



Group Awareness in Global Software Engineering

Filippo Lanubile, Fabio Calefato, and Christof Ebert

Insufficient team collaboration often challenges global software engineering projects. Group awareness can improve teams' trust, relationships, and efficiency. In this column, Filippo Lanubile, Fabio Calefato, and I survey the key technologies and tools that support group awareness and collaboration. The insights on technologies derive from discussions and presentations at related conferences, including the IEEE-sponsored International Conference on Global Software Engineering (ICGSE). I look forward to hearing from both readers and prospective column authors about this column and the technologies you want to know more about. —*Christof Ebert*

BOB AND ALICE are working in a distributed team that is developing a mobile app. Whereas Bob likes to frequently check his code changes and thus ensure growing functionality, Alice tries to see the bigger picture of usability and wants to first implement a consistent handling of all functions. In working with Bob, she's frustrated that he seemingly doesn't care about the overall performance—Bob doesn't reply to comments she inserts. Bob, on the other hand, sees no progress on her side and perceives the flood of comments without concrete change proposals as slowing

down their team's agile style. What's going on? Both have a different working style and neither adjusts to the other. They're unaware of each other's strengths, behaviors, and communication needs. Worse, they're unaware of the growing tensions, thus endangering the project, although both are individually trying to make it a success.

Group awareness has received considerable attention lately in distributed projects and global software engineering (GSE). From our research with GSE, we found that more than half of all distributed projects fail, most often owing to insufficient com-

munication and trust.¹ Awareness is necessary to coordinate group activities and ensure that individual contributions are relevant to the whole group. Paul Dourish and Victoria Bellotti described group awareness as “an understanding of the activities of others, which provides a context for your own activity.”² There are four types of group awareness:³

- *informal awareness*, also called presence awareness, provides information about who is around and their availability through IM and VoIP tools;

- *group-structural awareness* provides knowledge about team members' roles and teams' internal structures;
- *workspace awareness* provides information about team members' interactions with shared artifacts within a workspace; and
- *social awareness* refers to the information that team members maintain about others in a conversational context and the understanding that they have about social connections within a group.

Obviously, no technology can miraculously overcome culture challenges and poor management. However, a better awareness of what's going on in the group and better use of available technologies can help managers and their teams mitigate challenges with distributed projects.

Group Awareness Support in Application Life-Cycle Management Platforms

Application Life-cycle Management (ALM) is a continuous process of managing an application's life cycle through platforms that provide a project workspace with an integrated tool set encompassing all software development activities including requirements management, design, coding, testing, tracking, and release management.¹

Table 1 summarizes how the most popular ALM platforms support the four types of group awareness. All the listed platforms are native Web applications (except IBM Jazz and Microsoft Team Foundation Server, although these also come with rich desktop integrated development environments).

Trac

Trac is an ALM platform that takes a minimalistic approach to project life-cycle management. It combines an integrated wiki, an issue tracking system,



SocialTFS

Application Life-cycle Management (ALM) platforms either lack social awareness support altogether or, in the case of Jazz and GitHub, provide support that isn't as adequate as that for the other forms of group awareness. To fill this gap, Microsoft Research Software Engineering has recently funded the development of a community add-on for Visual Studio called SocialTFS (Team Foundation Server), under the Microsoft Innovation Foundation Award. SocialTFS aggregates teammates' content from popular social networks, such as Facebook, LinkedIn, and Twitter, into the developer's workspace. It doesn't require any explicit following action because followings are automatically added and removed, depending on what work items and artifacts are under change.¹

Reference

1. F. Calefato and F. Lanubile, "Augmenting Social Awareness in a Collaborative Development Environment," *5th Int'l Workshop Cooperative and Human Aspects of Software Eng. (CHASE 12)*, IEEE, 2012, pp. 12–14.

and a front-end interface to a Subversion source code management system, with plug-ins providing other features. Through email messages and RSS feeds, group-awareness features support workspace awareness by providing notifications of project events and changes. Regarding group-structural awareness, in line with its minimalistic approach philosophy, Trac only allows project administrators to manage developers' privileges to view, create, modify, and delete artifacts. To our knowledge, no plug-ins are currently available for adding either informal or social awareness to the environment. (For more on social awareness support, see the "SocialTFS" sidebar.)

Google Code

Google Code offers a project-hosting service similar to Trac. However, unlike Trac, Google Code is a hosted service and as such can't be extended through plug-ins. It supports both workspace awareness, by email and RSS notifications, and group-structural awareness, by managing permissions to edit artifacts. Google Code doesn't support social or informal awareness.

Assembla

Assembla is a Web-based ALM platform that hosts both open source and commercial software. Assembla improves on other Web-based environments in a few notable aspects. First, with respect to group-structural awareness, Assembla supports teams adopting agile development processes for running Scrum meetings. In addition, it provides a more sophisticated team-management feature, which lets project managers define teams and roles. Second, as for workspace awareness, notifications of changes within a workspace are available via Twitter, in addition to email and RSS feeds. Third, Assembla supports synchronous communication via instant messaging over the Extensible Messaging and Presence Protocol (XMPP)/Jabber protocol. However, Assembla doesn't support social awareness.

Jazz

IBM Rational developed Jazz as an extensible platform that's part of a development suite of tools called Collaborative ALM. Jazz has two clients: one is a Web application for project

TABLE 1

Group awareness in Application Life-cycle Management platforms.

	Informal awareness	Group-structural awareness	Workspace awareness	Social awareness
Trac URL: http://trac.edgewall.org License: BSD (open source) Service: owned Client: Web Pluggable: server-side		Yes	Yes (email, RSS)	
Google Code URL: http://code.google.com License: free to use Service: hosted Client: Web Pluggable: no		Yes	Yes (email, RSS)	
Assembla URL: www.assembla.com License: commercial Service: hosted Client: Web Pluggable: no	Yes (XMPP)	Yes	Yes (email, RSS, Twitter)	
Jazz URL: http://jazz.net/projects/rational-team-concert License: commercial (free ≤ 10 members) Service: owned Client: Web, Rational Team Concert Pluggable: client-side, server-side	Yes (XMPP, Skype)	Yes	Yes (email, RSS)	Partial (IBM Connections)
TFS URL: http://microsoft.com/visualstudio/alm License: commercial (free ≤ 5 members) Service: owned Client: Web, Visual Studio Pluggable: client-side	Yes (SIP/SIMPLE, Skype)	Yes	Yes (email, RSS)	
CodePlex URL: www.codeplex.com License: free to use Service: hosted Client: Web, Visual Studio Pluggable: no		Yes	Yes (email, RSS)	
GitHub URL: http://github.com License: free to use Service: hosted Client: Web Pluggable: no		Yes	Yes (email, RSS)	Partial (@mentions)

management tasks and the other is a full-featured desktop application called Rational Team Concert (RTC) that's built on the Eclipse Rich Client Platform (RCP) and oriented to developers. As for group-awareness sup-

port, Jazz provides workspace awareness through email notification and an RSS reader natively built in RTC. In addition, Jazz provides the most complete process- and team-management features to support group-structural

awareness among the reviewed environments. It also supports informal awareness by integrating with Lotus Sametime, Google Talk, and Skype instant-messaging networks. Finally, Jazz provides partial support of so-

cial awareness, thanks to its integration with the IBM Connections suite, because it creates a project community to which all project members are added as soon as they join the project. However, there's a risk of information overload because the team members' activity streams can't be limited to the awareness network—that is, the set of colleagues whose actions one monitors and to whom one's actions are displayed.⁴ This awareness network is highly dynamic because the set of colleagues continually changes over time depending on task assignments or software development phases.

Team Foundation Server

Team Foundation Server (TFS) is Microsoft's ALM platform. Its characteristics and features are similar to those of IBM Jazz; in fact, they both come with a lightweight Web client for management tasks and a rich client desktop for more development-oriented tasks. This desktop is called Team Explorer and is an extension of the Visual Studio integrated development environment. TFS provides support levels similar to Jazz with respect to group-structural and workspace awareness. It also supports informal awareness, thanks to a couple of community-contributed plug-ins that enable connections to Microsoft Messenger, Lync, and Skype networks. TFS lacks any support for social awareness.

CodePlex

CodePlex is a customized public TFS installation that Microsoft provides to host open source projects. It's similar to the other Web-based ALM platforms, supporting both workspace and group-structural awareness, but not social and informal awareness. Compared to other Web-based platforms, CodePlex's only peculiar feature is its ability to connect to Visual Studio.

GitHub

GitHub is a platform built on Git as the source code management system for both open source and commercial software projects. Like the other Web-based ALM platforms, GitHub offers lightweight group-structural and workspace awareness. GitHub, which describes itself as a “social network

and interpersonal connections with distant colleagues.

Lack of Group Awareness: A Typical GSE Scenario

Consider the following typical GSE scenario in which global software team members face challenges due to a lack of social awareness: Bob is a member

More than half of all distributed projects fail, most often owing to insufficient communication and trust.

for programmers,” aims to foster developers' collaboration by letting them fork projects and monitor development through a Twitter-like approach, giving them the chance to follow (watch) projects. Finally, GitHub partially supports social awareness by allowing developers to interact (via “@mentions”) with others.

Social Awareness and Global Software Teams

Social awareness would be most helpful to global software teams if they have collaborative development tools that support sharing personal and contextual information to increase the likelihood of successful interactions in global projects. Typically, team members connect through close interaction and face-to-face communication; such activity is largely reduced on global teams owing to distance. Therefore, collaborative development tools could be realized as plug-ins of extensible ALM platforms, such as TFS and Jazz, to provide global teams with “socially augmented” environments that simultaneously facilitate both development-related activities

of a team developing a mobile app for a home-automation project. He's also taking the responsibility of developing the app component that handles the home-security system settings. The distributed team is using an ALM platform to coordinate project development. After reviewing the first incremental release of the app, the customer raised some concerns about the current solution's usability—for example, regarding the user interface. Therefore, Tim, the project manager, decides to involve Alice, one of the company's usability experts; he assigns to her all the work items related to usability issues, including those Bob is already handling. Tim also requires that Alice approve all the commits with changes applied to the user interface.

Without social awareness support.

With the milestone integration deadline approaching, Bob performs an update of his local repository. He notices that Alice has committed some change sets to resolve the work items previously assigned to him. Looking at the comments in the change sets, he notices that she frequently pointed

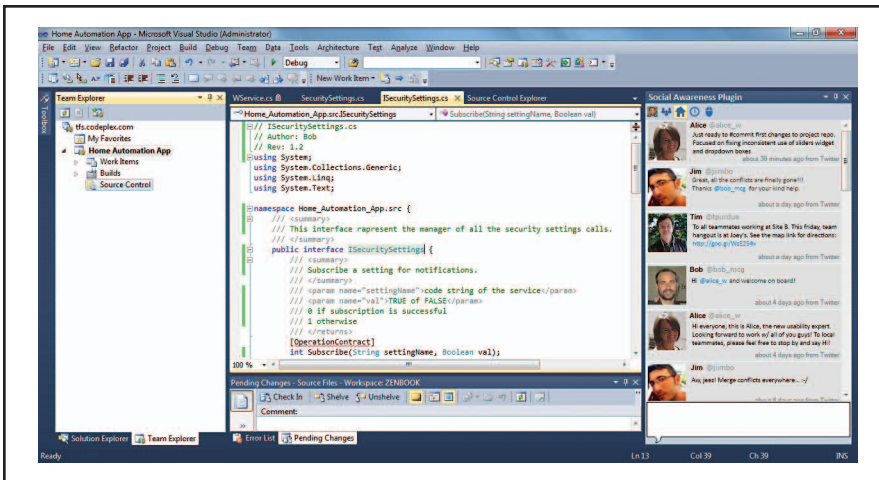


FIGURE 1. TFS and Visual Studio augmented with a social awareness plug-in.

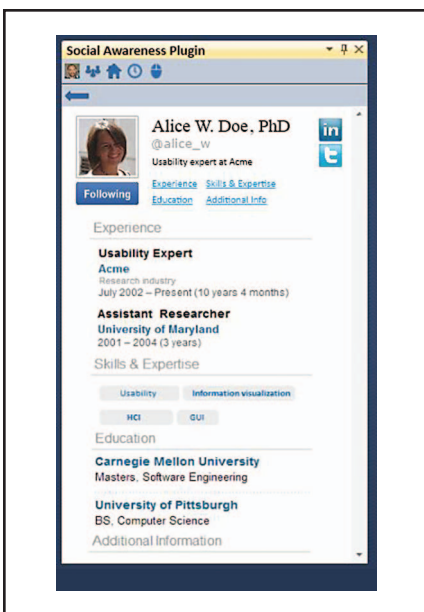


FIGURE 2. A user's profile with connected services, education, and work-related experience.

out that previous versions made an inconsistent use of slider widgets and drop-down boxes. Bob looks at some of the user interface views that Alice modified and sees no big difference compared to the ones he created in the first place. So, he decides to leave the new view he has been coding un-

changed and commit. With this behavior, Alice and Bob now face the same tensions as were prevalent a decade ago, when agile techniques were used ad hoc without awareness of other team members' reasoning.

With social awareness support. The social awareness plug-in installed in the ALM platform regularly updates developers' awareness networks to incorporate the posts from the people who are involved in the same work items (for example, those who have reported or commented on the same issue). So, Alice's posts begin to appear in Bob's workspace (see Figure 1). Other than Twitter, Alice has connected her LinkedIn account to the plug-in, so Bob can also see from her profile that Alice is very well known in the human-computer interaction community (see Figure 2). When Alice tweets that she's happy to have a chance to work with the team on the new mobile platform; Bob decides to reply to her tweet with a "welcome aboard" message. As they communicate more, Bob learns from Alice's posts that she has been busy fixing the user-interface views that Bob had coded. When Bob updates his local

repository, he notices that Alice has committed some change sets to resolve the work items previously assigned to him. Looking at the comments in the change sets, he notices that she has frequently pointed out that previous versions made an inconsistent use of slider widgets and drop-down boxes. Therefore, when Bob looks at some of the user-interface views that Alice modified, he realizes that she is right and that her reputation as a usability expert is well deserved. Therefore, he decides also to apply the same solution to the new view before committing. Alice then reviews and approves the change set committed by Bob.

Developers in distributed teams need to maintain both a general awareness of their entire teams and a more detailed knowledge of the people that they work with. Here are few hints for better group awareness:

- *Informal awareness.* Check regularly on who is around and communicate directly. Use instant messaging and VoIP tools for instant question resolution; don't rely on submitting comments via a change tool or emails.
- *Group-structural awareness.* Ensure a clear understanding about team members' roles and teams' internal structures. Use responsibility matrixes to visualize team responsibilities and track commitments. Avoid ad hoc assignments to simply get work done faster—this frustrates those who follow the process.
- *Workspace awareness.* Make clear who's working on which shared artifacts within a workspace. Use Scrum-like processes to get visibility on the current tasks and interfaces.
- *Social awareness.* Stimulate social

connections within the group—this isn't solely the manager's responsibility. As a group member, you can do a lot to connect and collaborate on personal levels. Even if socializing only happens remotely, it can be quite beneficial.

As Table 1 shows, ALM platforms still lack the necessary social awareness support. A socially augmented ALM platform would counteract such challenges by facilitating interpersonal connections and strengthening ties between distant colleagues—for instance, by securely sharing personal and contextual information within the workspace. Such technology for software development teams is still in its infancy. The current ad hoc use of social media in companies highlights that more concrete guidance and empirical research are necessary to both drive effective use and mitigate obvious risks related to security and inefficiency. ☞

References

1. C. Ebert, *Global Software and IT: A Guide to Distributed Development, Projects and Outsourcing*, Wiley, 2012.
2. P. Dourish and V. Bellotti, "Awareness and Coordination in Shared Workspaces," *Proc. Int'l Conf. Computer Supported Cooperative Work (CSCW 92)*, ACM, 1992, pp. 107–114.
3. C. Gutwin, S. Greenberg, and M. Roseman, "Work-space Awareness in Real-Time Distributed Groupware: Framework, Widgets, and Evaluation," *Proc. HCI on People and Computers XI (HCI 96)*, Springer, 1996, pp. 281–298.
4. C.R.B. de Souza and D.F. Redmiles "The Awareness Network: To Whom Should I Display My Actions? And, Whose Actions Should I Monitor?," *IEEE Trans. Software Eng.*, vol. 37, no. 3, 2011, pp. 325–340.



IEEE INTERNATIONAL CONFERENCE ON GLOBAL SOFTWARE ENGINEERING

The annual IEEE International Conference on Global Software Engineering (ICGSE) brings together industry and research from around the world, providing the leading forum for addressing topics such as how to make distributed teams more effective and efficient and how to cope with challenges created by such distributed teams, such as different methods and tools. The 2012 conference had participants from more than 20 countries with a third of the papers from industry.

ICGSE 2013 will take place 26–29 August in Bari, Italy. Join the conference and learn how to overcome challenges in distributed software projects. For more information, visit www.icgse.org.

FILIPPO LANUBILE is an associate professor of computer science at the University of Bari. He is the General Chair of ICGSE 2013. Contact him at filippo.lanubile@uniba.it.

FABIO CALEFATO is a postdoc at the University of Bari where he received a PhD in computer science. Contact him at fabio.calefato@uniba.it.

CHRISTOF EBERT is managing director at Vector Consulting Services. He's a senior member of IEEE and the editor of the Software Technology department of *IEEE Software*. Contact him at christof.ebert@vector.com.



Now Available in Enhanced Digital Format

More value, more content, more resources

The new multi-faceted *IEEE Software* offers exclusive video and web extras that you can access only through this enhanced digital version. Dive deeper into the latest technical developments with a magazine that is:

- Searchable**
- Linked**
- Engaging**
- Mobile**

computer.org/software

IEEE **IEEE computer society**