Knowledge Management in Distributed Agile Projects

Mohammad Abdur Razzak
Rajib Ahmed

School of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona
Sweden
This thesis is submitted to the School of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 2*20 weeks of full time studies.
Contact Information:

Author(s):
Mohammad Abdur Razzak
E-mail: more10@student.bth.se

Rajib Ahmed
E-mail: rae10@student.bth.se

University advisor(s):
Darja Šmite
School of Computing

School of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona
Sweden

Internet : www.bth.se/com
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57
Abstract

Knowledge management (KM) is essential for success in Global Software Development (GSD) or Distributed Software Development (DSD) or Global Software Engineering (GSE). Software organizations are managing knowledge in innovative ways to increase productivity. One of the major objectives of KM is to improve productivity through effective knowledge sharing and transfer. Therefore, to maintain effective knowledge sharing in distributed agile projects, practitioners’ need to adopt different types of knowledge sharing techniques and strategies.

Distributed projects introduce new challenges to KM. So, practices that are used in agile teams became difficult to put into action in distributed development. Though, informal communication is the key enabler for knowledge sharing. But when agile project is distributed, informal communication and knowledge sharing is challenged by low communication bandwidth between distributed team members as well as social and cultural distance.

In the work presented in this thesis, we have made an overview of empirical studies of knowledge management in distributed agile projects. Based on the main theme of this study, we have categorized and reported our findings on major concepts that need empirical investigation.

We have classified the main research theme in this thesis within two sub research themes:

- **RT1**: Knowledge sharing activities in distributed agile projects.
- **RT2**: Spatial knowledge sharing in a distributed agile project.

The main contributions are:

- **C1**: Empirical observations regarding knowledge sharing activities in distributed agile projects.
- **C2**: Empirical observations regarding spatial knowledge sharing in distributed agile project.
- **C3**: Process improvement scope and guidelines for the studied project.

**Keywords**: Knowledge Management, Knowledge sharing, Agile, Distributed, Spatial School.
Acknowledgments

We are thankful to our supervisor, Dr. Darja Šmite, who was the real source of encouragement and motivation during the whole work of thesis. Her useful suggestions, support, guidance, advice, countless discussion and ideas to bottleneck problems encountered during this thesis work were just immeasurable. This thesis would not have been possible without her valuable time.

We would like to thank Dr. Marcel Bogers, Associate Professor at University of Southern Denmark and Samireh Jalali, PhD student at Blekinge Institute of Technology, for their valuable suggestion during this research work. Special thanks to our interviewee and Company Alpha for giving us a part of their precious time, the valuable information and useful feedbacks. We would also like to thank Dr. Finn Olav Bjørnson and Dr. Geir Kjetil Hanssen for lending us their Phd thesis template.

Finally, we are deeply grateful to our families and friends. We would like to mention few names Jannatul Ferdous, Pangkaj Paul, Utpol Quraishy and Shafiqul Islam, for their continuous motivation helped us to overcome all problems which we encountered.
Abbreviation

- C - Contribution
- RT- Research Theme
- BS- Background Study
- KM- Knowledge Management
- DAP- Distributed Agile Project
- GSE- Global Software Engineering
- GSD- Global Software Development
- KMS- Knowledge Management Schools
- DSD- Distributed Software Development
List of Figures

1.1 Evolution of research .................................................. 5
1.2 Study design ............................................................. 6
List of Tables

1.1 Focus of the studies .................................................. 6
2.1 Different teams in Alpha ............................................. 16
2.2 Research activities ..................................................... 16
## Contents

Abstract i  
Acknowledgments ii  
Abbreviation iii  

1 Introduction 1  
1.1 Problem Outline 1  
1.2 Motivation 4  
1.2.1 What are we studying? 4  
1.2.2 Why are we interested in it? 4  
1.2.3 Why should this be interesting to others? 4  
1.3 Research Questions 5  
1.4 Research Design 6  
1.5 Thesis Structures 7  

2 Research Approach 8  
2.1 Background Study 8  
2.1.1 Data Collection 8  
2.1.2 Validity Threats 9  
2.2 Study 1 9  
2.2.1 Aim and Objectives 9  
2.2.2 Research questions 10  
2.2.3 Study 1 outcome 10  
2.2.4 Research Methodology 10  
2.2.5 Validity Threats 12  
2.3 Study 2 14  
2.3.1 Aim and Objectives 14  
2.3.2 Research Question 14  
2.3.3 Study 2 outcome 14  
2.3.4 Research Methodology 15  
2.3.5 The Case 15  
2.3.6 Limitation and Validity of Threats 17
3 Summaries

3.1 Knowledge Sharing in Distributed Agile Projects (RT1) ........................................ 19
3.2 Spatial Knowledge Creation and Sharing in a Distributed Agile Project (RT2) ................ 21

4 Conclusion and Future work

4.1 Knowledge sharing in distributed agile projects ...................................................... 23
4.2 Spatial knowledge sharing in a distributed agile project ............................................ 24
4.3 Research Goal ............................................................................................................. 25
4.4 General Observation ................................................................................................. 26
4.5 Direction for future work .......................................................................................... 27

A Appendix

A.1 Interview Questions .................................................................................................. 28
A.2 Case Study Protocol ................................................................................................. 30
A.3 Invitation letter to participate on semi-structured interview ..................................... 32

References ...................................................................................................................... 33
Chapter 1

Introduction

1.1 Problem Outline

Software development is dependent on Knowledge management (KM), since it is a knowledge intensive job. This enforces software organizations to manage their knowledge and later use it in smarter innovative ways to solve problems [44]. It helps software development organizations to acquire and maintain competitive advantage. KM is crucial for success in global software development [39].

Global software development is “software work which is attempted or engage in different geographical location across the national boundaries in a coordinated fashion to involve synchronous and asynchronous interaction” [43]. In the globally distributed agile project, team members share project-specific knowledge through frequent face-to-face interaction, effective communication and customer collaboration [3]. In agile software development, collaboration and coordination depended on communication, which is central for successful software development [48]. Software development depends on the developer’s knowledge and experience [30]. So, success of agile projects relies on effective knowledge sharing among teams. Some studies identified knowledge sharing is difficult in distributed agile teams due to lack of face-to-face communication between teams [7, 26].

To foster dynamic knowledge sharing, improve productivity and coordination in software teams, agile approaches were introduced. Agile team shares knowledge through several practices [11]: pair-programming, release and sprint planning, customer collaboration, cross-functional teams, daily scrum meetings and project retrospectives etc. But, the authors [11] argue that, these practices are team-oriented and rely on face-to-face interaction between team members. These practices do not facilitate knowledge sharing in distributed agile teams but effective for collocated and small teams. In traditional software development, knowledge is stored explicitly in documentation, but in agile development, knowledge is tacit which is in human mind [29]. So, converting tacit knowledge to explicit knowledge is one of the greatest challenges of knowledge management [37]. Due to the absence of explicit documentation in agile software development, experts need to spend much time in repeatedly answering the same questions,
knowledge is lost when experienced developers leave project. This is cause informal communication sometimes cannot serve as recorded documents, less support for reusability and less contribution in organizational knowledge [29].

Nowadays, to reduce the development cost and capture the global market, many software organizations become global. Along that, agile projects also become distributed. Software development is considered as a complex, knowledge intensive and rapidly changing activity. Where number of individuals, teams and organizations involve fulfilling common goal, interest and responsibilities [13, 36]. Technological and strategic knowledge helps developers to communicate; so it is essential to keep the knowledge stored in the organization for the future reuse. Devenport and Prusak [14] define it as “a method that simplifies the process of sharing, distributing, creating, capturing and understanding the company’s knowledge”. As, size of the organization grows rapidly, it becomes harder to find where the knowledge resides. Research shows, if companies manage their knowledge in a better way, they can increase quality, and decrease the time and development cost [42]. To improve the organizational performance, it is important to manage knowledge in a structured way which will help to convey right knowledge to right people on right time.

One of the main challenges of KM in agile software development is to convert tacit knowledge to explicit knowledge, as well as explicit knowledge to tacit knowledge [30]. Hansen et al. [23] recognized two strategies that an organization can choose for preserving both tacit and explicit knowledge. Codification, is systematizing and storing information about the company that constitutes knowledge (Knowledge-as-object [1, 24, 34, 47]). This codification strategy develops an electronic document system that codifies, store and allows to reuse of knowledge. This codification strategy also helps the new team members to reuse stored knowledge [23]. In the company, anyone can retrieve the codified knowledge without having contact with the person who originally developed it [23]. Geographically distributed or dispersed teams are benefited from codified knowledge through reuse, learning and innovation. Personalization (Knowledge-as-relationship [9, 37]), supports the flow of information in a company which is centrally stored information about the knowledge source [4, 22]. Knowledge seekers do not need to search in the documentation because this type of knowledge heavily relies on experts. But in the distributed settings it might be difficult to understand tacit knowledge due to language barriers, cultural factors and term used. So, through deep understanding is required to share and capture the tacit knowledge.

Michael Earl [17] has classified knowledge management (KM) strategy into three categories: technocratic, economic and behavioral. Earl also divided these three categories into seven schools, Technocratic: System, Cartographic and En-
Distributed projects introduce new challenges to KM. So, practices that are used in agile teams became difficult to put into action in distributed development. In the agile collocated development, informal communication is the key enabler for knowledge sharing but when agile project is distributed, informal communication and knowledge sharing is challenging due to low communication bandwidth as well as social and cultural distance [31]. Due to spatial, temporal and cultural factors, communication also gets aggravated in the distributed settings [25]. Several studies [8, 26] reported that, knowledge sharing in the distributed agile project is difficult due the challenges in communication, specially face-to-face interaction between team members in different geographical locations. There are also a lot of knowledge resides in the office space such as white board, taskboards, innovation board and so forth. Those knowledge might be helpful for the distributed team members. However, there are also no study reported so far that, spatial knowledge fostering KM activities for globally distributed teams. To identify the state issues, this research aims to identify how not to loose the benefits that agile practices provide with respect to KM in distributed projects.
1.2 Motivation

The overall perspective of this research was:

*How not to lose the benefits that agile practices provide with respect to Knowledge Management (KM) in distributed projects?*

We formulated following questions, to identify specific studies and research questions from overall research topic, goal.

1.2.1 *What are we studying?*

- We studied knowledge sharing approaches (in particular how practitioners applied knowledge sharing techniques and strategies in distributed agile projects) from knowledge management perspective in distributed agile projects.

1.2.2 *Why are we interested in it?*

- Because software development is a knowledge intensive job and it highly depends on the team members’ (developers, software architect, QA etc.) knowledge and experience. Distributed projects introduce new challenges to KM. So, practices that are used in agile teams became difficult to put into action in distributed development and that affects success in distributed agile projects. To identify the stated issues, we wanted to find out how knowledge creation and sharing activities are performed in distributed agile projects.

1.2.3 *Why should this be interesting to others?*

- Both research community and practitioners will get benefits from this research. Both will gains deeper understanding about knowledge sharing activities in distributed agile projects. Practitioners might get help by actively applying different types of knowledge sharing techniques and strategies in distributed agile projects.
1.3 Research Questions

Knowledge management is a vast area of research. Due to resource constraints, this research will not cover all related areas. Instead, it will focus on knowledge sharing in distributed agile projects. This thesis presents two studies where knowledge sharing activities were studied from knowledge management perspective in distributed agile projects. Based on the main theme of this research, we have grouped this research into two sub-themes (see in Figure 1.1). Our first research theme concerns knowledge sharing activities in distributed agile projects and second research theme is spatial knowledge creation and sharing activities in distributed agile projects.

![Figure 1.1: Evolution of research](image)

<table>
<thead>
<tr>
<th>Research Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>RQ1: How knowledge sharing activities are performed in distributed agile projects?</td>
</tr>
<tr>
<td>RQ2: How spatial knowledge creation and sharing activities are performed in a distributed agile project?</td>
</tr>
</tbody>
</table>
1.4 Research Design

This research has been divided into two parts (see in Figure 1.2). In addition to the two research papers, we have also developed two background studies related to the knowledge sharing in distributed agile projects. Later, which helped us to establish our study. Background 1 has been used as input of study 1 and study 1 has been used to answer research question 1. Likewise, background study 2 and study 1 both have been used as input of study 2 which answered research question 2.

![Study design](image)

The two studies covered the following topics see in Table 1.1:

<table>
<thead>
<tr>
<th>Study</th>
<th>Focus</th>
<th>Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study 1</td>
<td>The intention of this study is to find out knowledge sharing techniques for both knowledge creation and sharing, strategies applied and challenges faced by the practitioners in distributed agile projects.</td>
<td>P1</td>
</tr>
<tr>
<td>Study 2</td>
<td>This study presents a case study that discovered spatial knowledge sharing activities in a distributed agile project.</td>
<td>P2</td>
</tr>
</tbody>
</table>

*Figure 1.2: Study design*
1.5 Thesis Structures

The remainder of the thesis consists of two parts.

PART I - Summary of studies

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 - Research Approach</td>
<td>This section describes the research goal and related research questions. This chapter also gives details overview on research approach, how data have been collected and analyzed.</td>
</tr>
<tr>
<td>3 - Summaries</td>
<td>Summary of both studies are discussed in this chapter.</td>
</tr>
<tr>
<td>4 - Conclusion</td>
<td>This chapter conclude through overall knowledge gained in this thesis by answering results of research questions and providing suggestions for possible direction of future research.</td>
</tr>
</tbody>
</table>

PART II - Included papers


- We are planning to submit both papers to 8th IEEE International Conference on Global Software Engineering, Bari, Italy, 2013.
Due to difference in nature of each study of this research, different approaches were followed to address them individually.

Because this research addresses an issue is rather under-investigated, for this reason the study takes an explorative approach. Exploratory research helps to find out what is happening, seeking new insights and gathering ideas [41]. In exploratory research, typical techniques like case study, observation and historical analysis are used, which provides both qualitative and quantitative data. This research consists of two snowballing literature reviews, series of semi-structured interviews and a single-case study.

2.1 Background Study

The reason of choosing literature review during background studies is to summarize existing evidence, identify gaps in the current research and provide background to position new research activities. In the field of information system Webster et al. [49] proposed an approach called snowballing systematic literature studies as the main method to find relevant literature. The authors also highlight both backward snowballing (from the reference lists) and forward snowballing (finding citations to the papers). Jalali and Wohlin [27] conducted a study on database searches vs backward snowballing (which is based on their previous work) and authors conclude that, in both studies they got similar set of results which inspired us to perform snowballing literature study rather than a systematic literature study in this thesis.

2.1.1 Data Collection

According to Webster et al. [49] and Samireh Jalali et al. [27], snowballing search method can be summarized into three steps:

- Start the searches in the leading journals and/or conference proceeding to get starting set of papers.
Chapter 2. Research Approach

- Go backward by reviewing the reference lists of the relevant articles found in step 1 and step 2 (iterate until no new papers are identified).

- Go forward by identifying articles citing the articles identified in the previous steps.

In this research, we performed two snowballing literature review. At the beginning, we evaluated the relevancy of papers and then in order to find additional sources we went through the reference list of the relevant papers. The process was stopped when we could not add any further relevant papers published in the time period 1992-2012 (study 1) and 2002-2012 (study 2). The reason of choosing 20 years period for study 1 and 10 years period for study 2, was to establish theoretical consideration of this research.

Data Retrieval

Due to limitations of the Google Scholar, search area was limited to the title of the papers. Along that other limitations were made e.g. language to be English; the publication year to be 2012; only within Engineering and Computer Science; and it had to be at least summaries; and only articles and patents. Finally, through two snowballing literature survey we found 37 papers for study 1 (Section 2.2) and 3 papers for study 2 (Section 2.3), that were selected as primary papers for the data extraction and synthesis. The background of this research was built up based on those findings.

2.1.2 Validity Threats

We searched in Google Scholar (only once) using search terms and then limiting the search to 2012 to identify a starting set of papers for the backward snowballing. All papers are categorized as “relevant”, “irrelevant” or “maybe relevant” based on the evidence found in the title, abstract or keywords implicitly or explicitly. Without showing previous judgments both “irrelevant” and “maybe relevant” papers were given to second researcher for further analysis. Both researchers’ involved with this process which helped to mitigate reliability threat.

2.2 Study 1

2.2.1 Aim and Objectives

The aim of study 1 is to identify how distributed agile teams contribute to knowledge creation and sharing activities.
Chapter 2. Research Approach

- O1.1 Finding techniques applied by the practitioners to create shared knowledge among distributed project.
- O1.2 Finding KM strategies applied by the practitioners to share knowledge among distributed project.
- O1.3 Finding challenges faced by the practitioners to share knowledge in the distributed environment.

2.2.2 Research questions

RQ1: How knowledge sharing activities are performed in distributed agile projects?

- **RQ1.1:** How do team members contribute to knowledge creation in a distributed agile project?
- **RQ1.2:** How do team members share knowledge in a distributed agile project?
- **RQ1.3:** What are the challenges faced by the practitioners when sharing knowledge in a distributed agile project?

2.2.3 Study 1 outcome

- **Out1:** Description of knowledge management activities (creation and sharing) in distributed agile projects.

2.2.4 Research Methodology

Because this research addresses an issue which is rather under-investigated, this study takes an explorative approach. Exploratory research helps to find out what is happening, seeking new insights and gathering ideas [32, 41]. In some qualitative research, data collected through observation or interviews are exploratory in nature. So, extensive interviews might be helpful to handle this situation [45]. This type of exploratory research was also helpful to achieve goal by analyzing similarities and difference among the cases [12]. The primary focus of this study was to discover the knowledge sharing practices in distributed agile projects in order to identify techniques, strategies and challenges.
Chapter 2. Research Approach

Sampling

Convenient sampling was used to select the interviewees. The selection criteria for these interviewees were based on kind of company they work at, the experience of the company in distributed agile development (more than 2 years), their role in the distributed team as well as in the company, project duration and project distribution. The participants of this research were Project manager, Team lead, Software Architect, Line manager, Senior Software developer, System developer and Scrum master in different countries involved in distributed agile projects, located in different countries i.e. Sweden, Norway, Germany, Ukraine, China, India, Bangladesh, USA, and Latvia. To get the rounded perspective of this research phenomenon we included different roles from the agile team.

Data Collection

There are three types of interview techniques namely structured, semi-structured and unstructured [19]. Due to qualitative nature of this study we used semi-structured interviews for conducting series of interviews in software industries involved in distributed agile projects. According to Robson [40], an in-depth semi-structured interview is helpful for finding out what is happening and seeking new insights. Because of the exploratory nature of this study, seven semi-structured interviews were conducted in order to identify how practitioners are creating, storing and sharing knowledge related to software development in geographically distributed agile teams. These semi-structured interviews were combination of both open and focused questions. It helps both interviewer and interviewee to discuss on topic in more details. The reason of choosing semi-structured interview is to prompt and probe deeper into the situation. It also helped the authors to get information from individuals about their own practices, believes, and opinions which included both past or present experience. Before the interviews started, the researchers discussed about overall goal of this research to interviewee. The interview questions were descriptive and with the base questions there were follow up questions asked based on the discussion. We were concerned about some key terms: shared knowledge creation, knowledge transfer, strategies and challenges which later helped us for data analysis and those terms also evolve with interview questions (see in Appendix A.1). We conducted Seventeen semi-structured interviews from Six different companies. Selected companies are involved with software product development with different organizational setting and structure located in different countries. The duration of these interviews were on average 60 minutes and the interview sessions were tape recorded. Among seventeen semi-structured interviews nine were conducted through Skype and eight were face-to-face depending on distance between interviewer and interviewee.
Analysis and Synthesis

In qualitative research, data analysis is the most difficult and crucial aspect. According to Basit [2], raw data can not help the reader to understand the social world or participants view unless such data is systematically analyzed. To organize and make sense of collected data we adopted thematic analysis [10] technique during analysis. Thematic analysis is used to identify, analyze and report patterns or themes within data. It minimally organizes and describes data set in-details. In thematic analysis a theme that captures data with relate to research questions and represent them into a pattern within the data set [10]. This analysis performed through a process which maintain six phases to establish meaningful patterns of the data set. Braun and Clarke [10] provides an outline through the six phases of analysis. These phases are: familiarization with data, generating initial codes, searching for themes among code, reviewing themes, defining and naming themes, and producing the final report.

In first stage, we transcribed all collected interview data into written form in order to conduct a thematic analysis. It helped us to identify possible theme, patterns and develop potential codes [21]. Second phase start with initial codes from the extract data. There are different types of Coding techniques suggested in different studies such as open, axial, selective, descriptive/topic and pattern/analytic [46, 38, 35]. In our case, we applied Open coding technique and went through all transcribed textual data by highlighting section of the selected codes. That also helped us to relate coded data with research theme and research questions. In third stage, we analyzed broader level of theme rather than codes that helps to sort different codes into potential themes [10]. As Braun and Clarke suggested, to code as many potential themes/patterns as possible because initially some themes seems to insignificant but later they may be important in the analysis process. Later, mindmaps tool were used to represent them into theme-piles. This stage gave us sense of the significance of individual themes. Stage four is reviewing themes. In this stage we identified irrelevant (not enough or diverse) data with relate to different themes and broken down into separate themes. After refining all themes we identified “essence” of each theme and different aspects of the data each theme captures in stage five. At the end, in stage six, we provided extract data with relate to research questions and present some dialog that connected with different themes in support of results and discussion sections.

2.2.5 Validity Threats

To handle validity threats it is important to identify all possible factors that might affect the accuracy or dependability of the results.
Chapter 2. Research Approach

Internal Validity

Internal validity for qualitative research mostly relates to the researchers biasness and interpretation of data [6]. For selecting similar knowledge level of our interviewees we went through interviewee profile in Linkedin and their years of experience. After all basic findings, interviewer sent formal mail to interviewee with invitation letter to involve with this research. To mitigate threat of diverging to our biasness interview questions were designed to have majority open ended questions. Also to address our inexperience in interview question design we took help from our experienced advisor. Every interview started with similar introduction and some clarification questions. Then this recoded interview was transcribed immediately after the interview to reduce the risk of missing some information. Furthermore, researchers sent interview report to interviewee in order to check whether interview data correctly transcribed and confirm the content that indicates participants thoughts, viewpoints, feelings and experiences. In qualitative research it is an important part to understand interviewee inner words. To maintain reliability during data analysis we used thematic qualitative data analysis technique, that helped to identify, analyze and report themes within data. The extracted data from transcribed data is checked twice for any discrepancy by two researchers.

External Validity

External validity is aim of interest in topic or experience that can affect to fulfill research outcome [12]. This validity threat is more applicable to research that are quantitative and which tries to generalize outcome of the research. In our research we do not conclude to any generalized statement from interviews of seven different agile teams. This research is a social inquiry artifacts used in distributed agile software development teams for knowledge management.

Qualitative Validity Threats

This research encountered a major risk during information gathering from industry. Involving with academic research its totally depends on willingness and availability of industrial responsible. There are three types of validity threats with relevant to qualitative research [33], these are descriptive validity, interpretive validity and theoretical validity [28]. Descriptive validity indicates accuracy about descriptive reporting or information such as, events, objectives, behaviors, settings and places [28]. In qualitative research description is an important issue to maintain descriptive validity. Investigator triangulation techniques applied to mitigate this issue where both researchers together collect, transcribed and analyzed the data. The findings from each researcher also reviewed by other one to ensure the similarities of their conclusion. Along the investigator triangulation, we also maintained data triangulation, negative case sampling, reflexivity and
pattern matching [28]. Data triangulation helps to understand phenomenon from multiple sources (interviewee). In order to compare different participants answer from same organization we conducted multiple interviews.

2.3 Study 2

2.3.1 Aim and Objectives

The aim of the study 2 is to find out, how spatial knowledge facilitate KM activities in a distributed agile project.

- **O2.1** Finding spatial KM items in the distributed agile projects.
- **O2.2** Finding tools and techniques applied by the agile team to create both local and global knowledge by using office space.
- **O2.3** Finding techniques to share office space knowledge among distributed agile team members.
- **O2.4** Finding common understanding between team members.

2.3.2 Research Question

RQ2: *How spatial knowledge creation and sharing activities are performed in distributed agile project?*

- **RQ2.1:** What spatial knowledge creation strategies are practiced locally in a distributed agile team?
- **RQ2.2:** What are the approaches agile team practices to share spatial knowledge with remote team members?

2.3.3 Study 2 outcome

- **Out1:** Description of spatial knowledge sharing activities in distributed agile project.
- **Out2:** Process improvement scope and guidelines for the studied project.
2.3.4 Research Methodology

This case study had focuses on addressing RQ2 by developing insights into understanding of spatial knowledge sharing activities in distributed agile project. This research entail as an exploratory single-case study. Exploratory case study best suited for situation where in-depth and detail studies are unavailable. Case study research works with multiple variables in social context [50]. This provide researchers with rich descriptive data and can identify patterns based on the results. Case studies can be used for both qualitative and quantitative research. But it is uncommon to find case studies with quantitative or statistical methods.

There are two types of observation such as participant: being involved and structured: watching from “outside” during case study[15]. So, to observe the flow of spatial knowledge in the selected company participatory case study was performed. Participatory research is method where researcher takes part in the daily routines, events, rituals and interactions, that provides a deeper understanding [15].

2.3.5 The Case

This section describes in details about the subject of this case study that were conducted during the period June, 2012. The investigated company we named as Alpha for anonymity. An overview of Alpha is given in Table 2.1. Alpha was established in 2001 as an internet startup company. Briefly, the goal of the alpha was to provide consumers with unbiased information of any kind of consumer products. Now in year 2012, it has grown to be the second largest software product in the world among its competitor. It operates with four different offices from Sweden, Germany, United States and China. Alpha is medium sized having 80-100 employee distributed in different countries. Though alpha operates world-wide, the software product is developed in two locations. Most of the software development takes place in China and some features are developed in Sweden. There are three different teams in Sweden sites are working on three different department of development. These teams are i) Data-flow, ii) Data analysis and iii) Application development. This case study is based on the application development team. Currently, application development team is involved with China team in one collaborative project. Though, alpha adopt Kanban to coordinate different departments in Sweden sites, but application development team uses scrum method to coordinate between globally distributed developers in China. The team only practices daily Scrum from scrum method.
Table 2.1: Different teams in Alpha

<table>
<thead>
<tr>
<th>Departments</th>
<th>Sweden</th>
<th>China</th>
<th>Method(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Development Team</td>
<td>Team Leader(1), Developers(2)</td>
<td>Developers(5)</td>
<td>Scrum</td>
</tr>
<tr>
<td>Data Flow Team</td>
<td>Team leader(1), Developer(1)</td>
<td>Developers(3)</td>
<td>Kanban</td>
</tr>
<tr>
<td>Data Analysis Team</td>
<td>Team Leader(1), Developers(2)</td>
<td>Developers(4)</td>
<td>Kanban</td>
</tr>
</tbody>
</table>

Data Collection

To address the both research questions we performed participatory observational case study over a week period of time, and then to get in-depth evidence we also performed semi-structured interviews. An overview of this single-case study activities is given in Table 2.2.

Table 2.2: Research activities

<table>
<thead>
<tr>
<th>Activities</th>
<th>Participants</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observation</td>
<td>Swedish team- One Week</td>
<td>Focuses on daily activities of a Agile team.</td>
</tr>
<tr>
<td>Interviews- 3</td>
<td>Project lead and developer</td>
<td>Access of shared knowledge, missing items and challenges</td>
</tr>
<tr>
<td></td>
<td>- Swedish team</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Developer- Chinese team</td>
<td>during collaborative development.</td>
</tr>
</tbody>
</table>

Yin[50] represents, six sources of data in case studies: documents, archived record, structured interviews, direct observation, participatory observation and artifacts. Firstly, both researchers performed participatory observation to collect and acquire multi-faced data from daily development activities with development team. This Participatory observation is a common data collection method used in office setting [20]. Participatory observation gave us access to events, processes and physical artifacts. Both researchers maintained dairy to note down daily observational activities, that helped later to link between different findings. We also collected data from documents which are shared among distributed agile team using Google Documents. And analyzed issue tracking tool used by the company. Along that, we also examined physical space, documents, tools and physical artifacts used by the agile team. During the case study, one of the researcher participated on daily software development activities with the application development team. That gave opportunity to the researcher for continuous communication with remote team members.
Secondly, we have then interviewed three team members from the agile team for further triangulating the derived data. Two interviews were held face to face with collocated project lead and one developer, moderated by both researchers. One team member from China was interviewed in a video conference via Skype. Semi-structured interviews were conducted to find out what is happening and giving us new insights [40]. Data collected from the interviews were used for triangulation while comparing or contrasting them with data collected during participatory observations. These semi-structured interviews were combination of both open and focused questions. It helps both interviewer and interviewee to discuss on topic in more details. The reason of choosing semi-structured interview is to prompt and probe deeper into the situation. It also helps the interviewer to get information from individual about their own practices, believe, and opinion which included both past or present experience. The interview questions were descriptive and with the base questions there were follow up questions asked based on the discussion. Before interview start the researcher gave overall goal of this research to interviewee. These interviews duration were on average 60 minutes and tape recorded the interview session.

Data Analysis

We had different data sources feeding inputs to this research. Organized data helps reader to understand the context of research [2]. The result of each interviews were documented and report sent to interviewee to check whether interviews data correctly transcribed. Data collected from different sources were analyzed with the help of qualitative data analysis technique called thematic analysis technique [10]. The recorded interviews data were revisited multiple times and that was triangulated with the observation data during the analysis phase. Both researchers involved during analysis phase and performed six phases thematic qualitative data analysis. That helped both researchers to relate data sets among research theme and research questions. Later, we used mind mapping tools to visualize the data patterns. The data presented in this research for both research questions are based on the dairy, documents, overall observation and interviews. That interviews data were helped for further confirm of our observation.

2.3.6 Limitation and Validity of Threats

This research entails as exploratory case study, thus it has some limitations. There are two types of validity (in addition to Reliability) that have to be considered, namely construct validity and external validity [50]. Construct validity involves creating correct operational measures for the concept that are measured in the case study. Multiple sources of evidence are collected by both researchers during data analysis phase. Later, those multiple sources of data helped for
data triangulation within the data sets. During the interview sessions researchers applied chronological order in the discussion to maintain the chain of evidence. Researchers sent the case study draft report to key informant (the interviewee) in order to check whether case study report is correct representation. In addition to interviews, some follow-up questions were asked on different issues, maps and charts of the geographical characteristic or layout of office space. The goal of reliability is to minimize the errors and biases in the study. Furthermore, a case study protocol were sent to all interviewees before interviews actually being conducted.

The external validity needs to be obtained which helps to refer, “the domain to which the findings of the case study can be generalized”. The selection criteria played an important role in the creation of external validity [18]. And this is one of the limitations of this exploratory single-case study. We do not make any generalized claim and our findings might not be universally applicable to all distributed agile projects but in the distributed settings dispersed feature team might get help from our findings and results.
Chapter 3

Summaries

To support this thesis, this chapter presents a summary of the results and discussion from the two studies in the two papers. Based on the two studies this chapter is divided into two sections and each section gives an introduction describing the aim of each study and paper, along with main results and discussion. Further details and information about this study can be found in the papers in part II of this thesis.

3.1 Knowledge Sharing in Distributed Agile Projects (RT1)

Our first research theme was chosen to give us an empirical observation regarding knowledge sharing in distributed agile projects. Our main contribution towards RT1 is C1, an empirical observation on this field. During this research, we were driven by the three research questions which helped to achieved the goal of this research. To identify the empirical evidence we conducted series of semi-structured interviews.

Software organization is a knowledge intensive area which enforces software organization to manage their knowledge in smarter ways to solve problems and KM is essential for success in global software development. Success of agile project relies on knowledge sharing among team members. This study resulted in one paper, focusing on important techniques, strategies applied and challenges faced by the practitioners’ in distributed agile projects. From the series of interviews, we have identified different techniques practices by the practitioners to perform knowledge creation and sharing activities in distributed agile projects. Based on the team size and team settings, different teams adopt different types of techniques to perform knowledge creation and sharing activities among globally distributed team members. We have also analyzed those gathered data with consideration of both current research and research gap.

Through reuse, learning and innovation, geographically distributed team get-
ting benefits from codified knowledge. Both *tacit* and *explicit* knowledge are practiced by the team members to manage knowledge locally and globally. Sharing of explicit knowledge helps team members to get right knowledge at the right time. Communication, coordination and collaboration is the key to foster knowledge sharing between team members in agile software development. However, we have seen knowledge sharing in distributed agile projects is challenging due to factors like communication, language and cultural barriers. To mitigate those challenges and succeed in knowledge sharing within and across the border, practitioners adopt different types of techniques to manage knowledge both locally and globally.

Our empirical observation showed, in distributed agile project teams are using different type of techniques and strategies to perform KM activities. We also relate this empirical findings with Earl’s framework [17]. Our study describes, knowledge sharing strategies which are in practice to manage knowledge both locally and globally. Those practices are associated with systems, cartographic, engineering, organizational and spatial schools. Though systems, engineering and organizational schools are explicitly in practice, spatial school have less concern to manage knowledge in distributed agile project. With close observation between software engineering and schools Bjørnson and Dingsøyr found, there are heavy focus on systems and engineering schools [4]. There are also limited number of study focusing on organizational school but no studies was found in software engineering that focuses on spatial aspect [4]. Agile is more closely connected to the *socialization* that also includes spatial schools concept as knowledge sharing strategies. By using our contribution, researchers who are interested in doing research within this field can use our study to identify what types of knowledge sharing techniques and strategies are in practice.
3.2 Spatial Knowledge Creation and Sharing in a Distributed Agile Project (RT2)

The second research theme was chosen with the basis in the theories of Earl framework [17], which states different types of knowledge management schools. Earl classified KM strategy into three categories and also divided these three categories into seven schools. RT2, deals with the Spatial school. The intention of spatial school is to encourage socialization as a means of knowledge exchange. Spatial school is more concerned with the development and utilization of the social capital which is developed from interaction between people both formal or informal and repeatedly over time. Through a single-case study, this study resulted in contribution C2 and C3 respectively.

Nowadays, Agile development methodology is a popular choice among practitioners for distributed development. Knowledge creation and sharing is an integral part of success of any software development project but in distributed settings are challenging because agile methodology rely on lightweight documentation and informal communication. Knowledge often resides in the office space and most of this knowledge is not preserved due to lack of concern. There are a lots of knowledge created around whiteboards, taskboards, innovation boards and so forth in office space. For this empirical study, we investigated one medium scale software company in Sweden. For data collection, we observed a distributed agile team through participating daily development activities with development team. We examined physical space, documents, tools and physical artifact used by the agile team. We have then interviewed three team members from the agile team for further triangulating the extracted data. We also, collected data from documents which are shared among distributed agile teams using Google Documents. And er analyzed issue tracking tool used by the company.

We were driven by the two research questions during this research which contributed to a research paper. The intention of these research questions was to identify strategies of creating local knowledge by using office space and sharing techniques this knowledge among distributed agile team. From the case study we found that, studied team uses different physical objects to create local knowledge around it locally. In office space knowledge is created around agile boards or sometimes in backboards of conference room. Spatial knowledge is created from social interaction involving this physical items. Collocated team using this items creates this knowledge spontaneously but codifying this knowledge is difficult. The physical items related to this spatial knowledge can be codified. But as knowledge is context depended asset, simply codifying spatial knowledge loses the contextual information. We observed spatial knowledge created in collocation can be partially shared by using of internet based tools and communication
media. From Earl’s framework, spatial school is depended on systems school and organization school. We have also observed, knowledge sharing techniques applied by the team members in distributed agile project. We observed, both local and distributed teams depend on the tools to share and manage knowledge.

In support to $C_2$ and $C_3$, this single case study only revealed a part of the whole picture in terms of knowledge management initiatives practiced in industry. Every organization applies different strategies to manage knowledge asset. From our observation and interviews, it was found that developers in agile team is satisfied with their KM initiatives locally. The implementation of KM activities is not always directed as KM initiative, rather a solution for communication and knowledge exchange problem. In conclusion, practitioners are focused on knowledge management with system and organizational schools rather than spatial school due to the lack of effective, inexpensive softwares, hardwares and processes.
The overall goal of this thesis has been to understand knowledge sharing activities in distributed agile project. To establish this understanding, we were driven by two studies for approximately one year period of time.

The main focus of study 1 was, how knowledge sharing activities are performed in distribute agile project. Through this study we became aware about knowledge sharing techniques and strategies applied by the practitioners in distributed agile project. The results of study 1 helped to initiate study 2 which was focused on spatial knowledge sharing in distributed agile projects. Through both studies, the research questions’ answer also developed to fulfill goal of this thesis. We now sum up our main findings and outline possible future works based on our results. Along that, We have also made a couple of additional observation apart from main conclusions of this thesis.

### 4.1 Knowledge sharing in distributed agile projects

Our first research theme investigated knowledge sharing activities in distributed agile projects. We have one major contribution in this theme:

**C1:** Empirical observation regarding knowledge sharing activities in distributed agile projects. Through a series of semi-structured interviews, we gained insight into knowledge sharing activities in distributed agile projects. We found different types of knowledge creation and sharing techniques applied by the practitioners to perform KM activities in distributed agile projects. Along different types of techniques, we have also found that, practitioners adopted different types of strategies to perform knowledge sharing activities. Though codification is one of the crucial part for knowledge sharing, surprisingly we found from studied cases that, practitioners are concerned about knowledge codification. Teams are sharing codified knowledge among remote team members through repositories. During knowledge sharing among distributed team members, practitioners faced different types of challenges, such as, language, communication, misunderstanding, visualization, cultural, technological and time zone difference. To mitigate
those challenges, practitioners also apply different types of mitigation techniques, such as, informal communication, cultural exchange, common platform, tools, visual prototyping, common chat room, rotation, and overlapping hours. Through this contribution we realized the importance of knowledge sharing in distributed agile projects, that helps agile teams to get benefits from KM practices.


text:

4.2 Spatial knowledge sharing in a distributed agile project

Our second research theme investigated spatial knowledge sharing activities in a distributed agile project. We have two major contribution in this theme:

**C2:** Empirical observation regarding spatial knowledge sharing activities in a distributed agile project. Through a single-case study we gained insight into spatial knowledge sharing in distributed agile project. We have found, studied agile team adopted different types of knowledge creation strategies in order to use office space locally. we found, studied team does not directly share any spatial knowledge among remote teams by using any visualization tools. But, team shares most of the spatial knowledge items through repository. Knowledge which is spread across office space can partially be codified and stored in this repository. Because of the perception and discussion, physical boards or other objects are not codified. So, when studied team tried to share spatial knowledge, it then had to use repository and other internet aided tools which is classified as Systems school in Earl’s classification of KM. Also, for this spatial knowledge to be accessible and understandable to remote team members, the agile team relies on practices from Organization school i.e. using instant messaging to explain issues or new feature. So, from this case study we found that organization are more focused on usage of internet based tools rather than investing on expensive physical devices to create sense of common space or sharing knowledge stored in office space.

**C3:** Process improvement scope and guidelines for the studied project. We proposed studied team to use visual board, record video of presentations for spatial knowledge sharing. Visual aids is helpful to for distributed team members to get information at the right time, that may increase the teams’ productivity. This type of visual aid also help to increase trust and confidence between distributed team members. We have also recommend studied team, to be aware about spatial knowledge and team building during distributed agile project.
4.3 Research Goal

Returning, finally, to our research goal for this thesis: *How not to lose the benefits that agile practices provide with respect to Knowledge Management (KM) in distributed projects?* In distributed development, informal communication and knowledge sharing is challenging due to low communication bandwidth. We found that by taking knowledge management perspective on distributed agile projects, we could identify software process improvements that help team members to improve their practices. Our studies showed that, in distributed agile projects, team practices different types of techniques and strategies that help to maintain knowledge sharing activities. Through practicing those techniques and strategies, agile teams getting benefits in distributed agile projects. Our studies also showed, there are a lot of knowledge resides in office space that foster knowledge sharing. With respect to KM, agile team uses office space for knowledge creation and sharing. Our studies have contributed towards the state-of-the-art by contextualizing agile practices with respect to KM that applied by the practitioners’, but there are still a lot of possibilities for research within the field.
Chapter 4. Conclusion and Future work

4.4 General Observation

We have made a couple of additional observation apart from the above presented in main conclusions of this thesis. Though it is not in our research scope, but we have found these observations interesting enough to present, that might be important for future research.

Methodology We noticed from the semi-structured interviews and case study that, organizations are not practicing agile methodology in the full extent. Everyone adopt tailored methodology with mixed of agile and convenient processes. That gives them more freedom and flexibility. In one case, we noticed that organizations are actually not driven by methodology rather by its results. Depending on the situation they adopt features from different methodology whether it is agile, waterfall or opensource. The ideology of their work environment is “what works”.

Overcome barriers Success of agile software development relies on the communication, coordination and control. We noticed in one case, project manager joins daily standup meeting at night (Team distribution: USA-Bangladesh). Its helped that dispersed feature team to maintain team glue and increase the confidence of the team members in their work environment.

Tight schedule Sometimes its tough to share knowledge among team member due to the tight scheduling, of delivering the project on time. We have also noticed that, developers do not properly maintain commits (during pair programming), which later takes much time to understand.

Rotation/Visit According to Lavy and Hazzan, knowledge sharing increases the respect and trust between team members. In distributed development this is one of the integral part to succeed. In our study, we have observed that, due to budgetary constraints team could not frequently visit remote sites. Its also lead to decrease trust between team members.
4.5 Direction for future work

During the research we have found some interesting substances of knowledge management in both agile collocated and distributed project that might be valuable to explore in the future.

**Topic 1: Knowledge sharing in distributed agile projects: What is missing!**

In study 1 we identified, how do a team members contribute to knowledge creation and sharing in a distributed agile project. Findings of “What is missing” between distributed team might be interesting.

**Topic 2: KM in distributed agile projects**

The motivation of topic 2, is in-depth inquiry of how agile distributed project practices knowledge management activities with regard to spatial school. Multiple case studies can be interesting which will help to reveal more empirical data from the industry. Topic 2 might be direct extension of study 2.

**Topic 3: Applicability of agile collocated KM experience in distributed agile project**

It seems KM in collocated project is successful due to high range of face-to-face communication between teams and product owner. The motivation of this research is to observe, how agile collocated teams practices knowledge management activities and possibilities to simulate the collocated agile project experience in the distributed agile project. It will be also interesting to observe how closeness and cohesion facilitate KM activities in the agile project.
Appendix A

A.1 Interview Questions

Introductory Questions:

- What is your role in the organization?
- What are your responsibilities in the current project?
- How long have you been working with the organization?
- How long have you been working with globally distributed projects?
- Team Size and number of developing sites? Who does what? (Distributed, Dispersed or Combinational)

Domain Specific Questions:

- Which software development methodologies do you follow?
- Which agile practices are followed locally and globally? (e.g. Scrum, XP, Kanban, AUP etc.)
- Which types of knowledge do you share among local team members? (e.g. Tacit, Explicit, Embedded)
- How do you share knowledge across sites?
- What problems/challenges did you face while knowledge creation and sharing?
- What have been done to avoid those problems?
- How do you evaluate the success of knowledge sharing among team members in the same sites? Scale 1-10
• How do you evaluate the success of knowledge sharing among distributed teams?
  **Scale 1-10**

• Do you think the scope of shared knowledge is sufficient? Why or Why not?

• If sites share their physical spaces, do you think it will help to create knowledge and access the shared knowledge? Why and why not?

• Do you believe that knowledge sharing activities have positive impact on the team? Can you please describe?

• Do you like to add something, any success or problem scenario or anything?
A.2 Case Study Protocol

Background

All interview data will be used only for academic purpose and kept confidential. This interview is a part of our research about knowledge management in globally distributed software development teams. Knowledge management is essential for success in ever changing global software development. It helps software development organization to acquire and maintain competitive advantage. One of the objectives of KM is to improve productivity through effective knowledge sharing and transfer. Our brief research goals are -

- Finding types of knowledge resides locally in among distributed sites.
- Identifying how sites/locations use office space to create local knowledge.
- Finding preferences of knowledge management techniques and sharing approaches in different sites.
- Understanding the importance and necessity of shared knowledge creation.

Position:

Team Distribution:
Appendix A. Appendix

**Introductory Questions:**

- What is your role in the organization?
- What are your responsibilities in the current project?
- How long have you been working with the organization?
- How long have you been working with globally distributed projects?
- Team Size and number of developing sites? Who does what? *(Distributed, Dispersed or Combinational)*

**Domain Specific Questions:**

- Which software development methodologies do you follow?
- Which agile practices are followed locally and globally? *(e.g. Scrum, XP, Kanban, AUP etc.)*
- Which types of knowledge do you share among local team members? *(e.g. Tacit, Explicit, Embedded)*
- How do you share knowledge across sites?
- What problems/challenges did you face while knowledge creation and sharing?
- What have been done to avoid those problems?
- How do you evaluate the success of knowledge sharing among team members in the same sites?
  **Scale 1-10**
- How do you evaluate the success of knowledge sharing among distributed teams?
  **Scale 1-10**
- Do you think the scope of shared knowledge is sufficient? Why or Why not?
- If sites share their physical spaces, do you think it will help to create knowledge and access the shared knowledge? Why and why not?
- Do you believe that knowledge sharing activities have positive impact on the team? Can you please describe?
- Do you like to add something, any success or problem scenario or anything?
A.3 Invitation letter to participate on semi-structured interview

Study Title: Knowledge Management (KM) in Distributed Agile Projects

Dear Participant,
My name is Mohammad Abdur Razzak. I am a graduate student in the Software Engineering Department at Blekinge Institute of Technology, Karlskrona, Sweden. I am conducting a research study as part of the requirements of my MSc Thesis, and I would like to invite you to participate.

Knowledge management is essential for success in ever changing global software development. It helps software development organization to acquire and maintain competitive advantage. One of the objectives of KM is to improve productivity through effective knowledge sharing and transfer. Knowledge management activities for globally distributed agile teams need further exploration. The target of this research is to discover KM activities in agile software development team in distributed environment. Our intension is to find out, in distributed setting how agile teams contribute to share knowledge creation and transfer, scope of the shared knowledge, and also attempt to find how distributed teams access and use those spatially shared knowledge.

If you decide to participate, you will be asked to describe procedures of knowledge management activities in the agile distributed teams. In particular, we will discuss a) How agile teams contribute to shared knowledge creation and transfer activities in the distributed setting?, b) What is the scope of the shared knowledge? Is it sufficient?, and c) How sites access and use the spatially shared knowledge?

The Interview will mutually agreed upon time and place, and should last about one(1) hour. The interview will be audio taped so that I can accurately reflect on what is discussed. You may feel uncomfortable answering some of the questions. You do not have to answer any questions that you do not wish to. Participation is confidential.

We will be happy to answer any questions you have about the study. You may contact with my faculty advisor, DR. Darja Smite, +46-702-100-858, darja.smite@bth.se, if you have study related questions or problems.

Thank you for your consideration. If you would like to participate, please send me an email listed below to discuss participating.

With kind regards,
Mohammad Abdur Razzak
razzak@live.se
References


References


References


PART II – Included papers
Paper 1

Knowledge Sharing in Distributed Agile Projects: Techniques, Strategies and Challenges

Mohammad Abdur Razzak
School of Computing
Blekinge Institute of Technology
Karlskrona, Sweden
Email: morc10@student.bth.se

Rajib Ahmed
School of Computing
Blekinge Institute of Technology
Karlskrona, Sweden
Email: raae10@student.bth.se

Abstract—Knowledge management (KM) is essential for success in global software development. Software organizations are managing knowledge in innovative ways to increase productivity. In agile software development, collaboration and coordination depend on the communication, which is the key to success. To maintain effective collaboration and coordination in distributed agile projects, practitioners need to adopt different types of knowledge sharing techniques and strategies. There are also few studies that focuses on knowledge sharing in distributed Agile projects. This research identified the knowledge sharing techniques and strategies applied by the practitioners in the distributed agile projects. Challenges faced by the practitioners during knowledge sharing in distributed agile projects are also identified and discussed.

Index Terms—Knowledge management, knowledge sharing, distributed, agile.

I. INTRODUCTION

Software engineering is a knowledge intensive area. This enforces software organizations to manage their knowledge and later use it in smarter innovative ways to solve problems [1]. It helps software development organization to acquire and maintain competitive advantage. KM is crucial for success in global software development [2].

Global software development can be described as “software work which is attempted in different geographical locations across the national boundaries in a coordinated fashion to involve synchronous and asynchronous interaction” [3]. Software developers are knowledge workers who are dependent on each others’ work. In global software development this synchronization is dependent on KM. Some studies identified knowledge sharing is difficult in the distributed agile project due to lack of face-to-face communication between team members [4,5]. In the agile software development collaboration and coordination depends on communication, which is crucial to successful software development [6]. One of the major objectives of KM is to improve productivity through effective knowledge sharing and transfer [7]. So, success of agile projects relies on effective knowledge sharing among teams.

This research focuses on exploring knowledge sharing in the distributed agile projects. More specifically, this research attempts to identify knowledge sharing techniques and strategies practices between locally and globally distributed agile teams, and challenges faced by the practitioners in the distributed agile environment. We are driven by the following research questions:

RQ1: How do team members contribute to knowledge creation in a distributed agile project?

RQ2: How do team members share knowledge in a distributed agile project?

RQ3: What are the challenges faced by the practitioners when sharing knowledge in a distributed agile project?

The rest of the paper is organized as follows. In section II theoretical background of knowledge management in a distributed agile projects is described. In section III we present research methodology applied and followed by validity threats in section IV. The description of series of semi-structured interviews are described in section V. Results of the different findings are present in section VI. Discussion of the empirical studies is provided in section VII. Finally, section VIII concludes the paper with the summary of the major findings and future work.

II. RELATED WORK

Knowledge can have various meanings to different people [8]. In the britannica encyclopedia [9], knowledge is stated as “the fact or condition of knowing something with familiarity gained through experience or association; acquaintance with or understanding of a science, art, or technique; the fact or condition of being aware of something”. From the organizational perspective, knowledge is “a fluid mix of framed experience, values, contextual information and expert insights and grounded truth that provides a framework for evaluating and incorporating new experiences and information” [10]. Knowledge extracts from the information1 as information extracts from the data2, that means knowledge is a mixture.

1Information is data arranged in meaningful patterns [11].
2Data can be described as a series of meaningless outputs from any operation [11].
Software development is considered as a complex, knowledge intensive and rapidly changing activity. Where number of individuals, teams and organizations involve fulfilling common goal, interest and responsibilities [12,13]. Technological and strategic knowledge helps developers to communicate; so it is essential to keep the knowledge stored in the organization for the future reuse. Devenport and Prusak [10] define it as “a method that simplifies the process of sharing, distributing, creating, capturing and understanding the company’s knowledge”. As, size of the organization grows rapidly, it becomes harder to find where the knowledge resides. Research shows, if the companies manage their knowledge in a better way, they can increase quality, and decrease the time and development cost [14]. To improve the organizational performance, it is important to manage knowledge in a structured way which will help to identify the right knowledge to the right people at the right time. O’Del and Grayson [15] discussed that, knowledge management is not a vital methodology; it is a framework, a management mind-set which is based on past experiences and creating new wheels for exchanging knowledge.

Social interaction and conversation are the key elements of effective knowledge sharing in an agile process [16]. Verbal face-to-face interaction facilitates effective knowledge sharing between team members. Though face-to-face interaction is challenging in distributed software development. Research identified [17], team members (Developers or Quality Analysts) rotate or travel different location to work with other team members for a specific period of time, that enables effective knowledge sharing and increase team bonding even when team members are no longer working in the same location.

Software development relies on the team performance [18]. In traditional software development teams apply plan-driven approaches, maintain hierarchy and clear separation of roles. However, in agile software development, work is coordinated by the self-managing team and team decide how to coordinate work. There are three types of feature teams in distributed projects those are distributed feature teams, dispersed feature team, and distributed as well as dispersed teams [6]. Distributed feature teams are the combination of multiple feature teams and teams possibly reside at a different location. The internal communication between features teams are easier, because the nature of teams is collocated, but across the border it is harder due to temporal dispersion. Dispersed feature team members (architect, developers, tester and etc.) are spread over many sites. Team members are spread in the different geographical location, so much effort is needed to increase communication bandwidth between internal teams. In order to work as united team, joint objective is an important issue in dispersed feature teams. Distributed as well as dispersed teams, sometimes this type of teams called as hybrid team which is mixed of distributed and dispersed feature teams. Usually, large global projects adopt this type of team structure to facilitate both business and technological domain together [6].

To foster dynamic knowledge sharing, improve productivity and coordination in software development teams, agile approaches were introduced. Agile team shares knowledge through several practices [19]: pair-programming, release and sprint planning, customer collaboration, cross-functional teams, daily scrum meetings and project retrospectives. But, the authors [19] argue that, these practices are team-oriented and rely on face-to-face interaction between team members. These practices do not facilitate knowledge sharing in distributed agile teams but effective for collocated and small teams. In one study Dorairaj et al. [17] reported that in distributed agile project, team members practice sprint planning, daily scrum, sprint review and project retrospective meetings. Distributed agile team members share knowledge through effective use of knowledge management tools like Wiki, pair-programming and video-conferencing.

Bahli and Zeid [20] mentioned in an empirical study that, extreme programming (XP) improve tacit knowledge creation among the team members in the software development. It also facilitates higher degree of knowledge creation among team members through frequent interaction among team members, effective communication and collaboration in a development team. Knowledge creation in distributed agile teams enhance by the active collaboration among development team members and discussion with customer representative throughout the development phase [17].

KM is considered as software process improvement initiative [1,6]. KM in global software development projects is essential in order to cope with coordination, multiple knowledge sources, time and budgetary constrain [21]. There are two main strategies for knowledge management. Codification, is systematizing and storing information about the company that constitutes knowledge (Knowledge-as-object [16,22–24]). This codification strategy develop an electronic document system that codified, store and allows to reuse of knowledge. This strategy also helps the new team member to reuse stored knowledge [25]. Personalization (Knowledge-as-relationship [26,27]), supports the flow of information in a company which is centrally stored information about the knowledge source [28,29].

Michael Earl [30] has classified knowledge management (KM) into three categories: technocratic, economic and behavioral. Earl also divided these three categories into seven schools, Technocratic: Systems, Cartographic and Engineering, Economic: Commercial and Behavioral: Organizational, Spatial and Strategic. Both Codification [25] strategy and Systems school practice depend on the technology which apply Nonaka’s [31] externalization conversion technique to convert tacit knowledge into explicit knowledge. Research shows, technocratic schools are closely related with traditional software development and those who are developing software
through traditional approaches they are probably getting benefit from technocratic schools [32]. On the other hand, behavioral schools are more related with the agile approaches and agile teams are more benefiting from the behavioral school. A survey in traditional and agile companies shows, agile companies seem to be more satisfied with their knowledge management approaches compared to traditional companies [33]. In the agile software development, knowledge sharing happens through the interaction. Developers share knowledge by working together and through close interaction with customers; and more specifically pair programming, extreme programming, daily scrum meeting, and sprint retrospectives in Scrum. In the traditional software development, knowledge management relied primarily on explicit knowledge but in the agile software development KM relies on tacit knowledge [34]. In the agile software development, information radiator and collocating teams are related with spatial school [33].

In the traditional software development, knowledge stored explicitly in the documentation, but in the agile development methodology the knowledge is tacit [7]. Extracting tacit knowledge to explicit knowledge is one of the greatest challenges of knowledge organization [27]. Due to the absence of explicit knowledge in the agile software development, experts need to spend much time in repeatedly answering the same questions, experienced developers leave project, less support for re-usability and less contribution to organizational knowledge [7]. In the agile collocated development, informal communication is the key enabler for knowledge sharing but when agile project is distributed, informal communication and knowledge sharing is challenging due to low communication bandwidth as well as social and cultural distance [35]. Due to spatial, temporal and cultural factors, communication also get aggravated in the distributed settings [36]. Several studies [5,37] also point out that, knowledge sharing in the distributed agile project is difficult due the challenges in communication, specially face-to-face interaction between team members in different geographical location. To address those problems, we investigated how shared knowledge creation and transfer activities performed in the distributed agile projects. Along that, we also investigated what are the challenges faced by the practitioners when sharing knowledge among globally distributed agile team members.

III. RESEARCH METHODOLOGY

Because this research addresses an issue that has been rather under-investigated for that reason this study takes an explorative approach. Exploratory research helps to find out what is happening, seeking new insights and gathering ideas [38,39]. In some qualitative research, data collected through observation or interviews are exploratory in nature. So, extensive interviews might helpful to handle this situation [40]. This type of exploratory research also helpful to achieve goal by analyzing similarities and difference among the cases [41]. The primary focus of this study is to discover the knowledge sharing practices in distributed agile project in order to identify techniques, strategies and challenges.

A. Sampling

Convenient sampling was used to select the interviewees. The selection criteria for these interviewees were based on kind of company they work at, the experience of the company in distributed agile development (more than 2 years), their role in the distributed team as well as in the company, project duration and project distribution. The participants of this research were Project manager, Team lead, Software Architect, Line manager, Senior Software developer, System developer and Scrum master in different countries involved in distributed agile projects, located in different countries i.e. Sweden, Norway, Germany, Ukraine, China, India, Bangladesh, USA, and Latvia. To get the rounded perspective of this research phenomenon we included different roles from the agile team.

B. Data Collection

There are three types of interview techniques namely structured, semi-structured and unstructured [42]. Due to qualitative nature of this study we used semi-structured interviews for conducting series of interviews in software industries involved in distributed agile projects. According to Robson [43], an in-depth semi-structured interview is helpful for finding out what is happening and seeking new insights. Because of the exploratory nature of this study, seven semi-structured interviews were combination of both open and focused questions. It helps both interviewer and interviewee to discuss on topic in more details. The reason of choosing semi-structured interview is to prompt and probe deeper into the situation. It also helped the authors to get information from individuals about their own practices, believes, and opinions which included both past or present experience. Before the interviews started, the researchers discussed about overall goal of this research to interviewee. The interview questions were descriptive and with the base questions there were follow up questions asked based on the discussion. We were concerned about some key terms: shared knowledge creation, knowledge transfer, strategies and challenges which later helped us for data analysis and those terms also evolve with interview questions (see in Appendix IX). We conducted Seventeen semi-structured interviews from Six different companies. Selected companies are involved with software product development with different organizational setting and structure located in different countries. The duration of these interviews were on average 60 minutes and the interview sessions were tape recorded. Among seventeen semi-structured interviews nine were conducted through Skype and eight were face-to-face depending on distance between interviewer and interviewee.

C. Analysis and Synthesis

In qualitative research, data analysis is the most difficult and crucial aspect. According to Basit [44], raw data can not
help the reader to understand the social world or participants view unless such data is systematically analyzed. To organize and make sense of collected data we adopted thematic analysis [45] technique during analysis. Thematic analysis is used to identify, analyze and report patterns or themes within data. It minimally organizes and describes data set in-details. In thematic analysis a theme that captures data with relate to research questions and represent them into a pattern within the data set [45]. This analysis performed through a process which maintain six phases to establish meaningful patterns of the data set. Braun and Clarke [45] provides an outline through the six phases of analysis. These phases are: familiarization with data, generating initial codes, searching for themes among code, reviewing themes, defining and naming themes, and producing the final report.

In first stage, we transcribed all collected interview data into written form in order to conduct a thematic analysis. It helped us to identify possible theme, patterns and develop potential codes [46]. Second phase start with initial codes from the extract data. There are different types of Coding techniques suggested in different studies such as open, axial, selective, descriptive/topic and pattern/analytic [47–49]. In our case, we applied Open coding technique and went through all transcribed textual data by highlighting section of the selected codes. That also helped us to relate coded data with research theme and research questions. In third stage, we analyzed broader level of theme rather than codes that helps to sort different codes into potential themes [45]. As Braun and Clarke suggested, to code as many potential themes/patterns as possible because initially some themes seems to insignificant but later they may be important in the analysis process. Later, mindmaps tool were used to represent them into theme-piles. This stage gave us sense of the significanse of individual themes. Stage four is reviewing themes. In this stage we identified irrelevant (not enough or diverse) data with relate to different themes and broken down into separate themes. After refining all themes we identified “essence” of each theme and different aspects of the data each theme captures in stage five.

At the end, in stage six, we provided extract data with relate to research questions and present some dialog that connected with different themes in support of results and discussion sections.

IV. VALIDITY THREATS

To handle validity threats it is important to identify all possible factors that might affect the accuracy or dependability of the results.

A. Internal Validity

Internal validity for qualitative research mostly relates to the researchers biasness and interpretation of data [50]. For selecting similar knowledge level of our interviewees we went through interviewee profile in Linkedin and their years of experience. After all basic findings, interviewer sent formal mail to interviewee with invitation letter to involve with this research. To mitigate threat of diverging to our biasness interview questions were designed to have majority open ended questions. Also to address our inexperience in interview question design we took help from our experienced advisor. To handle threat of diverging to our biasness interview questions were designed to have majority open ended questions. Furthermore, researchers sent interview report to interviewee in order to check whether interview data correctly transcribed and confirm the content that indicates participants thoughts, viewpoints, feelings and experiences. In qualitative research it is an important part to understand interviewee inner words. To maintain reliability during data analysis we used thematic qualitative data analysis technique, that helped to identify, analyze and report themes within data. The extracted data from transcribed data is checked twice for any discrepancy by two researchers.

B. External Validity

External validity is aim of interest in topic or experience can affect to fulfill research outcome [41]. This validity threat is more applicable to research that are quantitative and which

<table>
<thead>
<tr>
<th>Projects</th>
<th>Project Distribution</th>
<th>Team Size</th>
<th>Team Types</th>
<th>Agile Position/Roles</th>
<th>No. of Interviewees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha</td>
<td>Sweden-Germany</td>
<td>6-7*</td>
<td>Dispersed</td>
<td>Team Lead Developer</td>
<td>2</td>
</tr>
<tr>
<td>Beta</td>
<td>Norway-Bangladesh</td>
<td>5-6*</td>
<td>Dispersed</td>
<td>Project Manager Developer</td>
<td>2</td>
</tr>
<tr>
<td>Gamma</td>
<td>USA-Bangladesh</td>
<td>12-16**</td>
<td>Distributed</td>
<td>Head of Engineering Senior Developer</td>
<td>3</td>
</tr>
<tr>
<td>Delta</td>
<td>Sweden-Bangladesh</td>
<td>16-18**</td>
<td>Distributed</td>
<td>Software Architect Developer</td>
<td>2</td>
</tr>
<tr>
<td>Epsilon</td>
<td>Latvia- Ukraine</td>
<td>11-15**</td>
<td>Distributed</td>
<td>Project Manager Developer</td>
<td>2</td>
</tr>
<tr>
<td>Zeta</td>
<td>Sweden-China</td>
<td>26-35***</td>
<td>Distributed</td>
<td>Line Manager Software Developers System Developer</td>
<td>4</td>
</tr>
<tr>
<td>Eta</td>
<td>Sweden-India</td>
<td>45-55***</td>
<td>Hybrid</td>
<td>System Developer; Scrum Master</td>
<td>2</td>
</tr>
</tbody>
</table>

In Table I *,**,*** indicates Small, Medium and Large scale teams respectively.
tries to generalize outcome of the research. In our research we do not conclude to any generalized statement from interviews of seven different agile teams. This research is a social inquiry artifacts used in distributed agile software development teams for knowledge management.

C. Qualitative Validity Threats

This research encountered a major risk during information gathering from industry. Involving with academic research its totally depends on willingness and availability of industrial responsible. There are three types of validity threats with relevant to qualitative research [51], these are descriptive validity, interpretive validity and theoretical validity [52]. Descriptive validity indicates accuracy about descriptive reporting or information such as, events, objectives, behaviors, settings and places [52]. In qualitative research description is an important issue to maintain descriptive validity. Investigator triangulation techniques applied to mitigate this issue where both researchers together collect, transcribed and analyzed the data. The findings from each researcher also reviewed by other one to ensure the similarities of their conclusion. Along the investigator triangulation, we also maintained data triangulation, negative case sampling, reflexivity and pattern matching [52]. Data triangulation helps to understand phenomenon from multiple sources (interviewee). In order to compare different participants answer from same organization we conducted multiple interviews.

V. SEVEN DISTRIBUTED AGILE PROJECTS DESCRIPTIONS

This section describes empirical cases that were conducted during the period May-June 2012. There are three types of feature teams in distributed projects, those are: distributed feature teams, dispersed feature team, and hybrid team [6]. Based on the team size, we have divided cases into three categories-Small, Medium and Large scale. Table I shows the overview of different projects.

A. Small Scale

Table II: Small scale teams

<table>
<thead>
<tr>
<th>Projects</th>
<th>Team Distribution</th>
<th>Development Method(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha</td>
<td>Sweden-Germany</td>
<td>Scrum and Kanban</td>
</tr>
<tr>
<td>Beta</td>
<td>Norway-Bangladesh</td>
<td>Feature-driven development</td>
</tr>
</tbody>
</table>

1) Alpha: Alpha is Stockholm based software company and they have offshore sites in different countries. The current project is a collaboration between Stockholm and Germany teams. Stockholm team is responsible for overall project management, testing, features development, bugs fixing and release planning. German team is responsible for core development and testing. The interviewees are both from Stockholm project management team and Germany development team (green color indicates that the interview conducted from the Stockholm and German team see in Figure 1). Team structure- Dispersed feature team (see in Figure 1).

As development methodology Alpha practices Agile which is a combination of both Scrum and Kanban. Stockholm team begins the day with a startup meeting. After the meeting, all task related information is kept on the Scrum board. Redmine is strictly maintained by the team members for documentation and to share knowledge among both local and distributed team members. There are no Scrum and Kanban board at the Germany site. All the information from Scrum and Kanban board are codified into Redmine by the Swedish team to share and exchange knowledge among the German team members. Skype is used as the major communication tool among the remote team. Every week they have 1-2 hours weekly workshop with the local team; that helps team members to share their thoughts with each others. Though Germany team does not have Scrum board, the team accesses the information through redmine.

2) Beta: Beta, is Oslo based software company. The company has three teams in two countries and all teams are working on same project. In Oslo, there are two teams. Team A is responsible for project management and part of development. Team B is responsible for testing. On the other site Bangladesh team is responsible for core development. Team structure- Dispersed Feature Team (see in Figure 3).
Fig. 3: Team distribution in Beta project

As development methodology Beta team practices Agile, more specifically Feature-driven development (FDD). Redmine, gitHub and Skype are used to share features & scheduling, code and screen respectively in the local team. Wiki tool used for documentation. The team do not have any physical board in their local site, team members share all information through redmine and wiki for both local and remote use.

Fig. 4: Beta project KM activities

Project Manager said “...I am not satisfied with the current tools; It’s tough to describe design to new team members. Visual aid is helpful during the discussion”

B. Medium Scale

TABLE III: Medium scale teams

<table>
<thead>
<tr>
<th>Projects</th>
<th>Team Distribution</th>
<th>Development Method(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gamma</td>
<td>USA-Bangladesh</td>
<td>Scrum</td>
</tr>
<tr>
<td>Delta</td>
<td>Sweden-Bangladesh</td>
<td>Agile; Waterfall</td>
</tr>
<tr>
<td>Epsilon</td>
<td>Latvia-Ukraine</td>
<td>Scrum, XP, and Kanban</td>
</tr>
</tbody>
</table>

1) Gamma: Founded by a web visionary and entrepreneur of Silicon Valley, California. Gamma team consists of business leader with proven track records and technical leaders with solid technology expertise and background. The core business of this company is Design and Development. Gamma, closely observed software development life cycle (SDLC) and relied on a lightweight, productive and client-friendly processes. At the beginning they start with traditional waterfall SDLC model but it is no longer with this company due to today’s fast-paced industries where change is the only constant. Now, Gamma teams are using an interactive agile model; continuously iterate from the initial requirements until initial launch and onward until the final launch. Though design and development is the core business of Gamma, with that they also provide Testing and QA, Integration and IT, Tech support, Maintenance and Refactoring. Working with the leadership team, a small project management, business analyst and architecture team in San Jose, CA and a large engineering, IT and technical support team is in Dhaka, Bangladesh. That means the team is distributed feature team (see in Figure 5).

Fig. 5: Team distribution in Gamma project

In one of the current project with Dhaka team USA team is responsible for core design and decision making. As a development methodology they are practicing is Agile, more specifically, they use Scrum. Dhaka team is responsible for development and testing. So, they start day with a short meeting with the USA team. USA team holds the main Scrum board and Dhaka team holds the sub-scrum board (for Specific Sprint). To share the central knowledge with local team they maintain a wiki as a knowledge management tool and documentation. In the distributed environment they maintain wiki (documentation sharing), gitHub (code sharing) and a Skype (communication). To share physical scrum board with the remote team the USA team upload pictures every day in the wiki.

The scope of the knowledge sharing is described by the interviewee as “...It’s ok for a small team, but an electronic white board will be helpful to share knowledge among remote team members. From the business perspective knowledge shar-
ing is a bottleneck and it also helps to reduce redundancy.”

2) Delta: Delta, another example of Dispersed feature teams is distributed between Sweden and Bangladesh. The Swedish team is responsible for both specification and design phase; on the other site, Bangladeshi team is responsible for core development.

As development methodology they adopt Agile but not specifically any practices like Scrum or Kanban. Their motto is What works!. For the task related knowledge sharing, they maintain GreenHopper Atlassian JIRA, they do not have any physical board in their offices. Confluence is used for knowledge management. If someone faces any problem in the local team they directly ask Sage\(^3\) person to get on the fly answer rather than searching. But for the remote team task related knowledge is kept in the sprint management tool and for the domain related knowledge they maintain a common chat room to resolve this specific problem.

The Software Architect of this company said “...Visual aid is helpful if diagram or other issues are hard to describe”

3) Epsilon: Epsilon is Riga-based IT Company providing banking and finance solutions. Epsilon is one of the fastest growing IT service providers in Latvia. Epsilon is specialized in providing IT consulting, as well as IT systems development, integration and software maintenance services.

One of the current projects running is the one with collaboration between Latvia and Ukraine. Latvian team is responsible for project management, requirement analysis, design and verification; and Ukraine team is responsible for the core development. The team uses agile, which is a mix of Scrum, XP and Kanban but not in a strict way. It is tailored and flexible.

Latvian team starts the day with a short meeting. All project and domain related information is kept in the wiki and JIRA. The main challenges faced by Epsilon team to share knowledge among distributed team are distance and visualization. So, to mitigate those challenges they maintain a common platform (same work environments), common chat room and visit remote site (sometimes).

One of the interviewee said “... It is good to have all the information in the electronic board for distributed teams”

C. Large Scale

<table>
<thead>
<tr>
<th>Projects</th>
<th>Team Distribution</th>
<th>Development Method(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zeta</td>
<td>Karlskrona-China</td>
<td>Scrum &amp; Kanban</td>
</tr>
<tr>
<td>Eta</td>
<td>Stockholm-India</td>
<td>Scrum &amp; Kanban</td>
</tr>
</tbody>
</table>

1) Zeta: Zeta, is one of the leading companies in telecoms industry founded in Sweden. One of the current projects is collaboration between Sweden and China where two distributed feature teams are working on one project.

\(^3\)Sage is usually an older person of experience. Can give advice or guidance to the younger team members.
As development methodology both teams practice Agile, more specifically, Scrum and Kanban. To share knowledge with the local team they maintain a wiki, pair programming, technical presentation and technical discussion forum. But to share knowledge among remote team they perform daily conference, sometimes visit remote site, and a technical forum, which is called as knowledge hub.

![Diagram](Zeta project KM activities)

The Line Manager said “...we are planning to buy a digital scrum board to increase communication bandwidth which will help distributed teams to get information at early stage”

2) Eta: One of the current project is collaboration between six distributed feature teams and every team has two remote team members in India. Team structure is hybrid and one of the Stockholm team is responsible for the project management.

As development methodology the teams using Agile which is a mix of both Scrum and Kanban but not in a strict manner. To share knowledge among distributed dispersed teams both formal and informal processes are followed by the team members. Most of the time workshops and seminars are arranged on different topics such as, technologies, tools, design methods to create local knowledge. Also, team meetings and on-demand meetings are maintained for knowledge sharing. To share knowledge across the sites the teams uses electronic media (such as, electronic boards, wikis, forums, documentation, etc.), online conferences, online meetings, and site visit.

One Interviewee mentioned “.....Seems, people sitting nearby places find it easier to communicate and apparently access to knowledge and sharing of knowledge increase. However, electronic mediums are functioning well enough to share knowledge. Electronic boards are in place containing the tasks list to perform and latest information. Along with that, necessary technical and business information are regularly updated in wiki, internal web, etc. Online forums are also used to discuss.”

The interviewee also suggested that “.....Knowledge sharing, within the team and between the teams, is a necessary function to be able to perform effectively. This effectiveness includes timeliness, accuracy in design & development, and following the right technical and business processes.”

VI. RESULTS

In this section, we describe different findings (techniques, strategies and challenges) from Seven cases that promote effective knowledge creation and sharing activities in distributed Agile projects.

A. Knowledge Creation: Locally and Globally

We have found, in distributed agile project team practices different types of techniques for both local and global shared knowledge creation. Pair programming, Customer collaboration, Scrum/Kanban board and Community of practice are explicitly practices by the teams to perform both local and global shared knowledge (see in Table V).

1) Pair Programming: Pair programming is used for both local and global knowledge creation. From the series of semi-structured interviews we have found that, both local and remote team members work together in one workstation to solved specific problem. They help each other to share their thought and create knowledge through discussion. In two cases, we have found, teams does not perform pair
programming for shared knowledge creation among remote team members. In Epsilon (ε) project, all development team members are in one site for that reason they do not need to perform pair programming for global shared knowledge creation. Although, Zeta (ζ) project is collaboration with Chinese team on the same product but development teams does not have any dependency. The development teams working on different module and later core developers merge all module together for specific release. But, the local team perform pair programming in Zeta (ζ) project.

2) Pre-planning game/Customer collaboration: In the development cycle customer has an important role. Customer collaboration helps team to build up technical-business collaboration on a project and also help to set the direction of the project. In the agile software development customers always involve with development teams by providing project requirements and performing acceptance testing. Through customer collaboration agile teams participate to create local knowledge. Evidence also found from different cases that, customer also involve with the remote development teams to create knowledge through continuous discussion and features feedback. We have also found that, customer also involved in issue tracking system, that helps both project manager and developers for early iteration. In two cases (δ and ε), we have found that, customer collaboration perform only in the local sites for shared knowledge creation.

3) Scrum/Kanban boards: The are two types of boards used by the office to create knowledge and common understanding. Scrum board is used for teams that plan their work in sprint. Kanban board used for to manage and construct team work in progress. In table V it is depicted that, teams are using Scrum and Kanban board to for shared knowledge creation among both local and globally distributed team members. In two cases (γ and η), we have found that, teams are using board both locally and globally. In Gamma (γ) project, remote teams has sub-scrum board which is replica of main scrum board. Along that, local team (in γ-project) upload picture of main Scrum board everyday in repository. But, in Eta (η) project, teams are using visual Scrum board to perform shared knowledge creation among distributed team members.

4) Innovation boards: Most of the innovative ideas are kept in the human mind as tacit knowledge. Due to continuous work load sometimes it is impossible to discuss with team member or knowledgeable person. So, rather discussing with someone, people share their ideas through innovation board in explicit way. In the one interview researchers found, teams are using innovation boards to share their ideas with both collocated and remote team members.

5) Workshop/Seminars: Weekly or monthly workshop and seminars are arranged with collaboration between business teams, technical teams and customers to share knowledge about project and latest technologies. This kind of workshop facilitate common understanding and communication between different team members. Workshop also helps to facilitates tacit knowledge sharing through socialization. In studied cases, we have only observed large scale teams practices this techniques locally to create shared knowledge. Later, theme of the workshop/seminars shared among remote team members through repositories.

6) Community of practice: To succeed in the agile project, learning is an important asset for the agile teams. Agile teams practice two mode of learning Peer learning and Community learning. In Peer learning, team members start learning through interacting and collaborating with team members. Community learning, access and conceive information available in knowledge archive or in discussion forum. We have found, community of practice in different projects to perform shared knowledge creation among local and remote team members.

We have also found that, to create shared knowledge team perform technical presentation. But, this activities only performed in the local site and later slides or documents share among remote team members. Technical forum is also in practice to perform shared knowledge creation between local and remote team members.

B. Knowledge Sharing: Locally and Globally

Knowledge exchange is always challenging in the distributed agile team due to a lack of face-to-face interaction among team members. Practitioners and researchers are trying to mitigate those challenges by initiating different kind of techniques and tools. From the studied cases we observed that, practitioner maintain different types of tools and techniques to share knowledge among globally distributed teams. Based

<table>
<thead>
<tr>
<th>Techniques</th>
<th>α</th>
<th>β</th>
<th>γ</th>
<th>δ</th>
<th>ε</th>
<th>ζ</th>
<th>η</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair programming</td>
<td>LG</td>
<td>LG</td>
<td>LG</td>
<td>LG</td>
<td>—</td>
<td>L</td>
<td>LG</td>
</tr>
<tr>
<td>Customer collaboration</td>
<td>LG</td>
<td>—</td>
<td>LG</td>
<td>L</td>
<td>—</td>
<td>L</td>
<td>LG</td>
</tr>
<tr>
<td>Scrum/Kanban boards</td>
<td>L</td>
<td>—</td>
<td>LG</td>
<td>L</td>
<td>—</td>
<td>L</td>
<td>LG</td>
</tr>
<tr>
<td>Innovation boards</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>LG</td>
</tr>
<tr>
<td>Workshops/Seminars</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>L</td>
<td>L</td>
<td></td>
</tr>
<tr>
<td>Community of practice</td>
<td>—</td>
<td>—</td>
<td>LG</td>
<td>LG</td>
<td>L</td>
<td>LG</td>
<td></td>
</tr>
<tr>
<td>Technical presentation</td>
<td>—</td>
<td>L</td>
<td>—</td>
<td>—</td>
<td>L</td>
<td>L</td>
<td></td>
</tr>
<tr>
<td>Technical forum</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>LG</td>
<td>LG</td>
<td></td>
</tr>
</tbody>
</table>

*In Table V, L indicates Locally, G—Globally and “—” not in practice

Dispersed teams- α, β, δ: Distributed teams- γ, ε, ζ, Hybrid team- η

**TABLE V: Knowledge creation techniques: Locally and Globally**
on the findings those knowledge sharing techniques are listed in Table VI.

All studied projects are concerned about repositories to share knowledge between local and remote team members. Most of the task and product related knowledge is kept in the repositories, that are easy to access by the remote team members. Different teams also depends on Daily scrum, Weekly sprints status, Discussion forum, online conference and Common chat room to share knowledge between local and remote team members. Electronic board is helpful to share knowledge across remote teams. Only one case found, those share knowledge between both collocated and distributed team through electronic board.

1) Repositories: To share knowledge among distributed sites local team used different types of repositories like Wiki, JIRA, Redmine, Confluence and Github etc. This type of repositories provide efficient mechanisms to access the codified knowledge. From the gather data it is evident that (see in Table VI), practitioners are most dependent on repositories to share knowledge among both local and distributed team members.

Wiki, according to Ulrike Cress [53], provides new opportunities to learn and collaborative knowledge building & sharing through social interaction and individual learning. In different cases we have found that, Wiki is helpful to start a new threat and discuss about issues with other team members. It is also helpful for new team members, that (wiki) provides details information about feature, documents and so forth.

Project and Issue tracking, Nowadays, almost all medium and large distributed or dispersed agile teams are using JIRA/Redmine to track issues, bugs, tasks, deadlines, code and hours. As collaboration and content sharing tools practitioners used Confluence to share docs, files, ideas, specification, diagrams and mockups. During the interview one project lead said,

“...Most of the time we share tacit knowledge between both local and global team. After that, the information is converted through redmine to make it explicit...” Project Lead- Alpha project.

It is also evident that, in one case we found local team uploads Scrum board pictures, slides and workshop information in the repositories which is codified and easy to access by the remote team members.

2) Pair programming: Pair programming plays an important role to create and share developers knowledge in both locally and globally distributed project. In pair programming, two developers work together at one computer with common goal [54]. In the studied projects, we found teams are using pair programming technique to share knowledge among remote team members. Team member use Skype to share screen among remote team members. Along that we have also found, team use TeamViewer and VPN services to share same computer screen among remote team members to perform pair programming.

3) Daily Scrum/Weekly sprints status/Online conference: Scrum meetings are source for sharing project progress information among team members. Usually a Scrum standup meeting is held in collocation. From the gather data we found, distributed teams practices Scrum standup meeting with Internet Relay Chat (IRC), Skype or other group chatting software. Through daily scrum/weekly sprints status/online conference local team share knowledge among distributed team members. In one case (ζ), we have found, due to less dependency development team does not need to perform Scrum meeting/Weekly sprint status. But team (ζ) maintain online conference to share knowledge among remote team members (if need). However, unless Beta(β) project, other projects are explicitly maintain online conference globally for knowledge sharing among distributed team members.

4) Common chat room: Common chat room is useful to exchange knowledge among distributed teams. From the empirical findings we observed that, for faster and quick communication among distributed team members medium and large scale teams maintain common chat room.

In one case, Software Architect said that, “.....Sprint man-
agement system handle all task related knowledge but for the
domain related knowledge sharing we maintain common chat room. Which help us to resolve specific problem within short time” -Software Architect, Delta project

But, in another case we found that, it is not an efficient way to communicate among distributed teams due to language barriers, common understanding, technological factors and so forth. Frequently misunderstanding occurs and things get wrong. To mitigate this types of problem practitioners also suggested different types of mitigation techniques.

5) Technical forum: The theme of technical forum is learning through sharing knowledge. Technical forum is like communities of practice that creates a network between technical team members. It is a self-organizing group that consist of individuals who share information, experience and technical skill on a specialized discipline [55]. Technical forum assist distributed teams in quick problem solving and reduce development time since team members will not be stuck to recur issues. Building trust between team members in the distributed environment is challenging; so knowledge sharing through technical forum can build trust between developers. Technical forum helps to create and share both local and distributed knowledge. We have found that, large scale team members practices technical forum techniques to share knowledge among remote team members.

6) Electronic board: The office boards hold lot of knowledge which is difficult to share among distributed teams. From the interviews it is revealed, practitioners are using electronic boards to share and access knowledge both locally and globally. Electronic boards hold the tasks lists to perform, latest information and along with that necessary technical and business information are regularly updated in wiki. Electronic boards help to decrease communication overhead.

In one case, interviewee said “...I am not satisfied with the current tools; It's tough to describe design to new team members. Visual aid is helpful during the discussion” -Project Manager, Beta project.

7) Rotation/Visits: The primary intension of team members rotation between different sites is knowledge sharing. Due to frequent face-to-face interaction with product owner, on-site team members gets more business and domain related information than offshore team members [56]. Lacking face-to-face meetings and poor socialization also cause of lacking trust among distributed team members [57]. Rotation between on-site and distributed team members promotes to share business and domain related knowledge across the teams. From the gathered data we have found that, both distributed and hybrid team visit remote sites and rotate team members to increase the trust and communication bandwidth between team members. But, studied dispersed teams never visit and rotate remote team members.

One of the distributed teams Line manager said “Visit on remote sites is highly cost. So, we rotate team members and mostly, duration of the rotation between team members is 3-6 months.” -Line Manager, Zeta project

We have also found, teams are practices Version control, Screen sharing and Discussion forum to maintain knowledge sharing among both local and remote team members.

C. Knowledge Sharing Strategies

According to the literature, there are two types of knowledge management strategies: codification and personalization. Codification strategy encourages people to create documents by using repositories. On the other hand, personalization encourages people to share knowledge through conversation or social interaction. Nonaka’s model [31] represent, personalization helps to facilitate tacit to tacit knowledge sharing through socialization. Table VII represent types of knowledge management strategies applied in the different projects to manage knowledge both locally and globally.

<table>
<thead>
<tr>
<th>Projects</th>
<th>Locally</th>
<th>—</th>
<th>—</th>
<th>Globally</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tacit</td>
<td>Explicit</td>
<td>Tacit</td>
<td>Explicit</td>
<td></td>
</tr>
<tr>
<td>Alpha</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Beta</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Gamma</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Delta</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Epsilon</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Zeta</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Eta</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

[+/-] In practice, [-/+] Not in practice

According to the seven cases evidence (see in Table VII), there are two types of knowledge that reside in the office: tacit and explicit. In the agile software development most of the knowledge is tacit which reside in the human mind rather documentation. These codified tacit knowledge shared among between locally and globally distributed team members through tools. Knowledge sharing approach varies between team members due to experience level. This types of problem leads search availability and hard to find right knowledge at the right time. We have also found, to share tacit knowledge between remote team members, teams maintain common chat room and online conference (see in Table VI). In one project (B), we have found, team does not share tacit knowledge among dispersed team members. But, based on the situation, sometimes team perform pair programming through screen sharing to resolve problems.

D. Challenges faced by practitioners during knowledge sharing among distributed teams

Challenges faced by the practitioners during knowledge sharing among distributed team members are shown in Figure 15. Mitigation techniques applied by the practitioners also shown in the same Figure 15.
Fig. 15: Knowledge sharing challenges and mitigation techniques

In Figure 15, arrows indicate the mitigation techniques applied by the practitioners for a specific challenge. Based on the severity Communication, Language and Cultural challenges frequently faced by practitioners during knowledge sharing in distributed agile project (see in Figure 15). In distance teams are also struggling with Misunderstanding and Visualization challenges.

Fig. 16: Success of knowledge sharing

Though teams are facing different types of challenges during knowledge sharing among distributed team members but we identified the success of knowledge sharing in both locally and globally distributed agile teams from studies seven cases. Based on the seven cases the above graph (see Figure 16) is drawn. Ordinal scale is used to map the interviewee satisfaction with theirs KM activities in both local and global teams. Figure 16 depicts that, in project (Alpha) interviewees think knowledge sharing activities are more succeed among distributed team members than in the local team due to language barriers (non-native English speaker) between local team members. It is also evident from the gather data that, dispersed feature teams more success about theirs knowledge sharing among distributed team members rather than distributed and hybrid teams.

VII. DISCUSSION

A. Codification of Knowledge

According to Polanyi [58], “Individuals know more than they can say”. Polanyi classified human knowledge into two categories. Tacit knowledge, that is very difficult to describe or express and this type of knowledge is transferred through demonstration. Tacit knowledge has an importance of cognitive dimension which consists of mental model, beliefs and perspectives [31, 59, 60]. So it cannot be easily characterized by clear expressive language. Explicit knowledge, which is easily written down and codified. Explicit knowledge is possible to easily characterize by textual or symbolic form. This kind of knowledge resides in textbook, memos and technical documents. Codification of knowledge is the conversion of tacit knowledge into explicit knowledge in a written, verbal or visual format. The extraction process of tacit knowledge into explicit is called externalization. Tacit knowledge cannot be interpreted fully even by an expert [11]. This type of knowledge is more deeply placed in action and stiff to express in word [61]. Nelson et al. [62] conclude, it is impossible to describe all necessary aspects of organizational tacit knowledge for successful performance. In the organization, most of the tacit knowledge is work related, that is learned informally during the team works [63]. Codification is challenging to extract tacit knowledge into explicit; so an expert needs to understand essence of the tacit knowledge to increase degree of explicitness of knowledge. Surprisingly, we found from our results that, all studied cases are concern about knowledge codification. To codify tacit knowledge, teams are using wiki, JIRA, confluence etc. In local sites, technical presentation and discussion forum are also taken into account as knowledge codification strategies. Later, teams share codified knowledge among remote team members through repositories and that is helpful for the remote team members to reuse codified stored knowledge.

B. Knowledge Management Strategies in Practices

We found that, knowledge management schools are in practice according to the results in Table V and VI. We used Earl’s [30] framework to select type of strategies or “schools” or practices in the different projects that applied to manage knowledge locally and globally.

Based on the evidence from different cases, we found that knowledge management schools were in use to manage knowledge both locally and globally. It is also evident that Systems, Cartographic, Engineering, Organizational and Spatial schools are practices in distributed agile projects to manage
knowledge both locally and globally (see in Table VIII). Both Commercial and Strategic schools are focused on business (patent, copyright, trademark, know-how and intellectual asset [30]) perspective and there is also no evidence found within gathered data sets that indicates those schools (Commercial and Strategic) were in practice. For that reason those schools are not taken into account in this research.

1) **Systems school:** This school’s philosophy is to codifying knowledge with the help of technology. Organizations use repositories for storing and sharing knowledge. These knowledge repositories usually store domain specific information. The codification of knowledge can be compared with externalization of knowledge by Nonaka [32]. It is easy to realize the benefits of knowledge bases and system school is the most researched school [28]. These knowledge bases become richer and more useful overtime. As shown in Table VIII, System school is in practice in all cases to manage knowledge locally and globally. Though search functions is a difficult issue in system school, practitioners depends on that because in distance this school effectively perform knowledge sharing activities using repositories.

2) **Cartographic school:** This school focuses on mapping of organizational knowledge and aims to build knowledge directories by disclosing “Who knows what” [30]. This is sometimes achieved by “yellow-pages”, which ensures the accessibility of knowledgeable person within the organization to others for knowledge exchange. Though knowledge maps and directories on company intranets might be helpful for distributed team members to have idea about who knows what, but in distributed project it seems challenging to put into practice. Because it needs joint effort and commitment from both local and remote team members. In collocation, it seem easier to find out knowledgeable or experienced person because they knew each other well. In globally distributed project “who know what” and “what is where” are important issues for effective knowledge exchange. We have found that, Cartographic school is practiced by the different projects (δ,ζ,η) to exchange knowledge both locally and globally. This strategy is also practiced in different companies by introducing knowledge brokers idea and that helps other developers to consult with knowledgeable and experienced software engineers [1]. Knowledge brokers are knowledgeable and experienced software engineers who will communicate with other developers with information or will listen to them.

3) **Engineering school:** This school of knowledge management focuses on business process re-engineering [28] and knowledge flows in organizations. This school has more empirical attention than other schools, that focuses on managing knowledge about software development processes and improvement of software development processes. More specifically, this school focuses on formal routines, mapping of knowledge flows, project reviews, and social interactions. Software process improvements like CMMI can be regarded as stimulus for knowledge flow throughout the organization. This school supports explicit knowledge sharing and in distributed project, temporal distance does not affect this school. In globally distributed project coordination is one the major challenges and engineering school focuses on coordination process, that aims to ensure knowledge flow within organization though shared databases. The process of using tools (i.e installation manual of GitHub or SVN with eclipse), quality code writing techniques, testing and reviews are documented in repositories to share among distributed teams. However, practice of this school is found from the studied projects.

4) **Organizational school:** The philosophy of Organizational school is to create network by collaborating between communities to share or pool knowledge. This school of knowledge management focuses on the organizational structure. These structures are often referred as “knowledge communities” [28]. This is a networking approach for people to communicate and share knowledge. Based on the seven cases, this school is in practice for knowledge sharing both locally and globally.

5) **Spatial school:** The intention of spatial school is to encourage socialization (tacit to tacit knowledge) as a means of knowledge exchange [64]. Spatial school is more concerned with the development and utilization of the social capital which develops from people interaction, formal or informal, repeatedly overtime [30,64,65]. Spatial school focuses on designing office space to promote knowledge sharing [23,32]. Organization uses different office settings to promote people to communicate. For example in the case of software organizations agile methodologies may use boards, charts or other
tools to create spatial knowledge. Even sometimes common spaces like conference rooms, dining rooms or place for refreshment activities are also place where knowledge can be shared. Five cases were found those practice Spatial school to manage knowledge locally and only case (Hybrid team) was found that practices Spatial school to manage knowledge both locally and globally. That Hybrid team is using visual board to communicate among collocated and distributed team members.

VIII. Conclusion and Future Work

The aim of this research was to discover the knowledge sharing techniques, strategies applied and challenges faced by the practitioners in distributed agile projects. This research has addressed the following research questions. (1) How do team members contribute to knowledge creation in a distributed agile project? (2) How do team members share knowledge in a distributed agile project? (3) What are the challenges faced by the practitioners when sharing knowledge in a distributed agile project?

To perform knowledge management activities in a distributed agile projects, different teams practice different types of approaches. But, in general, we found different types of knowledge creation and sharing techniques applied by the practitioners to perform knowledge management activities in distributed agile projects. Along with, we also found different types of strategies practiced by the team members to manage knowledge both locally and globally.

For the first research question, we found that:

- To perform shared knowledge creation in a distributed agile project, team members practice pair programming, customer collaboration, Scrum/Kanban boards, innovation boards, workshops/seminars, learning, technical presentation and technical discussion techniques.
- In globally distributed agile projects, teams practice different types of strategies to perform shared knowledge creation such as systems, engineering, organizational and cartographic schools. Though we have observed that, spatial school is in practice of local knowledge creation but while project is distributed, this school has less concern due to expensive tools.

For the second research question, we found that:

- To share knowledge among distributed sites, team members, practices different type of techniques: repositories, pair programming, version control, screen sharing, daily scrum, weekly sprint status, common chat room, technical forum, discussion forum, electronic boards, online conference, rotation/visit etc.
- Systems, Engineering and Organizational schools strategies are explicitly in practice to share knowledge among distributed team members. Those strategies foster team members for effective knowledge sharing activities in distributed agile projects. In distributed development, “who know what” and “what is where” is important to be known by employees for effective knowledge sharing and that is associated with cartographic school. But, in distributed agile projects, this school has less concern due to social-cultural distance. Spatial school facilitate knowledge sharing by using office space but in distributed agile projects this strategy is not explicitly in practice to share knowledge among remote team members.

For the third research question, we found that:

- During knowledge sharing among distributed team members, practitioners faced different types of challenges, such as, language, communication, misunderstanding, visualization, cultural, technological, time zone difference and lack of information.
- To mitigate those challenges, practitioners also apply different types of mitigation techniques, such as, informal communication, cultural exchange, common platform, tools, visual prototyping, common chat room, rotation and overlapping hours.

Through a series of semi-structured interviews from Agile practitioners, we investigate knowledge sharing activities in distributed agile projects. Communication, coordination and collaboration are the keys to foster knowledge sharing between team members in Agile software development. However, we have seen knowledge sharing in distributed agile projects is challenging due to factors like communication difficulties, language barriers and cultural barriers. To mitigate those challenges and succeed in knowledge sharing within and across the border, practitioners adopt different types of techniques to manage knowledge both locally and globally. Along those techniques, we have also noticed that, practitioners adopt different types of strategies to manage knowledge both locally and globally. Though systems, engineering and organizational schools are explicitly in practice, spatial school have less concern to manage knowledge in distributed agile project. With closer observation between software engineering and schools Bjørnson and Dingsøyr found, there are heavy focus on systems and engineering schools [28]. There are also limited number of study focusing on organizational school but no study was found in software engineering that focuses on spatial aspect [28]. Agile is more closely connected to the socialization that also includes spatial schools concept as knowledge sharing strategies. There are lots of knowledge resides in the office space and office space foster knowledge sharing through spatial knowledge management strategy. In future, it will be interesting to find out practices of spatial school in distributed agile projects.

IX. Appendix

Introductory Questions:

- What is your role in the organization?
- What are your responsibilities in the current project?
- How long have you been working with the organization?
In-depth Questions:

- How long have you been working with globally distributed projects?
- Team Size and number of developing sites? Who does what? (Distributed, Dispersed or Combinational)

- Which software development methodologies do you follow?
- Which agile practices are followed locally and globally? (e.g. Scrum, XP, Kanban, AUP etc.)
- Which types of knowledge do you share among local team members?
- How do you perform shared knowledge creation among distributed team members?
- How do you share knowledge across sites?
- What problems/challenges did you face while knowledge creation and sharing?
- What have been done to avoid those problems?
- How do you evaluate the success of knowledge sharing among team members in the same sites?

Scale 1-10

- How do you evaluate the success of knowledge sharing among distributed teams?

Scale 1-10

- Do you think the scope of shared knowledge is sufficient? Why or Why not?
- Do you believe that knowledge sharing activities have positive impact on the team? Can you please describe?
- Do you like to add something, any success or problem scenario or anything?

REFERENCES


Paper 2

Mohammad Abdur Razzak and Rajib Ahmed. *Spatial Knowledge Creation and Sharing Activities in Distributed Agile Project.*
Spatial Knowledge Creation and Sharing Activities in a Distributed Agile Project

Mohammad Abdur Razzaq  
School of Computing  
Blekinge Institute of Technology  
Karlskrona, Sweden  
Email: morc10@student.bth.se

Rajib Ahmed  
School of Computing  
Blekinge Institute of Technology  
Karlskrona, Sweden  
Email: raae10@student.bth.se

Abstract—Knowledge management (KM) is key to success for any software organization. KM in software development has been the center of attention for researchers due to its potential to improve productivity. However, the knowledge is not only stored in repositories but is also shared in the office space. Agile software development teams use the benefits of shared space to foster knowledge creation. But it is difficult to create and share this type of knowledge, when team members are distributed. This participatory single-case study indicates that, distributed team members heavily rely on knowledge codification and application of tools for knowledge sharing. We have found that, studied project did not use any specific software or hardware that would enable spatial knowledge creation and sharing. Therefore not codified parts of the knowledge items were destined to be unavailable for remote team members.

Index Terms—Knowledge management, knowledge sharing, KM schools, spatial school, agile.

I. INTRODUCTION

Software development is dependent on Knowledge management (KM), since it is a knowledge intensive job. This enforces software organizations to manage their knowledge and later use it in smarter innovative ways to solve problems [1]. It helps software development organizations to acquire and maintain competitive advantage. KM is crucial for success in global software development [2]. Global software development is “software work which is attempted or engage in different geographical location across the national boundaries in a coordinated fashion to involve synchronous and asynchronous interaction” [3]. In the globally distributed agile project, team members share project-specific knowledge through frequent face-to-face interaction, effective communication and customer collaboration [4]. In agile software development, collaboration and coordination depended on communication, which is central for successful software development [5]. Software development depends on the developer’s knowledge and experience [6]. So, success of agile projects relies on effective knowledge sharing among teams. To foster dynamic knowledge sharing, improve productivity and coordination in software teams, agile approaches were introduced. In traditional software development, knowledge is stored explicitly in documentation, but in agile development, knowledge is tacit which is in human mind [7]. So, converting tacit knowledge to explicit knowledge is one of the greatest challenges of knowledge management [8]. Due to the absence of explicit documentation in agile software development, experts need to spend much time in repeatedly answering the same questions, knowledge is lost when experienced developers leave project. This is cause informal communication sometimes cannot serve as recorded documents, less support for reusability and less contribution in organizational knowledge [7].

According to Earl’s framework [10], the philosophy of spatial school is Contactivity that means contact and activities. The intention of spatial school is to encourage socialization as means for knowledge exchange [9]. This school also can be labeled as the social school, because its’ intention is to encourage socialization for exchanging knowledge using office space. In agile development paradigm, social interaction is the major artifact to share knowledge in an effective way. Different literature claims, knowledge management activities in traditional teams fail due to pure codification strategy. Spatial school is more concerned with the development and utilization of the social capital which is developed from interaction between employees both formal or informal and repeatedly overtime [9–11]. There usually are lot of formal or informal discussions around white boards, taskboards, innovation boards, in front of coffee machine and so forth which create spatial knowledge. And in distributed project this knowledge can be helpful for the remote team members. Spatial school focuses on designing office space to promote knowledge sharing [12,13]. However, there is little empirical evidence about creating and sharing spatial knowledge with different KM activities in globally distributed agile team. The target of this research is to discover KM activities in distributed agile project focusing on spatial school.

The rest of the paper is organized as follows. In section II, theoretical background for spatial knowledge creation and sharing activities in distributed agile projects. In section III, we have presented design of this research along overview of the research subject and validity threats. In Section IV, we have presented results of different finding. Discussion of empirical study is provided in section V. Finally, section VI concludes...
the paper with the summary of major findings and future work.

II. BACKGROUND

Knowledge is context dependent asset and it depends on both time and space [14]. Knowledge is an individuals’ interpretation of some learning from social, cultural or historical context. Sharing this knowledge is not an easy thing to do. Getting people to talk and share their know-how is one of the biggest challenges to effective knowledge sharing between team members [15]. In the agile software development social interaction is an important issue to share knowledge in effective way. This type of social interaction happens between team members through tacit knowledge sharing. But in organizations it is tough to find out the right knowledge at the right time; due to knowledge fragmentation, overload and de-contextualization [16]. So, it is important for the globally distributed organization to ensure and preserve knowledge management strategy for both tacit and explicit knowledge [16].

Tacit knowledge is a type of knowledge that we acquire from our learning and personal experience; so tacit knowledge can be gained from discussion between individual; more specifically, the personalization strategy focuses on discussion between individual; more specifically, this strategy relies on person to person interaction [21]. In this case, knowledge seekers do not need to search in the documentation because this type of knowledge heavily relies on experts. But in the distributed settings it might be difficult to understand tacit knowledge due to language barriers, cultural factors and terms used.

Knowledge management is an interdisciplinary field. To unify the work in the field, Michael Earl [10] proposed a classification of knowledge management practices by organizations into different schools. This classification helps organizations and researchers to have vocabularies and understanding of knowledge management. Michael Earl [10] has classified knowledge management (KM) into three broad categories: technocratic, economic, and behavioral. These are then divided into seven schools: Systems, Cartographic and Engineering (Technocratic), Commercial (Economic) and Organizational, Spatial and Strategies (Behavioral). The schools are not competitive and their practices are not mutually exclusive [12]. To understand spatial school of KM we also need to understand it’s relation to Systems and Organizational schools. Systems school’s philosophy is to codify knowledge with the help of technology. Organizations attempt to use knowledge repositories for storing knowledge and sharing knowledge. These knowledge repositories usually store domain specific information that then can be used for decision making. The codification of knowledge can be compared with externalization of knowledge by Nonaka [12]. It is easy to realize the benefits of knowledge bases. That is why system school is the most researched school [22]. Spatial school focuses on the usage of organizations space to promote spontaneous discussion, organizations use different office settings to promote this. For example, in the case of software organizations agile methodologies may use boards, charts or other tools to encourage discussion which creates spatial knowledge. Even sometimes common spaces like conference rooms, dining rooms or place for refreshment activities also are places where knowledge can be created and shared from social interaction. Organizational school focuses on the organizational structure. These structures are often referred as “knowledge communities” [22]. This is a networking approach for people to communicate and share knowledge. The main aim is to collaborate and create knowledge for the organization.

KM is an important asset for software development like in any other organizations. Bjørnson and Dingsøyr [23] mentioned compared to the traditional companies with their KM approaches agile companies seem to be more satisfied. The authors also enclosed on this study that medium sized companies are getting more benefit with their KM approaches rather than small sized companies. Agile teams are more related with the spatial school which facilitates KM activities through using office space and knowledge exchange within and across the teams. In the same study, Bjørnson and Dingsøyr mentioned, agile companies may benefit from the behavioral schools. In general, there is a great deal of

1wiki’s, redmine, JIRA etc.
knowledge floating in the office space from social interactions. Due to lack of concern and explicate process, companies are losing this knowledge. So, it is important to identify, how practitioners are compensating the space aspect of knowledge management while teams are in a distributed environment. But very little empirical evidence identified on how distributed agile team create, share and access spatial knowledge within and globally distributed team members. This research is a step towards finding usage of space to create and share spatial knowledge in distributed agile teams.

III. RESEARCH METHODOLOGY

This research focuses on exploring office space settings which foster knowledge creation and sharing activities in a distributed agile project. More specifically, this research attempts to identify types of strategies applied to create and share knowledge by using office space between both local and distributed Agile teams members. Thus, we are driven by the following research questions:

- **RQ1**: What spatial knowledge creation strategies are practiced locally in a distributed agile team?
- **RQ2**: What are the approaches agile team practices to share spatial knowledge with remote team members?

This case study had focuses on addressing both **RQ1** and **RQ2** by developing insights into understanding of spatial knowledge sharing activities in distributed agile project. This research entail as an exploratory single-case study. Exploratory case study is best suited for situation where in-depth and detail studies are unavailable. Case study research works with multiple variables in social context [24]. This provide researchers with rich descriptive data and can identify patterns based on the results. Case studies can be used for both qualitative and quantitative research. But it is uncommon to find case studies with quantitative or statistical methods. There are two types of observation such as **participant**: being involved and **structured**: watching from “outside” during case study [25]. So, to observe the flow of spatial knowledge in the selected company participatory case study was performed. Participatory research is a method where researchers take part in the daily routines, events, rituals and interactions, which provides a deeper understanding [25].

### A. The Case

This section describes, details about the subject of this case study that were conducted during the period June, 2012. The investigated company we named as *Alpha* for anonymity. An overview of *Alpha* is given in Table I. Alpha was established in 2001 as an internet startup company. Briefly, the goal of the *alpha* was to provide consumers with unbiased information of any kind of consumer products. Now in year 2012, it has grown to be the second largest software product in the world among its competitor. It operates with four different offices from Sweden, Germany, United States and China. *Alpha* is medium sized having 80-100 employee distributed in different countries. Though *alpha* operates worldwide, the software product is developed in two locations. Most of the software development takes place in China and some features are developed in Sweden. There are three different teams in Sweden site are working on three different department of development. These teams are i) Data-flow, ii) Data analysis and iii) Application development. This case study is based on the *application development* team. Currently, application development team is involved with Chinese team in one collaborative project. Though, *alpha* adopt Kanban to coordinate different departments in Sweden sites, but *application development* team uses scrum method to coordinate between globally distributed developers in China. The team only practices daily *Scrum* from scrum method.

In this collocated environment there are three teams working together consisting of three team leaders and five developers. Total eight developer in Sweden site working in colocation. All three teams have remote members in China. Three teams have common product owner for communication and clarification. From the Table I there are eight team members in Sweden and twelve team members in China.

As part of this case study we observed only the application development team for one week. This team is responsible for developing and maintaining the web service. The team consists of seven developers and a team leader. This team is globally distributed as five of the developers are located in China and other three members are located in Sweden. Responsibilities of team leader are coordinating team, task distribution, testing, communicating with clients for service integration and communicating with managers involved as product owners in the project. Responsibilities of developers are developing and maintaining features in test driven way.
TABLE II: Research activities

<table>
<thead>
<tr>
<th>Activities</th>
<th>Participants</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observation</td>
<td>Swedish team- One Week</td>
<td>Focuses on daily activities of a Agile team.</td>
</tr>
<tr>
<td>Interviews- 3</td>
<td>Project lead and developer- Swedish team</td>
<td>Access of shared knowledge, missing items and challenges during collaborative development.</td>
</tr>
<tr>
<td></td>
<td>Developer- Chinese team</td>
<td></td>
</tr>
</tbody>
</table>

The team has experience of programming with web related technology ranging from two to five years.

Application development team has been operating as a dispersed team more than three years. Figure 1 shows team distribution in Alpha. Both yellow and green colored teams are part of dispersed application development team. Yellow color indicates that, participatory observational case study held in Sweden sites. This team uses tailored Scrum as their development methodology. They do not specify sprints for this software service. So, it requires frequent communication between developers in Sweden and China.

B. Data Collection

To address the both research questions, we performed participatory observational case study over a week period of time. Then to get in-depth evidence, we also performed semi-structured interviews. Overview of this single-case study activities are given in Table II.

Yin [24] represents six sources of data in case studies: documents, archived record, structured interviews, direct observation, participatory observation and artifacts. Firstly, both researchers performed participatory observation to collect and acquire multi-faced data from daily development activities with development team. This Participatory observation is a common data collection method used in office setting [26]. Participatory observation gave us access to events, processes and physical artifacts. Both researchers maintained dairy to note down daily observational activities that helped later to link between different findings. We also collected data from documents which are shared among distributed agile team using Google Documents. And analyzed issue tracking tool used by the company. Along that, we also examined physical space, documents, tools and physical artifacts used by the agile team. During the case study, one of the researchers participated on daily software development activities with the application development team. That gave opportunity to the researcher for continuous communication with remote team members.

Secondly, we have then interviewed three team members from the agile team for further triangulating the derived data. Two interviews were held face to face with collocated project lead and one developer, moderated by both researchers. One team member from China was interviewed in a video conference via Skype. The selection criterion of those interviewee based on theirs experience in distributed agile project, their role in the distributed agile project as well as in the company. Semi-structured interviews were conducted to find out what is happening and giving us new insights [27]. Data collected from the interviews were used for triangulation while comparing or contrasting them with data collected during participatory observations. These semi-structured interviews were combination of both open and focused questions. It helps both interviewer and interviewee to discuss on topic in more details. The reason of choosing semi-structured interview is to prompt and probe deeper into the situation. It also helps the interviewer to get information from individual about their own practices, believe, and opinion which included both past and present experiences. The interview questions were descriptive and with the base questions there were follow up questions asked based on the discussion. Before interview we started the researcher, we gave overall goal of this research to interviewee. These interviews’ duration were on average 60 minutes and recorded the interview session.

C. Data Analysis

We had different data sources feeding inputs to this research. Organized data helps reader to understand the context of research [28]. The result of each interviews’ were documented and report sent to interviewee to check whether interviews data correctly transcribed. Data collected from different sources were analyzed with the help of qualitative data analysis technique called thematic analysis technique [29]. The recorded interviews. data were revisited multiple times and that was triangulated with the observation data during the analysis phase. Both researchers involved during analysis phase and performed six phases thematic qualitative data analysis. That helped both researchers to relate data sets among research theme and research questions. Later, we used mind mapping tools to visualize the data patterns. The data presented in this research for both research questions are based on the dairy, documents, overall observation and interviews. That interviews data helped us for further confirm of our observation.
D. Limitation and Validity of Threats

This research entails as exploratory case study, thus it has some limitations. There are two types of validity (in addition to Reliability) that have to be considered, namely construct validity and external validity [24]. Construct validity involves creating correct operational measures for the concept that are measured in the case study. Multiple sources of evidence are collected by both researchers during data analysis phase. Later, those multiple sources of data helped for triangulation within the data sets. During the interview sessions researchers applied chronological order in the discussion to maintain the chain of evidence. Researchers sent the case study draft report to key informant (the interviewee) in order to check whether case study report is the correct representation. In addition to interviews, some follow-up questions were asked on different issues, maps and charts of the geographical characteristic or layout of office space. The goal of reliability is to minimize the errors and biases in the study. Furthermore, a case study protocol were sent to all interviewees before interviews actually being conducted.

The external validity needs to be obtained which helps to refer, “the domain to which the findings of the case study can be generalized”. The selection criteria played an important role in the creation of external validity [30]. And this is one of the limitations of this exploratory single-case study. We do not make any generalized claim and our findings might not be universally applicable to all distributed agile projects but not make any generalized claim and our findings might not be generalized”. The selection criteria played an important role in the creation of external validity [30]. And this is one of the limitations of this exploratory single-case study. We do not make any generalized claim and our findings might not be universally applicable to all distributed agile projects but

IV. Results

In this section we outlined findings from the observation and in-depth semi-structured interviews that were collected through a single-case study. An overview of different findings are listed in Table III. Table III, is a brief representation of evidences found from our case study.

<table>
<thead>
<tr>
<th>Dimensions</th>
<th>Findings</th>
</tr>
</thead>
<tbody>
<tr>
<td>KM Strategies in practice</td>
<td>Locally: Systems; Engineering; Spatial Organizational</td>
</tr>
<tr>
<td>Local knowledge creation</td>
<td>Whiteboard; Daily Startup meeting; Weekly workshop</td>
</tr>
<tr>
<td>Knowledge sharing tools</td>
<td>Redmine; Google docs; Google Calendar</td>
</tr>
<tr>
<td>Challenges</td>
<td>Technological; Language; Cultural; misunderstanding and visualization</td>
</tr>
</tbody>
</table>

Everyday in Swedish office, there is a standup meeting called “incident meeting”. All collocated developers report any problems in the software service which are interrupting the standard flow of operation. There is an incident board where all these incidents are listed with sticky notes. Incident board usually holds issues which need to be addressed immediately i.e. database servers are performing poorly. These incidents are then assigned to teams through this meeting. This meeting allows everyone in the office to be informed about the most recent issues about the software.

The application development team communicates with remote team members using instant messaging tool, Skype. Everyday there is a daily scrum meeting in Skype where team leader distribute tasks and shares knowledge with brief discussion. This informal conversation and discussions are automatically stored as chat history. Brief summary of discussion or plans may be explicitly stored as online documents. All the tasks and their related information are stored in a central repository called Redmine. Each task on this repository holds written discussions, limitations, status and information about people involved with it. Some of these tasks are also added to kanban board with colored sticky notes. Because, the Swedish office use Kanban as their main software development process to coordinate collocated software developers.

The daily development starts with a standup meeting after the incident meeting. This meeting bridges development team and managers in Sweden. Beside these boards, there are three conference rooms in the office. Every week there is a Monday meeting with all offices connecting with conference call. In this meeting all employees from all four operating countries gets to share their weekly achievements. And also the owner and managers shares the insights of company growth and achievements. This allows transparent knowledge sharing across sites. Second weekly meeting is held for knowledge sharing. This is a two hours long meeting were anyone can present about new technologies or ideas. Presentation topic for this meeting are decided week before the meeting by voting. This meeting is only held on Sweden office. But the presentation slides are stored and shared using online tools.

A. Spatial Knowledge Items

There are total four boards in the office, two teams have their own agile board, an common incident board and a common Kanban board. As Kanban is the main development method connecting three teams, the Kanban board can be considered as the most important physical artifact. The application development team uses incident and kanban board.

In office space, knowledge is created around agile boards and sometimes in white boards of conference room. In figure 2 shows, teams in Sweden office use these boards to provide brief overview of tasks with colored notes. Each note have tracking number, short title and responsible person name. These notes have detailed information stored in the issue tracking system. Chinese developers do not have boards on their office space, they rely on the issue tracking system for acquiring knowledge. There is a weekly technology workshop held in Sweden office. This technology presentation duration
is two hours. Developers share knowledge about new technologies, technologies used for developing the software service and improvement possibilities etc. Sometime they write notes backboards to explain the presentation. This knowledge management activity is partly shared with other office by sharing the presentation slides. But discussions and writings from these presentations are not documented. Therefore, knowledge is lost due to lack of explicit process. There is a bookshelf with few technology books in Sweden office. There is also a digital library for same books for other offices to access. Figure 3, shows participation of two sites to create shared and local spatial knowledge.

Though, the office uses traditional boards to visualize development activities. For persevering company wide knowledge it uses internet based software. For storing and distributing knowledge among China and other offices it uses Redmine issue tracking software. All the features, bugs and support requests are stored to this central repository with detailed information. Then Kanban board is synchronized manually from the digital repository with minimal information.

In the organization, spatial knowledge is mostly created by the developers from their intercations infront of agile boards or sitting next to each other. Also the interactions between management and developers create spatial knowledge. But this knowledge is hierarchical, it is propagated to the developers from their team leaders.

The importance of the visual knowledge can be realized as an snapshot of what is happening throughout the company and what is important for the company. This visual knowledge is easy to personalize rather having to go through vast digital knowledge repository. When this knowledge is codified, it looses some of attributes which makes it less attractive. Also, knowledge management in agile team is not document driven. This spatial school or social school of knowledge plays a major role in being agile or lightweight.

B. Scope of shared spatial knowledge

One of the main objectives of this case study and several interviews was to obtain information about spatial knowledge management. It is evident from our case study that, there is less scope of sharing knowledge created in physical space. The primary tools for the organization to explicitly share knowledge is the issue tracking system and online document sharing service. The physical Kanban board is a reflection of issue tracking system. From the figure 2, we can see that agile team in Sweden uses the boards but boards hold the brief redundant information which is already stored in issue tracking system. The company recently moved to online based document management from communicating knowledge through emails. The developers in Sweden use Skype for communicating distributed developers and among themselves, because they are not closely seated. Figure 3, outlines the sources of knowledge in distributed agile team. Sweden and China office use internet based tools for sharing knowledge which is represented as the green area. But we found the Sweden office is using office space to create knowledge more than the China office. The spatial knowledge created by developers in Sweden is partially shared with developers in China.

C. Success of knowledge sharing

The level of externalization and internalization are not same in between teams. So, frequently misconceptions happen if the shared knowledge is not properly codified. Agile distributed teams highly faced visualization problem. In the distance it is hard to explain diagram which lead to misunderstanding. So, visual aid is helpful to demonstrate the diagrams and other related issues.
From figure 4, we see developers from both office highly satisfied with knowledge management when developers are in collocation. This is related to ease of communication, trust, sharing space and cultural similarities. But knowledge management in distributed setting shows lesser satisfaction then collocation due to lower bandwidth of communication between developers, lack of codified knowledge, codified knowledge lacks in contextual information. Project manager in Sweden has different opinion about local knowledge management, we observed this is related to communication problem with newly employed collocated developers. It is difficult to build an idea from scratch with conference calls. It takes more than usual time to explain the idea and then other developers to contribute discretely. It would have been easier if all the developers share same room and boards. Then creating spatial knowledge would be easier. *Ordinal scale* used to maps the interviewee satisfaction about theirs KM activities between both local and global teams.

From the figure 4, we can observe that two of the interviewees are more satisfied with their knowledge sharing with local team members. One interviewee was less satisfied with knowledge sharing with local team member and more satisfied remote team members. For explaining this difference we identified it is due to the fact interviewee is working with remote team member for longer time than the colocated team member. Also language barrier played part for his dissatisfaction. Because three colocated developer are from different countries and not native english speakers. One developer said “...Knowledge difference between developers can be frustrating. As we share knowledge few developers misunderstands or do not understand sometimes. This is same for local and remote team. Also there is not enough “organized” explicit knowledge stored in this organization, its really difficult come up to pace.”

In one KM meeting one developer said “... I would like to know more about mocking and stubbing for testing.”. In response to this another developer started explaining it on the boards. This knowledge transfer was one time and only available for collocated developers in Sweden.

We observed some knowledge transfer using space is situational. Codification of this knowledge in that point of time is not possible. Also, text chatting do not carry the expression of communicator so some unspoken gestures are missed. Cultural difference and language barrier makes it difficult to communicate. So it also affects both spatial knowledge creation and sharing.

**V. DISCUSSION**

**A. Visualization of spatial knowledge**

Visualization can be helpful during internalization and externalization process. Based on both observation and interviews, we argue, distributed agile team always faces visualization problem due to temporal and geographic dispersion between teams which also leads to decrease in project productivity. Visual aid can always be helpful for distributed team members to get the information at the right time that may increase the team members’ productivity. Sharing visual board with the distributed team members can build trust and increased the confidence. Also organization can use projectors to cross-project meetings. This can help to share spatial knowledge in distributed agile teams to a maximum. We suggest the team to use visualization tool to share kanban board and use skype to try pair programming. These techniques will clear a lot of misunderstanding, give a visual feedback of progress, thus cutting down communication time. It is also important to have presentation that includes remote team members. If real time video presentation is not possible then this can be achived by recording video of presentation and then making it available to remote team members. This way the social intercations are fully documented for later use.

**B. Knowledge sharing strategies in practice**

Though the concern of this study is spatial knowledge creation and sharing in distributed agile project. During the case study we have observed that, studied team are using different types of KM practices that are associated with the systems, engineering, organizational and spatial school to foster spatial knowledge sharing both locally and globally. It is evident from our observation that, colocated part of studied team use office space for knowledge creation from there social interactions. But not all spatial knowledge are codified and shared with remote team members i.e. *developers explaining ideas on a white board in collocation*. We believe that, all spatial knowledge is not possible to be codified and shared among distributed team members. Interestingly, we have observed that, team do share some spatial knowledge among distributed team members through repository i.e. *replica of Scrum board through repository*. Therefore, we can say that there is a strong correlation between Systems, Organizational and Spatial school in distributed agile projects. All codified spatial knowledge is shared using communication tools with remote members, which then rely on community of practice. *Spatial school act to foster knowledge exchange by using...*
offices with distributed agile teams. From the KM perspective, practitioners should be more aware about spatial knowledge during the distributed agile environment.

C. Team building

We argue, team members in agile software development have to have a sense of shared ownership and responsibility in order to achieve goals. Building a cross functional team itself is a challenge when it comes to distributed agile teams; it is even more difficult. For knowledge to flow among the team members efficiently, teamness is essential. Knowledge creation and sharing also depends on the trust among team members. Thomas et al. [31] mentioned, software development is a social process and it is important to develop both organizational and individual trust within and across the teams. Trust on other developers also facilitates reusability of code and lead to more efficient knowledge creation and sharing [31]. Studied distributed team does not share all office space with each other, to create spatial knowledge, we can identify team building is an integral part.

VI. CONCLUSION AND FUTURE RESEARCH

In this study, we have investigated spatial knowledge creation and sharing activities in a distributed agile project, through a single-case study.

In response to RQ1 we found that, studied agile team adopted different types of knowledge creation strategies using local office space. Studied team uses kanban, incident and white boards to initiate discussions which then creates spatial knowledge in collocation. There is also one to one discussions and technical presentations that creates spatial knowledge too. Spatial knowledge is social knowledge, creation of this knowledge is not limited to these strategies only. But these are the most common artifacts that ignites spatial knowledge creation for agile teams in collocation.

In response to RQ2 we found, studied team does not directly create any spatial knowledge with remote team by using any visualization tools. But, team creates and shares most of the spatial knowledge items through repository and instant messaging. But knowledge which is created across office space from the interactions of collocated team members can partially be codified and stored in this repository. Because of some perception and discussion surrounding physical boards or other objects are not codified. So, from this case study we found that organization are more focused on usage of internet based tools rather than investing on expensive physical devices to create sense of common space or sharing knowledge stored in office space. But there is knowledge that is created from social interaction in digital space too.

This single case study revealed a part of the whole picture in terms of knowledge management initiatives practiced in industry. Every organization applies different strategies to manage knowledge assets. Organizations are concentrated on the codification of knowledge. From our observation and interviews, it was found that developers in agile team are satisfied with their KM initiatives. The implementation of KM activities is not always directed as KM initiative, rather a solution for communication and knowledge exchange problem.

In conclusion, practitioners are focused on knowledge management with systems and organizational schools rather than spatial school due to the lack to effective, inexpensive softwares, hardwares and processes. The motivation of this research was in-depth inquiry of how agile distributed project practices knowledge management activities with regard to spatial school. There are video projection methods, virtual boards and virtual environments that can enhance current spatial knowledge creation and sharing process in distribute agile teams. It would be interesting to conduct case study in a distributed agile team that uses these kind of tools and compare with this study. Also multiple case studies can be interesting for this research which will help to reveal more empirical data from the industry.

REFERENCES


