Facilitating Communication between Engineers with CARES

Anja Guzzi
Delft University of Technology
Delft, The Netherlands
a.guzzi@tudelft.nl

Andrew Begel
Microsoft Research
Redmond, WA USA
andrew.begel@microsoft.com

***Abstract*—When software developers need to exchange information or coordinate work with colleagues on other teams, they are often faced with the challenge of finding the right person to communicate with. In this paper, we present our tool, called CARES (Colleagues and Relevant Engineers' Support), which is an integrated development environment-based (IDE) tool that enables engineers to easily discover and communicate with the people who have contributed to the source code. CARES has been deployed to 30 professional developers, and we interviewed 8 of them after 3 weeks of evaluation. They reported that CARES helped them to more quickly find, choose, and initiate contact with the most relevant and expedient person who could address their needs.**

***Keywords*-coordination, CSCW, software engineering**

I. Introduction

Large software products are often challenging to develop due to hard-to-manage coordination requirements [1] and difficult-to-find information about the software [2], [3]. Long-lived software can especially frustrate developers because design decisions made in the past materially affect their work today [4].

Effective, efficient communication is the key to addressing these problems, but its realization in the software engineering domain has proven elusive [5]. One issue is that finding a person to speak with, who is relevant to one's information and coordination needs, requires a substantial amount of effort [6].

In July 2011, we surveyed 94 professional software developers at a large US-based software company about their communication practices working with engineers on teams with whom they collaborated [7]. Our results indicated that developers need to communicate quite often, 2–5 times per work task, and generally at least once a day during programming tasks. They found existing IDE and office communication tools made finding relevant engineers slow and tedious. First, they had to retrieve the set of people who contributed to a file or project, and then, look each person up, one-by-one, in the corporate address book to learn his or her name, email address, manager, office address, and department. Also, before contacting one of the contributors, they needed to look up his or her availability from the office calendar and instant messaging tools. We found that each developer uses a unique, fairly complex algorithm for combining the file's historical information (as used by Kagdi, *et al.* [8]), with organizational and availability information, simply to decide which person would be best and most expedient at helping fulfill his or her needs.

Related work in developing software-related communication tools, such as FastDash [9], Palantir [10], Jazz [11], and CollabVS [12], typically focuses on real-time collaboration and awareness, which presumes that everyone works on the same project at the same time, and has concordant goals. The developers that we studied, however, mainly collaborate on small feature teams that work independently on mutually dependent projects. Work-related communication between engineers on different teams occurs only as needed to transfer information or coordinate shared work.

To help developers like these to work more effectively with related teams, we created CARES, Colleagues and Relevant Engineers' Support, a new tool designed specifically for software engineers who want to communicate with others about source code they see in their IDEs. In each editor window of the IDE, CARES displays a context-sensitive list of photos of the engineers who are directly connected to the code in that file, and would be the best set of people to respond to another's request. To help developers select the most relevant person with whom to communicate, CARES augments each connected person's photo with a tooltip containing crucial information about him or her, such as his or her past checkins, place in the organization, physical location, and availability.

We enlisted 30 professional software developers from the same company to pilot CARES for 3 weeks [7]. Afterwards, we randomly selected and interviewed 8 of them. They reported that CARES made it easier and quicker for them to find and contact others, especially when they did not already know who the relevant people were or how they had contributed to the code. One said that by showing photos of engineers, CARES made it easier to figure out who people were versus just seeing a name or email address. Two thought that CARES would enable more people to ask *them* questions, as they worked on core parts of their product's code that everyone else in the organization depended on. Another showed it to his manager, who wanted to get everyone on his team to use it. None of the 8 uninstalled the tool.

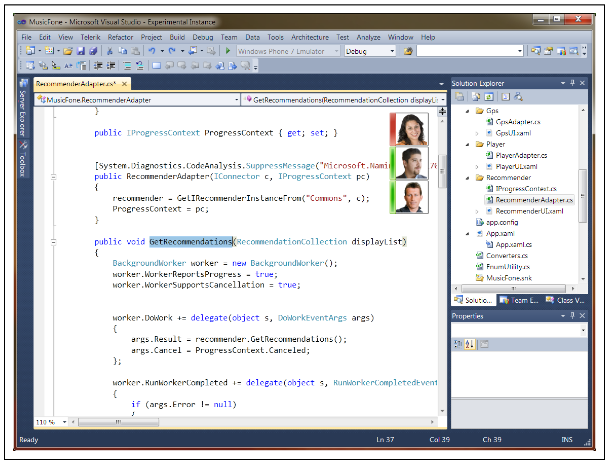
In this demo, we show how CARES can help developers discover and contact relevant colleagues to help find information and coordinate action. We explain how CARES works, and how it can be extended in the future to show other kinds of information relevant to developers' needs.

II. Demo Walk-Through

In this section, we introduce Jane, a fictitious character inspired by the developers we interviewed, and follow her as she performs one of her tasks, supported by the CARES tool.

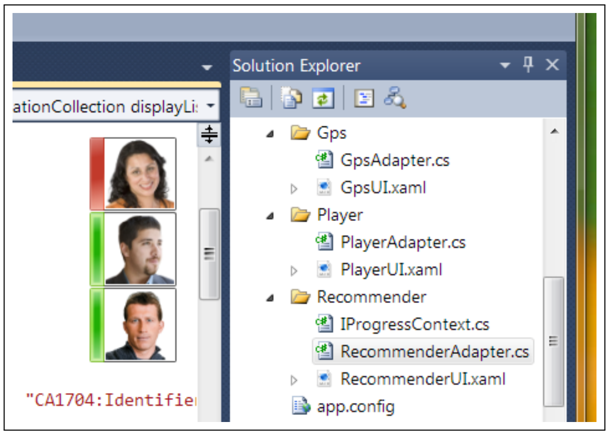
Jane is an experienced developer who has just changed teams for the first time in 3 years after successfully shipping her last product. She is assigned the job of designing a new architecture for a common infrastructure platform to support her new team's long-lived suite of internal software tools. To design this properly, she must find and understand the requirements, scenarios, implementations, and bugs of all the old tools. Various members of her new team have been helping her get started by sending her pointers to the top-level directories where each tool is located.

1) Jane opens an email with a URL in it that says very simply, "This directory contains the code for the IT administration tool." Jane is not sure what this tool really does, but she points her IDE at the directory in the repository, checks out the code, and opens up the project. Jane thinks to herself, "where should I begin?" Perhaps she can discover something about this tool's methods from the filenames. She sees one file that is suggestively named, `RecommenderAdapter.cs`, and decides to open it.

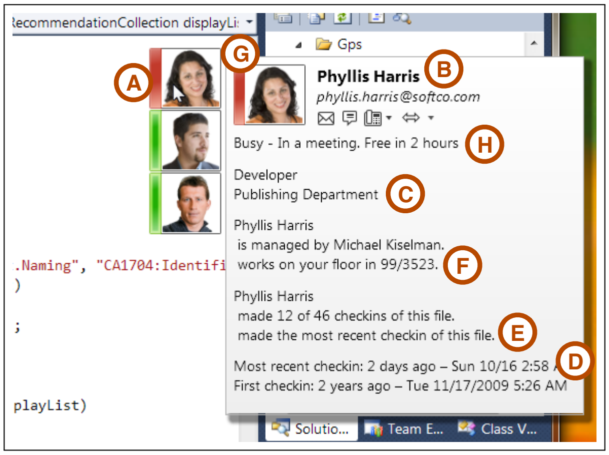

MusicFone - Microsoft Visual Studio - Experimental Instance
File Edit View Telerik Refactor Project Build Debug Team Data Tools Architecture Test Analyze Window Help
Windows Phone 7 Emulator
Debug
RecommenderAdapter.cs*
MusicFone.RecommenderAdapter
GetRecommendations(RecommendationCollection displayLi:
Server Explorer
Toolbox
Solution Explorer
Properties
110 %
Ready
Ln 37
Col 39
Ch 39
INS

```
}

public IProgressContext ProgressContext { get; set; }


[System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Nami
public RecommenderAdapter(IConnector c, IProgressContext pc)
{
    recommender = GetIRecommenderInstanceFrom("Commons", c);
    ProgressContext = pc;
}

public void GetRecommendations(RecommendationCollection displayList)
{
    BackgroundWorker worker = new BackgroundWorker();
    worker.WorkerReportsProgress = true;
    worker.WorkerSupportsCancellation = true;


    worker.DoWork += delegate(object s, DoWorkEventArgs args)
    {
        args.Result = recommender.GetRecommendations();
        args.Cancel = ProgressContext.Canceled;
    };

    worker.RunWorkerCompleted += delegate(object s, RunWorkerCompletedEvent
    {
        if (args.Error != null)
```

She sees a `GetRecommendations()` method that uses library methods she has never seen before. "What could this be for?" Jane wonders. "Maybe I can find a person who knows enough about the design to help me understand its purpose."



2) In the upper-right corner of `RecommenderAdapter.cs`' editor window, she sees the CARES visualization, which shows the photos of three people, a woman and two men. She knows the woman, but has never met the two men. She hopes that one of the three might be able to help.


A
B
C
D
E
F
G
H
ecommendationCollection displayLi:
Gps
.Naming", "CA1704:Identifi
)
;
playList)
Solutio...
Team E...
Class V...

Phyllis Harris
phyllis.harris@softco.com

Busy - In a meeting. Free in 2 hours

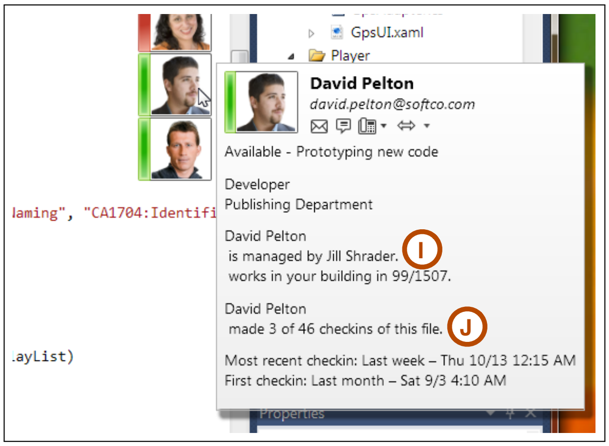
Developer
Publishing Department

Phyllis Harris
is managed by Michael Kiselman.
works on your floor in 99/3523.

Phyllis Harris
made 12 of 46 checkins of this file.
made the most recent checkin of this file.

Most recent checkin: 2 days ago – Sun 10/16 2:58
First checkin: 2 years ago – Tue 11/17/2009 5:26 AM

3) She figures that her friend, Phyllis, has worked on the file most recently (CARES sorts people's photos, with the person that made the most recent checkin on top), and hovers her mouse over her photo (A). A tooltip shows that she is Phyllis Harris (B), a developer in the same Publishing Department (C) that Jane works in. Phyllis worked on the code only 2 days ago (D), and is the most recent person to make a checkin (E). "Surely, Phyllis must know what this function is doing." Even better, Phyllis works nearby on Jane's floor (F) and it would be easy to walk over to meet her in person. But, Phyllis' presence indicator is red (G); she is in a meeting (H) and unavailable to help right now.



```
laming", "CA1704:Identifi
.ayList)
```

David Pelton
david.pelton@softco.com

Available - Prototyping new code

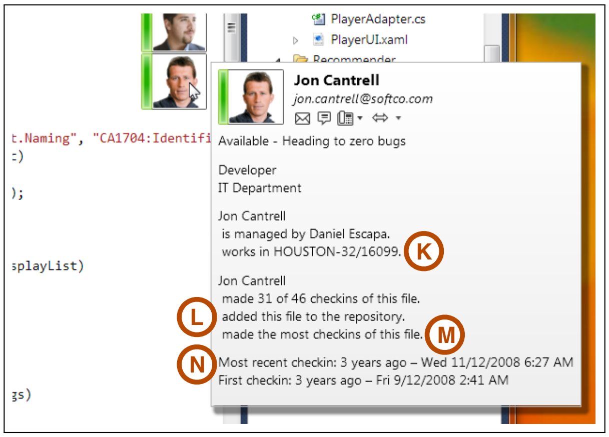
Developer
Publishing Department

David Pelton
is managed by Jill Shrader.
works in your building in 99/1507.

David Pelton
made 3 of 46 checkins of this file.

Most recent checkin: Last week – Thu 10/13 12:15 AM
First checkin: Last month – Sat 9/3 4:10 AM

4) Jane moves on to the next photo, David Pelton, another developer in the same department. David works for Jill (I), one of Jane's friends since she started working at the company. David worked on the code last week, but he has not done much (J); "he probably does not yet have a complete understanding of the code."



```
t.Naming", "CA1704:Identifi
:)
);
splayList)
gs)
```

Jon Cantrell
jon.cantrell@softco.com

Available - Heading to zero bugs

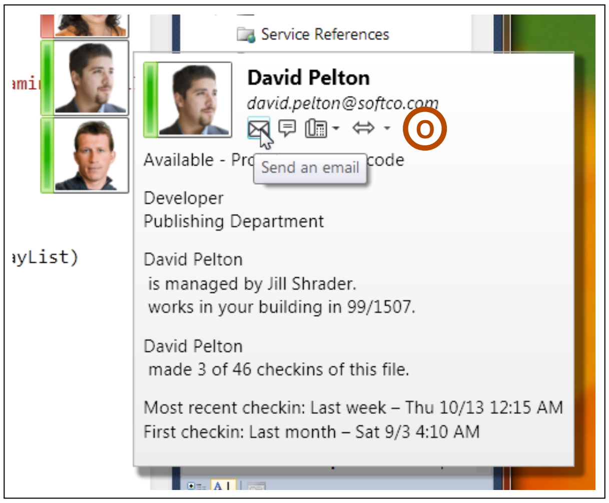
Developer
IT Department

Jon Cantrell
is managed by Daniel Escapa.
works in HOUSTON-32/16099.

Jon Cantrell
made 31 of 46 checkins of this file.
added this file to the repository.
made the most checkins of this file.

Most recent checkin: 3 years ago – Wed 11/12/2008 6:27 AM
First checkin: 3 years ago – Fri 9/12/2008 2:41 AM

5) The last person listed is Jon Cantrell, a developer in the IT department. He works in Houston (K), far away from Jane, and Jane does not know anyone in the IT department. Jon added the file to the repository (L), and made the most checkins of that file (M). Jane believes that he was the original author of the codebase, however he has not touched this code in three years (N). Jane concludes, "Since he has moved on to a different part of the company, he probably transferred ownership of the code to someone else. I am sure he would rather not be bothered about this unless it was absolutely necessary."



```
ami
ayList)
```

David Pelton
david.pelton@softco.com

Available - Prototyping new code

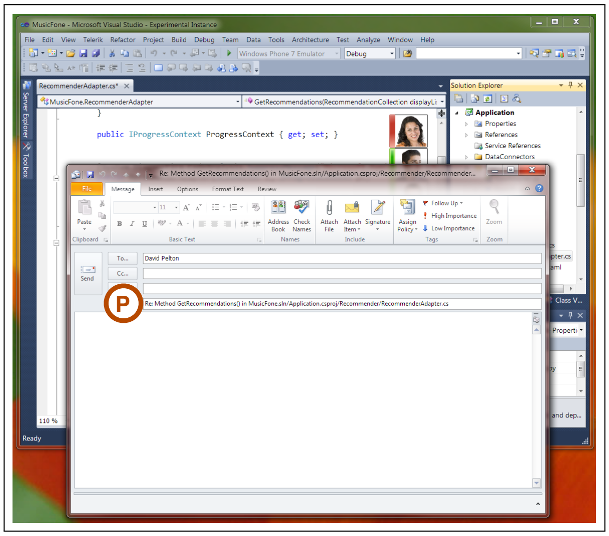
Developer
Publishing Department

David Pelton
is managed by Jill Shrader.
works in your building in 99/1507.

David Pelton
made 3 of 46 checkins of this file.

Most recent checkin: Last week – Thu 10/13 12:15 AM
First checkin: Last month – Sat 9/3 4:10 AM

6) Since Phyllis is not available, and Jon is no longer an option, Jane decides that she will contact David (O). Though he is available, Jane feels uncomfortable sending an instant message to a new colleague. Instead, she writes him an email to introduce herself, and then asks her question.



```
}

public IProgressContext ProgressContext { get; set; }
```

Re: Method GetRecommendations() in MusicFone.sln/Application.csproj/Recommender/Recommender...

To...: David Pelton

Re: Method GetRecommendations() in MusicFone.sln/Application.csproj/Recommender/RecommenderAdapter.cs

CARES helps Jane contextualize her email for David by filling in the current method and file name as the subject of the message (P). After a few minutes, David responds to Jane's email, and says that he is available for the next hour in his office. Jane heads over to meet David in person to learn more about the code.

By giving Jane the opportunity to read information about those who have worked on the source code, CARES made it possible for her to quickly discover and contact a relevant co-worker who can help her.

A video of this walk-through is available at *http://research.microsoft.com/~abegel/cares/demo.mp4*.

III. Implementation

CARES is the first instance of a series of lightweight tools in the Codebook family of applications [6]. It is realized as a Visual Studio 2010 editor component extension. Its list of photos is a view-relative, embedded editor adornment (*i.e.,* a graphic effect layered on top of the text view that always remains at the same position on screen) which shows as many photos as fits into the vertical space of the editor (though, during our deployment study, no one had files requiring more than 5 photos). To a first approximation, when a file from the source code repository is opened into the editor (detected using Visual Studio's built-in source code control extensibility object), a call is made in a background thread to the repository to fetch a list of all of the file's checkins. CARES collects the list of unique committers, and with each, associates the number and range of dates of their checkins, and fetches their company-managed, employee data (including their photo) from the repository. If the repository's committers also have Microsoft Windows identities, publically available extended information is pulled from the organization's Active Directory.

When the user hovers over a photo, a tooltip-like element (a Windows Presentation Foundation (WPF) Popup with a tooltip-like visual appearance and behavior) is shown that contains individual and aggregate information about the person, their relationship to the file and the user, and optionally, their relationship with the other people being shown. When the Visual Studio user is running the Microsoft Lync communication software, each person's picture is augmented with availability information drawn from the Lync instant messenger status. Buttons in the Popup can then enable the user to initiate communication with the person via email, IM, phone, and screen sharing.

To ease its adoptability, CARES needs no configuration. It pulls required information from Visual Studio's built-in, preconfigured extensibility objects. CARES is robust to the loss or recovery of connectivity to the source code repository and Lync server, gracefully degrading the user interface when needed data is unavailable.

As prototyped, this demo shows just one example of the information that CARES is capable of displaying. Its architecture employs the managed extensibility framework (MEF) to enable plugins, such as our source code repository plugin, to provide additional information about the current editor context. This context includes the current program element, file, project, solution, user, and any other associated people. Plugin authors can define alternative tooltip displays, such as showing number of changed lines per method instead of number of commits per file, using WPF XAML interface declaration files, without the need to alter the core source code of the tool.

For more information on CARES, please see *http://research.microsoft.com/cares*.

References

[1] K. Nakakoji, Y. Ye, and Y. Yamamoto, "Comparison of coordination communication and expertise communication in software development: motives, characteristics, and needs," in *JSAI-isAI*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 147–155.

[2] A. J. Ko, R. DeLine, and G. Venolia, "Information needs in collocated software development teams," in *ICSE*, Minneapolis, MN, USA, 2007, pp. 344–353.

[3] J. Sillito, G. C. Murphy, and K. De Volder, "Questions programmers ask during software evolution tasks," in *SIGSOFT/FSE*. New York, NY, USA: ACM, 2006, pp. 23–34.

[4] J. E. Burge, J. M. Carroll, R. McCall, and I. Mistrk, *Rationale-Based Software Engineering*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[5] R. E. Kraut and L. A. Streeter, "Coordination in software development," *CACM*, vol. 38, no. 3, pp. 69–81, 1995.

[6] A. Begel, Y. P. Khoo, and T. Zimmerman, "Codebook: Discovering and exploiting relationships in software repositories," in *ICSE*, Cape Town, South Africa, 2010, pp. 125–134.

[7] A. Guzzi and A. Begel, "Who cares about my work?" In Submission.

[8] H. Kagdi, M. Hammad, and J. Maletic, "Who can help me with this source code change?" in *ICSM*, October 2008, pp. 157–166.

[9] J. T. Biehl, M. Czerwinski, G. Smith, and G. G. Robertson, "FASTDash: a visual dashboard for fostering awareness in software teams," in *CHI*. New York, NY, USA: ACM, 2007, pp. 1313–1322.

[10] A. Sarma, Z. Noroozi, and A. van der Hoek, "Palantír: raising awareness among configuration management workspaces," in *ICSE*. Portland, Oregon: IEEE Computer Society, 2003, pp. 444–454.

[11] S. Hupfer, L.-T. Cheng, S. Ross, and J. Patterson, "Introducing collaboration into an application development environment," in *CSCW*. Chicago, IL: ACM Press, 2004, pp. 21–24.

[12] R. Hegde and P. Dewan, "Connecting programming environments to support ad-hoc collaboration," in *ASE*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 178–187.