
Agile Environments in Action

By Michael C. Marley, Steve Denning, and Mark W. Johnson

Illustrations by Mark W. Johnson

- » Creating your agile workspace
- » Rediscovering low-tech communication and using the right high-tech communication
- » Finding and using the tools you need

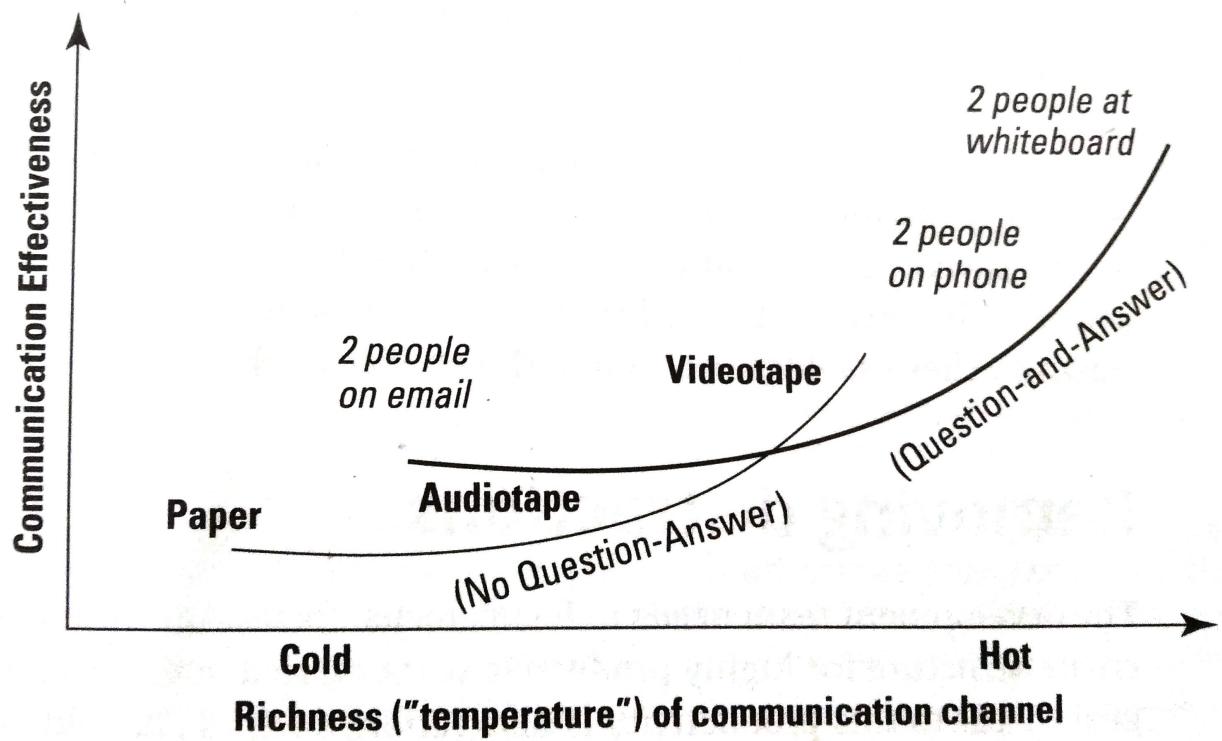
Creating the Physical Environment

Collocating the team

If at all possible, the scrum team needs to be *collocated* — that is, physically located together. When a scrum team is collocated, the following practices are possible and significantly increase efficiency and effectiveness:

- » Communicating face to face
- » Physically standing up — rather than sitting — as a group for the daily scrum meeting (this keeps meetings brief and on topic)
- » Using simple, low-tech tools for communication
- » Getting real-time clarifications from scrum team members
- » Being aware of what others are working on
- » Asking for help with a task
- » Supporting others with their tasks

FIGURE 5-1:
Better communication through collocation.



Setting up a dedicated area

Set up an environment where the scrum team can work in close physical proximity. If possible, the scrum team should have its own room, sometimes called a *project room* or a *scrum room*. The scrum team members create the setup they need in this project room, putting whiteboards and bulletin boards on the walls and moving the furniture. By arranging the space for productivity, it becomes part of how they work. If a separate room isn't possible, a *pod* — with workspaces around the edges and a table or collaboration center in the middle — works well.

Removing distractions

The development team needs to focus, focus, focus. Agile methods are designed to create structure for highly productive work carried out in a specific way. The biggest threat to this productivity is distraction, such as . . . hold on a minute, I need to take a call.

TABLE 5-1**Common Distractions**

Distraction	Do	Don't
Multiple projects	Do make sure that the development team is dedicated 100 percent to a single project at a time.	Don't fragment the development team between multiple projects, operations support, and special duties.
Multitasking	Do keep the development team focused on a single task, ideally developing one piece of functionality at a time. A task board can help keep track of the tasks in progress and quickly identify whether someone is working on multiple tasks at once.	Don't let the development team switch between requirements. Switching tasks creates a huge overhead (a minimum of 30 percent) in lost productivity.
Over-supervising	Do leave development team members alone after you collaborate on iteration goals; they can organize themselves. Watch their productivity skyrocket.	Don't interfere with the development team or allow others to do so. The daily scrum meeting provides ample opportunity to assess progress.
Outside influences	Do redirect any distractors. If a new task outside the sprint goal surfaces, ask the product owner to decide whether the task's priority is worth sacrificing sprint functionality.	Don't mess with the development team members and their work. They're pursuing the sprint goal, which is the top priority during an active sprint. Even a seemingly quick task can throw off work for an entire day.
Management	Do shield the development team from direct requests from management (unless management wants to give team members a bonus for their excellent performance).	Don't allow management to negatively affect the productivity of the development team. Make interrupting the development team the path of greatest resistance.

Going mobile

- » Use movable desks and chairs so that people can move about and reconfigure the space.
- » Get wirelessly connected laptops so that scrum team members can pick them up and move them about easily.
- » Have a large mobile whiteboard. Also see the next section on low-tech communication.

Low-Tech Communicating

The primary tool for communication should be face-to-face conversation. Tackling problems in person is the best way to accelerate production:

- » **Have short daily scrum meetings in person.** Some scrum teams stand throughout a meeting to discourage it from running longer than 15 minutes.
- » **Ask the product owner questions.** Also, make sure he or she is involved in discussions about product features to provide clarity when necessary. The conversation shouldn't end when planning ends.
- » **Communicate with your co-workers.** If you have questions about features, the project's progress, or integrating, communicate with co-workers. The entire development team is responsible for creating the product, and team members need to talk throughout the day.

- » The goal of the sprint
- » The functionality necessary to achieve the sprint goal
- » What has been accomplished in the sprint
- » What's coming next in the sprint

- » Who is working on which task
- » What remains to be done

Only a few tools are needed to support this low-tech communication:

- » A whiteboard or two (ideally, mobile — on wheels or lightweight). Nothing beats a whiteboard for collaboration. The scrum team can use one for brainstorming solutions or sharing ideas.
- » A huge supply of sticky notes in different colors (including poster-sized ones for communicating critical information you want readily visible — such as architecture, coding standards, and the project's definition of done).

A personal favorite is giving each developer at least one tabletop dry erase/sticky note easel pad combination, with a lightweight easel. These low-cost tools facilitate communication fantastically.

- » Lots of colorful pens.
- » A sprint-specific task or kanban board (described in Chapters 4 and 9) for tracking progress tactility.

RELEASE GOAL:	SPRINT GOAL:																	
RELEASE DATE:	SPRINT REVIEW:																	
		<div style="display: flex; justify-content: space-between;"> US = User Story </div> <div style="display: flex; justify-content: space-between;"> Task = Task </div>																
TO DO	IN PROGRESS	ACCEPT																
		<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>US</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td></tr> <tr><td></td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td></tr> </table>	US	Task	Task	Task	Task	Task	Task		Task	Task	Task	Task	Task	Task		
US	Task	Task	Task	Task	Task	Task												
	Task	Task	Task	Task	Task	Task												
		<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>US</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td></tr> <tr><td></td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td><td>Task</td></tr> </table>	US	Task	Task	Task	Task	Task	Task		Task	Task	Task	Task	Task	Task		
US	Task	Task	Task	Task	Task	Task												
	Task	Task	Task	Task	Task	Task												
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>Task</td><td>Task</td></tr> <tr><td>Task</td><td>Task</td></tr> <tr><td>Task</td><td>Task</td></tr> <tr><td>Task</td><td>Task</td></tr> </table>	Task	Task	Task	Task	Task	Task	Task	Task	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>US</td><td>Task</td><td>Task</td><td>Task</td></tr> <tr><td></td><td>Task</td><td>Task</td><td></td></tr> </table>	US	Task	Task	Task		Task	Task		
Task	Task																	
Task	Task																	
Task	Task																	
Task	Task																	
US	Task	Task	Task															
	Task	Task																
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>US</td><td>Task</td><td>Task</td></tr> <tr><td></td><td>Task</td><td>Task</td></tr> <tr><td></td><td>Task</td><td>Task</td></tr> <tr><td></td><td>Task</td><td>Task</td></tr> </table>	US	Task	Task		Task	Task		Task	Task		Task	Task						
US	Task	Task																
	Task	Task																
	Task	Task																
	Task	Task																

FIGURE 5-2:
A scrum task board on a wall or whiteboard.

High-Tech Communicating

When determining which types of high-tech communication tools to support, first consider the loss of face-to-face discussions. Some tools you can use follow:

- » **Videoconferencing and webcams:** These tools can create a sense of being together. If you have to communicate remotely, at the very least make sure you can see and hear each other clearly. Body language provides the majority of the message.
- » **Instant messaging:** Although instant messaging doesn't convey nonverbal communication, it is real time, accessible, and easy to use. Several people can also share a session and share files.
- » **Web-based desktop sharing:** Especially for the development team, sharing your desktop allows you to highlight issues and updates visually in real time. Seeing the problem is always better than just talking it out over the phone.
- » **Collaboration websites:** These sites allow you to do everything from sharing simple documentation so that everyone has the latest information to using a virtual whiteboard for brainstorming.

The purpose of the tool

When choosing tools, the primary question you need to ask is, “What is the purpose of the tool?” Tools should solve a specific problem and support agile processes, the focus of which is pushing forward with the work.

Above all, don’t choose anything more complicated than you need. Some tools are sophisticated and take time to learn before you can use them to be productive. If you’re working with a collocated scrum team, the training and adoption of agile practices can be enough of a challenge without adding a suite of complicated tools to the mix. If you’re working with a dislocated scrum team, introducing new tools can be even more difficult.