information and Software Technology,* vol. 50, pp. 511-533, 2008.

[24] L. Pilat and H. Kaindl, "A knowledge management perspective of requirements engineering," in *Fifth International Conference on Research Challenges in Information Science (RCIS),* 2011, pp. 1-12.

[25] J. Swan, *et al.*, "Knowledge management–the next fad to forget people?," in *Proceedings of the 7th European Conference on Information Systems*, 1999, pp. 23-25.

[26] K. Zachos, *et al.*, "Discovering Web Services to Improve Requirements Specifications: Does It Help?," in *Requirements Engineering: Foundation for Software Quality*, ed: Springer, 2008, pp. 168-182.

[27] K. Zachos, *et al.*, "Discovering web services to specify more complete system requirements," in *Advanced Information Systems Engineering*, 2007, pp. 142-157.

[28] K. Zachos and N. Maiden, "Inventing requirements from software: An empirical investigation with Web services," in *16th IEEE International Requirements Engineering, 2008. RE'08.*, pp. 145-154.

[29] J. Li, *et al.*, "Preliminary results of a systematic review on requirements evolution," in *16th International Conference on Evaluation & Assessment in Software Engineering (EASE 2012),* pp. 12-21.

[30] M. P. Papazoglou, "The challenges of service evolution," in *Advanced Information Systems Engineering*, 2008, pp. 1-15.

[31] E. Di Nitto, *et al.*, "A journey to highly dynamic, self-adaptive service-based applications," *Automated Software Engineering,* vol. 15, pp. 313-341, 2008.

[32] Y. Wang and Y. Wang, "A survey of change management in service-based environments," *Service Oriented Computing and Applications,* pp. 1-15, 2013.

[33] M. Glinz, "On non-functional requirements," in *15th IEEE International Requirements Engineering Conference, 2007. RE'07* pp. 21-26.

[34] S. Wang, *et al.*, "Quality of service measure approach of web service for service selection," *Software, IET,* vol. 6, pp. 148-154, 2012.

[35] M. Galster and E. Bucherer, "A taxonomy for identifying and specifying non-functional requirements in service-oriented development," in *IEEE Congress on Services-Part I, 2008*, pp. 345-352.

[36] I. I. Mitroff and H. A. Linstone, *The unbounded mind: Breaking the chains of traditional business thinking*: Oxford University Press on Demand, 1995.

[37] F. P. Brooks, "No silver bullet: Essence and accidents of software engineering," *IEEE computer,* vol. 20, pp. 10-19, 1987.

[38] M. Bano and D. Zowghi, "User involvement in software development and system success: a systematic literature review," in *Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering,* (EASE) 2013, pp. 125-130.

[39] J. He and W. R. King, "The role of user participation in information systems development: Implications from a meta-analysis," *Journal of Management Information Systems,* vol. 25, pp. 301-331, 2008.

[40] N. Seyff, *et al.*, "Using mobile re tools to give end-users their own voice," in *18th IEEE International Requirements Engineering Conference (RE),* 2010, pp. 37-46.

[41] S. Jones, *et al.*, "User roles in asynchronous distributed collaborative idea generation," in *Proceedings of the 8th ACM conference on Creativity and cognition*, 2011, pp. 349-350.

[42] L. V. Galvis Carreño and K. Winbladh, "Analysis of user comments: an approach for software requirements evolution," in *Proceedings of the 2013 International Conference on Software Engineering,* 2013, pp. 582-591.