

# Agile Record

The Magazine for Agile Developers and Agile Testers



**Distributed Team Management**

August 2013

[www.agilerecord.com](http://www.agilerecord.com)

free digital version

made in Germany

ISSN 2191-1320

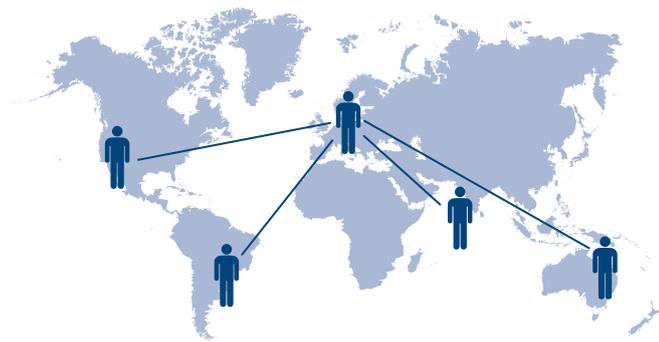
**issue 15**

# Thoughts on Distributed Teams

by *Huib Schoots*

***“Coming together is a beginning. Keeping together is progress. Working together is success.” – Henry Ford***

Right after my first cup of coffee, I log in to the company server from my kitchen table. I read my email and get myself another cup of coffee. Then I take a look at the issue management tool to see if the testers in Bangalore found many bugs while I was sleeping. I also check the online Scrum board to see the progress of the Sydney development team. While reading, a Skype call comes in and the team lead from Bangalore wants to talk. After talking for 15 minutes, I am fully updated on the testing done last night. The dashboard tells me development is on schedule and the nightly build of the interfaces was successful. This will be a productive day. I take another cup of coffee while I upload the latest documentation onto the server and prepare the meeting with the teams in Brazil and California later today ...



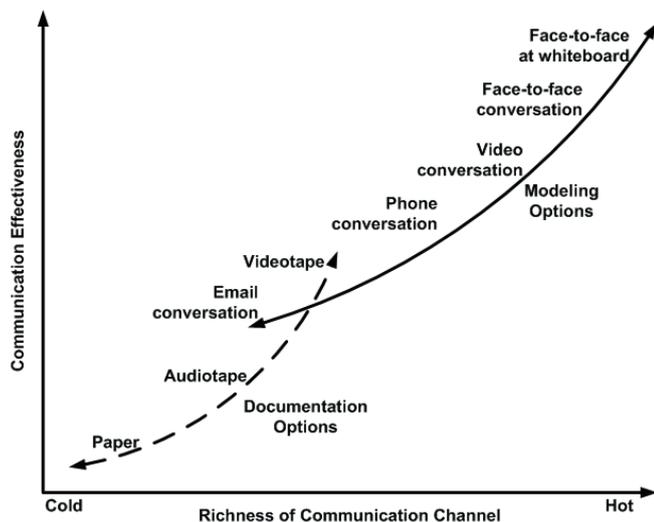
Do you recognize this story? This could be any regular day of an IT professional. Working from home while working with people all around the world is quite normal. Nowadays teams collaborate using evermore advanced tools. There are many different web-based and online tools: wikis, dashboards, Scrum boards, file sharing, knowledge sharing, workflow, conference, and meeting tools make collaboration via the internet easy, it seems. But is it really true? Let's have a closer look at distributed teams and the way they collaborate.

But first, what makes a team successful? Successful teams are all about people! Teams will succeed if the members trust one another, are persistent, and have passion and the willingness to help each other. This can lead to great collaboration, which will lead to success. If you google the success factors of collaboration you'll find many articles listing factors such as leadership, clear goals, principles, working process, accountability, transparency, information sharing, communication, relationship, membership, trust, and so forth.

Is all of this possible when people are in different time zones? Sure. But I don't think it will be very effective. Do you? When Steve Jobs worked at Pixar, he needed his employees to collaborate more. He decided to put everybody in the same building with only one pair of bathrooms. In this way, he forced people to run into each other seemingly accidentally. He needed people to mix, because he knew that mixing skills makes better movies. He also knew that the natural human tendency is to stay isolated and only talk to people just like yourself. His ploy worked! The results were instantaneous. Communication improved and with that so did collaboration.

So what does this anecdote tell us? It tells us that people need to run into each other to be really successful. Organizations need to make mixing as simple and obvious as possible. Get people together on regular basis, give them time to build trust and get to know each other. Effective communication is in the details, and to recognize these you have to know your team, so spend as much time together as possible.

I've worked on many international projects in my career. And without exception projects always went more smoothly when I was on-site. Collaboration improved, communication was much better, and work got done faster. A project in Jakarta I did last year had the same pattern: at the start I only was there for a week every month; later I was there full-time. The collaboration improved dramatically! Why? Because we got to know each other better and we took time to overcome our cultural differences.



Copyright 2002-2005 Scott W. Ambler  
Original Diagram Copyright 2002 Alistair Cockburn

When not in the same physical location, people can hide more easily. As I said before: people tend to isolate themselves. Communication gets more difficult and less rich, as the diagram above illustrates. The things everyone is working on become less transparent, and to overcome this issue we use our tools in combination with more comprehensive documentation. In my experience, teams rely too much on documentation and tooling. These can make it harder to keep everything up-to-date and information isn't shared as easily.

Yet another important aspect is working with a teammate. I love pairing employees and stimulate people by pairing them in every project I do. Team members can learn a lot from each other. I have seen many examples of two people getting into flow together, working quickly, and delivering awesome quality.

Such a simple change helps to increase discipline, quality, focus, morale, knowledge sharing, and learning – and it is much more fun. Pairing can only be done in the same location, so distributed teams can't do this effectively.

Distributed teams can work, but be aware of the consequences. Working together is about relations, communication, and people. When people are far away from each other but still have to work together, there are many excellent ways, including various tools, to make that possible. Yet, no matter how well the tools are working and no matter how passionate and talented your colleagues are, distributed teams are always suboptimal. ■

## > about the author

### Huib Schoots



Huib Schoots is an agile test consultant at code-centric, where he shares his passion for testing and agile through coaching, training, and presenting on a variety of subjects. With 15 years of experience in IT and software testing, Huib is experienced in many different testing roles. Curious and passionate, he is an agile and context-driven tester attempting to read everything ever published on software testing. A member of the Dutch Exploratory Workshop on Testing, blackbelt in the Miagi-Do School of software testing, and co-author of a book about the future of software testing, Huib maintains a blog at [magnifiant.com](http://magnifiant.com).

Curious and passionate, he is an agile and context-driven tester attempting to read everything ever published on software testing. A member of the Dutch Exploratory Workshop on Testing, blackbelt in the Miagi-Do School of software testing, and co-author of a book about the future of software testing, Huib maintains a blog at [magnifiant.com](http://magnifiant.com).

