A risk management framework for distributed agile projects

Suprika Vasudeva Shrivastava a,∗, Urvashi Rathod b

a Faculty of Business and Law, Department of Information Systems & Business Analytics, Deakin Business School, Geelong Waterfront Campus, Deakin University, Victoria, Australia
b Symbiosis Centre for Research and Innovation, Symbiosis International University (SIU), Lavale, Mulshi Taluk, Pune, Maharashtra 412115, India

Article history:
Received 15 February 2016
Revised 3 December 2016
Accepted 27 December 2016
Available online 5 January 2017

Keywords:
Distributed Agile Development (DAD)
Agile Software Development
Distributed Software Development (DSD)
Risk factor classification
Risk management framework in distributed agile development

Abstract

Context: Distributed agile development (DAD) approach has been adopted by the software companies for cost and time benefits. However, it causes significant challenges considering the contradicting nature of the agile and distributed development.

Objective: The objective of this study is to develop a risk management framework that comprises the perceived risks in DAD projects, their causes and the methods used in industry for managing those risks.

Method: This work is an extension of an exploratory study, wherein, DAD practitioners reported the risks they face in projects and the methods they use for managing those risks. The identified risks were further categorized based on their relevance to different aspects of DAD projects. In this extension, industry practitioners ranked the risks for their impact on DAD projects and rated the methods for the frequency of their use in projects. As the number of risks under each category was large for ranking, they were grouped under the risk areas within each category. The ranking of risk categories, risk areas and risk factors for their impact on DAD projects manifests their importance. The framework includes ranked risks, their causes and the risk management approaches. It was partially implemented in live projects in three different companies and was found to be beneficial.

Results: The perceived impact of the risk categories, ‘Group Awareness’, ‘External Stakeholder Collaboration’ and ‘Software Development Life Cycle’ on DAD projects has been found to be high and caused by the properties of Distributed Software Development (DSD). The partial validation of the framework in three companies reported the elimination of the majority of risk factors and/or reduction in their impact.

Conclusion: DAD projects provide significant benefits but hold substantial risks due to the contradiction between distributed development and agile practices. The reported framework could effectively minimize the DAD risks in practice.

© 2017 Elsevier B.V. All rights reserved.

1. Introduction

Distributed Agile Development (DAD) is intended to attain the quality and speed benefits of agile along with cost benefits of distributed software development (DSD). However, it induces significant risks due to differences in their key tenets [35]. Agile methods encourage frequent face-to-face communication and practices to build trust, while the distance in DSD signifies different organizational cultures, standards and policies, leading to reduced team cohesion [21]. It is observed that challenges in agile development and distributed development have been explored, but there has not been any comprehensive work on risks and challenges when these approaches are combined in practice [21]. Majority of research addressing the challenges in DAD are industrial reports [23,32,47], which is indicative of further need for investigation in this area.

This work is an extension of the exploratory research done by the authors on identification and classification of DAD risks and corresponding risk management methods in practice [40]. The exploratory study led to the formation of a list of risk factors that impact DAD project. A risk factor can be defined as a condition that can presents a serious threat to the successful completion of a software development project [41].

The classification of risk factors revealed five core risk categories, namely, ‘Software Development Life Cycle (SDLC)’, ‘Project Management’, ‘Group Awareness’, ‘External Stakeholder Collaboration’ and ‘Technology Setup’, which were then mapped to the components of the Leavitt’s model of organizational change [28] The purpose of the mapping was to facilitate the identification of organizational aspects, which must be addressed for managing risks in DAD projects. The exploratory study highlighted the role of the
intrinsic contradiction between the agile principles and the characteristics of distributed development in inducing risks in DAD projects. The exploratory work established the building blocks of risk management framework in DAD. These primitives have been deemed to be apt for providing deeper insights necessary for managing risks at the task/activity level by the practicing agile teams. However, there is a need for a systematic and structured approach for identification of the sources of risks and the methods used for managing them to facilitate DAD projects. The comprehensive framework must be able to manage risks in DAD projects at operational level. It is also expected to further the knowledge of the field for deriving new avenues for research and building dependable theories from experience. This study is the first step in this regard.

For the development of the framework, practitioners’ perception of the impact of the risk factors has been gathered using a self-administered questionnaire. The respondents had an experience of working at different levels in DAD project teams. At first, the practitioners ranked the risk categories for their impact on DAD projects. Then within each category, they ranked the risk areas and then, they ranked the risk factors within each risk area. The ranking revealed the most problematic aspects of DAD. Further, practitioners rated the extent to which they use specific techniques for managing risks in terms of the frequency of use. Based on the ranking and rating data, high impact risks and frequently used risk management techniques have been found. Further deliberation on the findings uncovered DSD properties that play significant role in inducing specific risks in DAD projects. Based on these findings, a DAD project risk management framework has been developed and subsequently, partially implemented in three different DAD projects. Substantial benefits have been achieved by the test DAD projects that validate the usefulness of the framework. Following are the contributions of this study to the existing knowledgebase of the field of DAD:

1. It has identified risk categories, risk areas under each risk category and risk factors under each risk area, which are perceived to have high impact on DAD projects.
2. It provides the risk management methods frequently used in practice for managing each risk factor in DAD projects.
3. Original exploration unveiled the DSD properties that conflict with agile principles and causes DAD specific risks. This extension study further established the role of those DSD properties with respect to the impact of DAD risks on projects and thus, equips the industry practitioners with the knowledge of the important aspects that must be addressed when certain DSD properties are more dominant than others.
4. The study resulted into the development of a risk management framework for DAD projects, which was validated by implementing a part of it on real world DAD projects. The results obtained after implementing the framework showed that it helped the team to perform effective risk management and hence, was useful.
5. Earlier research literature on the risks in DAD projects is either the studies based on literature reviews [8,22] or the case studies [33,33,44]. The findings of these works were limited and did not provide a roadmap for operationalisation in the real world. This research provides a comprehensive framework for managing risks in DAD projects that can be implemented by the practitioners.

In this paper, Section 2 describes the background of the research that shaped the research objectives and the methodology (Section 3). Section 4 is on Data Analysis and Results. Section 5 holds a discussion on results and the risk management framework. Findings of the validation study have been reported in Section 6. Section 7 describes the Limitation and Threats to validity of research. Finally, Section 8 holds the conclusion and future scope for the study.

2. Research background

Modern software development organizations prefer agile approach considering the benefits they achieve by offering better quality products quickly in market with improved customer satisfaction [32]. Due to the physical separation of the stakeholders in DSD, it becomes difficult to apply agile principles and practices effectively. There are studies which state that agile methods can be customized to suit the needs of distributed development [10,42].

However, there is a dearth of scientific research on the risks that arise when software development is done by combining agile and distributed development, commonly known as distributed agile development (DAD) [20,23]. There are some industry experience reports [43,49] and other case studies [32,33,44], which deal with risks in distributed agile development. Significant conceptual work in this area has been done by Hossain [20] in which the author has developed a framework for implementing scrum methodology in distributed development. Since these studies are based on review of literature, they do not consider the practical problems encountered in DAD projects in real life. There are other literature based studies, but they identify risks occurring in distributed development [8,22,52,53] or in agile development [11] separately. The existing studies consider some specific areas of concern and do not give an overall view of risks in DAD projects.

To fulfill the pressing need of consolidating the knowledge embedded in the minds of agile team members and developing a more dependable knowledge base for agile software development, a prior exploratory research, taken up by us, identified forty-five DAD risk factors categorized under five categories, namely, ‘Software Development Life Cycle’, ‘Project Management’, ‘Group Awareness’, ‘External Stakeholder Collaboration’ and ‘Technology Setup’ [40]. These risk factors were identified through exploration which involved conducting thirteen in-depth interviews with the practitioners who have experience in managing DAD projects, followed by an extensive study of work documents of twenty eight projects. The result of exploration revealed the way in which DSD properties such as Spatial Distance, Temporal Distance, Work/Development Culture, Language Barrier and Large Project Scope conflict with agile principles and cause the occurrence of risks. The risk management approaches that are used by the practitioners to manage the identified risks were also documented. Constant comparative method [12] was applied to analyse the content of in-depth interviews and work documents, which led to the comprehensive categorization of risk factors. The method is used for generating and suggesting the properties of a phenomenon, which in our case, is the Management of Risks in Distributed Agile Development (DAD) projects. Glaser [12] propagated this method for deriving the categories, properties and hypotheses about some general phenomenon from qualitative data by coding the initial incidents and identifying the relevant categories. Every new incident in the data is compared with the existing categories and placed under a category or a new category is created in the absence of a relevant category. As the process continues, the properties of categories start emerging. Further comparison between the categories and their properties leads to their integration. These evolved categories contribute to the emergence of new theory; further refinement requires reduction in terms of the items in categories, their properties and the number of categories. Such delimitation ultimately shapes the theory. Constant comparative method is useful in the development of categories, properties, hypotheses and theories based on qualitative data. However, it is not used for testing hypotheses. Categorization of risk factors involved executing all the stages of
constant comparative method. In the first stage, the transcription of each in-depth interview was reviewed carefully and pieces of information were extracted to form the risk categories [34]. The matrix was enriched iteratively with the data collected from every interview and the project work documents [39]. At this stage the risk categories and their properties started emerging. The second stage involves the iterative movement and deletion of risk factors across the risk categories and further deletion and addition of risk categories. Formation of concrete risk categories by grouping the risk factors leads to the reduction of categories and generalization enforced by constant comparison. This iterative process involved reading and re-reading of the transcripts and work documents. The end result was a comprehensive categorization of risk factors [40].

Although, the focus of in-depth interviews taken in the exploratory study was on DAD risks, arising specifically due to the combination of distributed development and agile approach, the respondents revealed many other risks, which can be attributed separately to each of the distributed development, agile methods or traditional software development. The researchers clarified with the respondents about the specific software development approach that they perceive to causing the risks and identified such risks and called them Non-DAD risks.

One example of such risk factor was ‘Customer not aligned to Agile Practices’ which was categorized under the risk category ‘External Stakeholder Collaboration’. This risk factor is attributed only to agile development approach and is caused when the customer is not accustomed to the way agile development works. The success of agile hinges on the active participation of the customers in the development process, which is possible when the customers are committed, collaborative and knowledgeable [3]. Another risk factor namely, ‘Switching of Context between Multiple Projects or between Development and Testing’, which was categorized under the risk category, ‘Software Development Life Cycle’ was considered as a Non-DAD factor. This factor is caused when developers have to switch task between multiple projects or switch between development and testing. Multitasking is one of the biggest drains and impacts the team performance significantly [4]. This risk can occur in any software development project irrespective of the methodology followed and so, was placed under the group of factors caused due to the traditional software development. Similarly, a risk factor which was specifically related to distributed development was ‘Lack of Experience in Managing Distributed Projects’, which was categorized under the risk category, Project Management. It was found that the team members do not have prior experience of managing distributed projects, due to which they are not aware of the challenges they may face while executing distributed projects [1]. In total such thirty-seven Non-DAD risk factors were identified, in addition to the forty-five DAD risk factors making the total count eighty-one.

Exploration is insufficient considering its inability to derive some conclusive guidance from the results and requires deeper investigation for more focused findings that can directly be used by the practitioners and researchers of the field. Practitioners in the field need to know the risks, which are perceived by the fraternity to have high impact on the success of DAD projects. They would also like to know the methods, which are regularly used by the DAD teams for managing the risks. This kind of knowledge along with the DSD properties that contribute to the emergence of such risks and the organizational dimensions that must be focused upon for the management of these risks can result in a framework that can be effectively used in practice. The findings of our previous exploratory research identified the problematic DSD properties and the organizational aspects (Leavitt’s model) that must be addressed and now, there is a need to know the risks that have high impact and the frequently used methods for the management of such risks. For achieving that, a descriptive study has been conducted that examines the risk factors for the degree of the impact that they are perceived to have on DAD projects, the frequency in which a risk management method is used in practice for managing a specific risk and the DSD properties, which are instrumental in causing the occurrence of risks that have high impact on projects.

3. Research objectives and methodology

The objective of this extension of the exploratory work has been the creation of a DAD risk management framework that comprises risk factors, which are perceived to have high impact on DAD projects and the risk management methods, which are frequently used to control the identified risks in practice. The framework also relates the DSD properties such as spatial distance, temporal distance, work/development culture, language barrier and large project scope with the risks that have severe impact on DAD project. Further, this study aimed at validating the risk management framework by implementing it in real world DAD projects executed in the industry.

To meet the stated objectives following research questions are addressed in this study.

(1) Which are the risk factors that have high impact on DAD projects?
(2) Which are the risk management methods, used frequently for managing risk factors in practice?
(3) Which are the DSD properties causing most of the risks to occur in DAD projects due to the contrast they have with agile principles and practices?
(4) Does the implementation of the risk management framework for DAD projects in real world projects provide any benefits?

As noticeable from the research questions, this study is descriptive in nature since it examines various dimensions of risks in DAD projects along with their perceived impact on the projects and the methods that can be used to control the risks [30]. Descriptive research approach examines the situation as it exists in its current state. It involves identification of attributes of a particular phenomenon based on observations or the exploration of correlation between two or more phenomena [30]. This descriptive study used a two-part self-administered questionnaire for capturing practitioners’ perception about the relative impact of the risk factors on DAD projects and how often the given risk management methods are used for controlling specific risk factors. In our initial exploratory study, practitioners were interviewed. Now, as we already had the identified risk factors with us, the survey of the perception of practitioners through questionnaire is considered to be appropriate. In the first part, ranks were given by the agile practitioners to the impact of risks on the success of DAD projects. In another part of the survey, ratings were given by the practitioners to the frequency of using a particular risk management method.

As the effectiveness of ranks can be ensured only when the number of items are eight or less [13], we observed that the number of risk factors under each category as identified in exploratory study was large, ranging from thirty-four (maximum) to four (minimum). The problem of large number of risk factors under each risk category could be resolved by further classification of risk factors within each risk category and subsequent introduction of risk areas under each category. We used constant comparative method [12] for identification of risk areas and grouped risk factors under different risk areas that subsequently fell under specific risk category.

The questionnaire listed five risk categories, the risk areas under each risk category and DAD as well as Non-DAD risk factors under each risk area making it a case of multi-level rank order
considered as 3.1. results collected controlling techniques respondents factors cation grouping questionnaire the institutes in 4. For. 3.1. Methodology for risk factor ranking

For collection of the rank data, we used purposive sampling because the responses were required to echo the experience of practitioners working on distributed agile development projects [5]. We set the criterion of minimum three years’ experience of working in DAD teams for the respondents in our sample as DAD approach is a relatively new phenomenon in software industry and three years in commercial software development is considered enough for understanding the intricacies of the process of software development. This helped us to obtain responses from the practitioners, who have executed multiple DAD projects and possess significant knowledge of risk management in such projects. This criterion restricted our sample to the chief executive officers (CEOs) of the companies (7), senior project managers (20), agile coaches (7) and business analysts (5) with minimum three years of experience of working on DAD projects. They belonged to various countries like UK, USA, Japan and India. We contacted 76 practitioners, out of which 11 respondents refused to give their responses or did not reciprocate to our correspondence. Therefore, we sent the questionnaire to 65 respondents, out of which 39 sent their responses back to us. Hence, we achieved a response rate of 60% for the first part of data collection. For some of the responses which had missing data, the respondents were contacted again through email or telephone to obtain the required data. The respondents belonged to 28 different multinational IT consultancy organizations ranging from small-sized to mid-sized to large sized. The respondents were asked to rank the risk categories on a scale of 1 to 5 which was defined as follows: 5: very strong impact, 4: strong impact, 3: some impact, 2: negligible impact, 1: no impact. The 'Don’t Know Condition' was also considered while collecting the ranking data for risk factors. The respondents were asked to give a rank '0' in case they are not aware of the risk, which was referred to the Don’t Know condition. The same scale was used to obtain the ranking for risk areas under each risk category and then, ranking for risk factors under each risk area. We limited the number of risk areas under each risk category and number of risk factors in each area to seven (or less) to avoid any bias introduced by the confounding effect of large number of items for ranking.

Kendall’s test of Concordance (W) [25] on the ranking data for risk categories, areas and factors has been used separately for assessing the extent of concordance in the responses. The average rank and statistical significance level (p=0.05) have been used for deciding on their impact and generalizability of findings, respectively. We chose Kendall’s test for analysis due to two reasons, 1) statistical significance of average ranks can be assessed and 2) the coefficient of concordance will be helpful in understanding the extent to which practitioners carry the same perception for a risk. As Distributed Agile Development is not very old approach of software development, we acknowledge that the practitioners might not have experienced the similar situation in DAD projects in a span of a few years. Therefore, in addition to average rank and statistical significance, it is prudent to know the extent of concordance in practitioners’ perception. Low levels of concordance will signify that less number of practitioners perceives exactly the same impact of a risk and average rank is representative of the similarity in their perception.

3.2. Methodology for risk management methods rating

The second phase of the survey involved the collection of data on the frequency of use of the risk management methods that were reported in the prior exploratory study. We contacted only those respondents who had provided us with the response for the first part of the questionnaire, since they were aware of the context of the study and also, had the minimum experience required for answering our questionnaire. Out of the first set of respondents, who had provided us the response for first part of the questionnaire, we could obtain the consent from twenty eight respondents for filling the second part of the questionnaire. Nineteen out of twenty eight respondents gave us responses for the second part of the questionnaire and hence we obtained the response rate of 67.85%. The respondents belonged to 17 different multinational IT consultancy organizations ranging from small-sized to mid-sized to large-sized companies. We obtained
responses from project managers (8), business analyst (2), agile coaches (4) and CEO of the companies (5).

The questionnaire had the list of risk management techniques corresponding to each risk factor. The respondents were required to provide the rating for risk management techniques based upon their ‘frequency of use in the industry’ on a scale of 5 to 1. The scale of 5 to 1 was defined as follows: ‘5’ for methods which are always used; ‘4’ for methods which are often used, ‘3’ for methods which are sometimes used, ‘1’ for those methods which are never used and ‘0’ (don’t know condition) if the respondents were not aware of the usage of risk management method. We applied Chi Square test of Goodness of Fit on the rating data for assessing the sample’s representativeness of the population. Goodness of fit tests the null hypothesis that the distribution of sample data is the same as the distribution of population data. High p-value backs the null hypothesis and so, indicates that there is a good fit in two distributions and so sample represents the population effectively.

4. Data analysis and results

The data analysis and results of ranking data and rating data are given in Sections 4.1 and 4.2, respectively.

4.1. Risk factor ranking data analysis

The application of Kendall’s test on the risk factor rank order data produced Kendall’s coefficient of Concordance (W), value of observed significance level (p-value) and the mean rank of risk factors within a particular risk area. The value of Kendall’s coefficient ‘W’ ranges from 0 to 1, where 0 reveals perfect disagreement and 1 reveals perfect agreement in the ranks given by the respondents [13]. We set the significance level to 0.05 for generalization of the findings at each level of the ranking. The detailed analysis performed at all the levels of the risk categorization is given below.

4.1.1. Analysis of risk categories ranking data

At the risk category level, the ranking was statistically significant (p-value 0.0000), although there was a low level of agreement amongst the respondents on the ranks given (W: 0.221). The risk category ‘Group Awareness’ was ranked highest while the category ‘Technology Setup’ was the lowest. The other categories were ranked in the decreasing order of the mean rank obtained as: ‘External Stakeholder Collaboration’, ‘Software Development Life Cycle’ and ‘Project Management’.

4.1.2. Analysis of risk area ranking data

Kendall’s test of concordance was then applied to the ranked data of the impact of risk areas falling under each risk category in order to identify the critical risk areas. The risk areas under each risk category have been arranged in decreasing order of the mean ranks given to them, as shown in Table 1.

The number in parenthesis along with the risk area name (in each cell) is the mean rank obtained by the risk area under respective risk category.

4.1.3. Analysis of risk factor ranking data

The risk factors under each risk area corresponding to a specific risk category were ranked based on their impact on project success. Table 2 shows the result of the Rank Test, wherein, the risk factors are arranged in descending order of the mean rank of impact they cast on DAD projects and segregated as per their significance level.

As mentioned in the section on Research Methodology, all DAD and Non DAD risk factors were ranked under each risk area and so, the average ranks, Kendall’s Coefficient of Concordance and p-values were obtained for all the listed risk factors. In Table 2, we have listed the average ranks of only DAD risk factors. This has saved the space and also, kept the focus of the paper on DAD risk factors only. This is the reason that some risk areas in Table 2 either do not show the ranks of any risk factor (e.g., the risk area ‘Managing the Project Team’) or list only one risk factor (e.g. risk areas, Customer Collaboration, Project Initiation, Infrastructure and Resources and Tool Selection).

As discussed previously in the Research Methodology (Section 3), there are certain risk areas for which none of the risk factors or only one risk factor has been listed due to the fact that either all or the remaining risk factors belong to Non DAD group of risks. Since, the ranking of risk factors was done by considering DAD and Non DAD factors collectively under their respective risk areas, we obtained the p-value and Kendall’s Coefficient for every risk area and reported that in Table 2.

4.2. Analysis of ratings to the frequency of using risk management methods

Corresponding to each of the risk factors identified, we obtained the rating for the frequency of the use of specific risks management methods on a scale of ‘1’ to ‘5’. We used ‘Chi Square test of Goodness of Fit’ in order to find out that the sample is true representation of the population. However, due to small size of sample, we could not fulfill the basic condition of having minimum expected frequency of each category as 5. This goes contrary to fundamental assumption of asymptotic distribution of population data. Although, Zeis et al. [53] showed that the underlying population distribution for likert type data is, in general, a continuous distribution. Instead of making ungrounded assumptions, we decided not to make any claim for specific finding from this even if we get high p-values, which, otherwise, lead to the acceptance of the null hypothesis of good fit of sample distribution to population distribution. We understand that as the basic condition of asymptotic distribution is not met due to the small sample size, the derived p-values could be spurious. Therefore, we decided to report the methods with high average rating for more frequent use in practice.

Chi-square test on the rating data generated three values (i) Chi-square test statistic (ii) Significance Level (p-value) and (iii) the average rating. For majority of methods, p-values are found to be high, which makes the null hypotheses of goodness of fit true and so, it can be concluded that the sample is representative of the population. However, the failure in achieving the basic requirement of asymptotic distribution for Chi Square test due to small sample size, we decided not to make any claim on the findings. For the framework, we segregated the risk management methods corresponding to DAD risk factors based on the ratings given to them depending upon the frequency with which they are used in the industry. The important risk factors and risk areas have been grouped together as ‘Risk Areas of Concern’ for DAD projects. We have provided these generic areas of concern along with the corresponding risk management methods which are most frequently (rating ‘4’ and ‘5’) used by the practitioners for controlling them in the ‘Discussion’ section of this paper.

5. Discussion

As distributed software development (DSD) and agile methods differ in their key tenets, significant risks arise when these two methods are combined for software development. Our exploratory work unveiled the properties of DSD which contrast with the principles of agile and hence cause risks [40]. We could relate each risk factor identified in DAD projects to one or more DSD properties, namely, spatial distance, work/development culture,
large project scope, temporal distance, and language barrier which contradict with the agile principles and practices.

Spatial distance refers to the geographic distance between the stakeholders [15], especially amongst team members and between the team and the customer representatives. Difference in work/development culture happens due to different team behaviour, perception of authority, hierarchy, planning and organizational culture amongst different teams [17,26]. Further, use of different tools, processes and practices is also related to this dimension adds to the challenges caused due to cultural differences [26]. Distributed projects tend to be large, more complex and have more number of contributors [14]. This causes more number of defects and the increase in the tendency of work going out of control [7]. Temporal distance is related to the difference in time-zones in which the multiple teams work in distributed environment [16]. Language barrier arises when the participants do not share common language or norms of communication [38].

On the other hand, agile methods emphasize on enhanced communication and trust building amongst the stakeholders in order to deliver enhanced business value to the customer. Agile principles and practices which can be represented by keywords like Customer Satisfaction, Embrace Change, Frequent Software Delivery, Customer Involvement, Team Motivation, Face-to-Face Communication, Working Software, Sustainable Pace, Simplicity, Self-organized Teams, Inspect and Adapt are difficult to achieve with distributed teams [40].

5.1. Impact of DSD properties on the ranking of DAD risk factors

In the stack of risk categories, the results show that the risk category ‘Group Awareness’ and ‘External Stakeholder Collaboration’ have the highest impact on the performance of DAD projects. Both these categories deal with the risk occurring due to poor communication and collaboration between the team members or between the teams and customers or other vendors. The risk category ‘Software Development Life Cycle (SDLC)’ was ranked third in terms of its importance as reported by the respondents. Software development in distributed agile environment requires the team members to deal with the risks arising due to the agile practices like short development cycles, lack of documentation, and dependency on social and interpersonal skill of team members for project development [29]. Distributed environment makes these issues even more difficult to manage and increase the complications in development process.

The next important risk category after SDLC was ‘Project Management’ which was followed by the risk category ‘Technology Setup’. Scrum or its variants are widely used Project Management Frameworks for agile development [46] and so, high comfort level of agile practitioners with those may be the reason behind the low importance given to ‘Project Management’. Similarly, most companies are investing on IT infrastructure and have well-established ‘Communication Infrastructure’ and ‘Tool Setup’, which are risk areas under the category ‘Technology Setup’ [50]. Hence, the risk category ‘Technology Setup’ has been ranked lowest amongst all the risk categories.

In this section, we elaborate the role of the aforementioned contradicting properties of distributed software development in augmenting the occurrence of the significant DAD risk factors. We provide relevant justification and reasoning for the ranks received by the risk factors with respect to the DSD properties based on the evidences in the existing literature and analysis of our exploratory work [40].

5.1.1. Spatial distance leading to most important DAD risk factors

Results show that spatial distance between the stakeholders has influenced the practitioners’ perception about the impact of the risk factors on distributed agile development projects (Fig. 1).

In each risk category and corresponding risk areas, the risk factors that have higher ranks are apparently related to the spatial distance. As can be seen in Table 2, in the risk category, ‘Group Awareness’, the risk factors including, ‘Lack of Communication between the Team Members’, ‘Poor Collaboration between Different Sites’ and ‘Lack of Trust between Onshore and Offshore Teams’ are highly ranked factors in their respective risk areas. These factors occur in DAD projects due to the inability of the team members and the customers to collaborate considering their geographic separation.

Similarly, we can see that ‘Customer Collaboration’ (one of the risk areas in this study) suffers due to the ‘Unavailability of Product Owner’ (risk factor) for the team, who can get clear requirements for system development [35]. Besides customer representatives, teams need to have constant communication with other vendors and third parties [35]. In the risk category, ‘External
Table 2
DAD Risk factors ordered as per the mean rank and segregated based on the significance value.

<table>
<thead>
<tr>
<th>Risk Categories ordered based on the Mean Rank</th>
<th>Risk Area ordered based on the Mean Rank</th>
<th>Statistically Significant Risk Factors (p-value ≤ 0.05) ordered based on Mean Ranks</th>
<th>Statistically Insignificant Risk Factors (p-value &gt; 0.05) ordered based on Mean Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Awareness **</td>
<td>(p-value: 0.467)</td>
<td>Statistically Significant Risk</td>
<td></td>
</tr>
<tr>
<td><strong>Communication</strong></td>
<td>(p-value: 0.020)</td>
<td>Lack of Communication between Team and the Client. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Coordination &amp; Collaboration</strong></td>
<td>(p-value: 0.819)</td>
<td>Statistically Significant Risk</td>
<td></td>
</tr>
<tr>
<td><strong>External Stakeholder Collaboration</strong></td>
<td>(p-value: 0.819)</td>
<td>Lack of Trust between different sites. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Multiple Vendor Involvement</strong></td>
<td>(p-value: 0.819)</td>
<td>Unavailability of Product Owners. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td>**Software Development Life Cycle **</td>
<td>(p-value: 0.000)</td>
<td>Statistically Significant Risk</td>
<td></td>
</tr>
<tr>
<td><strong>Requirement Elicitation</strong></td>
<td>(p-value: 0.000)</td>
<td>Requirements Unclear to the Team. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Release Management</strong></td>
<td>(p-value: 0.827)</td>
<td>Statistically Significant Risk</td>
<td></td>
</tr>
<tr>
<td><strong>Testing and Integration</strong></td>
<td>(p-value: 0.031)</td>
<td>Difficulty in System Release Management and Deployment. Temporal Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Coding and Integration</strong></td>
<td>(p-value: 0.000)</td>
<td>Statistically Significant Risk</td>
<td></td>
</tr>
<tr>
<td><strong>Design</strong></td>
<td>(p-value: 0.343)</td>
<td>Poor Technical Debt Management. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Requirement analysis and Specification</strong></td>
<td>(p-value: 0.343)</td>
<td>Inconsistency in Design Standards of Distributed Teams. Work development Culture</td>
<td></td>
</tr>
<tr>
<td><strong>Standards of Agile Ceremonies</strong></td>
<td>(p-value: 0.005)</td>
<td>Statistically Significant Risk</td>
<td></td>
</tr>
<tr>
<td><strong>Project Management</strong></td>
<td>(p-value: 0.077)</td>
<td>Higher Interdependency between the teams. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Project Organization</strong></td>
<td>(p-value: 0.216)</td>
<td>Working with Component teams instead of Feature teams. Spatial Distance</td>
<td></td>
</tr>
<tr>
<td><strong>Project Planning and Control</strong></td>
<td>(p-value: 0.005)</td>
<td>No common Definition of Done. Work/Development Culture</td>
<td></td>
</tr>
<tr>
<td><strong>Project Initiation</strong></td>
<td>(p-value: 0.000)</td>
<td>Differences in Agile Practices and Standard of Processes followed by Multiple Teams. Work/Development Culture</td>
<td></td>
</tr>
</tbody>
</table>

(continued on next page)
Table 2 (continued)

<table>
<thead>
<tr>
<th>Risk Categories ordered based on the Mean Rank</th>
<th>Risk Area ordered based on the Mean Rank</th>
<th>Statistically Significant Risk Factors (p-value &lt;= 0.05) ordered based on Mean Ranks</th>
<th>Statistically Insignificant Risk Factors (p-value &gt; 0.05) ordered based on Mean Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquistion and development of project team *</td>
<td>Statistically Insignificant Risk (p-value: 0.361) (W: 0.037)</td>
<td>Unavailability of Business Analyst, Work/Development Culture</td>
<td>Lack of Uniformity in Multisite Team’s Capabilities, Work/Development Culture</td>
</tr>
<tr>
<td>Managing the Project Team *</td>
<td>Statistically Significant Risk (p-value: 0.002) (W: 0.113)</td>
<td></td>
<td>(Non DAD Risk Factors since all factors belong to Non-DAD group of Risks)</td>
</tr>
<tr>
<td>Technology Setup</td>
<td>Statistically Significant Risk (p-value: 0.201) (W: 0.042)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>**(p-value: 0.162) (W: 0.050)</td>
<td></td>
<td>Lack of Communication Infrastructure, Spatial Distance</td>
<td></td>
</tr>
<tr>
<td>Infrastructure and Resources</td>
<td></td>
<td>Statistically Insignificant Risk (p-value: 0.180) (W: 0.046)</td>
<td>Inappropriate Tool Selection, Work/Development Culture</td>
</tr>
<tr>
<td>Tool Selection</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Significance value

***(Significance Value: p-value <= 0.05); obtained after applying Kendall’s test on all the ranks given to the risk areas under the marked category (Generic)***

**(Significance Value: p-value > 0.05 and <= 0.10); obtained after applying Kendall’s test on all the ranks given to the risk areas under the marked category (Semi-Generic)***

1. Those risk areas which do not possess any asterisk are the ones which have p-value > 0.10 (Specific)
2. Risk factors colored Blue: Statistically Significant Risk Factors (p-value <= 0.05) ordered based on Mean Ranks.
3. Risk factors colored Red: Statistically Insignificant Risk Factors (p-value > 0.05) ordered based on Mean Ranks.
4. The tags namely, Spatial Distance, Temporal Distance, Work/Development Culture, Language Barrier and Large Project Scope attached with each risk factor (superscript with the names of risk factors). They refer to the DSD property which is the primary cause for the particular risk to occur.

Stakeholder Collaboration’, the risk factors, namely, ‘Poor Coordination between Multiple Vendors’ and ‘Risk in Code Integration with Multiple Vendors’ are perceived to be important. Besides technical disparity, inability to coordinate the work at distant geographical locations poses major risk to the projects.

Further, software development activities such as requirement elicitation, coding and testing that involve close collaboration between the team members suffer due to their physical separation in distributed environment. Risk areas ‘Requirement Elicitation’, ‘Testing and Integration’ and ‘Coding and Integration’ in the risk category ‘Software Development Life Cycle (SDLC)’ (Table 2) have been ranked high and are statistically significant.

One of the highly ranked risk factors in the risk area, ‘Coding and Integration’ is ‘Poor Technical Debt Management’. Management of technical debt is critical for agile teams because they have to provide business value more frequently, adapt to the changing business needs and work at a sustainable pace. Distributed teams impede the speed and the ease of communication necessary for adapting to the changes. Another risk factor, ‘Unavailability of Requirement Documents for Testing’ has been ranked high because agile teams keep little documentation of requirements and the non-co-located DAD teams need requirements documents for testing. As apparent, both these factors become more perilous when teams are separated spatially and temporally.

A high ranked and statistically significant risk factor, ‘No Common Definition of Done’, in the risk area ‘Standards of Agile Ceremonies’ is caused as the team is unable to meet frequently to take decisions due to the physical distance between its members.

In distributed environment, team structuring is an important issue which helps to keep the team dependencies low and hence improve productivity and quality of work. The related risk factors fall under the risk area ‘Project Organization’, which is one of the highly ranked risk areas under the risk category, ‘Project Management’. One important risk factor under the risk area ‘Project Initiation’ is ‘Difficult to execute Fixed Price Projects’. This risk factor is related to difficulty in managing scope changes when the agile project is fixed price. The difficulty of managing changes in scope increases when the teams are not co-located. Agile teams need to conduct multiple levels of planning which makes project planning and execution in distributed environment difficult. Table 2 signifies that 16 out of 30 statistically significant DAD risk factors and 9 out 14 statistically insignificant DAD risk factors are caused due to the spatial distance.

5.1.2. Work/development culture leading to important risk factors

Differences in software development practices and organizational culture lead to significant risks in DAD projects. The results indicate that the difference in work/development culture is the next important factor after the spatial distance that induces risks to DAD projects. The risk factor, ‘Inappropriate User Story Estimate by Multiple Vendors’ in the risk category ‘External Stakeholder Collaboration’ and ‘Differences in Agile Practices and Standard of Processes followed by Multiple Teams’ in the risk category, ‘SDLC’ have been mentioned as important. Another important risk factor under the risk category SDLC is ‘No Common Definition of Done’ which occurs when DAD teams are not having a common definition of done at feature, sprint and release level.

Distributed projects are further adversely impacted when developers and testers lack collaboration which leads to ‘we versus they’ attitude between them. The risk factor ‘Lack of Collaboration between developers and quality assurance members’ is a highly ranked risk factor under the risk area ‘Collaboration and Coordination’. Agile team members on the other hand need to collaborate extensively in order to develop multiple skills and be cross functional.

Similarly, another risk factor, ‘Lack of Uniformity in Multisite Team’s Capabilities’ under the risk area ‘Acquisition and development of project team’, although statistically insignificant, occurs in DAD projects due to the commonly found difference in technical capabilities of different teams. Significant difference in the skill level of onshore and offshore teams leads to decreased productivity and hindered team work. In agile development, the team members need to build multiple skills while maintaining their own area of specialization which becomes difficult with teams spread out at various locations. Distributed structure of the team makes knowledge sharing amongst the team members difficult, hence leaving one of the teams with lesser knowledge.

We can see in Table 2 that 4 out of 30 statistically significant DAD risk factors and 4 out of 14 statistically insignificant DAD risk factors are occurring due to the difference in work/development culture amongst the multiple teams.
5.1.3. Large project scope leading to moderately important risk factors

Agile methods are designed for projects involving small and medium sized collocated teams to address the need for fast informal communication [31]. On the other hand, distributed projects involve large teams and increased number of stakeholders. This increases the communication lines and reduces the effectiveness of agile practices [45].

This is evident from the findings of this study. The risk factor ‘Unsuitability of Agile approach for Large Organization’ has been placed amongst the statistically significant risk factors under the risk area ‘Communication’ (Risk Category: Group Awareness). These teams find it difficult to perform the requirement prioritization (Risk Factor, ‘Inadequate Prioritization of Requirements’ under the risk area ‘Coding and Integration’ in the SDLC risk category) due to the large scope of the project [27]. Further, agile methods suggest the use of cross-functional teams, but projects with large scope need dedicated testing skills for effective testing of complex modules (‘Cross Functional Teams Insufficient for Testing of...’).
Large Projects’ under the risk area ‘Testing and Integration’ (Risk Category: Software Development Life Cycle).

As observed in Table 2, although the risk factors related to large project scope are statistically significant, they are ranked low as compared to other factors in their respective risk areas. The reason could be that many agile teams have recognized that agile methods were originally designed for small teams and they have to take extra efforts to facilitate the communication, for example, by conducting scrub-of-scrum meetings, using good communication tools, to manage large complex projects [23,48]. There are evidences in the literature too, which claim that agile approach has been successfully applied to large projects [10,33,48]. We can see in Table 2 and in Fig. 1 that 5 out of 30 statistically significant DAD risk factors and none out of the statistically insignificant DAD risk factors are occurring due to the large project scope. Hence, we can say that ‘Large Project Scope’ is leading to moderately important risk factor.

5.1.4. Temporal distance leading to less risk factors

Temporal distance follows the large project scope as a reason for posing risks to DAD projects. Under the risk category ‘Group Awareness’, the risk factors arising due to the lack of coordination have been found to be comparatively less important. Poor coordination between the distributed teams happens due to the asynchronous working hours of the team members [14]. Agile practices like daily scrum, sprint planning meeting, sprint review meeting help in improving team coordination, but temporal distance in distributed teams reduces their effectiveness, hence causing risks like ‘Poor Coordination between Multiple Teams’ [23].

Further, poor coordination in team affects the testing and integration of code [20]. In Table 2, the risk factor, ‘Code Integration across Multiple Sites’ has been ranked lower than few other factors namely, Poor Technical Debt Management and Unmaintainable Code (Non DAD risk factor), in the risk area, Coding and Integration.

It is observed that temporal distance leads to reduction in overlapping hours of team members and requires the team members to be flexible, but agile practices like pair programming reduce the problem of creating the time overlap. Increased communication in agile teams increases the commitment in team members to achieve the project success [19]. Moreover, distributed agile teams have started using practices like use of asynchronous communication tools like wiki, newsgroup, email, replacing direct communication with documentation, establishing several lines of communication between the teams, videoconferencing, desktop sharing, and scheduling regular travels which helps to resolve the difficulties arising in team communication and project execution due to temporal distance [23]. Hence, we observe that only 3 out of 30 statistically significant DAD risk factors and 1 out of 14 statistically insignificant DAD risk factors are caused by the temporal distance.

5.1.5. Language barrier leading to least important risk factors

Agile teams need to communicate frequently amongst each other and also with the customer for solution development. Since, agile methods prefer face-to-face communication over of documentation, difference in language amongst the team members cause significant challenges. In case of distributed teams, difference in language skills is bound to happen since the teams are dispersed across multiple locations across the globe. Distributed agile teams face difficulty in conducting effective daily standup meetings due to difference in language and culture (Risk Factor ‘Ineffective Standup Meetings’ in risk area ‘Standards of Agile Ceremonies’ and Risk Category ‘SDL’). Further, we can see in Table 2, the risk factor ‘Uncommon Language’ under the risk area ‘Communication’ (Risk Category ‘Group Awareness’), is placed at a lower rank than the risks caused due to the spatial distance.

Although, language barrier is a challenging aspect of distributed development, agile practices in DAD projects help to reduce the socioulture distance as they focus more on individuals and interactions. Practices like pair programming, collaborative planning activities encourage people from different cultures and backgrounds to communicate more effectively [19]. We understand that use of agile approach has caused very few factors being attributed to language barrier.

The extant literature claims that language barrier poses significant challenge to the success of projects [9,19,23]. However, according to these findings, the risks to DAD arise more due to the spatial distance and difference in work culture and less due to the language barrier. This finding could be attributed to the language advantage of the research setting. As the respondents for this part of the research have been from India, wherein, knowledge of English is reasonably good as compared to many other countries, they did not find the risk factors induced by the language barrier important. We concede that our respondents did not consider the usage of English difficult and would have found it easy to understand the accent of their counterparts from other countries. The other reason could be the extensive training given to the members of the distributed agile teams on the common language of communication.

It can be observed that 2 out of 30 statistically significant DAD risk factors and none out of the set of statistically insignificant DAD risk factors are occurring due to the language barrier.

5.2. Risk management methods for controlling the risks in DAD projects

As discussed in Section 4.2, for each of the risk factor, we segregate the risk management methods that are more frequently used (i.e. rating 5 or 4) from the ones that are less frequently used (i.e. rating 1, 2 or 3) to control the risks in DAD projects. In Table 3, we deduce some general areas of concern from the identified set of risk factors and present the risk management practices which are used more frequently to control them. Major risk areas of concern correspond to some problem area in DAD projects which is addressed by a group of risk management methods.

6. Validation study of the risk management framework

In order to check the applicability and the feasibility of the risk management framework, we conducted a validation study. We provided the partial framework to three different companies for implementation in their real time DAD projects. The framework implementation benefited the companies as they were able to identify the critical risks in the projects and use the suggested risk management methods for controlling the same. The details of the methodology used, data analysis and results obtained are given in this section.

6.1. Methodology for validation study of the risk management framework

The partial framework was implemented in real time DAD projects executed by three different Information Technology companies. The complete risk management framework, including all the risk categories, risk sub-categories and risk factors, was presented to the company representative for performing risk identification and control in distributed agile project. The company representative was asked to select the risk categories that were relevant for the selected project. Amongst the three companies, Company 1 selected the risk category, ‘Software Development Life Cycle (SDL)’ and Company 2 and Company 3 selected the risk category, ‘Project Management’. The risk categories selected by the companies were based on the risk factors that were important for the se-
Table 3
Major ‘Risk Areas of Concern’ in DAD projects and appropriate risk management practices for dealing with the risks.

<table>
<thead>
<tr>
<th>Major Risk Areas of Concern for DAD</th>
<th>Risk Management Practices</th>
</tr>
</thead>
</table>
| Communication for Collaboration     | Face-to-face interaction by having periodic collocation of team members at various distributed location and the client, especially during the initial stage of the project.   
Fostering team collaboration by using rich communication media like video conferencing, web conferencing and other collaboration tools.   
Scrum Masters to facilitate regular meetings like scrum-of-scrum and daily stand-ups between the distributed teams to encourage sharing of project status and issues in development.   
Use of Community-of-practice (CoP), which is an agglomerated forum for exchanging ideas and knowledge.   
Minimal documentation for knowledge sharing   
Training for team members for improving their communication skill and language skills.   
Enhancing team spirit by increasing collaboration between developers and testers, by involving testers throughout the project development.   
Encouraging the customer to have regular interactions with the teams through distributed meetings and occasionally collocate. |
| Third Party Management              | Synchronization amongst the multiple vendors to follow common standards and policies for software development.   
Use of vendor governance models which have ‘single point of responsibility’ like ‘Guardian Vendor Model’ to facilitate vendor coordination.   
Advance release planning and contract management and keeping the client informed about it, to manage the possibility of delays to other vendor or third party involvement. |
| Software Engineering Practices     | Explicit efforts to be taken for obtaining clear requirements by using rich communication media for interactions between the teams and the customer.   
Showing demonstrations of the working software to the customer regularly.   
If multiple customer representatives, then encourage them to have frequent communication to resolve the requirement issues. Having a single person responsible for deciding on the requirements (chief product owner) helps.   
Continuous integration of modules of different sites to enable the teams to make frequent and periodic releases.   
Use of appropriate tools to encourage pair programming in distributed teams. Carefully creating pairs of developers, ensuring their compatibility is necessary for good results.   
Creation of flexible designs to enable the teams to incorporate architectural changes.   
Creation of an architectural board composed of senior members to decide upon the design issues is useful.   
Backlog management and regular requirement prioritization.   
Distributed teams to have a common definition of done (DoD), which is an audit checklist of activities to be done for building the software. It is necessary for the teams as well as the management to agree upon the definition of done and use it throughout the process of development. |
| Team Organization and Management    | Use of long-lived feature teams who possess the skill and capability to deliver end-to-end solution, at each distributed site. Feature team is defined as long lived, cross functional, cross-component team that completes end-to-end features one by one [27].   
Minimize the dependency between the distributed team as far as possible.   
Ensuring that minimum team disruption happens during project development. If new members are added, ensure that they are provided with appropriate trainings to help them understand the complexities related to project and development environment.   
Having dedicated business analyst for each of the distributed team.   
Swapping of team members at various locations occasionally to enhance trust amongst the team members.   
Encouraging team members to do knowledge sharing. |
| Communication Infrastructure and Tools | Facilitating the team with appropriate tools for communication   
Use of common and good quality of tools by the distributed teams for development |

lected project. We understand that for the risk factors grouped under other risk categories, the companies had already taken suitable measures to control them or did not find the risk factors relevant to the selected project. Based on the risk categories selected by the company representative, we formalized the partial risk framework and sent it to the company for implementation in the DAD project. The risk category ‘SDLC’ was implemented in one of the companies while the risk category ‘Project Management’ was implemented in two other companies. For the considered risk category each company selected a set of risk factors, which were relevant to the ongoing project wherein the risk management framework was implemented.

We used two different self-administered questionnaires in order to measure the ‘Risk Exposure’ corresponding to the considered risk factors before and after implementing the framework.

The pre-intervention questionnaire was used to measure the ‘Initial Risk Exposure’ for the considered risk factors before the implementation of the framework. This required collecting ‘Initial Risk Impact’ and ‘Estimated Probability of Occurrence’ for the selected risk factors. The ‘Initial Risk Impact’ is the amount of loss that the risk may cause to the project as estimated by the practitioner and rated on a scale of 1 (low) to 10 (high). ‘Initial Probability of Occurrence’ is the probability that the risk factor is expected to occur and can also be rated on a scale of 1 (low, 0.1 probability) to 10 (high, 1.0 probability). ‘Initial Risk Exposure’ was calculated using ‘Initial Risk Impact’ and ‘Initial Probability of Occurrence’.

The companies considered the risk factors in the selected risk category and the suggested approaches for controlling those specific risk factors. The list of risk categories, risk areas and risk factors were ranked based on their expected level of impact. This helped the companies to foresee the risks which might impact the success of the project. Further, the teams implemented the relevant techniques from the given most frequently used risk management approaches for controlling the risk factors which may occur in the project.

In order to find out the benefits of the framework, we measured the ‘Current Risk Exposure’ of the considered risk factors after the framework implementation. After implementing the framework we obtained the ‘Current Risk Impact’ and ‘Current Extent of Risk Occurrence’ through a Post-Intervention Questionnaire. ‘Current Risk Impact’ is the amount of loss that the risk has caused to the project and rated on a scale of 1 (low) to 10 (high). ‘Current Extent of Risk Occurrence’ is the extent to which the risk actually occurred during the project execution with respect to the anticipation, can be rated on a scale of 1 (little occurrence) to 10 (substantial occurrence).
The benefit of implementing the framework was determined by the difference between the 'Initial Risk Exposure' obtained before the framework implementation and the 'Current Risk Exposure' for the considered risk factors after implementing the framework. The difference in Risk Exposure = Initial Risk Exposure – Current Risk Exposure

Subsequently, the following observations were made:

1. The number of risk factors which were totally eliminated due to the framework implementation.
2. The number of risk factors whose risk exposure (magnitude of risk) reduced: Those risk factors for which the difference in risk exposure was a 'non-zero positive value'.
3. The number of risk factors for which the magnitude of risk remained the same: Those risk factors for which the difference in risk exposure was 'zero'.
4. The number of risk factors for which the magnitude of risk inflated: Those risk factors for which the difference in risk exposure will be 'a negative value'.

The percentage of risk factors whose risk magnitude has reduced and the percentage or risk factors which were altogether eliminated, were used to measure the benefit of the framework.

6.2. Validation study: data analysis and results

We present the results obtained after implementing the partial risk management framework in three multinational IT consultancy companies. Table 4 shows the number of risk factors identified in each project got eliminated, the number of risk factors whose impact was reduced and the number of risk factors whose risk magnitude either remained the same or increased. The results show that there were higher percentage of risk factors in each company whose risk magnitude got reduced or eliminated than those for which the risk magnitude remains the same or increased. This indicates that the implementation of the framework led to effective risk control.

We present the benefits of implementing the risk management framework (partial) in three companies as follows:

6.2.1. Validation study in company 1

Majority of the risk factors (more than 50% –79.41%) in ‘Software Development Life Cycle’ risk category did not occur after implementation of the Risk Management Framework. There are certain risk factors (8.8%-33.33%) in SDLC risk category, whose Risk Magnitude (Risk Exposure) has reduced after applying the framework. Only one of the team (Team 3) has reported some risk factors whose magnitude of risk (risk exposure) has remained the same.

Although, there are some risk factors in case of all the teams, for which the risk magnitude has inflated, overall effect of implementing the framework was positive. As per the qualitative response on the benefit and usage of the framework, we received the following response (stated verbatim) from the three teams “Overall feedback from the team has been that the framework gives an insight into the challenges that the team might face during the project execution and can take care of those upfront by using the mitigation techniques provided.”

6.2.2. Validation study in company 2

As per the project manager who implemented the partial framework in the project, the risk management framework is able to bring significant risks which would impact DAD projects, to the forefront. The identification of the risk factors and the corresponding risk management methods was beneficial for their project.

Further, the ranking order of the risks under various risk areas in the risk category ‘Project Management’ was also helpful. It can be observed that there were 50% of the risks whose risk magnitude decreased. The result indicate that significant number of risk factors as suggested by the framework were considered important by the team and helped them to predict the challenges they may face while executing DAD project.

Although the framework was useful, due to the limited time and certain organizational constraints and policies, the team could not implement the framework with the required amount of rigor and seriousness. As a result, along with the risk factors whose risk magnitude reduced, there was significant number of factors whose risk magnitude remained the same (11%) or even increased (38.33%).

6.2.3. Validation study in company 3

The analysis shows that there were more than 50% of the risk factors considered for implementation whose risk magnitude reduced. There are 26% of the risk factors whose risk magnitude did not reduce and it remained the same, but there is no risk factor whose magnitude increased.

Table 4
Validation results of partial framework implementation in 3 IT companies.

<table>
<thead>
<tr>
<th>Company Name</th>
<th>Company 1</th>
<th>Company 2</th>
<th>Company 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Type</td>
<td>Application for Bank</td>
<td>Product Development</td>
<td>Product Development</td>
</tr>
<tr>
<td>Total No.</td>
<td>5 (Globally Distributed)</td>
<td>2 (Distributed within India)</td>
<td></td>
</tr>
<tr>
<td>Teams Implemented the Framework</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Team Size</td>
<td>7th</td>
<td>3rd</td>
<td>2nd</td>
</tr>
<tr>
<td>No. of Teams</td>
<td>45 (15 members per team)</td>
<td>20 members</td>
<td>20 members</td>
</tr>
<tr>
<td>Risk Category Implemented</td>
<td>34 Risk Factors</td>
<td>23 Risk Factors</td>
<td>23 Risk Factors</td>
</tr>
<tr>
<td>No. of Risk Factors Identified</td>
<td>Team 1 – 34 (Risk Factors)</td>
<td>Team 18 (Risk Factors)</td>
<td>Team 18 (Risk Factors)</td>
</tr>
<tr>
<td>No. of Risk Factors Eliminated</td>
<td>Team 1 – 34 (Risk Factors)</td>
<td>Team 18 (Risk Factors)</td>
<td></td>
</tr>
<tr>
<td>No. of Risk Factors whose Risk Magnitude Reduced</td>
<td>Team 12/18 - 50%</td>
<td>Team 12/18 - 66.66%</td>
<td></td>
</tr>
<tr>
<td>No. of Risk Factors whose Risk Magnitude remained the same</td>
<td>Team 1/0</td>
<td>Team 6/18 - 33.33%</td>
<td></td>
</tr>
<tr>
<td>No. of Risk Factors whose Risk Magnitude inflated</td>
<td>Team 1/0</td>
<td>Team 6/18 - 33.33%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Team 3/5/15 - 33.33%</td>
<td>Team 7/18 - 38.8%</td>
<td>Nil</td>
</tr>
</tbody>
</table>
Overall, we can observe that the risk impact of approximately (6 to 50) % of the risk factors in the considered DAD projects got reduced, while, approximately 50–80% risk factors in the SDLC risk category (Company 1) were totally eliminated.

This clearly shows that the framework was beneficial for the project in which it was implemented. As per the project manager who helped us in validation, the framework made some of the important risks discernible to the team, which helped them to take measure at the right time. He said that “although they are also having their own practices which they have been using for controlling those risks, but still the increased visibility of the risks, was beneficial for the project.”

7. Limitations and threats to validity

The authors emphasize this work’s significant contribution to the field through the risk management framework for DAD projects. However, there have been some unavoidable limitations, which must be stated for setting the relevance and significance of the framework. The large number of risk factors not only caused phased collection of data but also constrained online data collection as the respondents were not keen to fill the large questionnaire online. Majority of responses were collected through face-to-face meetings. This involved extensive travel and time and has definitely affected the number of respondents adversely. Since, India was the major location for data collection, outside respondents sent the response through e-mails. Further communication with such respondents on the data missing in the filled questionnaire resulted into second set of supplement responses. Due to repeated request for filling the data, the responses received second time might be close to the true perception but might not hold the same strength of thought as desired and might have impacted the validity of the study.

One threat to the validity of research is reflected by the low value of the coefficient of concordance, Kendall’s W, for the ranks given to the impact of the risk categories, areas and factors by the respondents on DAD projects. As distributed agile approach of software development is evolving and the respondents had experience of working on different research projects in varied environments, it is difficult to achieve high level of concordance in their perception about the risks. Some respondents might have been through the situations that others have not and this could be a major reason behind the low values of W. Ideally, experience of working on similar projects in similar environments could have made explicitly dependable revelations. However, that would have marred the generality of results, which could be achieved by this study considering the p-value for the ranks given to the majority of the risks. As the field of Distributed Agile Development evolves, we anticipate better concordance in responses due to the breadth and depth of the experience of respondents.

Further, the study could deal with the factors causing the occurrence of certain types of risks at conceptual level on the basis of the prevailing theories and related research literature. An empirical study to substantiate that ‘spatial distance’ is the main cause behind the risk factor, ‘inadequate communication about end user requirements’ or ‘large project scope’ leads to ‘requirements conflicts among multiple product owners’ is advised for addressing the threat to the internal validity of this research. The present work addressed this at conceptual level and could not be extended to the substantiation of such cause and effect relationship between the independent factors and the risks.

Another area of concern has been the validity of constructs being used in the study. Risk categories, areas and factors were defined and operational definitions were given in the questionnaire. Pilot test had helped in inclusion and exclusion of risks and further refinement of their meaning. The data was collected from majority of respondents during face-to-face interactions, in which, operational definitions were explained and the meaning of ‘impact’ on project was conveyed. However, quantitative methods [36] for establishing construct validity could not be used because of the paucity of the time and the inclination of the limited number of respondents for participation in this study. Future research can address these limitations by having an extensive pilot study with a concerted effort for removing the threats to the construct validity caused by the possibility of hypothesis guessing, evaluation apprehension, researchers’ bias and other such factors.

We faced significant resistance from the respondents for filling the second part of the questionnaire which was dealing with rating data for risk management methods. We decided to collect the rating data from the respondents who had provided us with the ranking of risk factors, since they were aware of the context of our study. As a result, the number of respondents who could help us for filling the second part of the questionnaire was less (19 respondents). Due to the time constraints, we had to stop the data collection of rating data and proceeded with the analysis. Consequently, average rating has been found to be statistically significant but spurious on account of small sample size. However, this was indicative of the possibility of true representation of the population by the sample used in the study. This can be verified further by collecting data from large sample. Although, we cannot generalize the findings related to risk management methods, but still they do give insights on the suitable methods which can be used by the practitioners.

Due to time and budget constraints, majority of data was collected from companies in India. This may have an impact on the extent of generalization and hence the external validity [24]. We tried to reduce the impact by collecting data from 5 different cities within India and also collecting some response from companies outside India. There were few respondents (4 out of 39) which belonged to countries other than India including U.K, U.S.A and Japan.

8. Conclusions and future scope

This study is an extension of an exploratory work on identification of the risk factors, which impact distributed agile development projects and risk management methods that are used in practice to control the risks [40]. The risk factors were grouped to form risk areas and high level risk categories. Risk categories includes, ‘Group Awareness’, ‘Software Development Life Cycle’, ‘Project Management’, ‘External Stakeholder Collaboration’, ‘Technology Setup’ [40] These risk categories were then related to the Lewitt’s model of organizational change, in order to give an organizational perspective to our findings.

We present the results of the descriptive study in which we obtained the ranking of the identified risk factors and rating of the corresponding risk management methods. The statistical analysis led to the identification of the most important risk categories, risk areas and risk factors. It was observed that the first two most important risk categories were ‘Group Awareness’ and ‘External Stakeholder Collaboration’. Both these categories have risks primarily induced by the spatial distance and the difference in work/development culture between the stakeholders.

The results show that the ‘spatial distance’ between the project stakeholders causes the maximum number of risks in DAD projects. Moreover, the risk factors which are having higher ranks are also the ones which are related to the spatial distance. ‘Differences in work/development culture’ is found to cause important risks which arise due to the uncommon processes, standards or understanding of definition of done of the product features. ‘Large project scope’ causes moderately important risk factors which occur when agile methods, which are actually intended for small project teams, are used for developing large projects.
comprising of geographically separated multiple teams. ‘Temporal distance’ causes less important risks since the use of contemporary communication tools and agile practices has helped to reduce the challenges caused by time zone differences amongst the teams. Very few and least important risks are caused due to the ‘Language barrier’ since agile development focuses on frequent team interactions which reduces the socio-cultural differences.

Hence, DAD teams need to adopt practices to reduce the impact of spatial distance between the stakeholders. Besides geographic dispersion, other properties including work/development culture, large project scope, temporal distance, language barrier, which have lesser impact on DAD project must also be considered for controlling the risks.

In this paper, we suggest the risk management practices which are most frequently being used by the practitioners to control the risks in DAD projects. The practices suggested by this study, focuses on reducing the communication gap between the teams and the other stakeholders including the customers, third parties and vendors. Besides that, focusing on following good engineering practices like strong release planning, continuous integration, pair programming, backlog management, having a common definition of done becomes absolutely necessary for DAD team. Distributed teams should be preferably long-lived feature teams with minimum inter-dependencies, which nurture trust, team-spirit and productivity. The recommendations for risk management are based on the views of the practitioners and hence may not represent the ideal set of practices for managing the identified risk.

The risk management framework was validated by implementing a part of it in three different multinational IT consultancy companies and it proved to be beneficial for identifying and subsequently controlling the risks in DAD projects. Although, the risk management framework developed in this study was perceived as beneficial for managing risks in DAD projects, but further researches should be performed to minimize the threats to validity of this research.

The comprehensiveness of the study opens many new avenues in terms of validation of the findings in the same and other geographies for strengthening generalization of results, deeper exploration of risk factors and their causes and formulation of more scientific and formal models to address the risks in DAD projects.

In this paper, we focus on the risk management methods that are more frequently used by the practitioners to control the risks in DAD projects. Although, the suggested risk management methods are used more frequently in practice, they may not represent ideal methods for controlling risks in DAD projects. Identifying and suggesting risk management methods that are good with respect to agile practices for DAD projects could be more useful for the practitioners.

Acknowledgments
We would like to thank all the participants from the corporate who participated in the interview session and gave their valuable inputs for this research work. This work has been funded by Project Management Institute Educational Foundation, PMI®E (USA).

Supplementary materials
Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.infsof.2016.12.005.

References