

Increasing Awareness in Distributed Software Teams: a First Evaluation

Kevin Dullemond
Delft University of Technology
IHomer
k.dullemond@tudelft.nl

Rini van Solingen
Delft University of Technology
Prowareness
d.m.vansolingen@tudelft.nl

Abstract—Similarly to co-located software engineering teams, teams in which developers work geographically distributed require a variety of information about the project, organization and team members to collaborate effectively. However, in such a setting, maintaining awareness at a sufficient level to collaborate optimally is generally more difficult. Therefore research has been done on the prioritization of information for distributed software engineers, based on their current activity and status. Based on this research a platform was developed incorporating a subset of the findings. In this paper we report on a first evaluation of the deployment of this platform in a distributed industrial case setting. The most important findings are that: (i) providing distributed software engineers with their information needs, appears to contribute to an increase in their overall level of awareness and (ii) information about project related communication with the customer and information about the happiness of colleagues adds the most value.

Keywords—Awareness, Global Software Engineering, Collaboration, Evaluation, Support platform, Iris

I. INTRODUCTION

In Software Engineering, for years it has been a trend for work to be done more and more distributed from each other. This trend is caused by the growing globalization of business [1] and because working from home is increasingly common [2]. However, Just like in co-located teams, software engineers in distributed teams need a variety of information about the project and their team members to collaborate effectively. In contrast with the co-located situation however, acquiring and maintaining sufficient awareness is more difficult. Earlier research [3] provided indications about the most valued types of information. In this study we have made a selection of the types of information found in [3] available in a company of distributed software engineers in a custom developed support environment for distributed software engineers. The main research question we aimed to answer with this is:

“Does the introduction of a selection of information needs of distributed software engineers in a single platform, improve the awareness of software engineering teams?”

To investigate this we have defined subquestions:

- What is the value in the availability of the different information types?
- How can the value of the availability of the available types of information be improved?
- How does the value of each of the information types differ depending on a distributed software

engineers’ relation with the project the information is about?

- The introduction of what types of information that are not available, will add the most value?
- To which extend is information overload encountered?
- To which extend is privacy an issue?
- What are the most important benefits for distributed software engineers of having all their information needs available in a single place?

This paper is structured as follows. First in section II we discuss background and related work of this research. Following this, in section III we discuss the research site and data collection methods, the findings in section IV and a discussion of these in section V. Finally, we discuss the threats to the validity of this study in section VI and present conclusions and discuss future research in section VII.

II. BACKGROUND AND RELATED WORK

Software engineering teams deal with a non-physical material: software. Due to the inherent difficulty of visual inspection of the product it is vital to stay up to date on each other’s status and overall progress. Overall, this knowledge is often referred to as ‘awareness’, defined by Dourish and Belotti [4] as: *“An understanding of the activities of others which provides a context for your own activity”*. In short: software engineering teams need to be continuously aware of the current status and progress of their collaborative work.

Adding distance to Software Engineering makes this even more important [1]. When a software team works distributed this significantly complicates making the information available in the team. As such, much attention needs to be placed in keeping each other up to date on changes that have occurred or are occurring [5]. This challenge has been recognized by tool developers in the GSE domain, who attempt to resolve this by making use of shared artifacts [6]. Firstly, there exist formal development tools which are also commonly used by co-located software teams such as issue management systems (e.g. Fogbugz¹ and Bugzilla²) and code repositories (e.g. Subversion³ and Git⁴). Secondly, there are specialized tools for distributed software engineers. Examples are FastDash [7] and

¹<http://www.fogcreek.com/fogbugz/>

²<http://www.bugzilla.org/>

³<http://www.subversion.tigris.org>

⁴<http://www.git-scm.com>

Expertise Browser [8]. Another example of a technique used by tooling to support the sharing of awareness information using shared artifacts is social tagging. Social tagging is the collaborative activity of marking shared content with tags as a way to organize content for future navigation, filtering, or search [9]. It can be found in Jazz⁵ and TagSEA [10]. This is just a sample of many of such tools with quite diverse isolated purposes.

Booch and Brown [11] propose the use of a Collaborative Development Environment: “a virtual space wherein all the stakeholders of a project - even if distributed by time or distance - may negotiate, brainstorm, discuss, share knowledge, and generally labor together to carry out some task, most often to create an executable deliverable and its supporting artifacts”. When moving to such an integrated solution this will take away the overload of software engineers in switching between a multitude of tools. However, there exists a different threat: information overload. One method of dealing with the overload of information for distributed software engineers is the use of the delta mechanism [3]. In fact, some tool developers have recognized this. For example, Jazz [12] uses RSS feeds to allow software developers to subscribe to a range of workspace-related events of potential interest, e.g. that a change has been made to a project artifact or that a project build has failed.

Using the delta mechanism however does not resolve the need to decide what information is important. With so many diverse topics covered, it is important to understand what information is important and when. Holmes and Walker agree [13]: “Awareness is often impeded at two ends of the spectrum: a lack of information, when the changes only become apparent when a build breaks or bugs appear; or an excess of information, where the changes are announced but the majority of the changes are not relevant to the developer in her particular project and context. The middle-ground is unpopulated: we lack automated support for developer-specific awareness (DSA)”. In our previous work we have already investigated and validated concepts for awareness increase among distributed software engineers, which find their way into the evaluation platform of this paper. Examples of such concepts are the overhearing of conversations [14], Mood-Based Microblogging [15], collaboration-centric tooling environments [16], and our work on the concept of Virtual Office Walls [17].

Finally, in [3] we have investigated the information needs of distributed software engineers and their mutual prioritization. The lists of most important information items found in that study are depicted in tables I, II and III. In this paper we introduce a selection of these to a group of distributed software engineers and evaluate the value of this.

III. RESEARCH SITE AND METHOD

A. Research site

This study is carried out at IHomer, a Dutch software engineering company founded in August of 2008 in which home is the default work location. The company currently employs 21 people, working on a variety of products, projects

TABLE I. PROJECT SPECIFIC INFORMATION ITEMS FROM [3]

Technological agreements <i>e.g. on programming language, frameworks or standards to use</i>
Requirements
Risks (project specific)
Process agreements <i>e.g. roles, stakeholders, the process type</i>
Issues (tasks)
System under construction
Source (repository)
Build status <i>e.g. build succeeded/failed</i>
Deployment Status <i>e.g. currently deployed version, is it running?</i>
Planning
Deadlines
Meetings
Status
Hours worked on the project
Milestones
Phase of project <i>e.g. starting up, active, commissioning, done</i>
Project related communication with the customer <i>e.g. mail, phone calls, transcripts</i>
Project related communication with the team <i>e.g. mail, phone calls, transcripts</i>

TABLE II. ORGANIZATION SPECIFIC INFORMATION ITEMS FROM [3]

Risks (organization specific)
Customer relations
Billing status <i>e.g. sent out, paid, overdue</i>
Organizational development
Action points <i>e.g. things that need to be prepared or researched</i>
Planning
Organizational meetings <i>e.g. date, content</i>
Business opportunities <i>e.g. possible new customers or projects</i>
Applicants <i>e.g. possible new colleagues</i>
Financials <i>e.g. liquidity, investments, Year-To-Date figures, forecast</i>

TABLE III. PERSONAL INFORMATION ITEMS FROM [3]

Contact information <i>e.g. mail, phone number, preferred means of contact based on specific situations</i>
Approachability <i>e.g. what means of contact are available? Can he/she be disrupted?</i>
Current activity
Planning
Agenda <i>e.g. planned activities</i>
Holidays
Idleness <i>e.g. when is there no billable work available for him/her?</i>
Seniority <i>e.g. junior, senior, years of experience</i>
Happiness <i>e.g. mood, is he/she content in general?</i>
Personal situation <i>e.g. what is going on in his/her life?</i>
Personal information <i>e.g. hobbies, name of spouse and children, age</i>
Team <i>e.g. in what team does he/she work?</i>
Knowledge/Skills/Expertise

⁵<http://www-01.ibm.com/software/rational/jazz/>

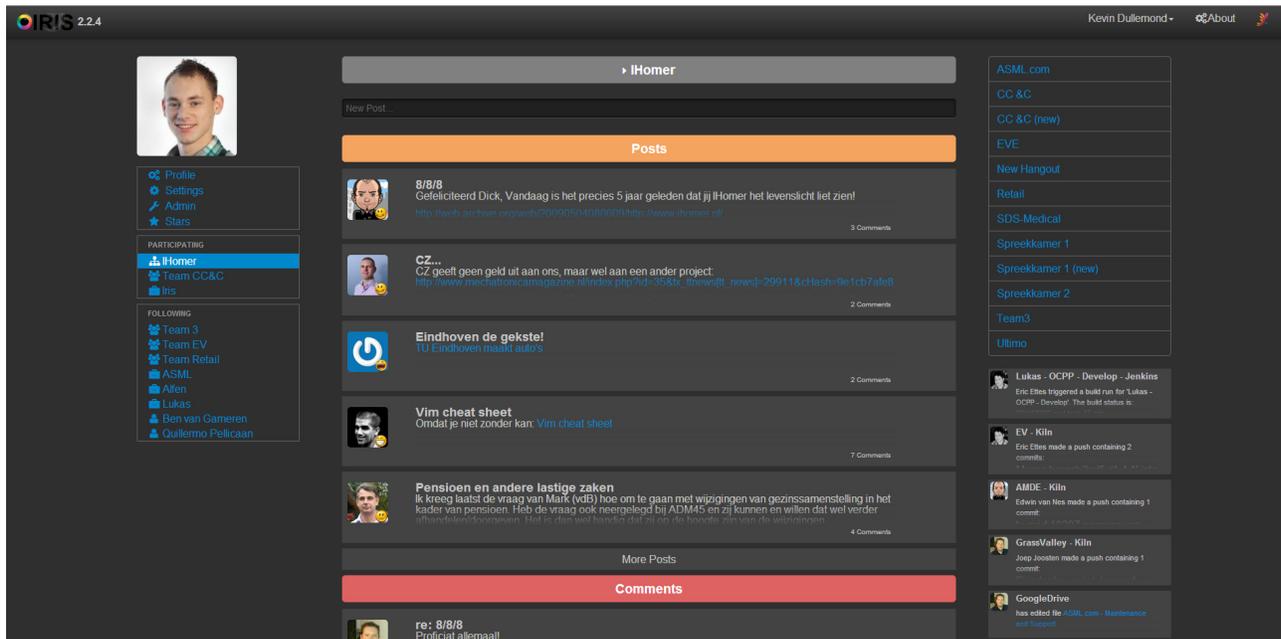


Figure 1. The main screen of Iris

and contracts. The largest team consists of 7 people working on related projects, but the overall group is very close with personnel moving between teams and teams exchanging projects as needed. Even though it is common practice in the company to work from home, the employees try to get together once a week on Tuesdays to meet face-to-face at the company office to stay connected. The company has grown over the past years and initially on Tuesdays everyone discussed what they were doing. This worked well until the company size reached 16, and then sub teams were formed to keep this face-to-face communication more tractable. Teams cluster according various factors such as projects and related technologies.

B. Evaluation platform

To be able to answer the research questions, we have investigated the use of a support environment called Iris in the company. Different versions of Iris have existed since January 2012 with different focus points and implementation technologies. The version we are reporting on in this paper has already been in use at the company for five months between March and August of 2013 and is still actively used today. In this section we will describe the process which was used to develop the platform, discuss which of the information items from tables I, II and III are supported, why these were chosen and how they are supported.

1) *Development process:* The development of Iris has been done in an iterative fashion with close collaboration with the users and continuous use of the system. Because of this, the evaluation is different than in most such studies, as functionality was continuously added during the use of the platform in the organization, based on what brought the most value for the least amount of effort. The reason we chose to structure the development process like this is threefold. Firstly, we needed to be able to deal with uncertainty and changing requirements since we are creating a genuinely novel product

and such projects are often faced with uncertainty regarding both requirements and implementation technologies. Secondly, we needed to involve the intended users as soon and as strongly as possible because they are experienced with working in a distributed setting and as such can give valuable feedback. Finally, we needed to stimulate usage as soon as possible since in earlier research [14] we found, that the value of awareness sharing technology is higher when a larger portion of the team uses it and that this is often a problem when introducing such tools in practical settings.

2) *What information items are supported:* As we discussed in the previous subsection, Iris was developed iteratively. In every iteration we decided to do first what we expected would bring the highest return on investment, based on the value the feature would bring to users and the amount of effort it required to implement. As such we went after the 'low-hanging fruits' first. With respect to choosing which information items to create direct support for, this meant focusing on the items that were already stored in a dedicated system and as such these items received a higher priority than one would expect based on their perceived value reported in [3] alone. Examples are information from issue management systems and repositories. It is important to note however that all items in tables I, II and III are valuable to distributed software engineers. The information items that are currently supported in Iris are depicted in table IV.

3) *How are these information items supported:* In Fig. 1 the main screen of Iris is shown. The screen consists of three main sections. On the left side of the screen the user can make changes to his settings and select the contexts he or she is interested in. The user configures which projects and teams he or she participates in and what projects teams and people, he or she follows. Subsequently the user can select one of the projects, teams or people and the rest of the user interface will show only that information that corresponds to that context.

TABLE IV. INFORMATION ITEMS CURRENTLY SUPPORTED IN IRIS

Issues (tasks)
Source (repository)
Build status <i>e.g. build succeeded/failed</i>
Deployment Status <i>e.g. currently deployed version, is it running?</i>
Project related communication with the customer <i>e.g. mail, phone calls, transcripts</i>
Project related communication with the team <i>e.g. mail, phone calls, transcripts</i>
Contact information <i>e.g. mail, phone number, preferred means of contact based on specific situations</i>
Happiness <i>e.g. mood, is he/she content in general?</i>
Personal situation <i>e.g. what is going on in his/her life?</i>
Personal information <i>e.g. hobbies, name of spouse and children, age</i>
Team <i>e.g. in what team does he/she work?</i>
Knowledge/Skills/Expertise

In the middle of the screen the most important information is shown: the posts and the events. In Fig. 1 the company context is selected and the posts and comments in that context are visible. Users are able to make posts in a specific context and attach an emotion to this (see the emoticons in the avatars of the posts). Additionally a specific post or comment can be clicked on resulting in the full post being shown including all comments on it in chronological order.

Events can also be shown in the middle of the screen. Events in Iris are notifications of things that occurred in dedicated systems that are relevant to software engineers. An example is the resolving of an issue in the issue management system. In Fig. 1 a concise version of the last five events that occurred in the selected context (in Fig. 1 this is the company) is shown in the bottom right of the screen. By clicking in this list all events belonging to the selected context are shown in full in the center of the screen, completely replacing the posts and comments. In Fig. 2 we show a (cropped) example of such a list of events. A final thing in Fig. 1 that is shown in the right side of the screen are links to all configured Google Hangout rooms with an indicator showing who are currently present in each of the rooms.

The information items depicted in table IV are reflected in Iris as follows. Firstly, the issues originate from an issue



Figure 2. Cropped version of the detailed event list

management system called Fogbugz. Changes to issues, such as the creation of a new issue and an issue being assigned to someone, are shown in Iris as events. Pushes or commits to the software repository are also shown as events. The software repositories that are supported at the moment are Subversion and Kiln. Furthermore also the results of automated builds and deployments using the Jenkins build server are available in Iris as events. Project related communication with the team and customer is supported in two ways. Firstly, it is possible to forward mails to Iris which will automatically share these, both as events and posts, in the correct context. Next to this, it is also possible to perform team communication and to discuss customer collaboration using the post functionality in Iris. Using this functionality, it is also possible to share information on personal information and for each post people need to supply their mood about what they are posting, giving insight in their happiness. Furthermore, there is also support for user profiles in which it is possible to define contact information, personal information, team information and information about knowledge, skills and expertise. Finally, also changes to files in the Google Drive folders of the company are shown as events. While this is not directly linked to a specific information type, information about specific unsupported information items is maintained in documents stored in the Google Drive folder. A specific example in the company is a document with the current sales leads for new projects.

C. Research Method

In this study we acquired the empirical data by performing six semi-structured interviews⁶. We chose to do semi-structured interviews because the strengths of this data gathering method include gathering insights, ideas, viewpoints and opinions of the interviewees [18]. The first author performed all six interviews which took roughly one hour each. To be able to analyze the results of the interviews at a later time, all interviews were recorded. We used purposive sampling to determine who to interview to make sure the gathered data would sufficiently reflect the viewpoints of everyone at the entire company. Firstly there exists three roles in the company structure. Most employees are software engineers but there are also two associates and 1 sales executive. In selecting the interviewees we made sure we covered all the different roles. Additionally, for the software engineers there exists four levels of seniority: self, project, customer and market and we made sure we selected one software engineer out of each of the seniority levels as well. In the remainder of this paper we will refer to the interviewees with fictive names depicted in table V. It is important to note that since all interviews were in Dutch all quotes reported in this paper are translations.

Finally, in the interviews we ask the interviewees for each of the supported information items whether they would recommend using it and if so what grade between one and ten they would give the strength of this recommendation. We ask this as a proxy to determine how valuable they think the feature is. This technique is called the Net Promoter Score (NPS) and the final score is arrived at by: (i) counting all scores of 9 and 10 as +1, all scores of 7 or 8 as 0 and all scores of 6 or lower as -1, (ii) summing these scores and (iii) dividing the outcome by the number of respondents. This calculation will always

⁶http://www.aspic.nl/collaboratecom2013/interviewstructure.pdf

TABLE V. THE INTERVIEWEES, THEIR ROLE AND THEIR SENIORITY

Name (fictive)	Role	Seniority (for software engineers)
Anthony	Software engineer	Market (Cat IV)
Brian	Associate, product owner	n/a
Charles	Software engineer	Self (Cat I)
David	Software engineer	Project (Cat II)
Edward	Software engineer	Customer (Cat III)
Frank	Sales executive	n/a

result in a NPS between -1 and +1 and any score above zero is a strong indication the corresponding feature is considered really valuable

IV. FINDINGS

In this section we present the results and findings of the interviews to answer the research questions defined in section I. We structure this section as follows: Firstly for each of the categories of information (project related, organization and personal) we in turn discuss (i) the value and absence of value the interviewees experienced with the supported information items, (ii) whether they would like to see different information about items related to projects they participate in than information related to other projects and (iii) which types of information they missed most and why. Following this we discuss how the interviewees experienced the problems of information overload and a lack of privacy and conclude by presenting how the interviewees overall felt about the support environment and what they consider to be the most important benefits.

A. Project related information items

1) *Most valuable information items:* As is depicted in Fig. 3, concerning project related information items, the interviewees considered project related communication with the customer (NPS: +0.50) and project related communication with the team (NPS: +0.17) to add most value. Brian commented the customer communication was valuable to him mainly because it was for the first time possible to have a complete insight in all communication with the customer in a lightweight fashion. David, Edward and Frank explicitly mentioned liking this made it easier for them to know what was going on with the project. Although the most common way to share communication with the customer was making a post (“*People return from a meeting and summarize the outcome*” - Brian), Brian also used the function to directly share emails and is convinced this has value because: “*Every time I shared customer communication, people would comment on it*”. Edward also mentioned he remembered seeing a post about a customer being very satisfied and getting a morale boost because of it.

	Anthony	Brian	Charles	David	Edward	Frank	NPS
Issues (tasks)	6.5	8	7	8	7	8	-0.17
Source (repository)	1	9	7	8	7	7	0.00
Build status	7.5	6	7	7	8	7	-0.17
Deployment Status	6	6	7	8	No	8	-0.50
Project related communication with the customer	No	9	7	9	9	9	0.50
Project related communication with the team	No	9	7	7	8	9	0.17

Figure 3. Net Promoter Score for supported project related information items

Regarding the project related communication with the team Charles, David and Edward gave the example that they used the platform to have a single place to do a weekly stand-up. At the start of every week on Monday they would announce to their team mates on what projects they were planning to work in the upcoming week. Edward did mention he did not really use the sharing of team communication beyond this because he would do most his team communication while in a Google Hangout with his team members and there was no way to share this communication directly in the platform. As opposed to his colleagues, to Anthony the sharing of project related communication with the team and customer was not valuable because in his team no-one really used it. He speculated this is probably due to habituation. The most important proposed improvements are: making the communication easier to search through (Brian) and adding some sort of structuring such as being able to add a question, poll or template for weekly stand-ups (Charles).

With respect to the other four information types: Source (NPS: 0.0), Issues (NPS: -0.17), Build Status (NPS: -0.17) and deployment status (NPS: -0.50) the interviewees saw less value. For automated builds and deployments this was largely caused by the fact that, most of the projects the interviewees worked on, either did not use automated builds and deployments at all, or did not configure them to be visible in the platform. For Anthony the issues and especially the repository information was less valuable because he was used to working with the dedicated tooling (the issue management system and the repository system respectively) visible at all times. Edward found the updates about the issues of limited value because he is used to working in a hangout with the other project members and because of that he always had a general idea what his colleagues were working on. Both Anthony and Edward did find issues valuable when it was about projects they were not part of.

Frank found different things valuable due to his role as a Sales executive. With respect to issues, he saw value in knowing on what projects work was being done, but other than that, he did not need information about the day to day struggles of the different projects. Brian especially liked the issues and source information because it made it easier for him to see what was going on. An example he gave was that he is the product owner of a specific product. Whenever there was an upcoming release, he would keep an eye on new pushes to the repository and accompanying comments to see whether it was appropriate for him to immediately check the new feature and give feedback to the person that made it. Charles liked these updates to get an overall sense of what everyone was doing. An example he gave was that he saw a new issue pop-up in the events list about something he heard another colleague talking about. He was able to connect the two colleagues and they were able to resolve the issue without the first colleague wasting much time. A final example of the issue sharing functionality being useful came from Edward. He talked about seeing a issue being resolved on a project which had been dormant for a while. This made it clear to him that this project had become active again.

2) *Value of information and its relation with the project:* Quite some different viewpoints arose regarding how the information that is valuable differs depending on whether the

information is related to a project you are part of. Four out of the six interviewees (Brian, Charles, David and Frank) saw value in seeing less detailed information about issues related to projects they are not part of. They suggest this can both be accomplished by omitting details in the item itself and by only showing the creation and closing of issues for projects you are not part of. Anthony had a contradicting opinion. Because he always had the issue management tooling opened for the project he was working on at the moment, only events about projects he was *not* working on at the moment had value to him.

With respect to changes to the repository the opinion was split three ways: Anthony and David are only interested in updates about the project they are working on right now, Edward and Frank only want to see updates about projects they participate in and Brian and Charles find updates about all projects valuable. For the build and deployment status Anthony, Brian and Edward only find them valuable for projects in which they participate. Charles believes there is value in seeing everything, Frank only needs to see planned builds and deployments to the customer and David reported only wanting to see production builds for projects he does not participate in but all deployments.

Finally with respect to project communication with the team and with the customer Brian and Frank suggest, a way to filter operational information for projects they do not participate in, would be valuable. Charles, David and Edward want to see all communication with the customer, but David only wants communication in the project team for his own projects and Edward would like a way to differentiate what information is only valuable to the team and what information is valuable to all, and use this to filter accordingly. Because Anthony did not use this functionality he did not express an opinion.

3) *Most missed information items:* With respect to the project related information items that were missed most, a large part of the discussions were about the fact that these were things that the organization is finding hard to control and document. Anthony for instance mentioned: *"I would be thrilled if any of the items on the list were consistently recorded at all"*. Brian and Charles specifically mention they missed all of the items equally and propose the creation of a dashboard page for each of the projects, depicting the information. They mention however it would be best to start using specialized project management tooling and using this as a source for the information in Iris.

David only actually missed having insight in the status to know how things are progressing and whether things are going according to plan. He thinks the other items are only useful when you are new on a project. Edward agrees and mentions he missed seeing when a milestone is due. For projects he is not part of this can help him in timing when to ask how things are going. Further he mostly missed having access to the technological agreements because he finds it tough to find this. He thinks it is interesting to see what different technologies are used throughout the company, for example for asking questions.

Finally, because of his role (Sales Executive), Frank took a different angle in answering this question. He is interested in the technological and process agreements to have insight

in what experience is present in the organization. Further, he is interested in an overview of the planning, both on a team and organizational level. Based on this information he can better decide what to focus on when looking for new projects. Because he is also actively involved in managing the customer relations, he also missed having insight in the risks of the various projects.

B. Organization specific information items

1) *Most valuable information items:* Because none of the organization specific information items are supported directly in the platform, there are no Net Promoter Scores available. However, Anthony, Brian and Edward reported that to them, business opportunities was supported in full by the post functionality and is one of the most valuable uses of the platform. The others agreed it was supported and valuable but would like it to be easier to search through. Charles suggests tagging the posts to be able to more easily search through them and being able to generate lists. He thinks it is important to stick with the post functionality instead of adding more structuring because he likes the personal nature of the posts. Frank on the other hand explicitly asks for an explicit list because to him the post functionality is too non-committing. Brian and Edward also report financials as being sufficiently supported and valuable. Charles even goes so far as to say he believes all items on this list are sufficiently supported and only need to be easier to search through.

2) *Most missed information items:* With respect to the most missed organization specific information, Anthony, Edward and Frank report missing an overview of the billing status with the various customers. They report this information to be valuable to them because it will help them to determine whether to take action based on whether the bill has been sent out or not and whether the customer has paid or not. Further, in contrast to Brian, Charles and Edward who reported finding financials sufficiently supported in the current system, Anthony reports missing financials because he finds it tough to locate the right information. Subsequently, Brian and Frank also report having missed action points as *"it isn't clear what happens after we agree to do something"* (Brian) and *"Sharing action points with posts is too non-committing. People should be accountable"* (Frank). Furthermore, Brian also reports missing information about organizational meetings because: *"this would be the last item to get rid of internal emails altogether"*. David, finally, misses an overview with all the different organization specific information types. The three types of information he finds most valuable in such a list are financials, business opportunities and billing status.

C. Personal specific information items

1) *Most valuable information items:* In Fig. 4 it is depicted that the interviewees considered happiness (NPS: + 0.50) the most valuable personal information item. Everyone of the interviewees agreed it had value in the context of the post. David and Frank report using the mood associated with the post to decide how urgent it is for them to read. When the mood is either very positive or very negative they read the post immediately. Frank gave an example for this when he read a post with a very sad mood immediately and it turned out to be about problems with cash flow and that this information

was important to him. Furthermore Frank says: *“I read posts completely differently if another emotion is associated with it”*. Edward also reports that he really likes sharing how he feels when he makes a post, because it helps him in expressing himself. Brian and David also use the mood indicators to get an overall feeling of the mood of their colleagues. Brian uses this to see whether and how (emotionally) he can approach someone. David even said: *“I saw someone was struggling with a task and felt unhappy about it so I contacted him and offered to help”*. All interviewees think it would be valuable to have insight in the overall mood of their colleagues. Brian proposes to improve the functionality by adding a diary function in which everyone reports at the end of each day what they did that day and how they feel about that. Charles proposes to create an overview with the mood of the last post made by each of his colleagues or some sort of aggregate function based on the mood of their last posts.

	Anthony	Brian	Charles	David	Edward	Frank	NPS
Contact information	No	6	No	No	8	8	-0.67
Happiness	7	9	7	9	9	8	0.50
Personal situation	No	8	8	10	9	8	0.17
Personal information	No	7	No	6	7	7	-0.50
Team	No	6	No	No	7	7	-0.67
Knowledge/Skills/Expertise	7.5	6	No	No	7	8	-0.50

Figure 4. Net Promoter Score for supported personal information items

The second most valuable personal information item, according to the interviewees, is personal situation (NPS: +0.17). With the exception of Anthony everyone finds this information valuable. David reported that to be able to share what is going on and knowing what is going on with his colleagues, helps him in collaborating with them. He said: *“You have to know what is going on in the lives of your direct colleagues to be able to collaborate”*. Charles reports liking being able to stay current on the personal lives of his colleagues. He mentions he could use this to deduce when someone would be less approachable and anticipate on this. Edward also said he thinks it is important to know when his colleagues are feeling okay or not and reports checking the platform several times a day on weekends when the spouses of his colleagues were expected to give birth to a baby. Frank reports that although he does not share much himself, because he finds it tough to decide what to share, he appreciates the insight in the personal lives of his colleagues. He said: *“I very much enjoy reading what my colleagues are doing. I like to know what keeps them occupied. This helps me to know what kind of person he is and helps me to get to know my colleagues better. To me, this is one of the most important benefits of Iris”*. Overall the interviewees do not see much reason for improvement for this functionality. Only Charles mentions he would like to be able to tag the different posts to be able to search through them.

The other four personal information items which are supported in the platform explicitly, are considered far less valuable by the interviewees: personal information (NPS: -0.50), knowledge/skills/expertise (NPS: -0.50), contact information (NPS: -0.67) and team (NPS: -0.67). The main issues with these types of information are caused by limited support in the platform, limited use by the users and because the interviewees already know about some of the information. Firstly, with respect to knowledge, skills and expertise the explicit implementation is the provision of a specific section

in the user profiles to indicate this. This was not used much because too few people maintained it properly (Brian and Frank) and because no-one maintained it to a useful level of detail (Edward). Anthony, Brian, Edward and Frank did report they find the indirect way of asking for specific expertise via the post functionality, to be valuable and sufficiently supported. Frank recalled searching for specialist database expertise necessary to secure a new contract and finding the necessary expertise in this fashion. Brian even reports: *“I was sometimes surprised by the depth of the expertise present in our organization”*. With respect to improvements the consensus was that as an organization they need to find a way to maintain this kind of information in a consistent way

Secondly, some interviewees saw value in having access to personal information. Brian calls it: *“valuable if you’re looking for it”*. Edward reports he did not know the information was there, but that it is definitely valuable to him. Charles, David and Frank say they think it will only be valuable when the company gets larger because at the moment they know everything they need to know about their colleagues. With respect to contact information, Anthony and Brian report that the company should have a single place or source to store this information, something that is not the case at the moment. Edward and Frank comment they think it should be made clearer how this information can be accessed. Finally, the interviewees did not see much value in the information about the teams, although Edward suggests it will be more valuable when the company gets bigger.

2) Value of information and its relation with the project:

Overall the interviewees report that differentiating between information that is related to projects you participate in and projects you do not participate in is much less important for personal information items than for project related information items. Only two differentiations are mentioned. Firstly, Brian suggests that while he wants to know information about all his colleagues, he would like the platform to make information about his team members more accessible, because he needs it more frequently. Conversely, Anthony and Frank suggest that with respect to the skills, knowledge and expertise they are specifically interested in information about non-team members because they already know this about their team members. Anthony said: *“You usually know about the expertise in your own project teams”* and Frank said: *“when you are searching for expertise it’s usually outside of your own team.”*

3) Most missed information items: The most missed information items are planning, current activity and approachability. Firstly, Charles, David, Edward and Frank report missing all the sub-items of planning (Agenda, Holiday and Idleness) and they see value in these to be able to anticipate correctly and to know when to coordinate. Anthony and Brian only missed the idleness portion of planning. Secondly, Brian and Edward report missing the current activity of their colleagues to be able to decide when to contact someone. Edward concedes this is partly implemented because it can be deduced from the posts and the events but that this insufficient for him. Conversely, Charles and David also report being able to deduce the current activity of their colleagues but they report that this offers significant value to them. Finally, Brian reports missing the approachability of his colleagues to be able decide whether and how to contact someone.

D. Information overload and privacy

Four out of the six interviewees report experiencing an overload of information often (Brian) or sometimes (Anthony, Charles and David). David commented “A whole chain of events clutters the time-line when large things happen” and Charles says it is particularly a problem “for sources that send more than one event about a single or related action”. Anthony also reports experiencing an overload because when he looks at the system it is always a snapshot of all the events that could be valuable to him and that he needs to manually search to look back in time. Finally he reports that in his team the sources that create the events were not properly configured and as such he could only access the events company-wide in an uncontextualized fashion, further aggravating the difficulties he had with searching through them. Edward and Frank report not experiencing an overload of information because “I used the events too little” (Edward) and “because I don’t follow the events all the time” (Frank).

None of the interviewees that experienced an information overload really had an approach to dealing with it other than the ‘ostrich tactic’ of putting their head in the sand and waiting for the issue to resolve itself. Anthony said: “I didn’t deal with it. I probably missed important events” and David said: “I would just wait for it to stop”. The interviewees propose the support environment can help prevent an information overload better by bundling events from the same user that are close together (Brian), combining related events (Charles and David) and by allowing users to filter based on the type of event (Anthony).

Finally, while everyone agreed with the thesis: *with Iris information has become more transparent and things are more visible and more easily accessible than they used to be*, no-one reported having ever experienced any threat to their privacy.

E. Overall view and main advantages

All of the interviewees reported seeing value in a platform such as Iris which aims to make all information a distributed software engineer needs, available in a single platform. To Anthony the main advantage is “To have an overview of everything that is happening in a project” and he is convinced that improving the contextualization is the most important thing to improve to achieve this. When asked about the most important advantage Brian reports “having all information in one place and being able to have an overview because of this”. To Charles it’s: “Knowing what your colleagues are doing without asking” and to David “The availability of information I couldn’t access before, most profoundly personal information”. Finally Edward reports the most important use of the platform is “A thermometer for the organization: how is it going and what is current” and Frank thinks “Not having to look in which system the information is, convenience and effectiveness” are the most important benefits. Additionally both Brian and Frank report they believe that being able to do away with internal emails altogether is a very important secondary advantage of Iris. Frank reports: “In my experience in other companies I always spend an enormous amount of time dealing with internal emails and this is far less if you use one public system in which all the information is available”.

V. DISCUSSION

In this section we will reflect on the findings by going over each of the sub research question and discuss what findings are applicable in answering them. We discuss each of these in a separate subsection.

A. What is the value in the availability of the different information types?

Overall the information items that brought the most value were project related communication with the customer (NPS: +0.50), project related communication with the team (NPS: +0.17), happiness (NPS: +0.50) and personal situation (+0.17). The project communication with the customer and team was primarily important to have lightweight insight in how things were going in a project. A number of the interviewees reported having insight in the issues, source, build status and deployment status helps them achieve this goal as well.

Furthermore, happiness and personal situation were primarily appreciated because of the insight this provided in who their colleagues are, what makes them tick and how they are feeling. The interviewees reported finding this valuable in collaborating with them and reported it is much harder to stay sufficiently informed about this in a distributed setting without such a support environment. Finally, information about business opportunities the support environment provided, was also reported by all of the interviewees to be valuable in staying informed about upcoming projects and the status of the work pipeline for the company.

B. How can the value of the availability of the available types of information be improved?

Most of the important information items discussed in the previous subsection are primarily shared using the relatively unstructured post functionality. The direct support for, for example sharing customer and team communication, is used less. Several causes for this are discussed. Firstly, some aspects of the support could be improved. For instance by showing what files exist in the change set for pushes to the repository and by being able to see an overview of the last issues dealt with by each user. Secondly there were problems with configuring the support correctly. For example, several interviewees reported either not using automated builds and deployments at all in their projects or simply not configuring the related events to be send to Iris. Thirdly, educating the users better would be an improvement as well. Both with respect to how to configure the sources for their project correctly, but also with respect to how and where different types of information are shared. In the interviews it occurred several times that an interviewee did not know about the availability of a certain type of information. An example is the availability of contact information, personal information and information about skills knowledge and expertise in the profile page of the users.

Next to improvements of the direct support, it was also discussed it should be easier to search through the posts because it was often difficult to locate the right information. As possible solutions to this, tagging and adding more structuring (such as templates) were suggested. Finally, it is important to recognize that making all information available in one location is not the only issue in this particular setting. In many cases the

problem lies with the fact that information is not consistently available, or not available at all. This makes it difficult for a platform which attempts to unify all information, to do its job. Therefore agreeing what type of information should be recorded and in what format, is an important step that this organization needs to take.

C. How does the value of each of the information types differ depending on a distributed software engineers' relation with the project the information is about?

The most important finding with respect to this research question is that the interviewees have quite contrasting opinions concerning this. An example of this is that while most interviewees like to see less detailed information about the issues related to projects they are not part of, to one of the interviewees precisely the information related to the project he is currently working on, offers no value. This is because it is his common work practice to always have specialized issue management tooling visible on his screen about the project he is currently working on. The main conclusions we draw from this are that: (i) we should support different options, test it out in practice and evaluate what works best and (ii) irrespective of what turns out to be the best configuration in general, it is probably important to make it configurable to the user to deviate from that standard configuration.

Another conclusion we can draw is that differentiating between information related to projects one participates in and projects one only follows, is primarily valuable for the project related information types.

D. The introduction of what types of information that were not available, will add the most value?

The most mentioned information items that the interviewees reported missing are the following. Firstly, with respect to the project several users report they miss all the information items on the list in the form of a project dashboard. They recognize that the main problem has to do with the fact that there is no consistent location or method of recording this information and one of the interviewees proposes the use of specialized project management tooling and using this system as a source for the support environment. When pressed about which of the items they missed most, primarily technological agreements and status are mentioned.

With respect to organization specific information items, billing status (customer relations) is a popular answer. Users mainly need this information to be able to decide whether bills have been sent to the customer on time and whether the customer is late with paying them. Additionally, action points are missed because it is difficult to have insight in what happens after it is agreed things are going to be taken care of. Finally with respect to personal information items, the main types of information that are missed are planning, current activity and approachability. The main value users see in knowing this information is that this makes it easier to know whether and how to contact people and to anticipate on actions of their colleagues in planning their own work.

E. To which extend is information overload encountered?

All the users that made extensive use of the event list, report experiencing an overload of information at least some of the time. The main cause of this is a single source system sending a lot of events shortly following each other about the same or related events. The interviewees did not have a proper way of dealing with this except for ignoring the event list in the period they experienced the overload. They propose the platform should deal with this by combining events that shortly follow each other and are related either because they are caused by the same user or are related with the same source system. Finally, one of the interviewees reported that he also experienced an overload of information because in his project the sources that create the events were not properly configured. Because of this, all the events related to his projects were not presented related to the correct projects in the platform.

F. To which extend is privacy an issue?

While all the interviewees agree the platform has made information more easily accessible and more transparent, no-one reported to ever have experienced the platform as a threat to his privacy. The most likely reason for this is that the company consists of a small and close community of people that trust each other.

G. What are the most important benefits for distributed software engineers of having all their information needs available in single place?

All of the interviewees reported seeing value in a platform such as Iris which aims to make all information a distributed software engineer needs, available in a single platform. Summarizing and aggregating what the interviewees say, the most important benefit of such a solution is: *While working in a distributed setting, (i) having insight in everything that is going on and (ii) how it is going, (iii) without having to ask for it or (iv) search for it, because it is available in a single location.*

VI. THREATS TO VALIDITY

In this study, a threat to external validity exists in the generalization from teams in a single software engineering company to the population of all distributed software engineering teams. To be able to better generalize beyond the setting we performed the study in, the study should be repeated in other companies as well.

Furthermore, there exist threats to construct validity in our study. Firstly, we attempted to mitigate threats to reliability by describing our research site and methods in sufficient detail and making the interview structure available online. Next to this we also make the platform we used in this study available upon request. We do this to make our data gathering methods repeatable. Subsequently, a threat to construct validity is mono-operation bias. Because we only used one specific support environment in our study one could argue the results only apply to the use of that tool. We mitigated this threat by providing an explanation of the platform we used and explicitly linking the functionality provided, to the information items that are supported. A final threat to construct validity in this study is that the interviews were conducted by the first author of the

paper who is also an employee of the company at which the study was performed.

Finally, there exist threats to internal validity. Firstly, while the default work location for the people at the company was home, on Tuesdays they mostly work together in a single location. Therefore they effectively worked co-located one out of each five potential work days, reducing the distributed nature of the software engineering team. Next to this, one of the people we interviewed also participated in the focus group in [3] and all of the interviewees have participated in the questionnaire of [3]. This could have biased the results due to a learning effect caused by repeated testing.

VII. CONCLUSIONS AND FUTURE WORK

In this paper we have reported on an empirical study about the support of the information needs of distributed software engineers. The main research question we have researched is:

“Does the introduction of a selection of information needs of distributed software engineers in a single platform, improve the awareness of software engineering teams?”

The main findings of this paper are:

- Providing information that distributed software engineers need in a single platform is valuable and appears to contribute to an increase in their overall level of awareness
- Information about project related communication with the customer and information about the happiness of colleagues adds the most value
- It is important to provide users with customization options to address personal preferences
- Providing training regarding use and proper configuration is important
- It is important for technological support to help users deal with an information overload
- It is important for an organization to decide how and what information they like to retain before attempting to make that information more accessible to distributed software engineers

Future work will concern processing the findings of this study. So, the technological support will be extended to support the most missed information items as well and the current support shall be improved based on the biggest limitations. Based on the information items that are missed most, the source systems we are planning to add first are: (i) project management tooling, (ii) the internal financial system used in the company (this includes the billing status) and (iii) the planning information stored in Google agenda. On the other hand, we are also interested in the Appreciative Inquiry approach which focuses on increasing what you do well over eliminating what you do badly [19]. This would concern investigating how the two most appreciated features (sharing project related communication with the customer and the happiness of colleagues) can be exploited further. Furthermore, we feel further research on the contextualization of information based on how related someone is to the information, is of high

importance as well. Finally, after implementing the findings in these proposed studies, the study presented in this paper should be repeated so the research questions can be re-evaluated in more detail.

REFERENCES

- [1] E. Carmel, *Global software teams: collaborating across borders and time zones*. Upper Saddle River: Prentice Hall PTR, 1999.
- [2] The Dieringer Research Group Inc., “Telework Trendlines 2009: A Survey Brief by WorldatWork,” 2009.
- [3] K. Dullemond and B. van Gameren, “What distributed software teams need to know and when: an empirical study (in press),” in *Proceedings of the 2013 International Conference on Global Software Engineering*. IEEE Computer Society Press, 2013.
- [4] P. Dourish and V. Bellotti, “Awareness and Coordination in Shared Workspaces,” in *Proceedings of the ACM 1992 Conference on Computer Supported Cooperative Work*. ACM Press, 1992, pp. 107–114.
- [5] C. Dentel, M. Nordio, and B. Meyer, “News and notification: Propagating relevant changes to developers,” *Software Engineering Laboratory: Open Source Eiffel Studio*, ETH Zürich, 2012.
- [6] J. C. Tang, “Findings from observational studies of collaborative work,” *International Journal of Man-machine studies*, vol. 34, no. 2, pp. 143–160, 1991.
- [7] J. T. Biehl, M. Czerwinski, G. Smith, and G. Robertson, “Fastdash: a visual dashboard for fostering awareness in software teams,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2007, pp. 1313–1322.
- [8] A. Mockus and J. D. Herbsleb, “Expertise browser: a quantitative approach to identifying expertise,” in *Proceedings of the 24th International Conference on Software Engineering*. ACM, 2002, pp. 503–512.
- [9] J. Yew, F. Gibson, and S. Teasley, “Learning by tagging: group knowledge formation in a self-organizing learning community,” in *Proceedings of the 7th international conference on Learning sciences*. International Society of the Learning Sciences, 2006, pp. 1010–1011.
- [10] M. Storey, L. Cheng, I. Bull, and P. Rigby, “Shared waypoints and social tagging to support collaboration in software development,” in *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*. ACM, 2006, pp. 195–198.
- [11] G. Booch and A. W. Brown, “Collaborative development environments,” *Advances in Computers*, vol. 59, pp. 1–27, 2003.
- [12] R. Frost, “Jazz and the eclipse way of collaboration,” *Software, IEEE*, vol. 24, no. 6, pp. 114–117, 2007.
- [13] R. Holmes and R. J. Walker, “Promoting developer-specific awareness,” in *Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering*. ACM, 2008, pp. 61–64.
- [14] K. Dullemond and B. van Gameren, “An industrial evaluation of technological support for overhearing conversations in global software engineering,” in *Proceedings of the 2012 International Conference on Global Software Engineering*. IEEE Computer Society Press, 2012.
- [15] K. Dullemond, B. van Gameren, M. Storey, and A. van Deursen, “Fixing the out of sight out of mind problem - one year of mood-based microblogging in a distributed software team,” in *Proceedings of the 2013 International Working Conference on Mining Software Repositories*. IEEE Computer Society Press, 2013.
- [16] K. Dullemond, B. van Gameren, and R. van Solingen, “Collaboration should become a first-class citizen in support environments for software engineers,” in *Proceedings of the 2012 International Conference on Collaborative Computing: Networking, Applications and Worksharing*. IEEE, 2012.
- [17] B. van Gameren, R. van Solingen, and K. Dullemond, “Auto-Erecting Virtual Office Walls,” in *Proceedings of the 2012 International Conference on Collaborative Computing: Networking, Applications and Worksharing*. IEEE, 2012.
- [18] A. Fontana and J. Frey, “The interview: From neutral stance to political involvement,” *The Sage handbook of qualitative research*, vol. 3, pp. 695–727, 2005.
- [19] F. Stowell and D. West, “The appreciative inquiry method,” in *Systems Thinking in Europe*. Springer, 1991, pp. 493–497.