UNIVERSITY OF CALIFORNIA,
IRVINE

Features of Technologies That Make Remote Work, Work

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Software Engineering

by

Tian Guan

Thesis Committee:
Professor David Redmiles, Chair
Associate Professor James A. Jones
Associate Professor of Teaching Hadar Ziv

2021

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

A huge amount of thanks goes to my thesis advisor David Redmiles. Thank you for gently guiding me in this whole research as well as the input and insights which gave me great support when I am working on this research. Thanks to my committee members James Jones and Hadar Ziv for the great suggestions and insights they gave on this study. I am also grateful for the support and help from Zhendong Wang for the suggestions in this study and for reviewing my thesis. Thanks to the interviewees who provide valuable data in the data collection process.

ABSTRACT OF THE THESIS

Features of Technologies That Make Remote Work, Work

By

Tian Guan

Master of Science in Software Engineering

University of California, Irvine, 2021

Professor David Redmiles, Chair

With the development of a wide variety of collaborative software research and tools, the distributed collaboration experience has made giant strides. Today, people conduct remote collaboration on an unprecedented scale. When technologies become our only means to communicate and collaborate without being able to be in a collocated office, it is time for us to take a closer look at how well current research and technology are helping people work together. As a previous software developer who worked on collaborative software development, and a current researcher, I am curious about what features of existing collaborative software tools are actually supporting remote collaboration. To learn this end, I first examined the research literature to learn the problems that researchers are facing and solutions they came up with; second, I conducted interviews to learn about how companies are utilizing technologies to bring people together for work, and their experience of using these technologies; third, I did an internship to gather data from participant observation; and, finally, assessed my final results in discussions with other researchers.

I found that some features of collaborative software are heavily relied on in distance work, and might be continued to be used even after people return to work in collocated offices. Further, the imitation of the face-to-face experience as some software is trying to do will always be inferior to the collocated experience. Such imitation may be immediately aban-

doned as soon as people can work together in collocated offices. We may need to pay more attention to collaborative software features that people would still be willing to use even when collocated. I also found some features are not always working as people expected. Unveiling and understanding the occasions when technology breaks down may shed some light on future collaborative software research and tools development.

Chapter 1

Introduction

Many efforts of studying and facilitating remote coordination focus on reproducing the same richness of synchronous interaction and information using technologies as if we were there [35]. This direction of solution can be viewed as an effort to use an imitation as a replacement for the face-to-face experience. For a person with a broken leg, the crutches can help people hobble around until they are back in shape [35]. The effort of imitations and reproducing the same visual and auditory experience using technological tools may orient us towards focusing on developing crutch-like tools. As technologies developed, many disadvantages of remote work have been alleviated. Video conference software like Zoom largely increases the efficiency and effectiveness of distant communication compared with audio-only media; communication platform like Slack provides various ways for online communication; 3D holographic devices can bring more details of facial impression and body languages, and create a more real sense of being in the same physical room; 5G and mobile devices, internet speed, high-resolution images, and the easiness of access to powerful computational devices nowadays. A potential instinct to view the progress we made today is to think that all disadvantages will be eliminated and there will be no differences in collaboration between remote and collocated colleagues someday in the future. Researchers, however, suggested

that the imitation brought by technologies may never be "close enough" to the face-to-face experience [35] and distance still matters today [44].

When people are collaborating face-to-face, the richness of the nuanced details conveyed and received is beyond what can be imitated. Also, people rely on context to figure out how they should behave, convince, negotiate, communicate through body language and emotional expression based on their past experience. In contrast, those kinds of information may not be easily observed in audio and video contexts, and the conversation and interaction tend to be less vivid in virtual communication. The passive information acquisition is strangled in the distributed collaboration environment either. For a newcomer or someone we have never been working with, trust can still be built by unintentional observation of how the collocated colleagues behave and interact with others in a collocated office. This trust-building process often happens without the conscious realization of the observers and people who are observed. Next time when we need to talk with the "stranger", their behavior pattern is somewhat already in our minds based on our passive observation experience. You may already trust this "stranger" because you recall that the stranger's group members were always going to ask for help from them whenever things went wrong. You may also know whom you should avoid consulting based on the past unconscious impression that they may always be busy. In distributed collaboration practices, however, there are fewer opportunities for people to obtain this type of awareness. In some of the interviews I conducted in this study, some interviewees mentioned that people only talk with each other purposefully when they need to discuss a certain topic or in a short group meeting. It is also nearly impractical to ask employees to open the camera and virtually be there eight hours a day. When you need to interact with someone with whom you have never worked, there's nearly nothing you could predict in your upcoming collaboration. Small mistakes may be mistakenly attributed to work attitude or ability issues due to the lack of passive knowing each other.

Besides imitating and reproducing the same experience of face-to-face collaboration, another

way to study remote collaboration is to explore what could go beyond being there. Humans have developed various mechanisms for effective physical proximate collaboration and communication, but there are also limitations. As some of the leading technology companies announce indefinite working from home policy, it is logical to anticipate an unprecedented scale of work style revolution. If we take a look at how remote work is providing extra dimensions of capabilities facilitating human collaboration that provide not only an imitation of social presence but also go beyond what can be done in face-to-face collaboration, we may get a better understanding of what challenges we're facing in the near future and provide some inspirations for future research.

From the 1980s, researchers and system builders in CSCW put great effort into utilizing technologies to solve the challenges in distance work. Today, giant strides in information technologies have been made. My motivation for this study starts from three years of face-to-face working experience as a software engineer who works in a collocated office environment. The projects I was working on were project management software which helps facilitate collaboration across all software development life cycles. As a previous collaborative software developer and a current student researcher, I am curious about 1. How does distance still matter in today's world? 2. How do contemporary technologies help people conduct distributed collaboration? These questions and my motivations lead me to three research questions:

1. What features of existing collaborative software support collaboration?
2. Which of those features and tools do practitioners believe would still be used after going back to the office?
3. What features would practitioners like to add to help distributed collaboration?

With the research questions in mind, I started to look into the literature to understand better the background and context in this area. When reviewing the literature, I started

to understand how researchers in CSCW (Computer Supported Cooperative Work) describe the problems and challenges in distributed collaboration and previous research addressed the challenges. In order to learn the real-world situations of technologies utilization, I conducted interviews with people from several companies who work in remote mode to understand how companies are using technologies to support remote work. As another part of the data collection process, I did an internship to obtain data via participant observation. Before composing the thesis, I analyzed what I found and present my analysis to researchers in this area to gather more insights as well.

This study is conducted in the following phases: 1. I started by reviewing my experience of working as a software engineer, participating in collaboration with collocated as well as remote colleagues, and developing software tools facilitating remote coordination. 2. I reviewed previous research and how related research has evolved for the past decades, understanding the critical challenges of distributed collaboration and the current focus of related research. 3. I conducted interviews with current software engineers in the software industry to gather data from people who work remotely, with some of those who used to work in a pure face-to-face office environment. 4. I also gathered data by observing while doing a remote internship with teammates located in different cities and countries. 5. I revisited the literature and analyze the data I collected. 6. I discussed my analysis result with researchers in this area to gather insights. The analysis and results are presented in chapter 6 and 8.

Chapter 2

Research Method

In my research, I first started to look back to my experience when I worked as a software engineer. I was developing features and tools to help facilitate collaboration. Most of the features I was working on was to develop more functionalities so people can text to each other, create backlog and tasks and invite assignee, log work hours on the timesheet to track progress, automatic notification when there is new progress to coordinate team members of different roles to work together. I was developing a project management tool and plugins to integrate with JIRA and Confluence. This part of past experience is discussed in chapter 3.

Second, I reviewed the literature to understand the research efforts in this area. I learned how researchers and system builders are defining the challenges in distance work and the solutions they came up with. Many valuable means to address the challenges in distance were proposed and invented including software tools (e.g. [64]), paradigms (e.g. [53]), and research methods (e.g. [62]).

Third, with all the research results, I wondered what is actually being adopted by companies to support their employees' distance work and how well do the technologies support distance work. Thus, I conducted interviews with employees who work in companies located in North

America. As another part of my data collection process, I did an internship to collect data via participant observation. The result of this data collection is presented in chapter 5.

Finally, I analyzed the results and correlated them back to the literature. I discussed my results and analysis with researchers in this area to obtain insights and suggestions. After the second round of analysis, I presented my research results to professors of my thesis committee.

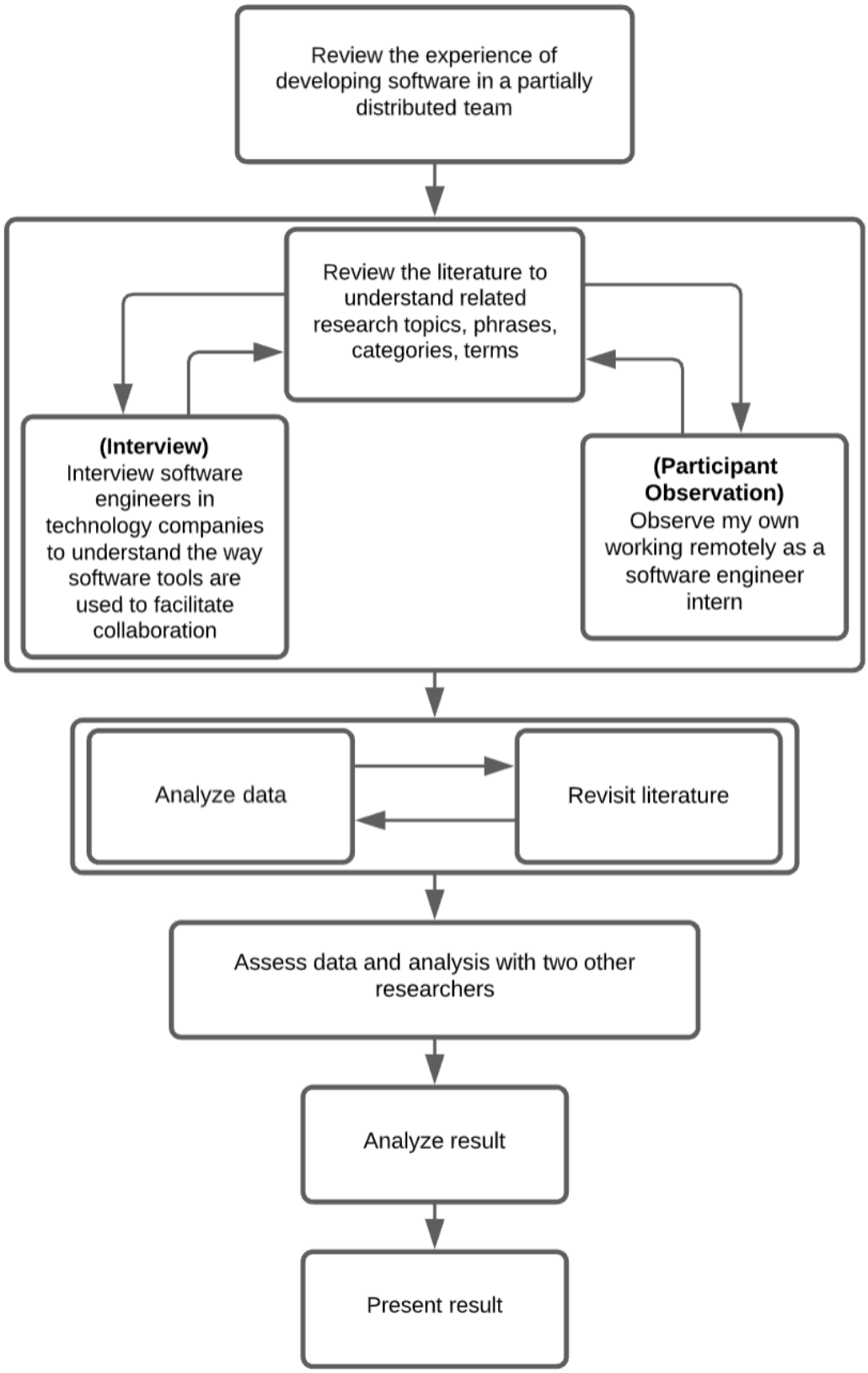

Figure 2.1: Research Process

Chapter 3

Experience and Motivation

3.1 Past Experience Working in an Face-to-face Environment

As a software engineer on a partially distributed team in a globally distributed company, I participated in both collocated teamwork and distributed collaboration. Because some team members spread across continents, we adopted and adapted some agile practices for more efficient communication, collaboration, and knowledge sharing between team members. As a member of the Tools Team, we also developed, adapted, used, and analyzed software tools to help solve problems brought by the lack of in-person collaboration opportunities.

I was in a team of 20, and most of my team members were located in three cities of two countries. We adopted agile practices as a framework for team and development management. A particular time was set up for both time zones and we hold daily stand-up meetings, weekly meetings involving stakeholders, events-related meetings via corporate business phone line services, and video conference software like Zoom. We developed a Project Management

tool and integrated it with customized JIRA and Confluence to provide a sharing workplace for team members. We hold regular scrum rituals to facilitate communication among team members, and also informal chat time to make up for the lack of informal communication opportunities with remote team members. Online chatting platforms are used for formal and informal conversations. Emails, comments, and notifications on JIRA and Confluence are also used for asynchronous communicating. Git is used as a version control system. The most common ways of communication when collaborating remotely are by emails, phone calls, video conferences, status changes, and information presented on JIRA and Confluence.

3.1.1 Challenges

One of the biggest challenges for distributed collaboration is communication. Because of the lack of water cooler chat opportunities, getting familiar with remote team members was slower than collocated ones. The trust-building process took longer, and the unfamiliarity makes the first contact and collaboration less likely to happen. Sometimes, people will turn to ask whomever they've already known, or ask them to relay messages. Frictions or unpleasant collaboration experiences usually made future collaboration harder. Without previous impressions built by observing and seeing each other unconsciously as in collocated offices, trust-building is slow and common mistakes can sometimes be attributed to people themselves, instead of accidental incidents.

Remote members may feel isolated and have less team unity, which leads to higher psychological costs for collaboration and friction resolving. Camaraderie building among distributed team members becomes more challenging too. Biweekly Team Building activities usually increase familiarity and camaraderie among collocated team members by doing games and storytelling together on weekends. But for remote members, this kind of opportunity is only possible when they travel to physical offices yearly.

Knowledge sharing is a great opportunity for team members to know each other and usually leads to later collaboration opportunities. We host book sharing, new project sharing, and technical trends sharing regularly. During this process, members get to know each other about personalities, skill sets, and project domains. People know whom to contact when they need help and what to expect when talking with others.

Conversations about decision making, design discussion, project planning, and technical solutions discussion need to be documented and synchronized up with remote members. Timely knowledge sharing should be ensured for efficient collaboration.

Schedules before deadlines could be difficult to coordinate due to asynchronous communication. Delays are usually expected. Coordinating a project is more challenging and time-consuming.

3.1.2 Our previous solutions

How did we overcome those difficulties using software tools? Our self-developed system is integrated with customized JIRA. Here I listed several features of JIRA that we use to coordinate collaboration. The content on each screen is only for illustration usage.

Show my activities on the Issue screen

Once an issue is assigned to me by selecting my name in the 'Assignee field', either by my manager or by myself, others can see that I take charge of this task (expert/owner/knowledge location, be aware of my tasks on hand). Others can see my project/ tasks participation as in the next slide.

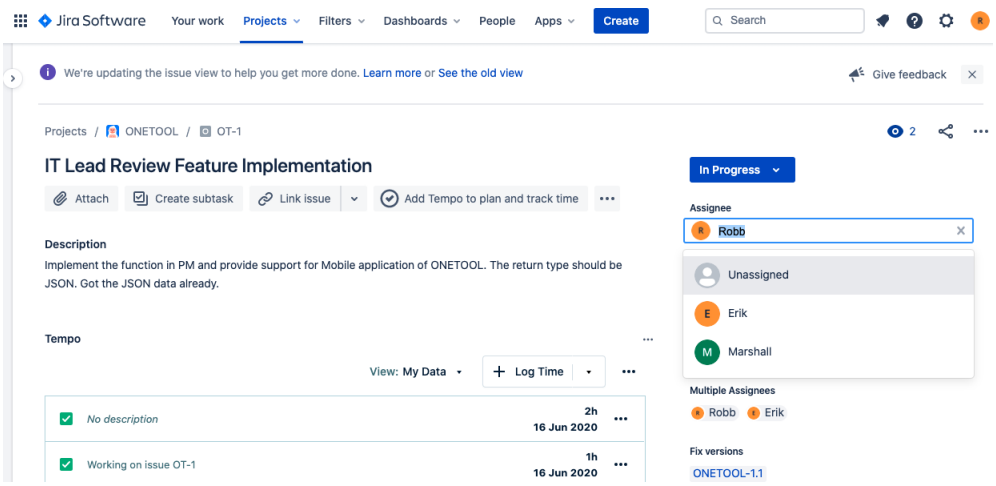

Figure 3.1: Show my activities on the Issue screen

Monitor others on the Dashboard screen

The Project Participation pie chart can be configured to visualize participation based on other metrics, e.g. Issue Type, Project, Sprint, Status, Multiple Assignees. Currently, this pie chart shows the participation information based on the 'Assignee' and in the project 'PM'. Since the Dashboard could be the first screen when you log in, you can configure it to show the information that is important to you by adding several gadgets on this screen.

Show my activities on the Issue screen

Log my timesheet and add some descriptions of my work. This can be used to track time costs and progress.

Monitor others' activities on the Issue screen

Observe information including the collaborators on this task, how many hours they spent, description of their work, and when did they work on this task.

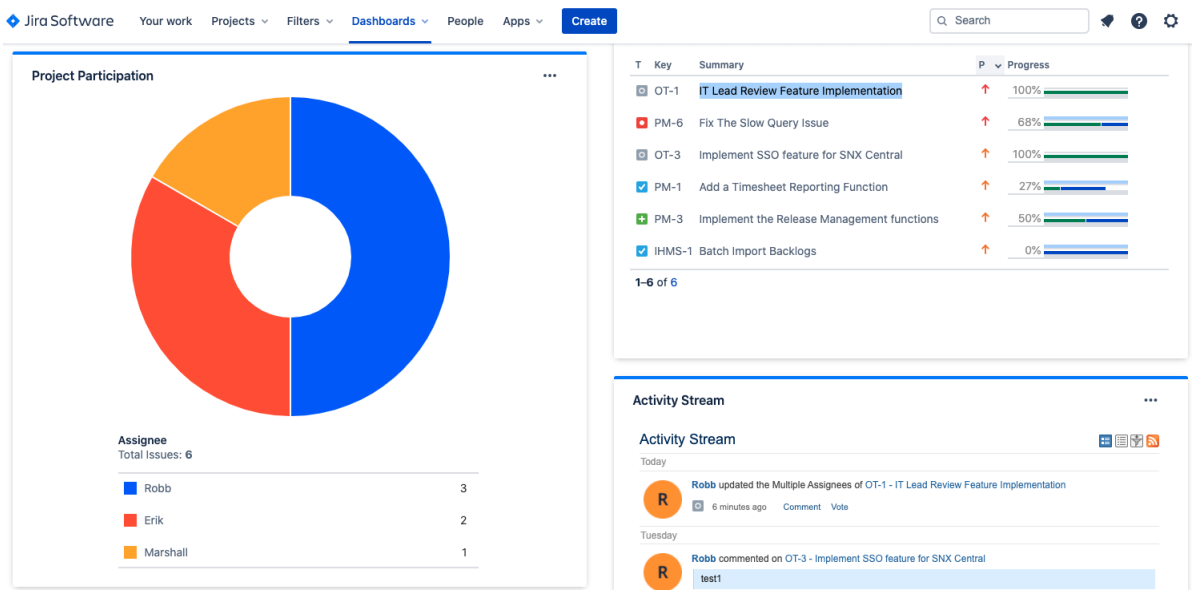

Figure 3.2: Monitor others on the Dashboard screen

Show my activities and monitor others' on the Issue screen

Each activity that was made on this issue will be recorded and shown in the 'History' tab with the time and the user name who made the change.

Show my activities and monitor others' on the Issue screen

The comment section can show the context and background information to others who wanted to know the design history, or information like what happened to this issue, later when they try to understand how this feature was implemented. Others can know when and why I was working on this issue, and I can also know who was working on this.

Monitor others' activities on the Dashboard screen

By adding an Activity Stream gadget on the Dashboard, you can have an awareness of what's going on whenever you log in and see this screen. Filters can be used to keep the information

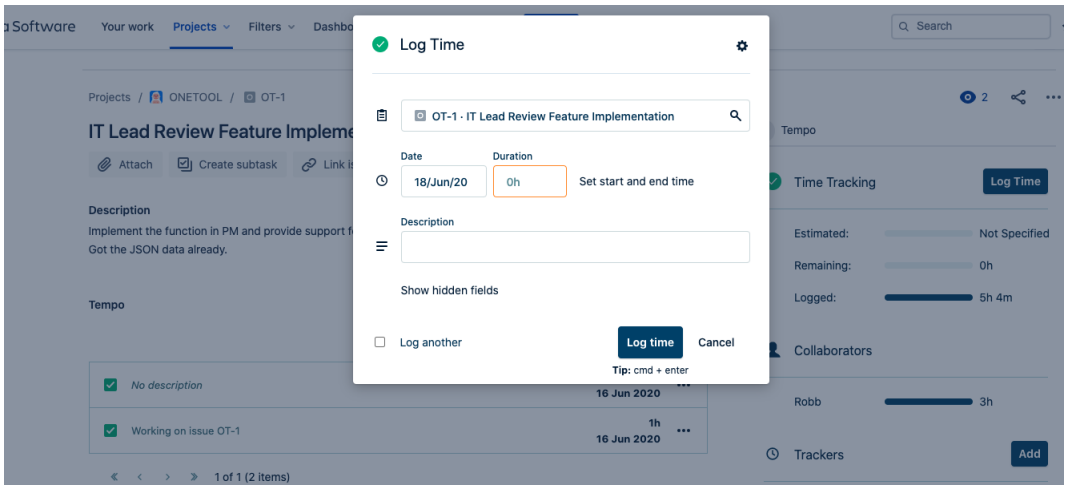

Figure 3.3: Show my activities on the Issue screen

we particularly care about. For example, I want to know Mia and Jackson's activities in the project 'PM' and the only Issue Type that I care about is 'New Feature', then I can filter out other information. A sample Dashboard configuration that I (as a Software Engineer) will see at first when I log in to JIRA.

Monitor others on the Dashboard screen

If I want to get a sense of how a sprint is going on, I can also add a Sprint Health Gadget on my Dashboard screen. By having a glance, I will be aware of the progress of this sprint, and the participation.

Monitor others on the Dashboard screen

If I want to know some specific information based on some criteria, I can use the Filters function to create a filter and then show the results on my Dashboard, I can get an awareness of what's going on by taking a look at my Dashboard together with other information that was introduced in the other slides. Now in this example, I want to get an issue list in project

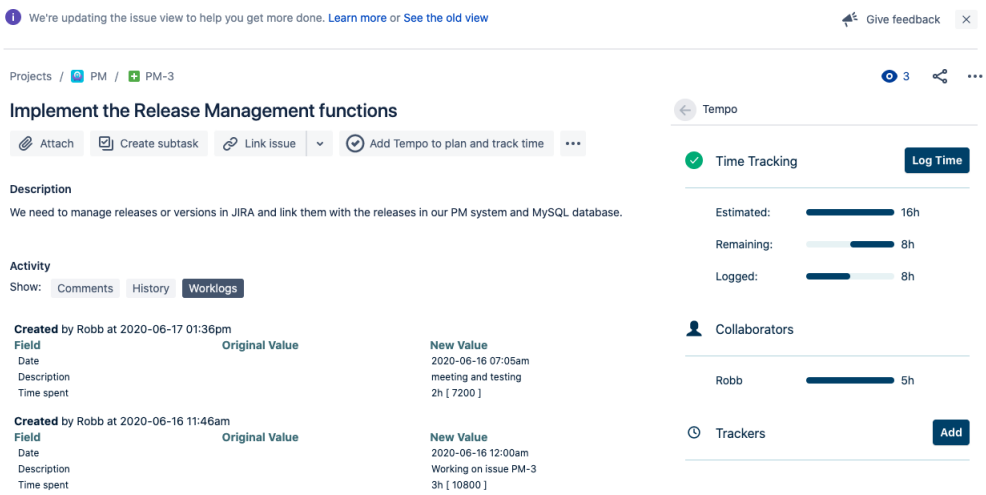

Figure 3.4: Monitor others' activities on the Issue screen

PM, issue type is New Feature, status is In Progress, you can also use JQL to do an advanced search.

Monitor others on the Dashboard screen

Now, add a gadget to show the result of my Filter on my Dashboard. This enables you to show the issues that matter to you, e.g. what are the ongoing tasks that my team is working on? What are the to-do lists? What is due next week?

Monitor others on the Dashboard screen

If you are a manager, you can add many other gadgets on the Dashboard screen to get a sense of the progress of your team, e.g. the average number of days issues have been unresolved or have been in status, etc.

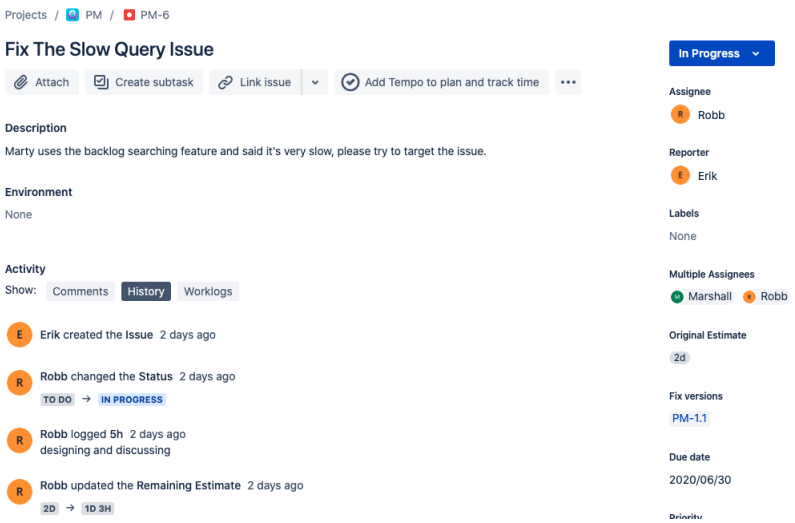

Figure 3.5: Show my activities and monitor others' on the Issue screen

Monitor others on the Project board screen

on the Project board or Kanban board, you can get a sense of the general progress in a release. To-Do list, In Progress list, Done list. You can also show the result of your Filter.

Monitor others on the People screen

You can see what your teammates are working on by going into the People tab, then clicking to check a team. You can also click on a team member's name to check what he is recently working with.

Monitor others on the People screen

Now you can see your teammate Marshall's recent works. Worked on lists, projects he's been working on, and people he's been working with, his job title, department, and location (time zone).

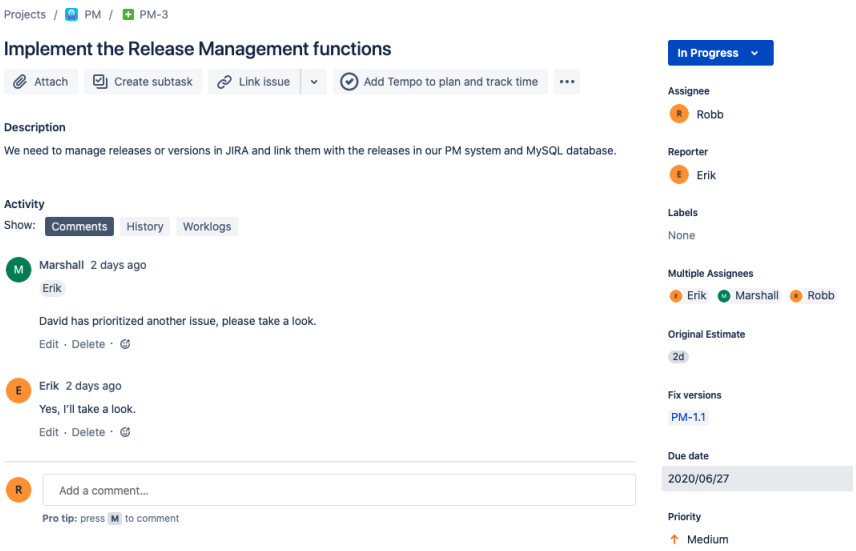

Figure 3.6: Show my activities and monitor others' on the Issue screen

3.2 Analysis from Background Study

An analysis of the limitations of collocated collaboration and what extra capabilities current technologies have brought to us when we're collaborating remotely could be a good starting point.

Synchronicity

Emails provide an asynchronous way for people to communicate, this allows flexibility for responses that could not be able to be given immediately. In face-to-face communications, responses are usually expected immediately which may not always be a good thing. A manager may be asked many questions in person when sitting in an office, a question is supposed to be answered within seconds when it is asked in person. Some questions may be answered better if the person who is asked can look them up before answering, which could be awkward in a face-to-face scenario sometimes. Because the in-person conversation is ephemeral, some questions may be asked repeatedly because previous conversations and

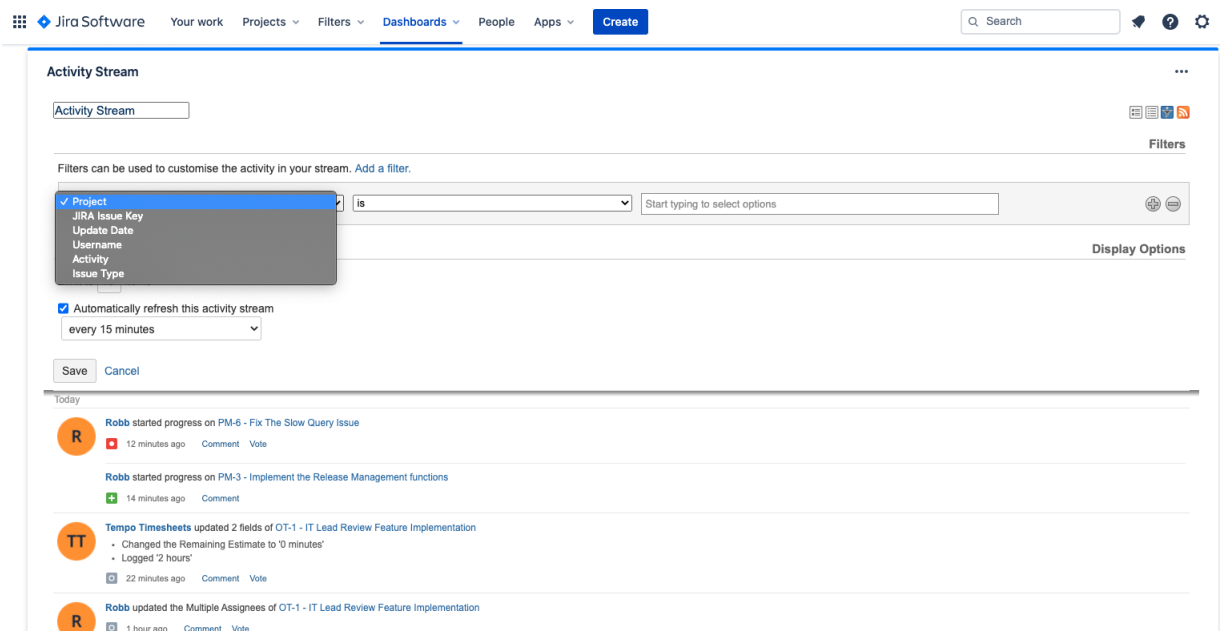

Figure 3.7: Monitor others' activities on the Dashboard screen

knowledge cannot be stored and searched later. However, when working remotely, many of the impromptu conversations may happen in an online communication platform, e.g. Slack. Although this brings inefficiencies for burning questions that need immediate responses, the content of the communication could be recorded and looked up by keywords later if someday a similar problem appears again. Also, team members in other time zones can get to know what happened when they are online. This asynchronous way of information exchanges allows flexibility of responding time and brings an efficient way to convey information despite limitations of time and space.

Visibility

Without being able to see each other in person, the nuanced details of communication are lost and it brings difficulties to letting people see what they are talking about, e.g. a line of code. The lack of opportunities to see each other in person may also cause impacts on familiarity and trust-building. However, some awareness and knowledge exchange and

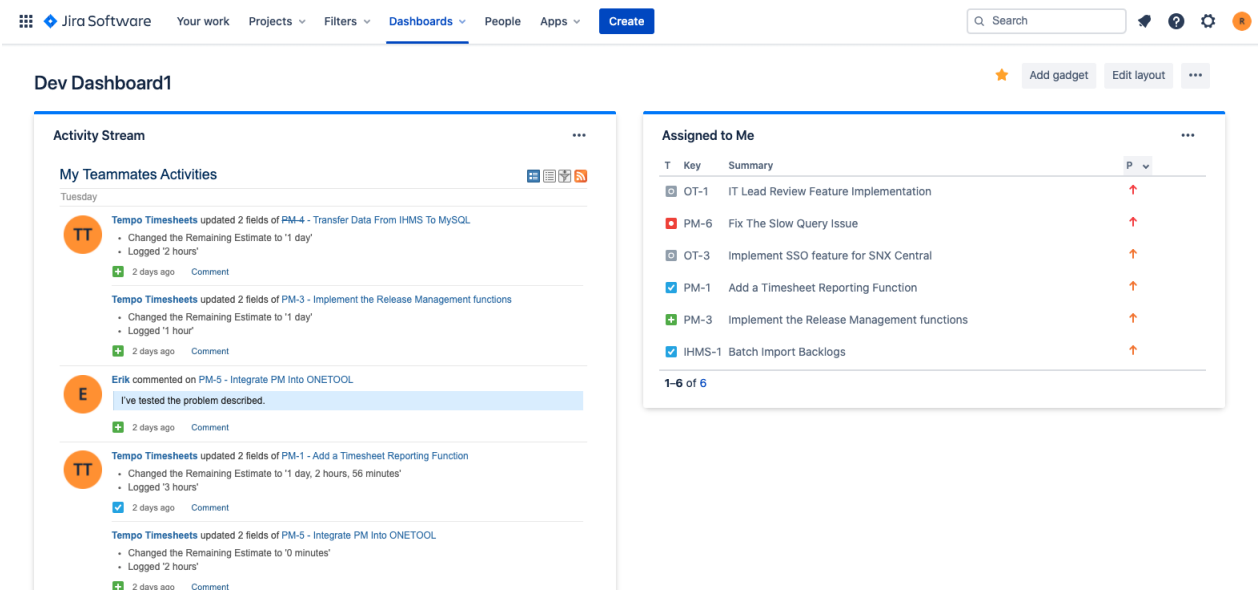

Figure 3.8: Sample Dashboard configuration

discussion can be better obtained in an anonymous environment. Platforms like Team Blind provide a great way for people to communicate voluntarily about something that they would be reluctant to talk about in a face-to-face environment. This extra channel of information exchanging may help managers understand what actually happened in a conflict, and help members understand others' real opinions about a topic.

Space

Remote collaboration breaks the boundaries of having the collaboration conducted in a shared limited space. Theoretically, virtual teams can accommodate an unlimited number of team members. And most importantly, they do not need to relocate to collaborate. People can work on a task together by staying in different geographic regions, as long as they work out a time frame that's workable for different time zones. Virtual teams benefit from being able to bring in people with different expertise and skill-sets from a global talent pool. The flexibility of no geographic distribution requirement for team members is another extra

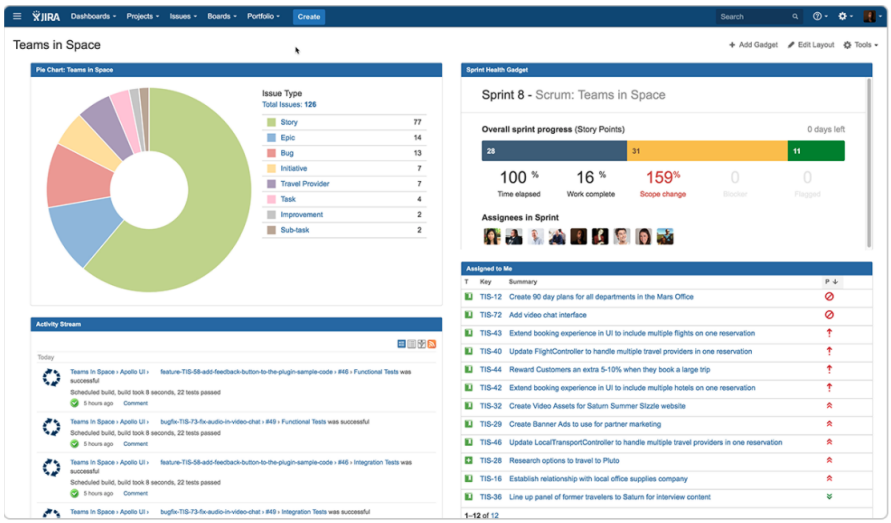

Figure 3.9: Monitor others on the Dashboard screen

advantage of remote collaboration to collocated work styles. Teams across geographies also allow for a 24-hour workday.

Communication

Technologies have made very successful progress in terms of imitating the in-person communication we conduct face to face using high-speed and high available internet, high-resolution and high-quality videos and audio telecommunication technologies, 3-D holographic technologies, etc. Annotations on a shared screen, emojis, remote computer control increases the richness of interaction that was lost compared to in-person collaboration. A logical expectation is to have more realistic media to convey detailed information more efficiently in the future. But we should think about what capabilities or experiences make remote collaboration so unique, and when it is superior to in-person collaboration.

One of the advantages is that every activity can be recorded easily and reviewed whenever necessary in the future. In a collocated office environment, a design plan meeting usually happens in a meeting room with a whiteboard. The diagram worked out collaboratively

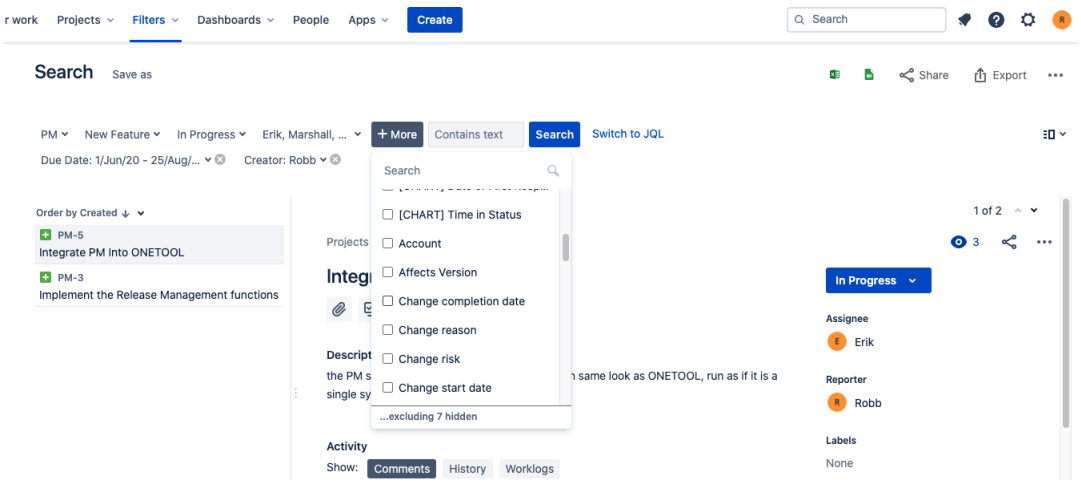

Figure 3.10: Monitor others on the Dashboard screen

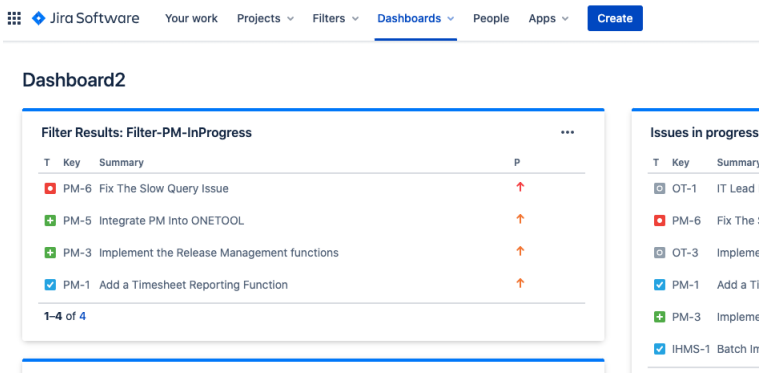

Figure 3.11: Monitor others on the Dashboard screen

and the important meeting notes may or may not be documented by attendees because of false confidence of "everyone who should know is already here and will remember what was discussed", which is usually not the case in reality. However, distributed teams have to document their design diagram and important meeting notes so those team members who are in other time zones will know the discussion result. The conversation can be recorded easily by clicking on the "Record" button of a video conference software instead of setting up a camera in a meeting room; the diagram will be drawn on an online whiteboard application and can be naturally saved as a file and spread across team instead of being wiped out on a physical whiteboard easily.

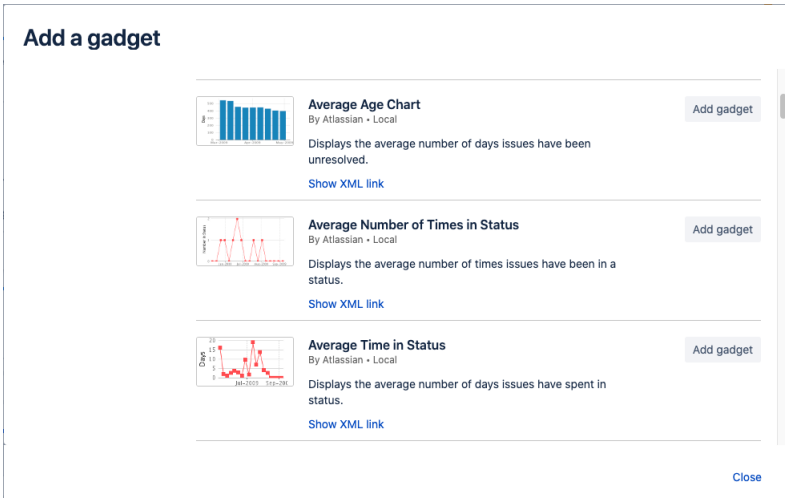

Figure 3.12: Monitor others on the Dashboard screen

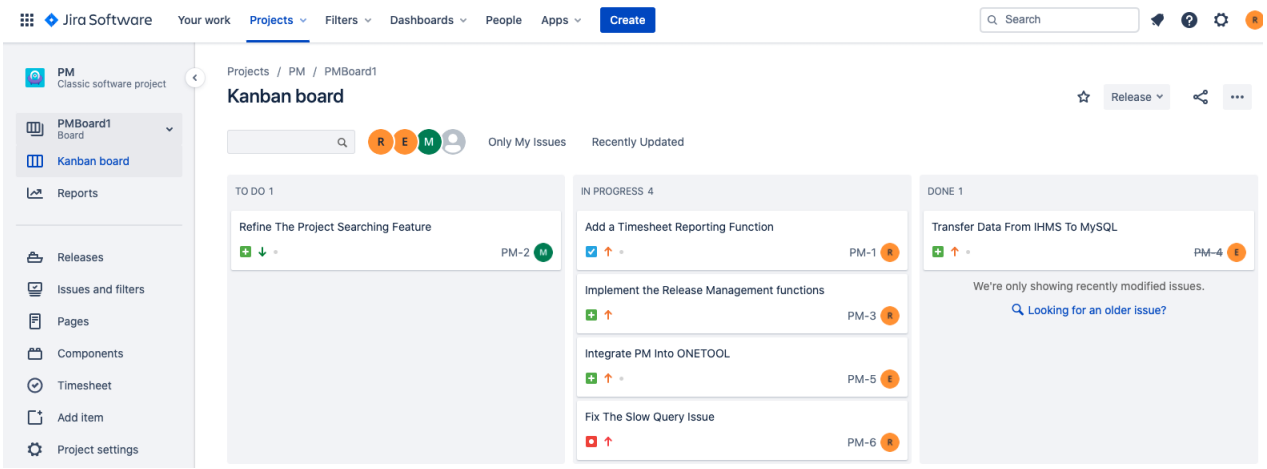

Figure 3.13: Monitor others on the Project board screen

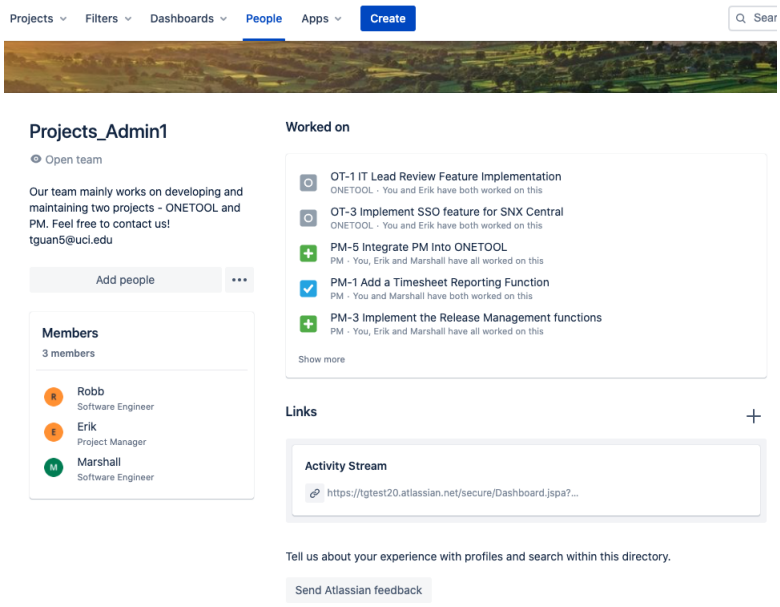

Figure 3.14: Monitor others on the Project board screen

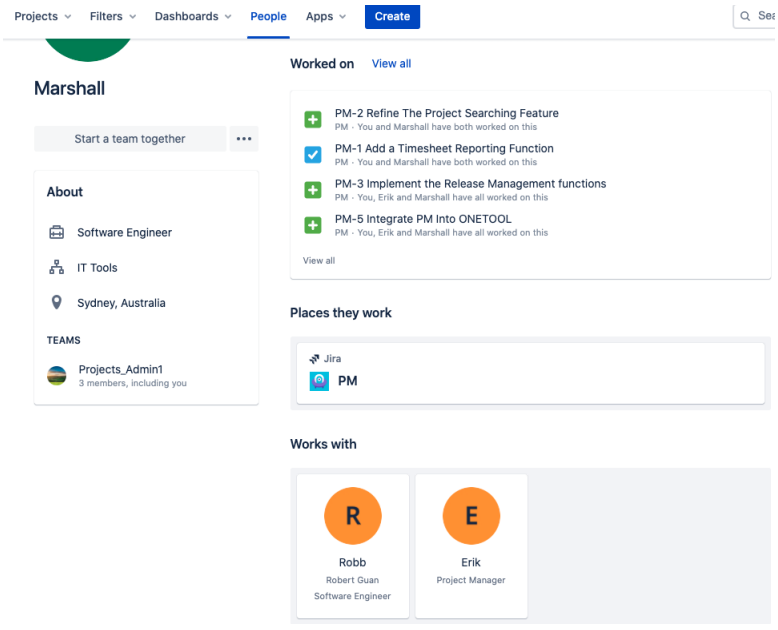

Figure 3.15: Monitor others on the People screen

Chapter 4

Literature Review

In order to integrate people in the right way to make teamwork efficient and productive, one of the most fundamental problems to solve is coordination. When it comes to software engineering, there seem to be more challenges due to the reasons including invisibility and constant change of software as Brooks [9] pointed long ago. When people work in a collocated environment, there are many coordination mechanisms people came up with naturally which make daily coordination during work viable and sufficient enough. However, when people work together remotely, those mechanisms we take for granted are gone. For decades, researchers in this area explored the challenges in distance work, pointed out the important factors in distance work, and developed a wide range of software tools as well as methods to help make distance work work and more efficient. In this chapter, I am going to present current research results about distance work by reviewing current literature.

4.1 Challenges in Distance Work

With the development of groupware technologies, people expect to communicate more efficiently and easier [44]. Global teams become more common. four social-technical conditions are suggested in Olson and Olson's work [44]: common ground, coupling of work, collaboration readiness, and collaboration technology readiness. They believe albeit the possible technology readiness and advanced software tools we might be able to use in the future, distance still matters. The differences in a local physical context, time zones, culture, and language continue to be issues as long as the existence of distance. Key characteristics of collocated synchronous interactions include 1. Rapid feedback; 2. Multiple channels; 3. Personal information; 4. Nuanced information; 5. Shared local context; 6. Informal "hall" time before and after; 7. Coreference; 8. Individual control; 9. Implicit cues; 10. The Spatiality of reference. These key characteristics of collocated synchronous interactions have long been what researchers and software developers were focusing on.

There are also other difficulties for distributed teams, including 1. Blind and invisible; 2. Time-zone differences; 3. Crossing institutional or cultural boundaries [47]. Researchers address these difficulties based on different situations. Technologies become an integral part of remote work support tools. Classifications of technologies have been proposed (e.g. [47]). These authors classify technologies into Communication Tools, Coordination Tools, Information Repositories, and Computational Infrastructure.

Despite the wide range of software tools we have today, there are still stubborn challenges we may need to face in distance work: 1. Out of sight, out of mind; 2. Trust; 3. Other cultures; 4. Different time zones. Also, collaborations over distance often face the challenges of the lack of ample and subtle interactions we take for granted in collocated teamwork. Some research shows a strong positive correlation between delay in cross-site work and the degree to which remote workers help each other when work-load is heavy [32]. In the collocated environment,

there are many factors helping people coordinate work with each other. Context, peripheral awareness, incidental properties of artifacts, and informal talks. In distance work, instead, these are all gone and have to be either obtained by technologies or overcame by some means.

Because of the belief in the efficacy of imitating face-to-face communication, many software tools were developed to provide a higher richness of interaction in distance work. However, some research suggested that imitation may always provide an inferior experience compared with face-to-face experience [35]. We as humans have developed a variety of mechanisms to help in social interactions and they meet our needs most of the time when it comes to initiating and maintaining friendships, discussing, planning, and negotiating. One must take into account the technical dependencies among engineering tasks when we are thinking about the social-technical phenomenon in software engineering [31]. Another key challenge faced by distance workers is the lack of sense of connection to the rest of the team [43]. And the lack of connections between colleagues contributes to the hardness of establishing trust. In geographically dispersed collaborations, trust becomes extremely difficult to establish. Trust affects distributed team performance in two important ways: 1. through the traditional view of effects on performance; and 2. via affecting moderating effects on other determinants [20].

Global teams have to manage various aspects of software development such as designing, development, and maintenance, which made software development a non-trivial task, especially when it comes to global software development. Five categories have been classified for the challenges: team, control and coordination difficulty, loss of communication richness, loss of team spirit, and cultural differences [67]. An integral part of software engineering is developing a shared understanding of multiple artifacts [66]. Maintaining "mutual knowledge" is a fundamental problem for geographically dispersed collaboration and traces the results of failures to do so [13].

Computer Supported Cooperative Work (CSCW) emerged in the 1980s from shared interests

among software developers and researchers. The name CSCW had controversy and Computer Supported Collaboration (CSC) was also used [24]. Since then, researchers started to analyze problems that existed in global software development. Some suggest that there is an inherent gap between the social requirement of CSCW and the supporting technologies [1]. Factors that contribute to the problems in distributed collaboration are the lack of awareness and informal communication [53]. Although many channels existed, the primary channel for communication of awareness is text-based channel [29]. Researchers explored the problems and proposed many solutions. Dr. Redmiles proposed Continuous Collaboration Paradigm [53] Some researchers also suggested design principles for software tools to support distributed teams focusing on trust factors, collaborative traces, and visual representations [63]. Extensive research has been done on trust, common ground, and social context [8]. Perhaps the problems of technologies supporting distance work are not due to technologies. Grudin suggests that many failures are due to the misunderstandings of groupware supporting distance work [25]. Some also say analyzing the problems of technologies supporting distance work should not only look at one tool once at a time. Instead, they need to consider all aspects of work. They found that groupware affected people's commitments and the nature of the work distribution [48]. Discontinuities in geography, time zone, organization, national culture, work practices, and technology could be the root cause the problems in distance work [12][39][38].

Another challenge in utilizing technologies to support remote work is the effort and time needed for training. Over-worked workers are reluctant to spend time to learn new software tools. As a result, they often rely on primitive but inefficient methods to communicate and collaborate [65]. Distance also changed some aspects of the research process. For example, research groups have fewer meetings. However, more sharing of data collection activities present [57].

As a result of all these challenges in remote work, collocated teams could be twice as pro-

ductive as remote teams [49]. The reason could be the cost of coordinating developers in different geographical locations [18][33]. Grudin also suggests that one of the factors that contribute to the failure of supporting the application is the disparity of who will benefit from an application and who must do additional work to support it [23].

4.2 Important Factors in Distance Work Success

Shared information space seems to be an integral part of distributed virtual environment [40]. Also, trust, awareness, media richness, and management are also considered fundamental factors in distance work success.

Trust

Trust is deemed as a determining factor to make interdependent actors work together effectively [41][33]. Teams can be successful without trust, but usually, additional cost such as monitoring teammates and backing up their work is needed [64][41]. Trust is not only the key challenge but also the primary element in knowledge sharing [2]. One of the most important issues in distance work is trust. Trust is often an issue in remote teams because people may be expected to work with people they do not meet in the whole lifetime of the project. This is especially the case in large teams to deliver innovative work [3][42]. Organizations do not usually allow enough time for people to know each other and build trust [5]. In a seminal study of sixty-two projects sponsored by NSF, researchers found that the greater the number of institutions involved, the less well-coordinated a project was [4][45]. In such teams, coordination has heavily relied on trust and shared communication software tools [37]. Managing such distributed teams is also a challenge for managers and team leads because team members have different time zone, working styles, values, and cultural orientations [58].

Awareness of the group members is an integral part of the construct of team cognition, which is crucial for distributed groups to collaborate well [27]. The concept of workspace awareness was suggested as a key to support the richness of interaction in a face-to-face working environment [28]. As one of the prerequisites of successful team collaboration, knowledge sharing is positively related to inter-organizational arrangements [50][36][60]. Virtual teams are often more diverse in personnel, resources, and work. Members often represent different cultures, languages, and organizational allegiances [51] (Fisher and Fisher 1998). Building trust is especially difficult and important in virtual teams. Although prior experience may reduce the negative impact of distance [14], newly formed teams or teams with newcomers may still suffer from the difficulties of building initial trust.

Awareness

Some research effort categorizes awareness into six types: Collaborations, Location, Context, Social, Workspace and Situation [54]. Awareness is not just about displaying the information to others. Researchers discussed about some concerns in displaying information in To Whom Should I Display My Actions [18]. The concept of awareness affect many design of collaborative software including: Ariadne [15][64], CollabCVS [30], FastDash [7], Jazz [11], and Palantir [55].

Media Richness

Researchers found that media richness serves as an important role in supporting distance work [59][21]. They provide a rational criterion to select media by ranking communication media. Improvements in information technologies facilitate communication among remote workers. Some research focused on improving basic tools like email, while others are trying to provide a better integration experience with the tasks performed by individuals [10][6][26].

4.3 Utilizing Technologies in Distance Work

Many research studies provide technological solutions to address problems in distance work. Maxfield presents a distributed virtual environment that support distant collaboration for multidisciplinary teams [40]. Collaboration Moderator Services are designed to address issues relating to knowledge based collaboration, by providing functionalities to raise users' awareness of opportunities [61]. TeamSpace is designed to support remote teams by managing shared work processes and maintain shared artifacts [22]. Many visualization systems are designed to help share awareness among remote team members [19][62][52][63][34][17][16].

Researchers also presented several ways to classify the software tools that are used to support distance work. One classification of collaboration technology is called Coordination Pyramid [56] (see Table 4.1). Coordination Pyramid provides a framework that recognizes four paradigms of coordination and classifies collaboration technology software based on the paradigm the software support. The Pyramid clearly presented the trend of how collaboration software tools evolved and what can be expected in the future. Another classification of collaboration technologies is Olsons' "Classification of Technologies to support distance work" [47][46] (see Table 4.2). This classification emphasizes the types of technology that are useful and why. The specific technology may be replaced with newer versions in the future, but the types of those features may continue to consist of the infrastructure that supports distance work. By studying the classifications and methodologies in these research studies, one can have a better understanding of the fundamental components of contemporary collaboration technologies.

Levels of Sophistication

	Continuous Coordination		
Contextualized information	1.Interruption and overload management; 2.Automatic expertise recommendation	1.Workspace awareness 2.Artifact recommendation	----
Information discovery	1.Communication gap analysis 2.Expertise location	1.Project visualiztation 2.Artifact tagging	1.Task-centric development 2.Executive project summarizing
Structured processes	Automated message triggers and notifications.	1.Software configuration management 2.Shared editing	1.Process definition and enactment 2.Workflow and issue tracking
Basic functionality	Basic computer-mediated communication	1.Archival and retrieval of common artifact sets 2.Access to artifacts in central repository	Rudimentary task-allocation and planning tools
	Communication	**Artifact management**	**Task management**

Table 4.1: Sarma's Coordination Pyramid adapted from [56]

Classification Of Technologies To Support Distance Work	
Communication Tools	1.Email and texting 2.Voice and video conferencing 3.Chat rooms, forums, blogs, and wikis 4.Virtual worlds
Coordination Tools	1.Shared calendars 2.Awareness tools 3.Meeting support 4.Large visual displays 5.Workflow and resource scheduling
Information Repositories	1.Databases 2.Shared files 3.Blogs or wikis 4.Laboratory notebook (online)
Computational Infrastructure	1.System architecture 2.The network 3.Large-scale computational resources 4.Human computation

Table 4.2: Olsons' Classification adapted from [47]

Chapter 5

Data Collection

5.1 Results from Interview

In order to learn more about the existing collaborative software tools used by people and how they are helping on distributed collaboration, I conducted six interviews with people who work in fully or hybrid remote teams. By learning about how they are using collaborative software tools to finish daily work, I was able to take a closer look at the collaborative software tools and the features which are being used in remote collaboration. The interview questions are designed based on three research questions:

1. What features of existing collaborative software support collaboration?
2. Which of those features and tools do practitioners believe would still be used after going back to the office?
3. What features would practitioners like to add to help distributed collaboration?

Potential interviewees were asked to fill out a survey to understand their background in-

formation including job title, years of experience, primary work location, etc. The survey was anonymous, and identifiable information including the name of their company, team, or colleagues was not asked nor mentioned. The background survey questions are listed in Appendix A. Before the interview, the interviewees were informed about the background and protocols of the study. Verbal consent was read aloud and the interviewees were asked whether they would like to participate in the interview. The verbal consent script is in Appendix B.

For participants that were selected for interviews, during the interview, after verbal consent, the interviewees were first asked to describe the software tools that they use in their daily work, and how do they use the software tools or features in their daily work. The software tools the interviewees mentioned would be written down. As a follow-up question, the interviewees would be asked to talk about whether they use some other categories of software if they did not mention those categories. The classification of the software tools being used in this study is from 2014 Olson's work "How to Make Distance Work Work", as shown in Table 4.2 [47]. The fifth question was "What are the new software tools or features you started to use or used more frequently since remote work started?" The above questions were asked to mainly answer the first research question. Then, interviewees were asked to talk about software tools and features that they would like to continue to use after going back to the collocated working environment. Finally, the last question of the interviews was: "Tell me about the occasions when technologies were not working as you expected. What would you like to change of them." The last two interview questions were majorly designed to help answer the second and third research questions of this study. The interview questions are listed in Appendix C.

In this section, I documented the result from each interview and my participant observation as a presentation of the data. The analysis is made in chapter 6.

	Job Title	Years of Experience	Workplace	Team Member Distribution Situation	Technologies Mentioned That Support Distance Work
Interviewee I	Java Backend Developer	2 years	3 days at company office 2 days at home	50% US, 50% India	JIRA, Confluence, Email, Apple iCloud Calendar, Slack, Zoom, Skype, Online Storage services, Hard Drive
Interviewee II	Senior Software Engineer	5 years	at home	Japan, Canada, UK, US, Italy	Slack, Outlook Calendar, Outlook Email, Github, Collaborator, Confluence, JIRA, Draw.io
Interviewee III	Software Development Engineer	6 months	at home	US	Quip, Pager, Internal Appointment Scheduler, JIRA, Slack, Github, Chime
Interviewee IV	Software Engineer	5 months	at home	US and India	Slack, JIRA, Confluence, Zoom, Outlook Calendar, Notebook
Interviewee V	Software Engineer Intern	3 months	at home	US	Outlook Calendar, VCS, Chime, Slack, Sprint Board Application, Bot for reporting updates, internal project wiki
Interviewee VI	Software Engineer Intern	3 months	at home and office	US	Chime, Google Doc, Slack, Asana, Internal Rating System

Table 5.1: Interviewee Information

5.1.1 Interview I

Daily work of the interviewee includes coding, researching, meeting, and documenting. They use Chime as video conferencing and texting software. Chime also can show the status of contacts. By knowing the status of another colleague, people can know whether it is a good time to reach out. Org information of colleagues can also be accessed through Chime. Slack is also used as a texting software and the status feature is powerful so people can tell whether others are online, offline, on vacation, in a meeting, etc. For documenting discussion and design, they use Google Docs. The commenting feature is used for discussion over specific details. By using sharing features, they are able to generate links and share among group members, they also have control over access rights and range, e.g. who can view, comment, or edit. They use Asana to know what projects others are working on as well as sprint status.

The interviewee believes there are difficulties while using the software to work with others remotely too. First, for some important meetings, the meeting recordings are not always be shared with everyone, the interviewee believes it would be helpful if every participant in a meeting will receive the recording by default. Also, the video recording is not a fast way to review the gist of a meeting, generating text-based meeting notes and summaries would be a great feature to have. Second, although video conferencing and texting software enable people to talk over distance, they feel less connected compared with face-to-face working environments. People usually will not talk to each other unless they have some "formal" reasons to do so. Also, talking over computers and video conferencing software is more challenging in terms of expressing a difficult idea and understanding each other compared with being able to talk face to face.

Finally, while some features already exist for a while, they were not used in the same way. Some new norms help people use the features of collaborative software as they are supposed to be used. For instance, people can update status on Slack and Chime to reflect whether

they are busy and it is a good time to be reached out. However, this feature was not able to reflect people's real status for a long time because people did not update their status timely. A new norm starts to be adopted after their manager asks them to update the status hourly. This new norm of using the status feature help reduce a lot of hesitations when people are trying to reach out.

5.1.2 Interview II

The interviewee uses a group-wide shared calendar to know others working status. By observing whether calendar events like meetings, people can know when would be a good time to reach out. When using Chime, there will be a mark that shows whether a message is read or not. This feature helps people have a better understanding of people's status in a conversation. In texting conversation, Slack has an ample collection of memes and emojis and also supports customized pics, which facilitates a friendly and vivid vibe in a conversation.

When comparing with face-to-face collaborating experience, the interviewee also observed some obstacles in remote collaboration supported by technologies. First, sometimes the overhead before being able to actually start to collaborate makes the collaboration less likely to happen. For example, comparing with walking by a colleague's workstation and starting the conversation, one should ping the other on Slack to check whether it is a good time to talk, open a video conferencing software to create a link, and send the link over to the other colleague, join the video conference and adjust microphone and camera settings. Sometimes, system privacy settings, software updates, and network issues make the overhead effort even time-consuming and people may be reluctant to talk unless have to do so. Besides, fewer opportunities to interact with people are also mentioned due to the lack of random talk opportunities over the water-cooler or hallway. Also, because of the nature of remote

work, people would have fewer opportunities to interact with colleagues who do not have overlapping on the project. Without being able to see each others' faces, people feel less connected as well.

Some new norms are also invented to cope with the difficulties of remote collaboration and build up team morale. For instance, they have a happy hour session every Friday afternoon for people to chat randomly about cooking, lifestyles and pets sharing, etc.

5.1.3 Interview III

The interviewee talked about several software features which help them collaborate remotely. Without being able to talk to people in person, Slack is frequently used as the primary communicating tool. Besides using basic features of Slack including group chat, remote control, and video conferencing, they were using Slack as an entry point for several other usages. For instance, JIRA is integrated with Slack, so they are able to create JIRA tickets directly in Slack without having to open several extra JIRA webpages. Also, the content in email replies will be automatically imported into the JIRA ticket as comments, which helps people understand the context of the ticket. JIRA Kanban is used as a Sprint Board which help team members understand better other members tasks on hand and who to reach out to when necessary, which is especially important when team members are all working remotely and the lack of opportunities to know each other's task on hand in random talk.

Occasions that technologies break down are mentioned too. With the help of file storage and synchronization services like Google drive, they are able to store important data online using those services. However, hard disk drives are also used to store and share data because of the lack of reliability and confidence of the online storage services. Besides, when the primary communication service Slack goes down, the team switches between other similar services like Zoom or Skype. Slow response, screen freezing, unstable internet, and system

error happen from time to time, which creates difficulties for remote workers especially when technologies are the only resort to them.

5.1.4 Interview IV

Without being able to discuss designs and plannings in a meeting room, a commonplace to document important discussions, designs, and plannings become especially important. The interviewee talked about how JIRA and Confluence are used in their them to manage tasks and document important teamwork. A JIRA task and a Confluence page will be created to document task description, context, designs, etc. By commenting on a specific line or a discussion thread, remote team members exchange ideas asynchronously. The Confluence page is also presented in a video conference by sharing a screen just like how people are able to write on a whiteboard in a meeting room. During a video conference, a participant will take notes on a Confluence page or use any notebook software. But when it comes to expressing ideas by drawing in a video conference, they have not adopted any online whiteboard software.

Some challenges are mentioned by the interviewee albeit the conveniences brought by the technologies mentioned above. First, people who are relatively introverted may have fewer opportunities to talk with colleagues except for formal and purposeful interactions. In contrast, talking to people next to you in a collocated office would be much easier. Second, team building and getting to know team members becomes more challenging. Although they have a Friday happy hour intending to get people involved and talk, fewer and fewer people are attending. Sharing life or playing a game is just more difficult when people are geographically distributed.

5.1.5 Interview V

The interviewee mentioned the integration of Slack with Zoom which makes instant meetings less time-consuming because they do not need to switch to, open, and log in to Zoom while communicating with team members in Slack. All they need to do is click the profile of a team member and a Zoom meeting will be started right away.

There are also occasions when the software may not serve the purpose as people expected. The shared calendar they are using, for instance, is supposed to stop pushing work-related notifications when the status is set as on vacation, but the interviewee keeps receiving notifications. Sometimes the notifications will be sent after the meetings start. The thread organization is a mess which makes it easy to miss important emails, which should not happen as people are relying on technologies to communicate remotely. Switching software tools are also more difficult than expected because some members prefer using the old tools instead of learning the new ones even though the latter may be more powerful. And even though there are software tools available to help obtain may awareness information, people will not spend much time on it, unless the information can be obtained passively. Also, although they use Zoom annotation and whiteboard features, they feel those features are more cumbersome and are less likely to use compared with grabbing a pen and drawing on papers or whiteboard in a meeting room. When it comes to hardware issues, for instance, laptop water damage, the nature of remote work leads to much more delay and inconveniences because they have to wait for a new computer to be mailed out before continuing the work. Most importantly, when people are not sitting in the same office, emergent communication becomes more difficult. When they have a severe issue and need immediate collaboration to fix an error, pinging people on Slack and waiting for their responses is too slow compared with walking by a colleague's workstation.

5.1.6 Interview VI

The interviewee talked about how Slack is used more than a communication tool. Usually, the knowledge sharing by a team will be documented in Confluence. But because their team is completely distributed, most of the communication activities happened in Slack. One of the advantages is that important previous knowledge is stored and can be searched when necessary. Thus, Slack is used as an internal wiki and they can often find the solution which is in a previous episode of conversation by searching keywords in Slack.

There are occasions when features will not work well either. For instance, the interviewee mentioned there needs to be an easier way to manage access settings in Quip doc. And Slack will not tell whether the message is read which they believe is necessary when they need a timely response. The shared calendar they are using does not provide an easy way to view others' calendar events. Lastly, although Slack can show others' status, they need to ping others to know their actual status anyways.

5.2 Results from Participant Observation

As the second part of the data collection process, I did a 12-week internship to take a closer look at how technologies were helping people conduct remote work in the contemporary world. The internship was conducted in a fully remote mode, and the team members were all working from home. While working on a development project as a software engineer intern, I was able to observe how a company that owns several well-known collaborative software products was using those technologies to conduct remote work. The technologies we used in the companies and the way we were using those features helped answer the first research question of this study: What features of the collaborative software support collaboration. My experience of using those software tools and the conversations with my

colleagues contributes to the conclusion related to my second research question: Which of those features and tools do practitioners believe would still be used after going back to the office? The occasions I encountered in this internship when technologies could not meet all expectations helped answer my third research question: What features would practitioners like to add to help distributed collaboration? The experience of using the technologies, the ways that technologies support distance work, and the occasions when technologies could not meet the purposes are documented in the following sections.

5.2.1 Video Conference and the Recording feature

In the first half of the internship, one of the most important tasks was having meetings with stakeholders to understand and document project requirements. The stakeholders gave a verbal description of the context and technical details of the project and I drafted the requirement specification doc in a real-time collaborative document platform - Quip. However, this process was not as easy as it sound. As a newcomer who has not had much time and chances learning architecture and some implementation details of major components, understanding the problem and possible technical solutions the architect mentioned on the fly was difficult.

Fortunately, with video conference software, recording an important conference to watch over again later becomes much easier. After having verbal consent from meeting participants, all I had to do is click a button, so the conference video will appear on the cloud shortly after the meeting. Because of the readiness and less overhead of recording an ongoing meeting, I am more inclined to use this feature. It turned out that the video recordings were highly helpful. As I understand more of the problem and the implementation details of the program, I started to understand better the problem and possible technical solutions mentioned in the meetings as I watch them over again.

By contrast, in my previous working experience in a collocated office, the meetings usually happen in a conference room or in a workstation if the conference rooms are all booked as usual. Imagine asking for recording a meeting and passing the recording over to all participants after that, it may merely happen on some serious occasions. The overhead of setting up the video and audio input devices to record, managing the recordings after the meetings, and the cognitive load to bring up this relatively more unnatural request make it unlikely to happen in a collocated working environment. In an online video conference, however, the audio and video devices are usually already set up; and after a short period of time be aware of the meeting is being recorded, the recording mark becomes not so conspicuous and usually not noticeable as the participants focusing on the discussion.

If what makes remote collaboration irreplaceable is the conveniences it brings which make people prefer to use it even when they can also work in a face-to-face environment, then the less overhead and readiness of recording a meeting could be one of the reasons remote collaboration wins.

5.2.2 Asynchronous Collaboration

In remote collaboration, one of the most obvious characteristics is the adoption of a more asynchronous collaboration working mode. Without being able to walk over to the workstation, ad-hoc collaborations usually start from pinging colleagues on the internal communication platform via text messages, e.g. on Slack. Slack has the feature to show people's status including In Meeting, On Vacation, etc, people usually reply later when they see the message or finish another task on hand. When emergencies happen which need instant collaboration, this is not an ideal collaboration mode compared with being in the same physical office; however, it helps avoid frequent interruption when you need to focus on finishing a task on hand.

With an online collaboration document platform, like Quip, collaborations happen when you receive a notice that someone tags you on a doc. The doc can also be shared among stakeholders so they can collaborate on it by adding comments and making changes asynchronously. Quip docs also become a wiki and a place to document original design and discussion for later usages.

5.2.3 Integration

In the second half of my internship, I spent most of my time on development. My typical day starts from logging on to Slack to be online and then replying messages in threads using Slack, checking my Google calendar, having meetings with the team using Google Meeting or Zoom, doing development work, documenting important design/discussions in Quip doc, and logging timesheet in Workday. Emails are basically for event notification e.g. incidents alarm and become less important because almost all text-based conversations happen in Slack. With more dependency on Slack, it naturally becomes the place where collaboration starts. We send code snippets, quip doc, meetings invitation links over Slack, but switching between Github file (for viewing code), quip doc (link google doc), Google conference/ Zoom meetings become cumbersome. So Slack started to integrate those services.

Usually, to start a meeting, we ping a colleague on Slack to confirm the meeting time, add an event on Google Calendar to make this discussion aware to other people, create a Google Meeting, and then copy and paste the meeting invitation link to send over it to the meeting participants. This whole process is cumbersome and time-consuming.

With the integration, instead of switching between Slack, Google calendar, and Google conference to book an event on the shared calendar, copying and then sending over the meeting link on to Slack, we can start an instant Google meeting with team members by simply clicking a button on Slack UI. The meeting invitation will be sent automatically to the

invitee on Slack so they can join instantly by clicking the invitation link, a meeting event will be added to the shared google calendar so others will know these meeting participants are busy during the time, the Slack status of the meeting participants will also be changed to "in-meeting" automatically. This integration of Slack, Google calendar, and Google Conferences reduces overhead to having a meeting in the online world.

Slack also integrates some application widgets so it is not required to open a link in order to view the content. For example, code can be displayed in a code snippet box in a specific style in the message, quip doc can be displayed directly in the Slack chat box without having to open the link, JIRA ticket can also display in the chatbox directly. The integration helps facilitate information obtaining with less effort which boosts efficiency to some extent.

Slack also integrates bots to send event reminders to channels. But I think the SlackBot could be used in more useful ways. E.g. it will be better if the procedure of scheduling a meeting could be done by the SlackBot by sending to the bot some key information like "Schedule a Google meeting with Mike and Ashley at 5 pm and record the meeting". Besides all the steps mentioned above, it can also send reminders to the participants on Slack before the meeting start.

The uncertainty of technologies, "Can you hear me? Can you see my screen?"

We all shared the experience of having some issues with the technologies during a meeting, it could be a microphone or camera stopped working, hearing an echo when people talk, and screen sharing issues. The regular routine of starting a meeting is: can you hear me properly? Can you see my screen? After I had a problem switching windows when sharing screen, I started to ask the same question each time in an important meeting when I switch windows with different content to share: "Can you see my IDE now? I think you can see my terminal right?" The uncertainty of technologies in the online world makes the remote

collaboration experience feel primitive and less natural and real.

Chapter 6

Results and Discussion

Technologies and collaborative software connect distributed team members together. When people are talking about those features of the software tools, they tend to talk about how they are supposed to help on remote collaboration. In fact, however, many features would not serve the purpose as people expect. Future collaborative software and features can focus on addressing those pain points to improve the remote collaboration experience.

6.1 Technologies That Support Distance Work

Based on table 4.2, I summarized technologies mentioned in interviews and my participation observation and filled them into each category.

6.1.1 Email and Texting

Email is still being used in all interviews and in my participant observation. However, people seem to use Email only to check a notification or on very formal occasions such as for farewell

Classification of technologies to support distance work		Percentage of the interviews in which the category is mentioned	Software tools mentioned in the interviews	Software tools I observed being used in internship	Features of technologies that support distance work
Communication Tools		100%			
	Email and Texting	100%	Outlook Mail and other Email services, Slack, Chime, Pager, SMS message	Gmail, Slack, SMS	Mark of message have been read or not, emojis, conversation history searching
	Voice and Video Conferencing	100%	Zoom, Chime, Slack, Skype, Google Meeting	Google Meeting, Slack, Phone Call	virtual background, polls, hand raising, emojis, whiteboard, screen sharing, remote control, meeting recording, waiting room, break rooms, integration with other applications
	Chatrooms, Forums, Blogs, and Wikis	100%	Slack, Chime, Confluence, Quip	Github, Quip, Confluence	comment on line, replying in threads, notification via emails, search for knowledge, progress and tasks-on-hand management, awareness sharing, project knowledge management by documenting, supporting many media forms including UML or code snippet etc,
	Virtual Worlds	0%	----	----	----

Table 6.1: Technologies and Features That Support Distance Work - Communication Tools

letters before leaving a job. Many features in the Email application UI have never been used and people do not understand the functions of those buttons. For quick messages, people just switch to Slack, Chime, or other equivalents. When it comes to emergencies or out-of-work time, neither email nor Slack can grant immediate response, people tend to use text messages.

Voice and video conferencing tools are heavily relied on in distance work. People set up a virtual background in video conferences when they work at home. Meeting recordings can be stored online and shared among participants and those who cannot join the meeting. There are often whiteboards integrated with these tools which enable meeting participants to draw and express ideas. Polling can be conducted during a meeting so participants can vote for each option. During a meeting, people can click on a hand-raising icon to ask questions. Emojis is an important part of informal conversations because it helps build a relaxing vibe that facilitates communication and collaboration. Screen sharing is always used in team meetings. The remote control is used when people are seeking help from each other. The waiting room is enabled when the meeting host wants to have control of the participants of the meeting. When people need separate rooms to talk in small groups, break rooms are used. The integration with other applications usually gives users a pleasant and convenient experience because they do not need to jump between applications.

For Chatrooms, Forums, Blogs, and Wikis, I found that Confluence and Quip or equivalent software tools are always mentioned. Seniors comment on a line of code snippet to express concerns for others' code submissions. Exchanging ideas happen in the commenting threads. Asynchronous collaboration is common because once there is a new comment, people will receive email notifications. One of the most important functions of using the Wikis is to document team knowledge and for later use. When people encounter problems, they often search in the wikis to find solutions. A confluence page or quip page integrates with many media types such as code snippet format, UML, pictures, and many other applications

including JIRA. The pages can present that information without having users click a link and jump to a new browser tab. When people document tasks and assignments on the pages, they often mark the tasks as To-do, In-Progress, or Done. Thus, tasks status is shared among all page viewers. The rights control of those pages enables people to modify access and grant people with View, Comment, or Editing rights.

Although we have seen research mentioned about virtual worlds and explored the possibility of using VR in supporting distance work, I did not find any instance of using it. We can imagine when VR technologies become more common and mature, virtual worlds may bring better collaboration experiences with more richness of information in people's interaction due to the ability to "see" each other.

Shared Calendars are an integral part of distance work technologies. People use shared calendars to find other team members' availability before booking a meeting. The calendars can display important information including participants, description, time, and links. People also have control over what information to exclude. Reminders will be sent by integrating with email systems.

For awareness sharing, I can find it in tools like JIRA, Slack, Chime, and Asana. Usually, there are many dashboards for users to configure so information like who is working on what, how many hours have been spent, what are future sprint goals can be shared among team members. People often look for others' information in these tools before having a meeting with them in order to understand the context and more about each other. Kanban, Sprintboard, Timesheet are the most common features of these awareness tools.

Workflow and Resource Scheduling is seen in tools like JIRA and Github. Users can configure the workflow of software development, planning, or testing cycles so they can manage lifecycles automatically. For example, when planning is done, a tech lead clicks Finish Planning and then assigns the task to a developer. The developer will see the task is open for

Classification of technologies to support distance work		Percentage of the interviews in which the category is mentioned	Software tools mentioned in the interviews	Software tools I observed being used in internship	Features of technologies that support distance work
Coordination Tools		100%			
	Shared Calendars	67%	Outlook Calendar, Apple iCloud Calendar	Google Calendar	meeting information display, member status sharing, appointment booking, event reminders
	Awareness Tools	100%	JIRA, Slack, Chime, Asana,	Github, JIRA, Slack	issue tracking, task-on-hand sharing, task invitation, knowledge documenting, commenting, dashboard, kanban, timesheet, sprintboard
	Meeting Support	100%	Internal Appointment Scheduler	Google Calendar	meeting information display, member status sharing, appointment booking, event reminders, captions displaying and translation
	Large Visual Displays	0%	----	----	----
	Workflow and Resource Scheduling	100%	JIRA, Github	Github	configure and manage work flow, status sharing

Table 6.2: Technologies and Features That Support Distance Work - Coordination Tools

Classification of technologies to support distance work	Percentage of the interviews in which the category is mentioned	Software tools mentioned in the interviews	Software tools I observed being used in internship	Features of technologies that support distance work
Information Repositories	100%	Confluence, Github, VCS, Quip, Google Doc	Confluence, Quip	commenting, documenting knowledge, searching for posts, task status management, task assignment

Table 6.3: Technologies and Features That Support Distance Work - Information Repositories

development and start to log working hours on it. After the development is done, testing engineers have access to the task and can start to log work hours on the task.

Information Repositories include many software tools like confluence and quip. I found that the searching feature is very crucial for asynchronous knowledge sharing. People always search in the repositories to find answers and document their new solutions for knowledge to pass on.

6.2 When Technologies Break Down

6.2.1 Improper Use of Technologies

For instance, when it comes to the advantages of video conferencing software compared with audio conferencing software, people tend to say how being able to see each other even when they are not in the same location help reduce information missing while communicating

caused by distance. In reality, however, people do not usually open their cameras while in meetings, except for important formal occasions or 1-on-1 meetings with the manager. By contrast, face-to-face meetings in a collocated office help people convey and read more information including facial expressions, gestures, reactions, and emotions. Also, sometimes physically present in a collocated working environment help people get familiar with each other quickly, build team morale, and increase people's willingness to speak and contribute to group discussions.

6.2.2 Slow Response

Expecting instant responses become impractical in daily work when walking by a colleague's workstation is not a choice in remote work. Waiting for responses after sending a message in slack is expected and normal. This causes many inconveniences when people need a quick response and instant collaboration. Deploying a hotfix version of an online service is an example where immediate communication, deployment process approval, and collaboration are needed. Waiting for responses on Slack, Email or comment becomes too slow to finish the work. Traditional methods like phone calls and pagers are frequently used in these scenarios.

6.2.3 Too Much Overhead

Although we have many applications which help people conduct daily work, switching between applications and web browser tabs is sometimes a hustle that prevents people from using the applications. A typical scenario of scheduling a meeting can illustrate this. Before starting a meeting, one usually has to ask a participant on Slack or via Email about meeting time. Then, they may have to: 1. book a calendar event on a shared calendar like Outlook calendar, Apple iCloud Calendar, or Google Calendar; 2. open a video conferencing software like Zoom or Google Meeting; 3. enter meeting description, time, title, maybe selecting

contacts as participants, and then create a meeting; 4. copy and paste the link into Slack or Email; 5. open the video conferencing software and enter into the meeting room; 6. check audio and video input devices; 7. wait for the participants to join a meeting or allow them to join from the waiting room.

Integration of software may reduce the time spent on switching and jumping between software. For instance, with the integration of Google Meeting or Zoom in Slack, people can start a meeting without jumping out of Slack. However, integration may lead to too many web pages or screens within a software. For instance, JIRA integrates features that cover a wide range of areas in software development that covers issue tracking, requirements and test case management, and agile software development management. But some people may find it difficult and cumbersome to obtain certain information. For instance, in order to see a tasks distribution pie chart, one has to go through several steps by clicking on menu and sub-menu items before being able to jump to the right screen. A complex system sometimes means a complex menu system too, which might be confusing for new users to navigate. Thus, although JIRA can provide much more information than an issue tracking tool, people are not using those features because there are easier ways to obtain that information, or maybe the information is just not worth it to jump through that many screens.

6.2.4 Reluctance of Accepting New Technologies

Another reason that technologies may not work as people expected is due to the reluctance of accepting new technologies. The cost of training and the time needed to learn how to use new software tools are two reasons we saw in this study. In one of the interviews, an example was mentioned that some team members are reluctant to use Confluence simply because they have been using another internal wiki application for many years and are so used to it that all they can see about Confluence is how it is weird to use compared with the

old tools they were using.

6.2.5 Meeting on Purpose

In remote work, people start a video conference for a formal reason. Update sync-up, stand-up meeting, small group discussion over a topic, almost all of those occasions when people can meet each other are on purpose. The lack of opportunities for random chat in the hallway or near a water-cooler, a quick talk with a colleague sitting next to you, chatting while going out for a lunch together makes it more difficult for people to exchange ideas, share what tasks they are working on, and to get familiar with each other. Even worse, people in the same team but do not have overlapping on the work may never have a chance to talk to each other. Some teams have a new norm that there is a happy hour session every Friday afternoon when people share their lives, pets, cooking, or playing games to create an occasion when people can connect base on interests. But due to the nature of meeting via video conferencing software, the activities and games that can be chosen during these sessions are limited. Also, people are more likely not to attend a virtual happy hour compared with holding it in an office on a Friday afternoon when people almost finish a week's work.

6.2.6 Zoom Fatigue

In video conferencing, people are forced to place their heads in a small frame in front of a camera during a whole meeting. Also, being able to see your own facial expression and behavior is similar to watching yourself in front of a mirror. The unnatural feeling of seeing your face and keeping a good posture will last for the entire meeting. When people are working remotely, continuous meetings are pretty common because the time that is needed for traffic or going to another meeting room is no longer needed. However, the back-to-back

meeting may spend more energy compared with in-person meetings. Zoom platform CEO mentioned Zoom fatigue and talked about that he stopped doing back-to-back meetings after once he had 19 back-to-back meetings in a row. The reasons for Zoom fatigue or similar symptom of feeling tired of the continuous meeting may come in four ways: 1. Excessive amounts of close-up eye contact are highly intense; 2. Seeing yourself during video chats constantly in real-time is fatiguing; 3. Video chats dramatically reduce our usual mobility; 4. The cognitive load is much higher in video chats because, in video chats, we have to work harder to send and receive signals.

6.2.7 Software or Hardware Malfunctioning

As every team member works together remotely, the dependency on software and hardware reliability makes the remote collaboration even vulnerable to the outage of the infrastructure people are using to connect to each other. Whenever team members are experiencing issues like a communication service going down, internet connection unstable, laptop operating system issues, or water spilled on a laptop, the collaboration will be affected significantly. When a laptop has a hardware issue, it takes time for a new computer to be mailed out and delivered. And setting up a new computer and installing all the software and software development environment is a very time-consuming process and repetitive work.

6.3 Discussion

Virtual worlds and VRs are not new ideas in research, but in reality, I did not see the adoption of any of those technologies in people's daily work. Research have presented methodologies and paradigm of conducting remote collaboration, but none of them are mentioned in the data collection in this study. Many research focuses on how to deliver more rich-

Technologies Breakdown Reason	**Examples in Interview Result**	**Examples From My Participant Observation**
Improper Use of Technologies	Camera off most of the times; Don't update status frequently on Slack or Chime; Don't update Calendar events frequently;	Don't update status timely; Outdated automation tool; Outdated confluence page;
Slow Response	Wait for a long time before getting response from Slack, Chime, or Email	Waiting for response causes delay in all lifecycles;
Too Much Overhead	Starting a video conference needs too many steps;	Book a meeting; Share a link; Start a conference;
Reluctance of Accepting New Technologies	Prefer old tools over new ones.	----
Meeting on Purpose	Won't talk to each other unless having a formal reason	Only talked with half of the team for working except for initial introduction when joining the team
Zoom (or equivalent) Fatigue	People expecting shorter time in between meetings.	Too many meetings in a row in most of the days, many of them are not necessary.
Software or Hardware Malfunctioning	operating system issues; water damage to laptop	System breakdowns during a meeting.

Table 6.4: Technology Breakdown and Examples

ness of information in the software trying to imitate the real-world experience people have in face-to-face collaboration experience. However, even if we have more horsepower in the supporting technologies such as high-speed internet or more powerful computing units, people prefer face-to-face experience over imitations brought by technologies. More research on how distance work technologies can provide unique and irreplaceable characteristics to enrich people's collaboration efficacy is needed. Also, most of the classifications we presented above are based on the type of technologies, such as email, video conferencing, shared calendars, etc. However, there are more and more integrated software tools that provide all of those features. For example, JIRA can provide awareness information in task status, team member information, tasks participation, recent activities, location, etc. And Slack is more than a messaging tool, instead, it provides information like whether people are online, working, having a meeting, or on vacation. Also, because Slack can store people's daily conversation, people are using it as a wiki in some sense to search for solutions to problems that others may have encountered.

6.4 Limitation and Future Work

Due to the limitation of time and resources, this study conducted six interviews and one participant observation of 12 weeks. More data is helpful to have a more comprehensive result. Interviewees may have a similar background because they are software engineers and all work in companies in North America. Future work can have more diverse data by studying how companies in other fields are utilizing software to do distance work.

Chapter 7

Conclusion

In this study, I first reviewed the research literature to learn the problems that researchers are facing and solutions they came up with; second, I conducted interviews to learn about how companies are utilizing technologies to bring people together for work, and their experience of using these technologies; third, I did an internship to gather data as a participant-observer; and, finally, assessed my results in discussions with other researchers. I found that most of the technologies mentioned in the classification are heavily relied on in distance work, and might be continued to be used even after people return to work in collocated offices. I also found that although imitation of face-to-face experience may always be inferior to the real experience, the unique capabilities provided by collaborative technologies make them likely to be used even after going back to a collocated environment. Unveiling and understanding the occasions when technology breaks down may shed some light on future collaborative software research and tool development. Also, the research community of this area is pretty active. With the adoption of the 'hybrid' and 'work from home' model by more and more organizations, we can expect to see more valuable research on remote collaboration in the near future.

Bibliography

[1] M. S. Ackerman. The intellectual challenge of cscw: The gap between social requirements and technical feasibility. *Human–Computer Interaction*, 15(2-3):179–203, 2000.

[2] B. Al-Ani, S. Marczak, R. Prikladnicki, and D. Redmiles. Revisiting the factors that engender trust of global systems engineers. In *2013 IEEE 8th International Conference on Global Software Engineering*, pages 31–40. IEEE, 2013.

[3] B. Al-Ani and D. Redmiles. In strangers we trust? findings of an empirical study of distributed teams. In *2009 Fourth IEEE International Conference on Global Software Engineering*, pages 121–130. IEEE, 2009.

[4] B. Al-Ani, D. Redmiles, C. R. de Souza, R. Prikladnicki, S. Marczak, F. Lanubile, and F. Calefato. Trust in virtual teams: theory and tools. In *Proceedings of the 2013 conference on Computer supported cooperative work companion*, pages 301–306, 2013.

[5] D. Bandow. Working with the borg: Trust, systems development and dispersed work groups. In *Proceedings of the 1998 ACM SIGCPR conference on Computer personnel research*, pages 163–169, 1998.

[6] V. Bellotti, N. Ducheneaut, M. Howard, and I. Smith. Taking email to task: the design and evaluation of a task management centered email tool. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 345–352, 2003.

[7] J. T. Biehl, M. Czerwinski, G. Smith, and G. G. Robertson. Fastdash: a visual dashboard for fostering awareness in software teams. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1313–1322, 2007.

[8] P. Bjorn, J. Bardram, G. Avram, L. Bannon, A. Boden, D. Redmiles, C. de Souza, and V. Wulf. Global software development in a cscw perspective. In *Proceedings of the companion publication of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 301–304, 2014.

[9] F. P. Brooks. No silver bullet. *Software state-of-the-art*, pages 14–29, 1975.

[10] M. Cataldo, P. A. Wagstrom, J. D. Herbsleb, and K. M. Carley. Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, pages 353–362, 2006.

[11] L.-T. Cheng, C. R. de Souza, S. Hupfer, J. Patterson, and S. Ross. Building collaboration into ides: Edit $>$ *compile* $>$ *run* $>$ *debug* $>$ *collaborate*? *Queue*, $1(9) : 40 - -50, 2003$.

[12] K. M. Chudoba, E. Wynn, M. Lu, and M. B. Watson-Manheim. How virtual are we? measuring virtuality and understanding its impact in a global organization. *Information systems journal*, 15(4):279–306, 2005.

[13] C. D. Cramton. The mutual knowledge problem and its consequences for dispersed collaboration. *Organization science*, 12(3):346–371, 2001.

[14] J. N. Cummings and S. Kiesler. Who collaborates successfully? prior experience reduces collaboration barriers in distributed interdisciplinary research. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 437–446, 2008.

[15] C. De Souza, P. Dourish, D. Redmiles, S. Quirk, and E. Trainer. From technical dependencies to social dependencies. 2004.

[16] C. R. De Souza, T. Hildenbrand, and D. Redmiles. Toward visualization and analysis of traceability relationships in distributed and offshore software development projects. In *International Conference on Software Engineering Approaches for Offshore and Outsourced Development*, pages 182–199. Springer, 2007.

[17] C. R. de Souza, S. Quirk, E. Trainer, and D. F. Redmiles. Supporting collaborative software development through the visualization of socio-technical dependencies. In *Proceedings of the 2007 international ACM conference on Supporting group work*, pages 147–156, 2007.

[18] C. R. De Souza and D. F. Redmiles. The awareness network, to whom should i display my actions? and, whose actions should i monitor? *IEEE Transactions on Software Engineering*, 37(3):325–340, 2011.

[19] T. Düssel, H. Zilken, W. Frings, T. Eickermann, A. Gerndt, M. Wolter, and T. W. Kuhlen. Distributed collaborative data analysis with heterogeneous visualisation systems. In *EGPGV*, pages 21–28, 2007.

[20] R. Evaristo. The management of distributed projects across cultures. *Journal of Global Information Management (JGIM)*, 11(4):58–70, 2003.

[21] J. Fulk and C. W. Steinfield. *Organizations and communication technology*. Sage Publications, 1990.

[22] W. Geyer, H. Richter, L. Fuchs, T. Frauenhofer, S. Daijavad, and S. Poltrock. A team collaboration space supporting capture and access of virtual meetings. In *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, pages 188–196, 2001.

[23] J. Grudin. Why cscw applications fail: problems in the design and evaluationof organizational interfaces. In *Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, pages 85–93, 1988.

[24] J. Grudin. Computer-supported cooperative work: History and focus. *Computer*, 27(5):19–26, 1994.

[25] J. Grudin. Groupware and social dynamics: Eight challenges for developers. *Communications of the ACM*, 37(1):92–105, 1994.

[26] D. Gruen, S. L. Rohall, S. Minassian, B. Kerr, P. Moody, B. Stachel, M. Wattenberg, and E. Wilcox. Lessons from the remail prototypes. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 152–161, 2004.

[27] C. Gutwin and S. Greenberg. The importance of awareness for team cognition in distributed collaboration. 2004.

[28] C. Gutwin, S. Greenberg, and M. Roseman. Workspace awareness in real-time distributed groupware: Framework, widgets, and evaluation. In *People and Computers XI*, pages 281–298. Springer, 1996.

[29] C. Gutwin, R. Penner, and K. Schneider. Group awareness in distributed software development. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 72–81, 2004.

[30] R. Hegde and P. Dewan. Connecting programming environments to support ad-hoc collaboration. In *2008 23rd IEEE/ACM International Conference on Automated Software Engineering*, pages 178–187. IEEE, 2008.

[31] J. Herbsleb. Building a socio-technical theory of coordination: why and how (outstanding research award). In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pages 2–10, 2016.

[32] J. D. Herbsleb, A. Mockus, T. A. Finholt, and R. E. Grinter. Distance, dependencies, and delay in a global collaboration. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pages 319–328, 2000.

[33] J. D. Herbsleb, A. Mockus, T. A. Finholt, and R. E. Grinter. An empirical study of global software development: distance and speed. In *Proceedings of the 23rd International Conference on Software Engineering. ICSE 2001*, pages 81–90. IEEE, 2001.

[34] I. Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on visualization and computer graphics*, 6(1):24–43, 2000.

[35] J. Hollan and S. Stornetta. Beyond being there. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, page 119–125, New York, NY, USA, 1992. Association for Computing Machinery.

[36] S. Jalali, C. Gencel, and D. Šmite. Trust dynamics in global software engineering. In *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, pages 1–9, 2010.

[37] S. L. Jarvenpaa, K. Knoll, and D. E. Leidner. Is anybody out there? antecedents of trust in global virtual teams. *Journal of management information systems*, 14(4):29–64, 1998.

[38] M. Lu, M. B. Watson-Manheim, K. M. Chudoba, and E. Wynn. Virtuality and team performance: Understanding the impact of variety of practices. *Journal of Global Information Technology Management*, 9(1):4–23, 2006.

[39] M. Lu, E. Wynn, K. Chudoba, and M. Watson-Manheim. Understanding virtuality in a global organization: toward a virtuality index. 2003.

[40] J. Maxfield, T. Fernando, and P. Dew. A distributed virtual environment for concurrent engineering. In *Proceedings Virtual Reality Annual International Symposium'95*, pages 162–170. IEEE, 1995.

[41] D. J. McAllister. Affect-and cognition-based trust as foundations for interpersonal cooperation in organizations. *Academy of management journal*, 38(1):24–59, 1995.

[42] N. B. Moe and D. Šmite. Understanding a lack of trust in global software teams: a multiple-case study. *Software Process: Improvement and Practice*, 13(3):217–231, 2008.

[43] S. Morrison-Smith and J. Ruiz. Challenges and barriers in virtual teams: a literature review. *SN Applied Sciences*, 2:1–33, 2020.

[44] G. M. Olson and J. S. Olson. Distance matters. *Human–computer interaction*, 15(2-3):139–178, 2000.

[45] G. M. Olson, A. Zimmerman, and N. Bos. *Scientific collaboration on the Internet*. The MIT Press, 2008.

[46] J. S. Olson and G. M. Olson. Working together apart: Collaboration over the internet. *Synthesis Lectures on Human-Centered Informatics*, 6(5):1–151, 2013.

[47] J. S. Olson and G. M. Olson. How to make distance work work. *interactions*, 21(2):28–35, 2014.

[48] J. S. Olson and S. Teasley. Groupware in the wild: Lessons learned from a year of virtual collocation. In *Proceedings of the 1996 ACM conference on Computer supported cooperative work*, pages 419–427, 1996.

[49] J. S. Olson, S. Teasley, L. Covi, and G. Olson. The (currently) unique advantages of collocated work. *Distributed work*, pages 113–135, 2002.

[50] N. Panteli and S. Sockalingam. Trust and conflict within virtual inter-organizational alliances: a framework for facilitating knowledge sharing. *Decision support systems*, 39(4):599–617, 2005.

[51] G. Paré and L. Dubé. Virtual teams: An exploratory study of key challenges and strategies. 1999.

[52] M. Petre, A. Blackwell, and T. Green. Cognitive questions in software visualization, 1998.

[53] D. Redmiles, A. Van Der Hoek, B. Al-Ani, T. Hildenbrand, S. Quirk, A. Sarma, R. Filho, C. de Souza, and E. Trainer. Continuous coordination-a new paradigm to support globally distributed software development projects. *Wirtschafts Informatik*, 49(1):28, 2007.

[54] D. A. D. E. Santo Saraiva, B. R. de Oliveira Rodrigues, F. H. Zaidan, and F. S. Parreiras. Quality assessment of awareness support in agile collaborative tools. In *2018 XLIV Latin American Computer Conference (CLEI)*, pages 21–30. IEEE, 2018.

[55] A. Sarma, Z. Noroozi, and A. Van Der Hoek. Palantír: raising awareness among configuration management workspaces. In *25th International Conference on Software Engineering, 2003. Proceedings.*, pages 444–454. IEEE, 2003.

[56] A. Sarma and A. van der Hoek. Categorizing the spectrum of coordination technology. *Computer*, 43(6):61–67, 2010.

[57] C. Schunn, K. Crowley, and T. Okada. What makes collaborations across a distance succeed? the case of the cognitive science community. *Distributed work*, pages 407–430, 2002.

[58] V. Seshadri and D. Elangovan N. Role of manager in geographically distributed team; a review. *Journal of Management (JOM)*, 6(1):122–129, 2019.

[59] M. E. Sosa, S. D. Eppinger, M. Pich, D. G. McKendrick, and S. K. Stout. Factors that influence technical communication in distributed product development: an empirical study in the telecommunications industry. *IEEE transactions on engineering management*, 49(1):45–58, 2002.

[60] S. Staples and P. Ratnasingham. Trust: The panacea of virtual management. *ICIS 1998 Proceedings*, page 12, 1998.

[61] R. Swarnkar, A. K. Choudhary, J. A. Harding, B. P. Das, and R. Young. A framework for collaboration moderator services to support knowledge based collaboration. *Journal of Intelligent Manufacturing*, 23(5):2003–2023, 2012.

[62] E. Trainer, S. Quirk, C. De Souza, and D. Redmiles. Analyzing a socio-technical visualization tool using usability inspection methods. In *2008 IEEE Symposium on Visual Languages and Human-Centric Computing*, pages 78–81. IEEE, 2008.

[63] E. H. Trainer and D. F. Redmiles. Foundations for the design of visualizations that support trust in distributed teams. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, pages 34–41, 2012.

[64] E. H. Trainer and D. F. Redmiles. Bridging the gap between awareness and trust in globally distributed software teams. *Journal of Systems and Software*, 144:328–341, 2018.

[65] M. B. Watson-Manheim and F. Bélanger. Support for communication-based work processes in virtual work. *E-Service*, 1(3):61–82, 2002.

[66] J. Whitehead. Collaboration in software engineering: A roadmap. In *Future of Software Engineering (FOSE'07)*, pages 214–225. IEEE, 2007.

[67] Y. Yue, I. Ahmed, Y. Wang, and D. Redmiles. Collaboration in global software development: An investigation on research trends and evolution. In *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*, pages 78–79. IEEE, 2019.

Appendix A

Interview Background Survey

1.What is your job title?

2.How many years (months) of work experience do you have?

3.Where is your current workplace?

4.When did you work at your last workplace and where was it?

5.What is the geographic distribution situation among your team members?

Appendix B

Verbal Consent Script

Hello, my name is Tian Guan from the University of California, Irvine. I'd like to ask you to participate in a research study about technologies and distributed collaboration.

If you agree to be in this study, we will ask you to answer several questions. The study will take about one hour to complete. We will keep all of your personal information confidential.

Participating in this study is optional, and you can tell me if you want to stop being in the study at any time.

Do you have any questions about the study?

Would you like to participate?

If you have questions about this study in the future, you can contact me or UCI ICS. If you have questions or concerns about your rights as a research participant, you can call the UCI Institutional Review Board at 949-824-6662 or email at IRB@research.uci.edu

Appendix C

Interview Questions

1.What does your daily work look like and what're your typical responsibilities?

2.What software tools do you use in your daily work?

3.How do you use those tools and features in your daily work?

4.(follow-up questions of question 3, only ask if not answered or not answered adequately previously:

1.Tell me about the communication tools you're using, what features facilitate your communication with other team members? (The categories of the communication tools will be given as examples when the interviewees ask for examples or not sure about the definition of communication tools.)

- email texting
- video/audio conferencing
- chatrooms, forums, blogs, wikis

- virtual worlds

2.How do you coordinate activities among team members? (The categories of the coordination tools will be given as examples when the interviewees ask for examples or not sure about the definition of communication tools.)

- shared calendars
- awareness tools
- meeting support
- large visual display
- workflow and resource scheduling

3.What features of what tools help you organize and manage shared information? (The categories of the communication tools will be given as examples when the interviewees ask for examples or not sure about the definition of communication tools.)

- Databases
- Shared files
- Blogs or wikis
- Laboratory Notebooks (online)