9 Agile Communications and Tools

AGILE COMMUNICATIONS PRACTICES

ONE OF THE MOST IMPORTANT values behind agile is *openness and transparency* (see Chapter 3). Many traditional projects in the past limited the sharing of information to carefully controlled channels of communication and bad news was sometimes hidden from view in order to present a favorable image of progress. An important role of a traditional project manager in that environment has been to manage that flow of information.

Since the flow of information in an agile project is more open and transparent, information can be allowed to flow much more easily and automatically using the concept of an *information radiator*. For that reason, the role of a traditional project manager to manage the flow of that information is less essential. In addition to the general importance of openness and transparency, there are a number of other reasons why communications is extremely important in an agile project:

- Information is rapidly and dynamically changing in real-time throughout the project and needs to be shared efficiently.
- Good communications is essential to support close collaboration among everyone on the project team as well as people who may be peripheral to the project team, especially if the team cannot be collocated—that is, working in the same area—which is often the case.
- Sharing of information with the customer and business sponsor is also essential for the same reasons to support a close and collaborative partnership relationship.

As agile projects become larger and more complex, using tools to help distribute information quickly and efficiently becomes essential.

Information radiators

An *information radiator* is a large and highly visible display of critical team information that is typically located in a spot where the team and others can see it constantly, and it is continuously updated

either during the daily standup meetings or even more frequently in real-time as work is completed. It could take on several different forms:

- In some cases, it is actually a large white board or a complete wall of a room that is used to track progress.
- It could also be in electronic form by using one of a number of different agile project management tools so that a much broader audience can view the information online.

The concept of an *information radiator* in agile is consistent with the value of openness and transparency:

- Everyone on the team is aware of the work of all other members of the team and how it contributes to the goals of the team, which is consistent with promoting a strong and unified teamwork approach.
- The product owner and other stakeholders outside of the team are also aware of progress and issues that might impact progress which is consistent with promoting a spirit of partnership and customer collaboration.
- A white board approach using a physical white board and colored "stickie" notes has been widely used in agile projects for a long time, but it has some significant limitations.
  - It doesn't work well with distributed teams.
  - It doesn't provide an ability to easily roll up information across multiple teams.
  - It is difficult to keep the information organized and it also doesn't provide a capability to sort and report on the information in different ways.

For those reasons, many agile projects are moving to more widespread use of online tools. Figure 9.1 shows an example of an information radiator using an online tool.

A big advantage of information radiators is that each individual on the project team can individually update the status of tasks that he/she is responsible for in near real-time and all of that information will be aggregated in the information radiator for sharing with others both inside and outside the project team. That reduces the role that a project manager has provided of aggregating and reporting project status information. An information radiator is typically used in conjunction with the daily standup meeting to discuss and review the work being done by the team. If an online tool is used, the information can also be rolled up at a number of different levels across multiple teams and even across multiple projects.

The use of information radiators is a very powerful aspect of agile, and if it is implemented correctly:

- It promotes openness and transparency which helps build stronger and more effective teamwork.
- It promotes customer collaboration and helps to build a spirit of trust and partnership between the development organization and the business users.

- It can remove a huge burden on management for attempting to control all aspects of an operation. In one situation, a senior manager commented on how much agile has removed the management team from resolving day-to-day issues that the project team now takes responsibility for, and it has enabled the company to focus on much higher-level strategic goals.

Face-to-face communications

Agile heavily emphasizes face-to-face communications wherever possible over other forms of communication.

> "The most efficient and effective method of conveying information to and within a development team is face-to-face conversation."[2]

However, that does not mean that documentation and other forms of communication are eliminated and this principle needs to be adapted to fit the situation, particularly in the case of widely

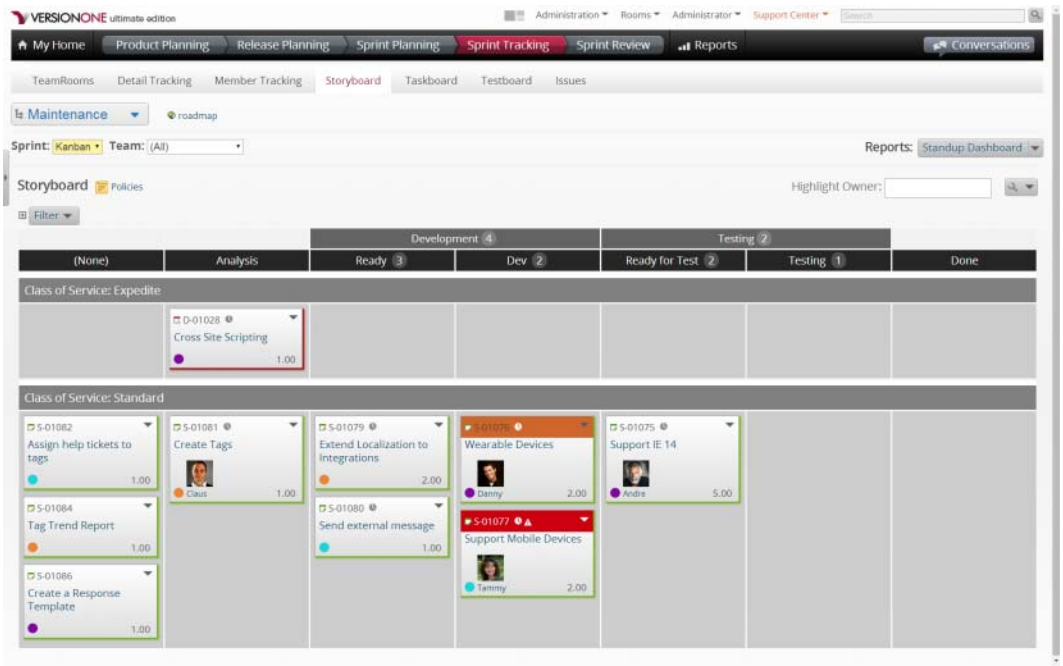

FIGURE 9.1 Example online information radiator[1]

[1]VersionOne agile Tracking Tools, http://www.versionone.com/product/agile-tracking-tool/
[2]"Principles behind the Agile Manifesto," http://www.agilemanifesto.org/principles.html.

distributed teams. Here are some examples of how face-to-face communications are used in agile projects:

- Ideally, agile teams are co-located in the same room to facilitate direct face-to-face communications
- User stories do not attempt to capture all possible details of customer requirements—they are considered to be a "placeholder for communications."

Daily standups

All of us have participated in long, inefficient meetings that are not focused, go on and on, and don't use people's time very well. Daily standups are a way of making meetings much more efficient in an agile project.

- They take place daily but they are limited to 15 minutes, and each person in the meeting typically stands up to encourage people to keep it brief.
- The meeting is focused on the tasks at hand and is typically held in front of a progress board.
- The meetings are structured so that each person answers three primary questions:
  - What did you accomplish yesterday?
  - What are you going to accomplish today?
  - What obstacles are in your way?

Naturally, the idea of a daily standup might need to be modified to fit the environment, particularly with distributed teams.

Distributed teams

Many agile communications practices (e.g., daily standups) are based on the assumption that the team is collocated. However, in many situations, it isn't possible to have that level of collocation and the communications practices need to be adjusted to support distributed teams that may not be in the same geographical location and could even be in different time zones, which is often the case with offshore development teams. In that case, you have to adjust the communications strategy to fit distributed teams:

- Video conferencing might be used in lieu of face-to-face communications.
- Online status boards might be used in lieu of physical wall boards.
- Wikis might be used for sharing of information.
- Daily standups might need to be extended or supplemented with other meetings to provide more communications.

Managing communications with offshore teams in an agile project can be very challenging. Some of the challenges are:

- Geographic separation
- Cultural and language differences
- Time zone differences

I was in a situation where the entire development team was in India and only the customer-facing people were in the United States. Informal meetings next to the water cooler will not work in this situation. But even in an agile environment where the team is collocated, a brief daily standup might not be sufficient. If it is the only (or primary) form of communication with the team throughout the day, it can be very difficult to limit the daily standup to just a 15-minute session. You must learn to adapt the communication practices to the situation.

AGILE PROJECT MANAGEMENT TOOLS

In my opinion, an agile project manager who doesn't know how to use one of the widely used agile project management tools like VersionOne, Rally, Jira, or others is equivalent to a traditional project manager who doesn't know how to use Microsoft Project. The Agile Manifesto value says:

> "Individuals and interactions over processes and tools."

Tools, however, still play a very important role in agile, especially as you start to scale projects to large, complex enterprise-level solutions and that's the area where you are most likely to find an agile project management role.

There are many tools that are used in agile ranging from very simple tools that are oriented around simple, team-based activities to more complete tools that include team-level capabilities but also go well beyond that level and provide a capability for scaling projects to enterprise-level projects. It is impossible to cover all of the agile tools that someone might consider using in this book—in order to bound the scope of the discussion and simplify it a bit, I'm going to do the following:

- First, since this book is about agile project management, I'm going to focus the discussion on tools that are most likely to be used by an agile project manager that would provide a full capability for handling large, complex, enterprise-level projects. Several that I've worked with, in particular, include VersionOne, Rally, Jira/Greenhopper, and Microsoft TFS.
- Since many of these tools have similar capabilities, I've decided to focus on using VersionOne Ultimate Edition (Version 14) as a representative example of all of them to provide a general understanding of how these tools are typically used. This is not intended to be an endorsement of the VersionOne tool; I just had to pick one tool to use as an example to simplify this topic. It is also not

intended to be an exhaustive tutorial on how to use the tool. Most tools such as VersionOne have online videos for that purpose.

Agile project management tools are fundamentally different from traditional project management tools like Microsoft Project:

- The typical way of modeling traditional projects is based on a fairly statically defined model composed of work breakdown structures, Pert charts, Gantt charts, and others. Those modeling tools work fine for a heavily plan-driven approach where the requirements and the overall design approach for meeting those requirements can be defined prior to the start of the project; however, they can be difficult and impractical to apply to a more dynamic and adaptive project approach where the requirements are much more uncertain and difficult to totally define upfront.
- A more dynamic and adaptive approach is based on the concept of a continuous flow of project requirements that are expected to change and be further defined as the project progresses and calls for a different modeling approach. An agile project manager needs to go beyond understanding how to develop a fairly statically defined project model based on a traditional project modeling approach and understand how to optimize the flow of a much more dynamic stream of changing requirements through a much more flexible and adaptive process model. That's what agile project management tools are designed to do.

The fundamental difference is that traditional tools like Microsoft Project are heavily oriented around defining and managing the *structure* of project activities while agile tools are more oriented around managing the *flow* of project activities through a much simpler and fluid structure.

Benefits of agile project management tools

At an enterprise level, agile project management tools provide some significant value:

1. *Ability to fully engage all members of the team in the process.* An agile project management approach is very different from a traditional, plan-driven project management approach. In a traditional, plan-driven project, the project manager is primarily responsible for planning the project as well as managing and reporting progress. Microsoft Project is well-designed to support that role because it is a stand-alone desktop tool and most of the information flows through the project manager. In an agile project, the responsibility for planning and managing the effort is distributed among everyone on the team, and agile project management tools are designed with that in mind.
2. *Ability to update progress in real-time and rapidly view status and issues (information radiators).* An agile project is also fast moving, and it's very important to be able to update information easily in real-time and for anyone to be able to view that information easily. That's the concept of an information radiator.
3. *Ability to scale projects to an enterprise level and provide a standardized way of reporting across projects.* As I've previously discussed, there are different kinds of information radiators.

Small, single-team agile projects might use a simple Kanban board on a white board with index cards or "stickies" on it to manage and track progress. That method works OK for small teams, but falls apart quickly for projects with teams that are not co-located or projects that require multiple teams. Most of the agile project management tools provide at least the capability for an online Kanban board to enable sharing of information across distributed teams.

Characteristics of enterprise-level agile project management tools

VersionOne has developed a nice summary of criteria for selecting a robust, enterprise-level agile project management tool, which is reproduced here with permission from VersionOne.[3]

Full Agile Process Lifecycle Coverage

"Agile management systems must span the full range of processes from integrated customer feedback through portfolio/product planning, code integration, testing and delivery.

- Product, release and iteration planning & tracking
- Agile portfolio management including epic boards, epic ranking, rapid epic breakdown and reporting
- Strategic goals, functional rollups (themes), goal assignment, impediment tracking and retrospectives
- Consolidated requirement backlog and defect repository
- Test management (manage, track and execute tests across multiple stories, defects and distributed projects)
- Integrated product road-mapping linked to releases/iterations

Team and Customer Collaboration

Collaboration capabilities should promote teamwork and expedite communication between team members, teams and organizations.

Social-media style communication portal for project teams

Contextual collaboration includes links to work items, team members and dedicated TeamRooms™

[3] "VersionOne Tool Evaluator Guide," http://www.versionone.com/pdf/agiletoolevaluator.pdf.

RSS feeds and email notifications for receiving message and alerts

Cross-project planning, tracking & reporting for distributed team members

Board Views are customizable, filterable and support flexible color coding to convey information

Integrated customer Idea Management platform (facilitating customer engagement, collaboration, and prioritization)

Visibility, Reporting & Analytics

Agile management solutions must include concise views into data, work estimates, actuals and trends.

- Executive-level dashboards with best practice metrics
- Advanced planning including: release forecasting, "what-if" analysis, and workload balancing
- Team burn-down, burn-up, cumulative flow, and test trend reporting
- Epic boards/Storyboards with drag-and-drop tracking of story cards, work-in-process limits, aging & cycle-time reporting
- Roll-up reporting for teams working across projects and projects with multiple teams
- Best-practice agile dashboards: Project, Sprint, Program, Team Member, and role-based dashboards
- Custom reports and graphs created via web-based, wizard-driven interface
- Agile visualizations—Epic bubble charts, Hierarchy/Tree charts, Relationship Visibility, Release Dependency Mapping

Simplicity & Ease of Use

Use a tool with a simple user interface and built-in process navigation.

- Built-in agile process navigation with customizable dashboards for team members to track their projects and work
- Drag-and-Drop ranking and whiteboard-style release and iteration planning
- Drag-and-Drop epic, story, task and test boards with customizable workflow processes and configurable card data
- Multi-select options for actions such as: Move; Close; Reopen; Delete; and Rank

- Multi-level estimation at the story/defect level and the effort tracking level
- TeamRoom™—dedicated team-based environment supporting the daily activities of development teams
- PlanningRoom™—dedicated environment for program-level managers to collaborate in focused planning sessions

Configurable Workspaces, Process and Terminology

Agile Tools should guide you through the agile process and help you implement "agile your way". Custom workspaces and terminology support unique process configurations without sacrificing visibility, reporting and analytics across the portfolio.

- Drag-and-drop epic, story, task and test boards with customizable workflow and configurable card data
- Customizable methodology templates (XP, Scrum, DSDM, Kanban, etc.)
- Extensive customization options including boards, fields, lists, value, grids, etc.
- Customizable folder structure for nested project/release hierarchy
- Customizable and filterable board views that support color-coded visual indicators

Agile Portfolio and Program Management

Agile tools should grow with you as your needs grow including portfolio-level planning for your strategic initiatives, program-level coordination and project-level story delivery with full traceability from high-level epics to development-level tasks.

- Portfolio-level business initiatives, release rollouts, progress and organizational velocity mapped against a timeline
- Program-level Epicboards, Epic Bubble charts, epic ranking, planning and roll-up reporting
- Integrated product roadmapping linked to releases/iterations
- Advanced planning including: release forecasting, "what-if" analysis, and workload balancing
- Cross-project team planning, tracking and reporting
- Release Dependency Diagram to better prioritize story completion
- Extensive support for Scaled Agile Framework® (SAFe™) PPM methodology (Planning, Process, Metrics, Reports, etc.)

Deployment, Security and Integrations

Agile tools should provide: flexible deployment options for ANY team size and ANY agile methodology; application-, role- and project-level security; and open integrations for simplified deployment and customization.

- Free trial software available for both On-Demand and On-Site options
- Four right-sized product editions that grow with you as your agile project management needs evolve
- SaaS and on premise deployment options with easy data portability should your requirements change
- Available integrations to 45+ ALM technologies
- JAVA and .NET SDKs and open, web-services API
- Password authentication and single Sign-On (SSO) across systems and third-party integrations
- Project-level and role-based security to provide the right access to the right data

That's a long list of capabilities and, naturally not all of those capabilities may be required in a particular implementation, but it's a good list, and it's always easier to take away something that's not needed rather than adding something that's not there.

SUMMARY OF KEY POINTS

Agile Communications Practices

1. Communications in an agile project is extremely important to support openness and transparency and it is essential to support teamwork and rapid and efficient coordination among the people on the project team, as well as close collaboration and partnership with the customer and business sponsor.
2. Information radiators are widely used in agile projects to disseminate information rapidly and efficiently in a very dynamic and fast-paced environment. The simplest form of information radiator is a white board with colored "stickies" to track progress of work. Online tools have a number of advantages over white boards and can be more effective in many environments for a number of reasons such as the ability to easily roll up information across multiple teams and the ability to support distributed teams that cannot be collocated.
3. Agile emphasizes face-to-face communications; however, in the real world, communications practices frequently need to be adapted to fit teams that cannot be collocated. In that situation,

there are a number of alternatives that can be used in lieu of or to supplement direct face-to-face communications. Those alternatives include video conferencing, online status boards, and Wikis.

Agile Project Management Tools

1. Agile project management tools have a very different orientation than traditional project management tools. Traditional project management tools are heavily-oriented around planning and managing the structural aspects of a project (Gantt charts, Pert Charts, dependencies, etc.). Agile project management tools are much more organized around planning and managing flow and the traditional emphasis on structure is considerably simplified or not needed at all.
2. Agile project management tools are also designed around a collaborative team approach where each individual on the team has direct access to the tool for planning and tracking their own work rather than all work being coordinated and managed by a project manager.
3. Many agile project management tools are very scalable and offer capabilities all the way from simple single-team agile projects to complex, large-scale enterprise-level capabilities.

DISCUSSION TOPICS

Agile Communications Practices

1. What are the major differences between communications practices in an agile project and a conventional, non–agile project? Why is communications so important in an agile project?
2. What is an information radiator? What are the advantages of online tools for managing communications in an agile project?
3. What are some of the challenges associated with distributed and/or offshore teams, and how would you go about resolving them?

Agile Project Management Tools

4. What agile tool capability do you think is most important in a typical agile project? Why?
5. What do you think is the most important factor in choosing an appropriate tool for an agile project environment? Why?