Technology Stewardship for Distributed Project Teams

John David Smith
Learning Alliances and CPsquare, USA

ABSTRACT

Distributed project teams and communities of practice face similar challenges in using technology to work together and "be together." However, these technologies are not only powerful enablers, they also present persistent challenges in and of themselves and in that they enable and filter multiple kinds of heterogeneity. These challenges are magnified by linguistic and cultural heterogeneity, multiple time zones, organizational boundaries, and the variance in technology literacy in project teams. Project teams need to adopt some of the practices that are found in technology-mediated communities to cope with their persistent technology challenges. The notion of a "digital habitat," which has been developed by Wenger, White, and Smith (2009), is used to describe the landscape of technologies and practices that in combination will enable a project team to accomplish its tasks. For some distributed teams, a project must include ensemble work to ensure that all team members can participate and contribute, that communication and collaboration practices exist or are being developed, and that what has been learned in the process of completing the work is assessed and retained. The technology steward is proposed as a new role and function for project teams that operate at this level. Activities that support the exploration of a digital habitat include the development of technology literacy and increase the learning capacity of project teams.

INTRODUCTION

Project teams are one of the dominant forms of organizing work, and project management is one of the essential skills for a company's survival. Throughout the history of project management, much of the literature on the subject has adopted a command-and-control model of project management, where learning was not an explicit or salient element. Project teams are now more complex because of factors such as globalization, the need to work across organizational boundaries, and the way that work is increasingly mediated by technology (Zuboff, 1988). But because of the way projects bring workers together in ad hoc groupings, they are differentially exposed to the challenges of distributed collaborative technologies, and technologies present their own learning challenges. In a knowledge economy, learning is a more important element of project work than in the industrial age. A learning model such as a community of practice framework augments more traditional project management thinking, particularly as it addresses technology use.

This chapter explores some of the connections between project management and learning, and connects them to the technology challenges that project teams face. It proposes that a technology steward role, which is significant in distributed communities, can support the learning and ongoing sense-making that is an important ingredient of knowledge-intensive and distributed project work. The conceptual discussion of a technology steward is complemented by two concrete examples of the kind of contribution that a technology steward makes to a project team. The first is a case study describing an exploration episode where a group collectively explores the use of a new tool for teamwork. The second illustrates how a technology steward breaks down the different functions of a software platform from a social and collaboration perspective that is derived from the study of communities of practice.

Pre-publication draft for K.L. Milhauser, **Distributed Team Collaboration in Organizations: Emerging Tools and Practices** (Hershey, PA: IGI Global, 2011)

BACKGROUND

Project management is probably as old as organized work itself. As a formal discipline it is a direct descendant of Taylor's "scientific management" approach (Cleland & Gareis, 2006). Tools such as Gantt Charts and methods such as Critical Path Analysis and Work Breakdown Structure all aim at the rational organization of discrete activities that can be known in advance, repeated, measured, and thus improved. Spreading from large civil engineering projects in the early 20th Century to defense contracts to software development, today project management techniques are commonplace and essential to corporate operations (Cleland & Gareis, 2006). Recent developments such as Agile methods for software development (Derby & Larsen, 2006) are innovative and relevant to knowledge-intensive projects because they make learning an integral part of the project management process.

Communities of practice are probably as old as human society as well. As a concept and a term, "community of practice" was proposed in Lave and Wenger (1991) as a way to account for the social organization of apprenticeship. Further conceptual elaboration occurs in Wenger (1998). Ito et al. (2009) place communities of practice in the larger context of a "social turn" in literacy studies, new media studies, learning theory, and childhood studies that moved away from the previously dominant focus on individual cognition and knowledge acquisition.

A working definition is provided by Wenger (2006), p. 1: "Communities of practice are groups of people who share a concern or a passion for something they do and learn how to do it better as they interact regularly." The early adopters of the community of practice framework were corporations managing knowledge (Wenger, McDermott, & Snyder, 2002; Wenger & Snyder, 2000) where technology by itself was inadequate as a means of managing this critical resource. In *Digital Habitats*¸ Wenger, White, and Smith (2009) develop a framework for thinking about technology as an integral component of the life of communities of practice and propose the role of technology steward as a key kind of community leader.

PROJECT TEAMS AND COMMUNITIES

Studies of the connection between project management and communities of practice are relatively recent. In her study of project based organization, Inkeri Ruuska (2005) showed how semi-formal community of practice and formal project management structures coexisted and mutually informed each other. Lloyd (2007) suggested that a large, distributed, open source software production process could be understood as a community of practice that contains many concurrent small projects staffed largely by volunteers within it. Justesen (2006) argued that project management must attend to the diversity of available knowledge in a project that aims at significant innovation. This can be accomplished by acknowledging and preserving the distinct voices and perspectives of the communities of practice that are represented in the team. Although they do not explicitly make the connection, the methods for agile project retrospectives that are proposed in Derby and Larsen (2006) are very much in line with good practice in a facilitated community of practice.

Wenger et al. (2002) provided a useful contrast between the two social structures. An excerpt from the Table 2.2 on p. 42 of their Cultivating Communities of Practice suggests these crucial differences.

Table 1. Selected comparisons from Wenger et al. (2002)

	Project teams	**Communities of Practice**
What's the purpose?	To accomplish a specified task	To create, expand, and exchange knowledge, and to develop individual capabilities.

Who belongs?	People who have a direct role in accomplishing the task	Self-selection based on expertise or passion for a topic
How clear are the boundaries?	Clear	Fuzzy
What holds them together?	The project goals and milestones	Passion, commitment, and identification with the group and its expertise

The primary distinction between the two has to do with the notion that project teams are about accomplishing an assignment and communities of practice are about self-organized learning, with ramifications such as friendships with colleagues and commitment to a community that identification with a domain can entail. Those fundamental differences have management implications. Beyond that the differences between the two social formations involve how outcomes are thought about (whether specified in advance or emergent), role definitions (whether designated in advance or discovered along the way), and process expectations (whether specified as part of a project plan or negotiated over time).

Much of the work in Hinds and Kiessler (2002) demonstrated how distributed work has a long history and showed how the use of information technologies both enable new possibilities and raise new challenges for organizations and communities. I will argue that heterogeneity of staff, knowledge, and other factors all increase as technology becomes more prominent in the work of a distributed project team, so project teams need behaviors, activities, and roles that are more characteristic of communities of practice.

Even setting aside the role of technology, the difference between a traditional project team and a community is a matter of emphasis and in fact there can be considerable overlap. In addition to the overlap and coexistence of projects and communities studied by Ruuska (2005) and Lloyd (2007), there are many project teams that spawn communities and many communities that launch projects. When a project begets a community it is often because the sense of accomplishment that people experienced in a project results in a recognition of each other's expertise and people want to stay connected after its end. When a community launches a project, it could be to produce an event, to explore a topic, to standardize a practice, or to provide the community with a technology advance. Figure 1 suggests how project teams and communities can have a reciprocal relationship, one providing the social setting that seeds a larger or smaller instance of the other. The diagram suggests that project teams typically have clearer boundaries around them than communities, which have larger and less distinct peripheries. In the following discussion I explore the cases where one provides the context or background for the other.

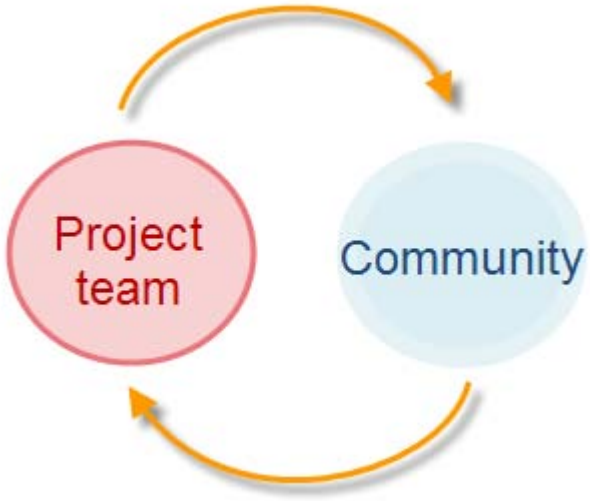

Figure 1 - Projects and communities beget each other

From the point of view of cultivation it may be most useful to contrast the two social formations because both participants and leaders must consider different issues. From the point of view of how people put technology to use, there may be differences, but I propose that it is quite useful to think of distributed project teams and communities of practice as reciprocal lenses on each other. Although project teams are the more common social formation in many organizational settings, the informality of self-organizing communities of practice can teach us a lot about social roles, adaptation, and combinations of tools and technologies that can support distributed project teams. Conversely, what might we learn about communities and their technology requirements if we looked at them as projects? This reciprocal question is outside the scope of this chapter but deserves further thought.

The social interactions that are distinctive in a community of practice involve the negotiation of meaning. As argued in *Digital Habitats* (Wenger, White, & Smith, 2009), communities of practice are adept at finding ways of bending technology to their needs: putting technology to work in the negotiation of meaning in the context of heterogeneous work histories, cultural backgrounds, maturity, practice settings, and even technology resources. Heterogeneity of just these factors is increasing as work on distributed teams becomes more global, scales up, and involves more learning.

Heterogeneity in communities and project teams.

In thinking about communities of practice we most often start with what is held in common, but it is actually the way people deal with heterogeneity in a community of practice that can make them productive. First I will characterize how communities and project teams deal with different kinds of heterogeneity and then argue that when collaborative technologies become a central enabler of the work of distributed project teams, heterogeneity increases and a community of practice framework will prove helpful in framing the issues that result.

To organize our thinking about how the two different social formations deal with heterogeneity, Table 2 applies the structural characteristics of a community of practice (Wenger, 1998; Wenger et al., 2002) to a project team.

Table 2. Heterogeneity in teams and communities

	Project teams	Communities of practice
Domain (topic)	Heterogeneous expertise applied to different parts of a problem.	Single domain seen from different perspectives because of members' backgrounds or participation in other communities.
Membership	Participation specified by roles, which are time-delimited and shaped by a project plan.	Member involvement changes with interest and mastery. Participation can be brief or long-lived.
Practice (being together)	Meetings, reporting, and involvement is planned in advance and managed by a project leader.	Ongoing negotiation of polarities such as togetherness and separation, participation and reification, or group and individual.
Practice (working in the world)	Different kinds of work is assumed to occur in different settings that are anticipated in the project plan.	Members practice independently, in contexts that over time are discovered to be the same or different.

Table 2 emphasizes the front-loading of planning and design that is characteristic of a managed project team. In the classic project management approach, a major goal is to remove as much uncertainty from a project as possible and a typical strategy to accomplish that goal is to anticipate heterogeneity of various sorts and reduce its impact by careful planning. For example, a work breakdown structure, staffing plans, and meeting schedules have a common aim of sorting out the differences in a social process in advance. See NASA (2001) as an example, emphasizing the importance of cost control. Communities of practice may confront similar heterogeneity, but they do not front-load the effort: negotiations around topic boundaries, membership, and how to talk about the practice are an ongoing feature of community life. Heterogeneity is more often acknowledged and put to use in a community of practice setting, where it is a resource. That's why negotiating and learning about the heterogeneity of technologies is an ongoing thread in the lives of so many distributed communities. In the communities we studied in *Digital Habitats* (Wenger, White, & Smith, 2009), we found that part of keeping the technology topic alive without letting it swamp all else was the role of a technology steward, which is discussed below.

Table 2 can be put to another use: suggesting why the opportunities that social technologies open up for distributed project teams also tend to increase heterogeneity. For example, distributed project teams can draw domain expertise and staffing from a more widely distributed and more diverse landscape. More distinct nomenclatures, languages and accents are therefore involved. More specific roles can be incorporated into a project plan. Technology provides more ways of being together, so more choices of how to work together are available, including the choice of whether or not to "work together" at the same time. Technology resources beyond those used for social purposes can greatly expand manufacturing, testing, or work resources as well. All of these possibilities increase diversity and the need to manage them increases the complexity that needs to be managed. The very elements that make technology essential for getting work done in a project team tend to make it more important to learn to use the technology well. A technology steward can play a productive role in keeping a project's tools relevant, the skills of its staff up to date, and its conversations and partnerships as effective as possible.

Project team as the background for community.

When learning is a significant component of the work to be accomplished by a project team, or when the heterogeneity in a project team is a significant project resource, community-like structures become important. When they do not exist in the social surroundings, they form "inside" a project, especially if the consolidation or development of new knowledge is important for accomplishing a project's goals. Justesen's (2006) study argued that preserving the diversity and voice of communities that are represented in a project team was a critical element of a project team's capacity to deliver significant innovation. Long-running project teams experience member turn-over, where there's a need to bring new members of the team into the team's culture and tell them the stories from the team's history: community behaviors are thus called for.

Figure 2 - Project team as the background for a community

Heterogeneity is an important factor within projects that may call for community-style interactions. But projects themselves are heterogeneous in their styles of communication, beyond the abstraction of a Gantt chart plan. If we look closely we see that, like communities, we find projects that are "oriented toward meetings," or "oriented toward access to expertise," or even "toward relationships" much like the orientations for communities that we propose in *Digital Habitats* (Wenger, White, & Smith, 2009). Since those orientations have technology implications, it stands to reason that project orientations might as well. The technologies that serve one project may not be those that serve the learning needs of small communities that exist inside or serve other projects with different orientations. See a discussion of future research suggestions later in this chapter.

Community as the background for project teams

Particularly as communities become mature and therefore more intentional, they tend to identify activities that require purposeful action over longer periods of time: thus projects sprout up. Figure 1.1 in *Digital Habitats* (The range of activities in which a community of practice engages) depicts a spectrum of formality along with two other dimensions (Wenger, White, & Smith, 2009). Informal activities coexist with the formal, providing a backdrop of resources, evaluation, and guidance for a formal learning project. In this sense, project teams can be seen to "report to" a community, which establishes goals, parameters, and may even help identify budget resources.

Figure 3. Community as the background for a project team

From another perspective, even when a community isn't sponsoring a project, sometimes the community is the critical sounding-board, source of informal advice, or even context for legitimizing a project. In fact, the surrounding community can provide a critical intellectual resource for a project team. Sometimes a key resource or contributor such as the community technology steward will be part of the network or surrounding community but not part of the formal project team. When the knowledge and skills required for a project are very cutting-edge or very diverse, project team membership sometimes can't be known in advance, much less specified. All of the discussion about permeable community boundaries will apply in those situations because team members may need to bring an expert into a few technology-mediated conversations, but not involve them in the whole project's work-space.

In some circumstances, conversations in a surrounding community contain information that is vital to a project's success. The heterogeneity that the community encompasses is a key resource for avoiding project team groupthink. In that case, a distributed, technology-mediated team may need team members to stay involved in those conversations or activities of that surrounding community (which have more fuzzy and ad hoc technology boundaries than what we normally think of as "a project area").

The project team is a community.

Traditional software development teams were organized like assembly lines with autonomous workers (Derby & Larsen, 2006). Agile software development teams place less emphasis on front-loaded planning and organization, so they function more like communities of practice in several ways. The way Agile projects organize work into short episodes followed by regular retrospectives serves to emphasize the importance of collective sense-making. Another factor that brings more heterogeneity into those conversations is that group boundaries are self-consciously negotiated over time, so that the voices of customers, managers and others who are not traditionally part of a development team participate in retrospectives. A key assumption in an Agile software development project is that the construction of software will generate new knowledge about user needs, so that specifications can't be completely determined in advance. The construction of emergent knowledge during the development process itself places sense-making activities characteristic of a community of practice squarely in the center of a project team's interactions.

The development of a range of socially-oriented methodologies like Extreme Programming or Pair Programming in the 1990's echoes developments cited in Ito, et al. (2009, p. 13) in the emergence of learning theory, literacy studies, and new media studies:

> "The 1980s and 1990s saw the solidification of a new set of paradigms for understanding learning and literacy that emphasized the importance of social participation and cultural identity, and that moved away from the previously dominant focus on individual cognition and knowledge acquisition."

The practices advocated by Derby and Larsen (2006) for Agile retrospectives are very much like those that are common in a community of practice meeting. Therefore it may be productive to think of a project team as if it were a community of practice, with fuzzy boundaries, a target that emerges and evolves, and negotiated work processes. Figure 4 suggests that the two views of social groups overlap considerably.

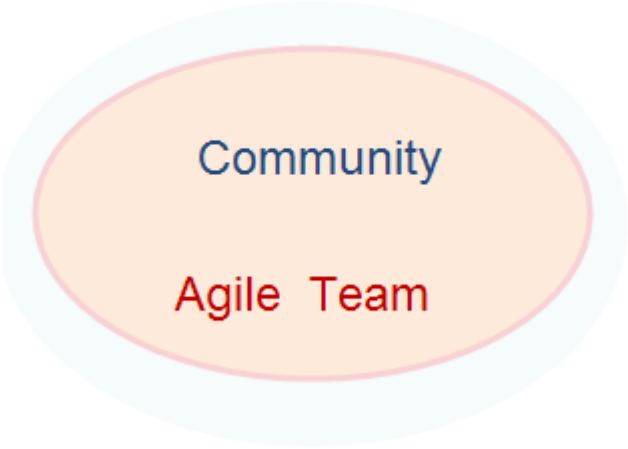

Figure 4. An Agile team works like a community

It is interesting to note that the treatment in Derby and Larsen suggests that Agile retrospectives are face-to-face meetings. While in some respects that may be the ideal, the reality is that many Agile teams are distributed and must use technologies of various sorts to be and work together.

Learning and Exploration As An Element Of Project Work.

As was discussed earlier, the increase in cultural and technological heterogeneity with which a project team must cope when it works in a global environment creates a significant learning challenge. This means that, in addition to accomplishing a project's work assignment, a part of a project team's work needs to focus on learning how to cope with all the differences in time zones, bandwidth, technologies, language, local organizational constraints, and customs regarding deadlines or commitments. Some of the heterogeneity that a project team may need to cope with is inherently technical. Cultural heterogeneity is not, but it may only be visible through the technology that supports the team's collaboration. Because of ongoing change in the technology environment, ongoing learning about technology becomes an important team activity beyond a project team's initial storming and norming stages (Tuckman 1965). The improvisational practices that we observe in communities of practice and the frameworks such as the polarities that are discussed in the context of one software platform below, help us think about how to get conversations to address tricky questions such as, "How do we work together?"

A NEW FUNCTION AND ROLE ON PROJECT TEAMS

During the years spent researching and writing *Digital Habitats* (Wenger, White, & Smith, 2009) we found instances where technology stewardship was an informal and widely-distributed function and we found instances where one person had the role of a technology steward who worked on behalf of the community. We found that it was useful to consider both the technology stewardship function and the

tech steward role or practice. Since a project team context emphasizes roles and assignments, that perspective is discussed here first.

The tech steward as a role on a project team.

Based on the communities we studied in *Digital Habitats*, we defined the role of technology steward as follows (Wenger, White, & Smith, 2009, p. 25):

> Technology stewards are people with enough experience of the workings of a community to understand its technology needs, and enough experience with technology to take leadership in addressing those needs. Stewardship typically includes selecting and configuring technology, as well as supporting its use in the practice of the community.

We found that technology stewards seemed to emerge naturally in distributed communities that survived. Some technology stewards came to that role because of having a technology background or interest but many took on the role because there was nobody else in their community who would do so. We observed many different activities, which we collapsed into the following categories:

- community understanding
- technology awareness
- selection and installation of tools and platforms
- tool adoption, community transition, and everyday use

In each of these activities, learning at an individual and community level are the main goals. In the context of distributed project teams, learning and the accomplishment of project tasks are interwoven with each other.

Changing the wording slightly to reflect a distributed team context we have:

> Technology stewards are people with enough experience of the workings of a distributed team to understand its technology needs, and enough experience with technology to take leadership in addressing those needs. Stewardship typically includes selecting and configuring technology, as well as supporting its use in the practice of the team.

In terms of the life cycle of a project team, the technology steward would play an important role in planning the project, in selecting the tools that are to be used initially, in launching group interactions and establishing group practices around the selected tools. As the project progresses, the technology steward would be watching for misfits between the project team's needs and its digital habitat, suggesting adjustments to project practice or technologies as needed.

As has been discussed previously in this chapter, in a globalizing workplace a project planner or manager can no longer assume uniformity in terms of tools (e.g., one tool or even one kind of tool), project personnel, or appropriate ways of collaborating. All the heterogeneity that a team will need to deal with cannot be anticipated or resolved in advance. So the technology steward, who is part ethnographer and part trouble-shooter, plays an important role after a project has been launched. Having someone pay attention to the team's needs and success in sense-making becomes a necessity. Partnership between project leadership and technology stewards becomes a key element in the translation of the tasks that we observed in communities of practice to a project team context.

A technology steward's role is shaped by the social context in both communities and in project teams. For example, in the very technical communities such as the Ubuntu community described by Lloyd (2007), everyone is concerned with technology in one way or another, so that interest in technology choices and uses are widely distributed throughout the community. In such a community a new member could turn to almost anyone and get informed guidance and support about what tools to use and how they are to be used. Therefore, technology stewardship in such a community is a broadly distributed function, not the role of a small group or one person. Another example of how technology stewardship can be widely distributed comes from a large-scale ethnographic study showing how learning about technology is intertwined with ongoing social interaction. In "Hanging Out, Messing Around, and Geeking Out: Kids Living and Learning With New Media" Ito, et al. (2009) describes how learning about technology propagates through a social system, along friendship lines. Megan Finn discusses the "techne-mentor" in depth (pp. 59-60):

> In conceptualizing the media and information ecologies in the lives of University of California at Berkeley freshmen, classical adoption and diffusion models (e.g., Rogers [1962; 2003]) proved inadequate. Rather than being characterized by a few individuals who diffuse knowledge to others in a somewhat linear fashion, many students' pattern of technology adoption signaled situations in which various people were at times influential in different, ever-evolving social networks. The term "techne-mentor" is used to help to describe this pattern of information and knowledge diffusion…. Techne-mentor refers to a role that someone plays in aiding an individual or group with adopting or supporting some aspect of technology use in a specific context, but being a techne-mentor is not a permanent role.

Thus a "technology steward" is a specific kind of techne-mentor, who works on behalf of a community, mentoring and also being mentored in the context of that community. People learn to use a tool from people that they know, learning when needed. Knowledge, whether about technology or other matters, spreads along the lines of trust. When a team has a culture of shared learning, technology stewardship may be a widely distributed function. However, a designated technology steward can compensate for a less supportive culture in teams not so blessed. In that case the role of the technology steward is to be the instigator of social processes that spread technology skills widely throughout the project team. The following discussion of a platform exploration event suggests how tools are assessed and knowledge about their use can be spread at the same time.

TECHNOLOGY EXPLORATION AS A PROJECT PRACTICE

To illustrate how a group in a project team might explore how a technology could serve their needs to be, learn and work together, I describe a meeting where an informal learning group meets to test a new tool. This case involves a group of eight people who had met during CPsquare's Foundations of Communities of Practice workshop, where the author is one of the workshop leaders. This event can be seen as an example both of technology stewardship enacted collectively and as the kind of event that a technology steward would organize to generate involvement and knowledge about a tool throughout a social group.

During the Foundations Workshop we seek to establish conversational practices using several different media, including forums, chats, and teleconferences, so the idea of conversational practices that adapt to and frame the use of a technology is well established. In addition, participants have had several experiences of a process where one conversational medium is the platform for the introduction and familiarization with another. In this meeting, the group had a specific agenda: to explore use of one tool as a support for community meetings using previously established group practices and known technologies as a platform from which to

explore.

The purpose of our meeting was to explore http://TokBox.com, a video meeting tool. TokBox™ is a free video conferencing platform that sets up a "Hollywood Squares" format on a web page so that participants can see each other using a video camera attached to their computer. See Figure 5. Two people from the group had met on TokBox™ beforehand to make sure the tool was viable. Their initial meeting found that audio feedback problems made it difficult to communicate, so it was decided to use a regular teleconference call for the session's audio channel. One member of the group sent out an email invitation to all the workshop participants, (whether they had recently participated in these meetings or not). The invitation named the teleconference phone bridge as the initial meeting point.

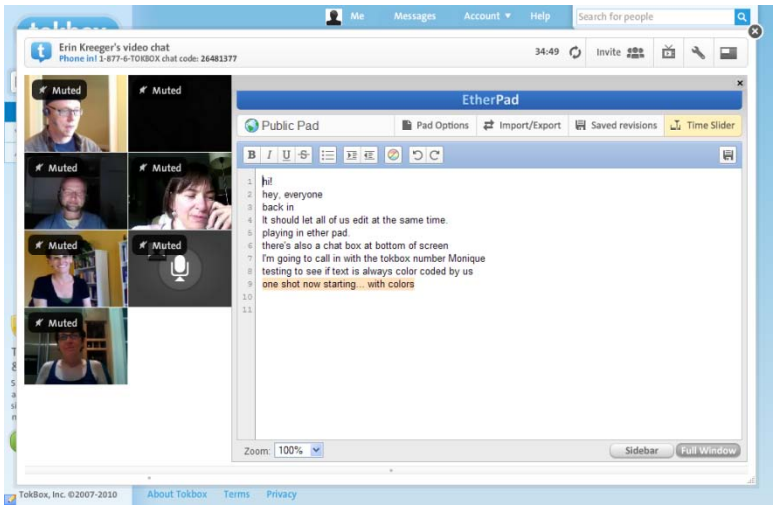

Figure 5. Being together on TokBox™

During the exploration, the phone bridge served as a platform to support our being together, although we also used several rapid-fire emails, Skype™ chats and eventually the TokBox™ tools themselves as channels to synchronize, facilitate and make sense of "being together." The first instruction that each person received on the conference call when they had logged on to TokBox™ was to find the TokBox™ mute button so as to avoid the echo through the phone bridge.

In addition to the phone bridge and backchannel connections, during the session we used several of the TokBox™ tools in turn, exploring their possibilities and comparing them to other alternatives. Some people didn't have a video connection (or preferred not to use one) and one just listened in to the conversation without connecting to TokBox™ at all, because they were on a mobile phone while driving. The group explored several TokBox™ tools: its chat tool, its etherpad, and some others. All of that made for a very complicated group structure. All of us could hear, but what each person could see was not the same.

The conversation was very much about observing out loud what we were seeing, considering how it worked for us, and thinking about how it would work for the several groups that each of us work with professionally. Was there value in seeing

other people's faces via the group video? (Answer: yes, for some, but not for all.) How would the tool work for a lecture versus a more open-ended conversation? What were the set-up issues in terms of inviting other people to join on the fly? Was there a difference between using the TokBox™ email invitation tool and sending the URL by some other means? (Answer: not much.) Although some web conferencing platforms completely lock down the structure and shape of the interface, TokBox™ lets you float video windows around, open and close apps like etherpad, and much more. What are the benefits of that kind of malleability? Does it also cause problems? (One of us kept getting dumped from the video connection whenever we entered an etherpad window. We never figured out why.) We compared TokBox to the many other tools that exist and which we have been exploring. (There are many related tools mentioned on the CPsquare wiki.[i])

It's obvious that to explore a social tool like TokBox™ you can't do it alone. You need partners. But to find out how it supports a conversation, you need to have a conversation that is as close to real work as possible, while being sure that you can detect and repair breakdowns. So you need other people who share your language, are willing to explore the tool, and can connect (and re-connect when the technology is confusing, misunderstood, or it simply fails). In particular it's helpful to have a robust and continuously available backchannel (the phone bridge, email and Skype™ chats in this case) to support the exploration. In this case, the phone bridge was also the standard against which we measured the tool (the null treatment, if you will) since it was well-known to all: our previous conversations had always been on the phone bridge.

In addition to organizing technology exploration, a technology steward brings a kind literacy about the social dynamics of collaborative technologies. Next we illustrate the kind of contribution that a technology steward brings to a distributed project team by examining one individual tool.

ONE PLATFORM: SKYPE™ FOR DISTRIBUTED COMMUNITIES AND PROJECT TEAMS

Technology stewards offer useful instruction to distributed teams about using individual tools, along the lines of a techne-mentor (Ito et al., 2009). But the nature of distributed teams may require that support be provided through the very technologies that a project team is trying to use. That frames the way support can be provided because it requires that open channels be identified in order to help someone explore a new one, very much along the lines of the technology exploration examples in the previous section. In addition to support, a technology steward can provide a higher level of service by helping a distributed team makes sense of its digital habitat: those insights come from stepping back from the specific characteristics of one tool in order to see a landscape of tools that, in combination, support a group's need to be and work together. In Wenger, White and Smith (2009) we propose technology polarities as key elements of technology stewardship literacy. These polarities are inherent tensions that address fundamental issues of learning and interactions which groups of people must face. They are useful for describing tools as they are used by small or large groups of people. They provide a framework for systematic communication about how tools work together with respect to communication and collaboration issues. These include issues such as how one tool duplicates another, how one might replace another, what a "missing tool" might be, and how tools might be used in combination. As the polarities focus attention on a tool's functions, they frame choices that are made at a platform level, where tools are aggregated, purchased, installed and updated.

The following discussion uses Skype™ to illustrate the issues that the polarities bring into focus. The following definitions of the polarities are paraphrased from Wenger, White and Smith (2009):

- **Rhythms** answer the question: *What are sources of togetherness and separation and is the rhythm optimal for the community or distributed team?* People live apart but want to practice and work together. Technology both enables new kinds of togetherness and creates new kinds of separation. In addition to the physical dimension, technology allows communities and project teams to interact at the same time or to "time-shift". Multiple time zones are an inescapable reality in global project teams.

- **Interactions** answer the question: *What kind of reification needs to be produced and how can this be integrated into appropriate forms of participation to generate both meaningful learning and productive work experiences?* People experience meaningful learning and work at the same time as they create representations or records of their experiences. Technologies afford new opportunities for teams and communities both to have new experiences and to represent them in new ways.

- **Identities** answer the question: *How much group participation (and conformity) does a community or project team want and how flexible does it need to be to accommodate individual choices and participation in multiple communities and projects?* Communities emerge as a result of an ensemble effort but are experienced individually. Technology provides new ways for groups to form but also for more individual differentiation.

Meaningful communication about a tool's use requires that we point to specific buttons and understand the context of use and experience. Given that we often have many tools to choose from, that we use them in specific combinations and that that the tools a group uses interact with each other in complex ways, communication at the right level of specificity or generality matters considerably. How a technology steward communicates about a group's tools and experience of them affects usability, learning, and collaboration. Although most people probably think of Skype™ as a personal or individual tool, it is complex enough to demonstrate the issues involved in understanding a collaboration platform. It is not just one tool. It is a platform that carries many different tools with it, whether they are used or not.

Skype™ is a phone.

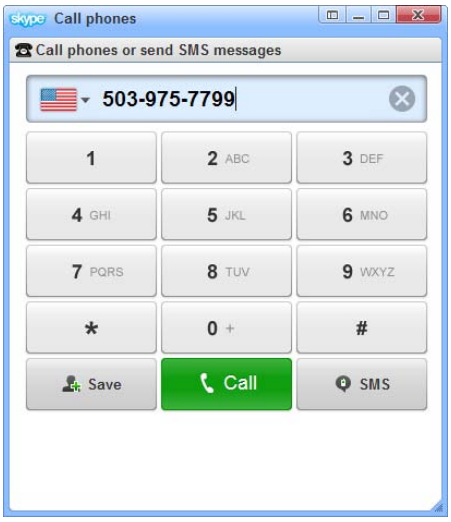

Figure 6. Skype™ as a phone on your computer

The most obvious thing about Skype™ is that it works like a phone, on a computer, over the Internet. One-to-one interaction on the spur-of-the-moment is the ideal way to resolve an issue. Skype™ is simple

to use: you click on a keypad or enter numbers in a dialog box to call people. See Figure 6. But it's not so simple: Skype™ is two phone tools that have useful features in common but they are different in that one is for calling a regular number (you pay a small fee) and another is for calling other Skype™ users (with a Skype™ ID, for free).

A phone (whether of the Skype™ variety or not) can readily be placed on the polarities diagram in Figure 7. The phone is on the participation end; you have to participate in real time, so it's synchronous; and it is a one-to-one experience, so it belongs close to the individual end of the spectrum. The placements in Figure 7 determine where a given tool will appear in a tool landscape diagram such as Figure 10. Notice that both *how a tool can be used* and *how a tool **is** used* changes its position in terms of the polarities. An example of the former is that a Skype-to-Skype™ call can easily be returned, but Skype™ can't receive calls from a regular phone number unless you buy a special service; that moves the tool along the *Identities* polarity toward the individual end of the spectrum. An example of the latter is that Skype™ makes conference calls very easy but their quality depends on the bandwidth of the initiator of the conference call (through whom everyone on the conference call communicates); if the initiator of the conference call has limited or unstable bandwidth the call will suffer from poor quality: easy conference calls move the tool toward the group end, but having to pay attention to the initiator's bandwidth pulls it toward the individual end.

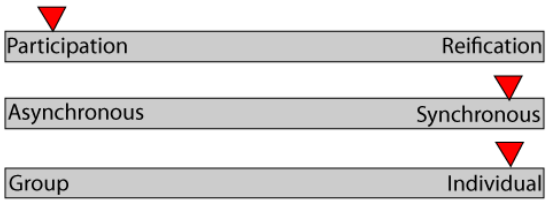

Figure 7. Polarities of Skype™ as a phone

Contact list tool.

Whether calling another Skype™ user or a regular phone number, you can use Skype's contact list tool. The contact list tool sits at the same place on the polarities that other contact list tools do, including Microsoft Outlook, Gmail, and mobile phone contact tools, but it has extra features that the others don't. For example, the contact list tool shows which other Skype™ users are currently "available," indicated by a green dot with a check-mark in it, so it works like a global presence or availability indicator. You know in advance whether someone is available for a call or not. Because the contact list tool is not just a static record of addresses, the availability indicator moves its position toward the participation end of the rhythm spectrum. You can easily rename contacts and group them, which makes the tool even more "individual." You can easily initiate calls and chats with groups of Skype™ users, moving the tool toward the group end of the identities spectrum. Therefore the Skype™ contact list tool, compared to Skype™ as a phone, is located more toward the reification end, it's more asynchronous, and somewhat more of a group tool. Figure 8 suggests where the Skype™ contact list tool sits on the three polarities.

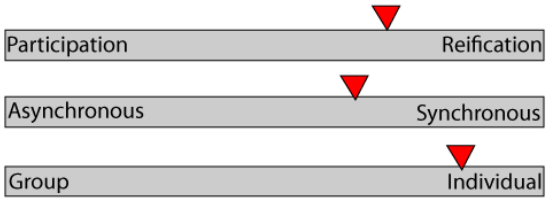

Figure 8. Polarities of the Skype™ contact list

The design of the contact list tool suggests that Skype™ has a strong design bias toward the individual end of the spectrum. An "advanced" option on the contacts menu lets you save all your contacts to a standard contacts file but it's not designed so that a project roster (e.g., a subset of an individual's contacts) can easily be saved. Skype™ makes it easy to share a group of contacts via Skype™ itself, however. The intricacies of sharing contact information at a group level are a critical concern of a technology steward.

Chat: a flexible tool spanning multiple polarities.

Chat is one of the most versatile collaboration tools that exist. A chat is useful for conversations, for alerts, for sharing resources, for negotiating meeting times, for announcing an impending meeting, and on and on. It is notable that there are many different and incompatible chat protocols and tools. Once you have a chat connection with someone the possibilities for collaboration increase dramatically.

Skype's text chat tool is similar to many such tools: it is a well-designed instant message tool for interaction with plain text. It is probably the most heavily used of all of Skype's tools after the phone. The polarities give us a way to discuss the tool's functions and some of the variant practices that it enables. Although Skype™ chat is an "instant message," it keeps chat transcripts on your machine permanently and you can search through them. Therefore we place it in the middle of the rhythm polarity, balancing participation and reification: everything you ever said in a chat or anyone said to you in a chat is available. Chats are synchronous in the sense that everyone receives a chat message when it is sent; however several factors tilt the interaction polarity somewhat toward the asynchronous: 1) Skype™ will deliver chat messages asynchronously to a member of a chat that is not logged on, 2) chat transcripts are persistent and searchable, and 3) unlike many chats, an individual is not dropped off the chat when they are inactive, allowing someone to just listen in. A chat is inherently a group activity and Skype™ does not really support a customized experience, so it is placed toward the group end of the Identities polarity in Figure 9.

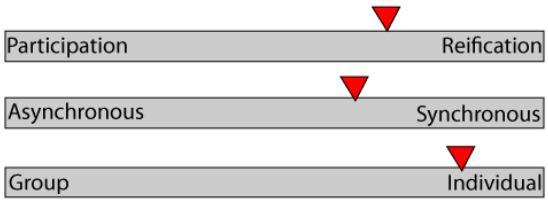

Figure 9. Location of Skype™ chat on the polarities map

To illustrate the idea that the polarities are a tool for understanding the use and the experience of use, I will show how specific uses of Skype™ move the chat tool in different directions from the balance shown in Figure 9. Figure 10 shows the three Skype™ tools discussed here along with three chat uses that are relevant to distributed teams which I discuss below.

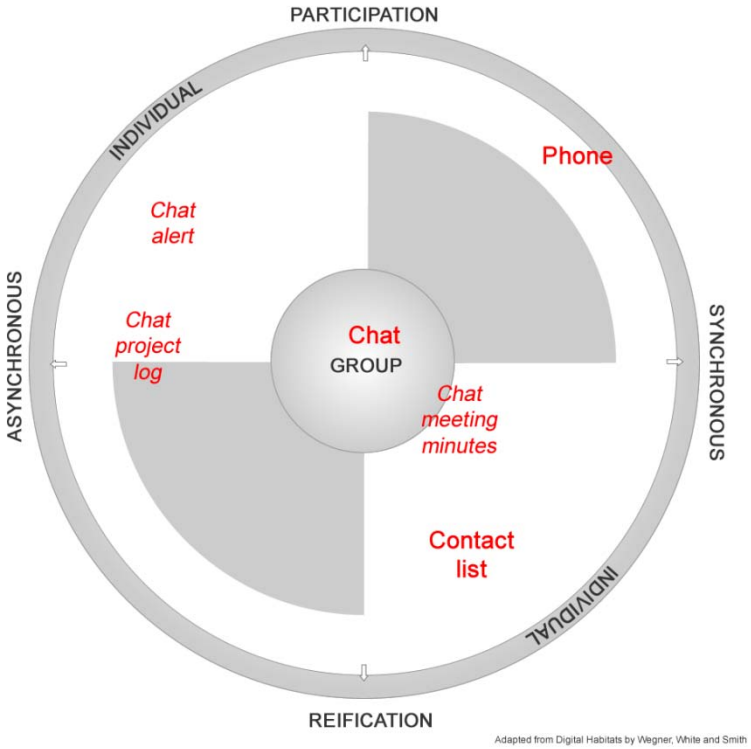

Figure 10. Three Skype™ tools and three uses of chat on the digital landscape

The first chat use variant to mention is a "Chat Alert." It involves using the Skype™ text chat as an alert – to drop notes off on another person's desk. The drop-off may be just a URL and the message is no more than "Hey, look at this!" A direct message on Twitter or the inbox feature on http://delicious.com would be obvious alternatives, but on a windows machine Skype™ blinks so that it is hard to miss. No response is required but an alert can lead to extended conversations. Because a response is not expected, this use puts chat more toward the asynchronous end of the Rhythm polarity. As a targeted drop-off, it is more individual. This is an example of the ubiquitous casual synchronization that makes distributed teams able to function.

The second use variant is a "Chat project log." When a small group is working on a project over a long period, for example, a long-running Skype™ chat is a good way to keep everybody connected and focused. This is particularly so when the project involves alternating synchronous and asynchronous interaction. When you turn on your computer in the morning, all the conversations between people in different time zones pop up. Because you can choose how to participate, whether to follow a conversation moment-by-moment, I show this variant at the asynchronous end of the rhythm polarity and the individual end of the identities polarity.

The third use variant is "Chat meeting minutes." A large number of people can be on a text chat, making it ideal for capturing both group consensus and individual variances. Making meetings self-documenting by ensemble note-taking can increase efficiency and can help groups bridge linguistic barriers. English, for example, is spoken with many different accents but its written form is relatively standard. Thus a

practice where notes are captured in Skype™ in real time helps people decode what others are saying. The idea of writing together during a meeting with the intention of producing minutes that will be used in the future moves this variant toward the reification end of the interaction polarity.

Implications.

The point of using these polarities and the feature-tool-platform-configuration scheme are to enable consideration of issues such as tool duplication, replacement, technology gaps, and combinations. The polarities represent natural categories that help a technology steward take a step back from the hands-on level and make sense of the experiences that enable a community to be together and to learn. This discussion of Skype™ as a community or project team tool illustrates some of the issues that distributed project teams must face and that a technology steward can shed light on.

Here are some further thoughts to consider in reflecting on the use of polarities:

- A tool's polarities are determined as much by their design as by their technological background and how they fit within a larger configuration.
- Tech stewards need to understand what it's like to use a tool and to be able to talk about the experience and the tool separately.
- Preferred, ignored, duplicated, or competing tools all make sense within a social and technical mix we call a digital habitat.
- Each software feature makes sense within the context of a tool, and each tool is framed by its position on a platform, which has meaning in the context of a configuration that's shared by a group of people.
- In a way it's all circular because you can't see a community or a project team's configuration (or digital habitat) directly or simply.
    - You can't stand outside of your own digital habitat.
    - You can't really see a community unless you're participating in their habitat.
    - Seeing their habitat as they see it requires relationships and access to their practices, habits, and cultural frame.

DIRECTIONS FOR FURTHER RESEARCH

The community orientations that were developed in *Digital Habitats* (Wenger, White, & Smith, 2009) are specific to communities, but may suggest lines of thought and research for distributed project teams.

National culture is another dimension that may be important when selecting technologies. Are individualist cultures more likely to prefer asynchronous tools and collectivist cultures more likely to want synch tools? How might cultural dimensions affect the polarities? The same questions could be considered regarding generational issues: people who have grown up with the support of techne-mentors have a very different attitude toward technology than those who received substantial instruction on the use of technology.

The technology exploration discussion suggests several overlapping questions:

- How can a project team explore existing variance in the use of a tool? What are the benefits of (or, possibly, the problems with) uniform competence in using a tool once a team has settled on it? In the example case described in this chapter, some people didn't want to use video at all or found that it didn't add much to their experience of closeness beyond what our phone bridge provided. For others it added quite a bit of context and sense of closeness that was useful.

- Is it always clear what tool to use as a platform for exploration **from**? Does that matter? Different project teams might use different technologies and will have different amounts of trust or determination to explore. In this chapter, we used email to get everyone on a phone bridge from which we all got into TokBox™. Stragglers got caught up via Skype™ chat.
- A final question is about what this process of exploration does to the group itself. Can it be outsourced? Can we leverage the experience of others? What are the implications of having others do the exploration for us, be they experts in a company's IT department or technology stewards or whomever? In the examples described in this chapter we were very much doing it for ourselves and that certainly colors our experience. How important is firsthand experience of exploration?

Using the polarities To talk in more precise terms about tools is a useful contribution. Accumulating a corpus of reflection on current technologies, such as is being developed in CPsquare (http://cpsquare.org/wiki/Technology_for_Communities_project) will both enable further research and be a research resource as well.

CONCLUSION

This chapter has described the similarities and differences between communities and distributed project teams. It examined several variants, where a community or project team as a social formation overlap, coincide or provide a background for the other. It has applied research originally carried out in a community setting and applied it to distributed projects. The concrete example of technology exploration illustrates the way that multiple technologies are used in a distributed project environment. The use of the polarities developed in *Digital Habitats* (Wenger, White, & Smith, 2009) shows how a technology steward can help a group understand its technology needs more precisely.

REFERENCES

Cleland, D. I., and Gareis, R. (2006). *Global project management handbook.* New York: McGraw-Hill Professional.

Derby, E. & Larsen, D. (2006). *Agile retrospectives: Making good teams great.* Raleigh, NC: The Pragmatic Bookshelf.

Hinds, P.J. & Kiesler, S. (2002). *Distributed work.* Cambridge, MA: MIT Press.

Ito, M., et al. (2009). *Hanging out, messing around, and geeking out: Kids living and learning with new media.* Cambridge: MIT Press.

Justesen, S. (2006). *Navigating the paradoxes of diversity in innovation practice*. Frederiksberg: Copenhagen Business School.

Lave, J. & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.* New York: Cambridge University Press.

Lloyd, A. (2007). *A system that works for me-an anthropological analysis of computer hackers' shared use and development of the Ubuntu Linux system.* Unpublished Masters Thesis, Department of Anthropology, University of Copenhagen. Retrieved August 2, 2010, from http://eskar.dk/andreas/lloyd_thesis.pdf

NASA. (2001). *NASA contractor financial management reporting.* Retrieved August 2, 2010, from http://nodis3.gsfc.nasa.gov/npg_img/N_PR_9501_002D_/N_PR_9501_002D__Chp2.pdf

Ruuska, I. (2005). *Social structures as communities for knowledge sharing in project-based environments*. Doctoral Dissertation Series 2005/3, Helsinki University of Technology, Department of Industrial Engineering and Management.

Tuckman, B. (1965). Developmental sequence in small groups. *Psychological Bulletin, 63*, 384-399.

Wenger, E. (1998). *Communities of practice: Learning, meaning and identity*. New York: Cambridge University Press.

Wenger, E. (2006). *Communities of practice: A brief introduction*. Retrieved July 1, 2010, from http://www.ewenger.com/theory/communities_of_practice_intro.htm

Wenger, E., McDermott, R. & Snyder, W.M. (2002). *Cultivating communities of practice: A guide to managing knowledge*. Cambridge: Harvard Business School Press.

Wenger, E. & Snyder, W. (2000). Communities of practice: The organizational frontier. *Harvard Business Review*, January-February, 139-145.

Wenger, E., White, N. & Smith, J.D. (2009) *Digital habitats: Stewarding technology for communities.* Portland, OR: CPsquare.

Zuboff, S. (1988). *In the age of the smart machine: The future of work and power*. New York: Perseus Books Group.

ADDITIONAL READING SECTION

Arnold A., and Smith, J. D. (2003) Adding connectivity and losing context with ICT: Contrasting learning situations from a community of practice perspective. In Marleen Huysman, Etienne Wenger and Volker Wulf, *Communities and Technologies; Proceedings of the First International Conference on Communities and Technologies*. Dordrecht: Kluwer Academic Publishers.

Arnold A., Smith, J. D., and Trayner, B. (2006). Narrative: Designing for context in virtual settings. In Antonio D. de Figueredo and Ana P. Afonso (Eds.) *Managing learning in Virtual Settings: The Role of Context*, pp. 197-218. Hershey, PA: Information Science Publishing.

Bernoff, J., and Li, C. (2008). *Groundswell: Winning in a world transformed by social technologies.* Cambridge: Harvard Business School Press.

Denning, S. (2010). *The leader's guide to radical management: Re-inventing the workplace for the 21st century.* San Francisco: Jossey-Bass.

Kimble, C., and Paul Hildreth, Eds. (2008). *Communities of practice: Creating learning environments for educators*, Volume 1 (PB). Charlotte, NC: Information Age Publishing.

Morgan, G. (1986). *Images of organization*. Newbury Park, CA: SAGE Publications.

Shaw, P. (2002). *Changing conversations in organizations: A complexity approach to change*. New York and London: Routledge.

Shirky, C. (2008). *Here comes everybody*. New York: Penguin Press.

Wenger, E. (2001). *Supporting communities of practice; A survey of community-oriented technologies*. Draft Version 1.3, March 2001. Retrieved August 2, 2010, from http://ewenger.com/tech/index.htm

KEY TERMS & DEFINITIONS

Backchannel: A communication channel that plays a supportive or background role in the overall communication process of a group. During a telephone conference call, for example, having a chat conversation in the background can help clarify terms, share details, or queue speaking order.

Chat: Same-time (synchronous) text interaction over the Internet. Chat can be a one-to-one interaction, such as with an Instant Message service, or a large group interaction, as in an IRC chat room.

Community of practice: A group of people who interact over time around a specific topic in order to develop individual and collective competence.

Configuration of technologies: The overall set of technologies that serve as a substrate for a community's habitat at a given point in time—whether tools belong to a single platform, to multiple platforms, or are free-standing.

Digital habitat: The same thing as a group's configuration of technologies.

Feature: A characteristic that makes a tool or a platform usable for a specific purpose. Some features define a tool; others add to its functionality or to the enjoyment of the experience. A phone without a microphone is not a phone, but a mute button is an element that adds functionality.

Platform: A technology package that integrates a number of tools available in the marketplace (for purchase or for free) that one can acquire, install, or rent. Vendors often organize a group of tools as a platform. Platforms offer teams and communities a simple entry into using a set of tools.

Social media: A general term to describe activities that involve social interaction, technology, and user-generated content.

Technology stewards: People with enough experience of the workings of a community to understand its technology needs, and enough experience with technology to take leadership in addressing those needs. Stewardship typically includes selecting and configuring technology, as well as supporting its use in the practice of the community.

Tool: An identifiable piece of technology that supports a discrete activity in a community (for example, a discussion board that supports online conversations) or bridges different types of activities (for example, recording a phone conversation for later use).

[i] See http://cpsquare.org/wiki/Technology_for_Communities_project