12-14-2013

Highlighting Communication Activities and Inefficiencies Between Agile vs. Waterfall Methods: An Agent Based Model of Knowledge Sharing

Meghann L. Drury-Grogan
Fordham University, mdrury@fordham.edu

Deanna M. Kennedy
University of Washington Bothell, dkennedy@uwb.edu

Highlighting Communication Activities and Inefficiencies Between Agile vs. Waterfall Methods: An Agent Based Model of Knowledge Sharing

Meghann L. Drury-Grogan
The Schools of Business
Fordham University
5 Columbus Circle, 11-08
New York, NY 10019
mdrury@fordham.edu

Deanna M. Kennedy
School of Business
University of Washington Bothell
18115 NE Campus Way
Bothell, WA 98011
dkennedy@uwb.edu

ABSTRACT

We employed agent-based simulation techniques to create a dynamic multi-level team environment to study communication activities as knowledge sharing occurred. We examined knowledge seekers and knowledge providers who act and react to one another's communication behavior on Waterfall versus Agile teams using rich versus lean media to answer research questions regarding inefficient use of team members. The simulation model was checked for validity against assumptions that project management method drives project schedule and communication media motivates the number of meetings. Results further indicate that (a) slightly more knowledge seekers exceed their knowledge need on Agile teams using lean versus rich media; (b) knowledge overage was reduced by utilizing a Waterfall rather than Agile method, and through the use of lean media; and (c) the maximum time wasted by team members who completed gathering knowledge to meet their initial needs was on Agile teams using lean media.

Keywords

Project Management Procedures; Knowledge Management; Communication; Team Communication; Waterfall Software Development, Agile Software Development; Agent-based Modeling.

INTRODUCTION

Successful projects depend on members to create, apply, and share knowledge (Fong, 2003; Nonaka and Takeuchi, 1995; Yuan, Zhang, Chen, Vogel and Chu, 2009). We define knowledge as "information possessed in the mind of individuals: it is personalized information (which may or may not be new, unique, useful, or accurate) related to facts, procedures, concepts, interpretations, ideas, observations, and judgments" (p. 109, Alavi and Leidner, 2001). The recognized need for knowledge sharing, especially among project team members, has created a burgeoning area of research about the practices and facilitation (e.g., Wang and Ko, 2012). However, while knowledge sharing has been conceptualized as a dynamic process with multi-level scenarios across individual, team, and organization (Nonaka and Takeuchi, 1995), little has been done to show how these conceptualizations could play out over time. As a dynamic process, time plays a critical role in understanding team processes and outcomes (Kozlowski, Chao, Grand, Braun, and Kuljanin, 2013). Additionally, interactive processes, such as knowledge sharing, may depend on the effectiveness of individual level activities (Hackman, 2003; Klein and Kozlowski, 2000). To address the gap in our understanding of knowledge sharing on project teams as a temporal and multi-level phenomenon, we aim to examine knowledge sharing between team members and gain insights about team outcomes.

An individual level activity for sharing knowledge is through team communication where knowledge providers and knowledge seekers connect to fill gaps in knowledge and coordination (Wang and Ko, 2012). Researchers acknowledge the importance of team communication for project team performance (e.g., Hsu, Shih, Chiang and Liu, 2012; Park and Lee, in press) over most other PMBOK (Project Management Body of Knowledge) knowledge areas (Peltoniemi, Jokinen and Mönkkönen, 2004). Recognizing the importance of communication practices, organizations may try to affect team communication through (1) the planned use of different communication media; and (2) the project structure which sets interactive meeting time constraints.

First, the communication media, and particularly the richness of the media available for team member interactions, may affect the duration of communication and capacity to convey cues (Daft and Lengel, 1986) that may influence the timeliness and amount of knowledge shared. Second, organizations often implement Project Management (PM), a system with the intentional purpose of promoting the gathering of knowledge from others (Gray and Meister, 2004). The Waterfall method brings team members together periodically to elaborately discuss milestone achievements by subject area (Jurison, 1999), while the Agile method calls for more frequent yet shorter interaction sessions (Cusumano and Smith, 1995; Hass, 2007). The PM method may set different schedules and frequency for meetings between members further affecting the timeliness of knowledge acquisition.

While organizational inputs may extend the time it takes some members to seek out knowledge, these delays may in turn lead other members to inefficiently utilize their time as they gather and process information beyond what they need to complete their own job. As such, some members may be subject to information overload which brings about fatigue, errors, and conflict (Edmunds and Morris, 2000) as they continue to meet others and gain information. Moreover, the energy wasted in attending unnecessary meetings can lead to burnout (Salanova, Peiro and Schaufeli, 2002). We seek to understand how communication media and PM methods differentially enable knowledge inefficiencies such that some members experience overload and wasted energies. To draw out the outcomes of interest, we track team communication where we assume that knowledge is being acquired during conversations between team members. We pursue three research questions (RQ):

(1) *To what extent does the available media's richness and PM method schedule, Waterfall or Agile, affect the number of team members exceeding their knowledge need?*

(2) *To what extent does the available media's richness and PM method schedule, Waterfall or Agile, affect the maximum amount of knowledge by which a team member exceeds his/her knowledge need?*

(3) *To what extent does the available media's richness and PM method schedule, Waterfall or Agile, affect the maximum amount of time a team member wastes after filling his/her knowledge need?*

Our contribution to the project management literature comes from taking a temporal and multi-level perspective of knowledge sharing to uncover how team members may be inefficiently utilized on projects depending on communication media use and PM method. Moreover, we are able to make novel contributions by studying teams staffed in similar ways, i.e., the same amount of knowledge seekers and knowledge providers with similar requirements while varying organizational inputs. To do so, we employ agent-based modeling to create a dynamic team environment. Researchers have increasingly employed simulation techniques to provide insights into complex interactions among organizational members and their tools (e.g., Espinosa and Carmel, 2003; Noguiera and Raz, 2006). Agent-based models are beneficial for investigating emergent phenomena that results from interactions among agents (Davis, Eisenhardt and Bingham, 2007; Harrison, Lin, Carroll and Carley, 2007). Herein, team members act and react to one another's communication behavior as knowledge sharing occurs. As such, our parsimonious model provides insights meant to inform future research about potential areas to target regarding knowledge sharing on project teams.

BACKGROUND LITERATURE

This section presents a temporally based, multilevel conceptualization of knowledge sharing to frame our research.

Knowledge Sharing

Teams are groups of individuals that work together, are dependent upon one another and have one or more tasks to perform in order to accomplish various goals (Hackman, 1990; Mayer, Davis and Schoorman, 1995). Specifically, team members rely on each other for knowledge to produce a successful output based on the collective contribution of all team members (Katzenbach and Smith, 2005). As Nonaka and Takeuchi (1995) suggest, teams are agents of knowledge creation whereby individual, team, and organizational contexts interact. Through dialogue team members can share explicit knowledge that can be converted into tacit knowledge (Nonaka and Konno, 1998) and, ultimately, collective learning (Fong, 2003). Establishing places, such as at meetings, where knowledge seekers and knowledge providers can open a dialogue may promote knowledge sharing (Nonaka and Konno, 1998). While the interaction between tacit and explicit knowledge has been conceptualized as a spiraling process, when spread out over time, the course of activities may fluctuate between explicit and tacit activities. Figure 1 demonstrates the conceptualization of the way knowledge creation plays out over time through the progression of explicit knowledge sharing at formal meetings (e.g., planning meetings, scrum meetings and retrospective meetings in agile or the planning meetings and milestone meetings in waterfall) and tacit knowledge creation when working individually. Moreover, we are interested in how these activities unfold under different organizational inputs including available communication media and meeting schedules per PM method. We discuss these organizational inputs in greater detail below.

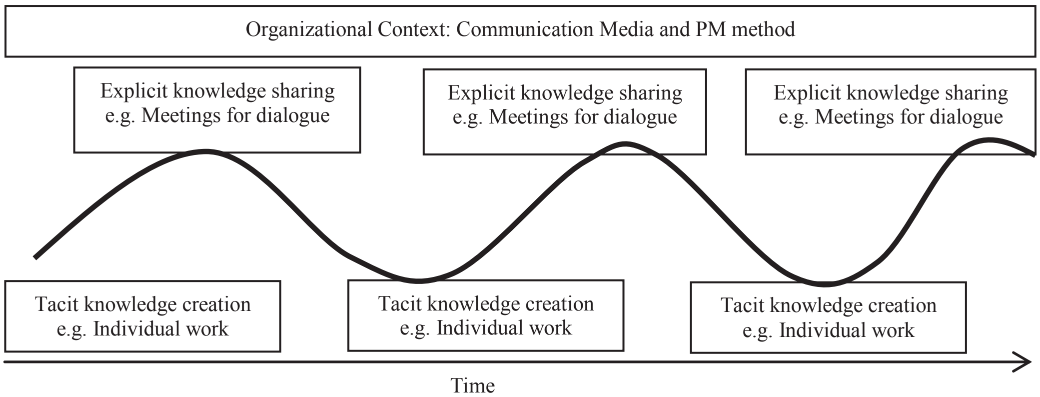

Figure 1. Knowledge Sharing Conceptualization

Media Richness

Temporal differences in interpersonal communication activities may be driven by the communication media being used for interactions. Media richness theory, advanced by Daft and Lengel (1986), suggests that communication channels differ in cue-carrying capacity (i.e., the types and amount of information made available to be processed into knowledge that can be effectively transmitted). As such, different communication media may be considered along a media richness continuum that is anchored by rich media and lean media (Chidambaram and Jones, 1993). Rich media generally has a high cue-carrying capacity because it allows for multiple types and amounts of information to be transferred (i.e., verbal, paraverbal, and nonverbal) (Daft and Lengel, 1986). The types of rich media teams use include face-to-face and other tools such as voice over internet protocol (VOIP) or video-conferencing that allow teams to collaborate synchronously as collaboration can occur via information technologies that allow people to interact with each other simultaneously (e.g., Baker, 2002; Drury and Williams, 2002), though synchronous rich media can increase communication time and costs on projects (Natu and Kennedy, 2012). Based on these findings about cue-carrying capacity and duration, we associate rich media with longer durations but more knowledge content being transferred between knowledge seeker and provider.

Lean media is limited in terms of cue-carrying capacity because it restricts non-verbal and paraverbal cues (Straus, 1997). As a result of this limitation, researchers suggest that teams may exchange shorter messages through these means than through richer media (Boyle, Anderson and Newlands, 1994). Interestingly, however, researchers suggest that when using lean tools, such as email or instant messaging, team members are prone to use them as much if not more often than richer media options (LaToza, Venolia and DeLine, 2006). Thus, we assign shorter messages when lean media is used with less knowledge content being transferred in any one exchange between seeker and provider than using rich media.

Project Management Methods for Software Development

PM methods are used to facilitate many software development (SD) projects as these types of projects are notably knowledge-intensive, requiring the integration of knowledge from various domains (Patnayakuni, Rai and Tiwana, 2007). PM methods set meeting schedules so that the team members adequately interact around the six project management functions of planning, organizing, executing, monitoring, reporting, and controlling to manage the knowledge quality (Gannon, 1994). As well, meetings allow members to effectively share knowledge because not all team members will possess the knowledge for each of these PM activities (Chau, Maurer and Melnik, 2003) as each has their own limited experience on which to draw (Drury, Acton, Conboy and Golden, 2011). Two methods that have received little direct comparison and are the basis for this research are the Waterfall and Agile methods.

Waterfall Software Development

The more traditional approach and foundational SD standard (Sommerville, 1996) is the Waterfall method where the optimal SD process is a linear, sequential series of project management phases (Cusumano et al., 1995). The classic Waterfall method was modeled after hardware design projects where engineers could predict how parts of the system would interact (Royce, 1970). It is a unidirectional, top down process with a sequence of activities that is non-iterative (Fitzgerald, 2000; Sommerville, 1996) where phases of the process move from one to the next. In this sequential design process, the progress of a project flows steadily downwards through the phases like a waterfall: requirements specification, design, implementation (coding), integration, testing and debugging, installation, and maintenance. One can only move onto the next phase when the preceding phase is completed accurately (Royce, 1970).

Waterfall is a traditional PM method with disciplined, deliberate planning and control methods. Such teams focus on the process of work rather than team members' creativity as work allocation clarifies what and how work will be done within a specific timeframe (Taylor, 1998). The project is planned up front as tasks are completed one after another in an orderly sequence in distinct project life cycle phases that are easily recognizable and not revisited upon completion (Hass, 2007). The design, development, and testing takes place by different teams, and these teams only put these modules together and test the entire system in the last phase of the project, which usually requires reworking the modules and writing new code to correct problems from the interactions of the modules due to unforeseen problems, mistakes, miscommunications, or design changes that occurred during the project (Cusumano et al., 1995; Jurison, 1999). This process is efficient so long as system requirements remain stable until the end of the project because all requirements are captured before any design and development occurs so customer feedback is not incorporated (Chau et al., 2003). Thus, the major advantage of Waterfall is that it provides a structure for organizing and controlling a SD project, though it must identify user requirements accurately at the beginning of the project (Cusumano et al., 1995).

Agile Software Development

However, past research shows Waterfall was not delivering cost-effective and user-driven software to customers (Lyytinen, 1987) as projects were unpredictable and rarely followed a sequential flow where customers could identify all requirements upfront (Hass, 2007). Agile allows for iterative development that simultaneously designs products and processes (Ballard and Howell, 2003) on small collaborative SD teams (Dybå and Dingsøyr, 2008). These teams work under extreme time pressure to develop and deliver working software to customers in short iterations (Fitzgerald, Hartnett and Conboy, 2006; Fowler and Highsmith, 2001), a time-boxed period of fixed length, often two weeks' duration (Schwaber and Beedle, 2002). Iterative development is the process of building a system within a short period of time (Larman, 2004). Because of these short iterations, Agile teams are able to respond quickly to changes in the business environment, technology, and customer requirements by continually redesigning and adapting development processes (Henderson-Sellers and Serour, 2005) as they only plan for one iteration at a time.

Agile team interactions focus on communication to convey knowledge (Cockburn and Highsmith, 2001). Researchers report that the most efficient and effective method of conveying knowledge for Agile teams is face-to-face communication (Fowler et al., 2001), so teams work in close proximity to foster face-to-face communication, timely feedback, and informal social interaction, though they can experience a lack of team engagement when members feel a lack of decision ownership and empowerment (Drury, Conboy and Power, 2012). Proximity refers to the actual physical distance between people (Hinds and Kiesler, 2002) as collocation is a key Agile tenet (Green, Mazzuchi and Sarkani, 2010).

Another key tenet to Agile is frequent and continuous communication with the type of communication medium key. There is an emphasis on richer communication medium rather than lean ones, particularly during the beginning of the development project (Green et al., 2010). While lean technologies of today have created more cost-effective ways to communicate over vast differences, they cannot fully replace the power of rich communication (Carmel and Agarwal, 2001). Nevertheless, Agile teams like other teams must balance between rich and lean medium as organizations strive to reduce development timelines, deliver products to market faster, and leverage cheaper SD resources across the world (Green et al., 2010).

METHOD

In this study, agent-based simulation is employed to study a 2 (PM method) x 3 (communication media scenario) design. We use this approach because an abstract model that characterizes real-world environments can provide preliminary insights for these other methods that direct better use of time and budgets towards viable and meaningful effects in other settings (Harrison et al., 2007). The scope of the current model is to reproduce the basic and general communication patterns of team members working on one stage of a SD project. When designing the model we followed practical suggestions (Davis et al., 2007) and established guidelines for decision making and model construction (Rand and Rust, 2011). Figure 2 illustrates the progression through the design decisions. Table 1 reports the decisions for each of the simulation elements utilized in simulation procedures. To ensure our program has face validity, we (1) model parameters after published parameters, theorized relationships, and noted insights from past studies, and (2) provide a preliminary test in the results section to show that the parameters have the expected effects that characterize real-world situations of Waterfall and Agile projects, and projects using rich and/or lean media. The simulation is programmed in NetLogo (Wilensky, 1999).

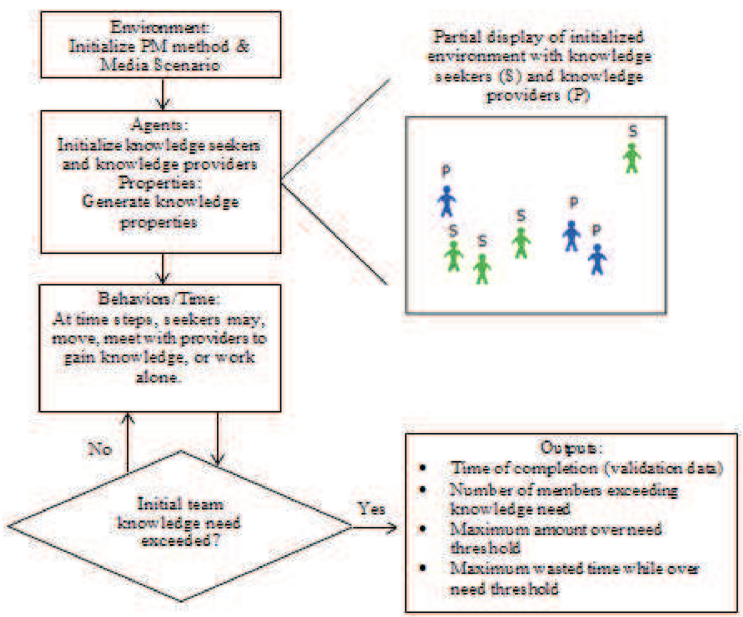

Figure 2. Project Management Simulation Procedures

Decision Area	Application
Environment	• ***PM method*** **to schedule time allotment for meeting and for individual work where knowledge is not shared, with the Waterfall method taking longer than the Agile method (Cusumano et al., 1995). We therefore make the following time allotment for the two PM methods:** **W) Waterfall method allots 16 hours (time steps) for meeting, 20 hours for individual work.** **A) Agile method allots 4 hours for meeting, 10 hours for individual work.** • ***Media Scenario*** **to determine the available media richness. The different scenarios set meeting duration (how much of available meeting time used) and content amount (how much of the provider's knowledge is shared) (Chidambaram et al., 1993; Daft et al., 1986; LaToza et al., 2006).** **R) Rich media scenario: 100% meeting duration, 100% content.** **L) Lean media scenario: 25% meeting duration, 50% content.** **M) Mix media scenario: equal probability of using rich or lean media and commiserate duration and content quantity.**
Agents	• ***Team members*** **on a SD project.** **Knowledge seekers: 16 members (i.e., 2 sub-teams) who need knowledge.** **Knowledge providers: 16 members (i.e., 2 sub-teams) who provide knowledge to seekers when not already in a meeting with another knowledge seeker.**
Properties	• ***Knowledge need*** **of knowledge seekers randomly drawn from a Poisson distribution with mean of 40.** • ***Knowledge expertise*** **of knowledge providers randomly drawn from a Poisson distribution with mean of 5.** **(**This approach assigns positive integer values to knowledge seekers that are sufficiently large enough to require communication with at least one knowledge provider; such interactions may reflect a project with

	inherent interdependence that requires cross-functional collaboration.)
Behaviors	At each time step knowledge seekers are either moving to meet a knowledge provider, in a meeting with a knowledge provider agent, or are working individually and not gaining knowledge from another member. • *Movement*: Knowledge seekers will move in random directions until a knowledge provider is encountered. Movement reflects the difficulty that researchers have noted in members' abilities with locating and accessing knowledge resources (e.g., Chang, Yen, Chiang and Parolia, 2013). • *Meeting*: A meeting is initialized and collaboration commences when a knowledge seeker meets an idle knowledge provider. After the meeting, the knowledge seeker's need is reduced by the provided amount. The knowledge seeker must then move away to work individually for a given amount of time or engage with another knowledge provider. • *Knowledge Sharing*: Over the course of the project, until stage completion, we allow knowledge seekers to continue gaining knowledge, even when their own need is fulfilled. This behavior maps to real-world activities where teammates may be assuming responsibilities for other members or are otherwise gaining and processing unnecessary knowledge (Salanova et al., 2002).
Timing	• *Time steps:* Steps map to the hours of time that team members may spend collaborating with teammates or working individually on the project.
Stopping Criteria	• *Criterion:* When the cumulative initial knowledge need for all knowledge seekers is exceeded by the cumulative amount of knowledge gained over time. At the time step when the simulation stops, the project stage is considered complete.
Outputs	• *Preliminary Tests (Validation):* Length of the project in terms of speed and effective meetings. • *Members Exceeding Their Knowledge Need:* Number of knowledge seekers that went over their need. • *Maximum Amount of Knowledge Gathered Exceeding Knowledge Need:* The maximum amount any knowledge seeker went over his/her need. • *Maximum Time Wasted Filling Knowledge Need*: The maximum wasted time that a knowledge seeker spent over their need limit.

Table 1. Project Management Simulation Decisions

Herein we model independent behaviors over time as members have opportunities to meet and knowledge seekers (i.e., novices) can acquire knowledge from providers (i.e., experts). The need and supply of knowledge are determined based on Poisson distributions where seekers' needs average 40 knowledge units representing a need across multiple areas, and providers supply average 5 knowledge units representing an expertise in one area. The meeting opportunities occur per a schedule directed by the PM method. As such, the PM method determines when and for how long team members have to interact (i.e., allocated hours for meeting and individual work). Waterfall allows for longer open meetings (16 hours or 2 days) and longer non-meeting times when members are expected to work alone (20 hours) than the Agile method (4 hours of open meeting time followed by 10 hours of individual time). During open meeting times knowledge seekers can contact a knowledge provider and if the provider is free they communicate. The communication media is selected based on set availability; for example, in the rich media scenario, members always meet using rich media (e.g., face-to-face) where 100% of the knowledge given is absorbed, but the time consumed takes 100% of the meeting length. Alternatively, if the mix media scenario is used, each time team members meet they have a 50/50 chance of using rich or lean media. The selection decision is not stochastic, that is, previous use of rich media does not preclude rich media from being selected in the next meeting. The meetings and individual work continue until the culmination of knowledge needed by knowledge seekers is overcome. As such, all teams satisfied exactly or exceeded the team's cumulative knowledge need (i.e., created overage) by the time the project stage completed. This threshold indicates that the needed knowledge was sufficiently distributed among team members, although any particular knowledge seeker may end up below or above their initial need.

RESULTS

The methods described above were used to simulate six datasets, based on the experimental design, each with 1000 runs. The data collected was then analyzed using MANOVA with Student-Neuman-Keuls (SNK) rankings. The first analysis was conducted to assess the model that the generated activities reflect expectations of real-world projects using Waterfall or Agile methods and rich and/or lean media. The second analysis examines output data to inform insights about the RQs.

Preliminary Tests

To demonstrate that our model provides results that could be expected of projects using the PM methods or communication media of interest, we compare the time of project stage completion across conditions. In terms of time, the results in Table 2 show that teams using the Waterfall method, regardless of the communication media, take longer than teams using an Agile method, which is expected based on our parameters and what we know of Waterfall and Agile teams method (Cusumano et al., 1995) and therefore serves as an accuracy check for our model. The fastest teams were those using Agile in a rich media scenario (*speed* $\mu_{A\text{-}R}$ = 254.29). The slowest teams were those using Waterfall in a lean media scenario that took almost two and a half times longer than the fast group (*speed* $\mu_{W\text{-}L}$ = 636.21). The results also show the number of meetings was strongly driven by the media scenario with lean media requiring the most meetings (*meetings* $\mu_{W\text{-}L}$ = 256.51 and *meetings* $\mu_{A\text{-}L}$ = 256.44). This also makes sense as lean media provides less information and therefore requires more meetings compared to rich media (Boyle et al., 1994. Thus, we proceed to utilize this model to uncover the interactive effects of organizational inputs on inefficiencies from members seeking knowledge.

	Speed (hours)	SNK	Meetings (meeting-count)	SNK
Waterfall - Rich Media	497.89	*e*	128.81	*a*
Waterfall - Lean Media	636.21	*f*	256.51	*c*
Waterfall - Mix Media	448.00	*d*	171.30	*b*
Agile - Rich Media	254.29	*a*	128.90	*a*
Agile - Lean Media	422.46	*c*	256.44	*c*
Agile - Mix Media	286.96	*b*	171.47	*b*
F-stat	22708.91**		28220.60**	
Df	5.00		5.00	

*+ p < 0.10, * p < 0.05, ** p < 0.01*

Note. Mean values reported in table. Those means assigned the same letter were not significantly different.

Table 2. Preliminary Tests

Output Tests

Table 3 presents the MANOVA and SNK results of output comparisons across datasets. Based on the results we provide insights to our RQs.

Our first RQ asked about the number of team members exceeding their knowledge need. The results indicate that teams were almost equal in terms of the number of knowledge seekers over their knowledge need. The test reports a marginally significant difference (*F-stat* = 2.09, p = 0.06) between Agile teams in the rich media scenario (*members* $\mu_{A\text{-}R}$ = 7.75) and Agile teams in the lean media scenario (*members* $\mu_{A\text{-}L}$ = 7.91).

When looking at the maximum amount of knowledge overage by knowledge seekers that exceed their own need (RQ(2), we found many differences across PM method and media conditions. The members with the most knowledge beyond their need were on Agile teams in the rich media scenario (*overage* $\mu_{A\text{-}R}$ = 24.27) and Agile teams in the mix media scenario (*overage* $\mu_{A\text{-}M}$ = 24.10). These members gained almost one and a half times more knowledge than members with the lowest amount of overage on Waterfall teams in the lean media scenario (*overage* $\mu_{W\text{-}L}$ = 17.26). Taken together, these results indicate that overage is reduced by utilizing a Waterfall rather than an

	Members (number of members exceeding knowledge need)	SNK	Overage (maximum knowledge overage)	SNK	Waste (maximum time spent over, in hours)	SNK
Waterfall - Rich Media	7.80	*ab*	21.21	*c*	4.32	*A*
Waterfall - Lean Media	7.86	*ab*	17.26	*a*	6.94	*E*
Waterfall - Mix Media	7.86	*ab*	21.50	*c*	5.59	*C*
Agile - Rich Media	7.75	*b*	24.27	*d*	4.92	*B*
Agile - Lean Media	7.91	*a*	19.04	*b*	7.67	*F*
Agile - Mix Media	7.80	*ab*	24.10	*d*	6.32	*D*

F-stat	2.09+	209.57**	674.11**
df	5.00	5.00	5.00

*+ p < 0.10, * p < 0.05, ** p < 0.01*
Note. Mean values reported in table. Those means assigned the same letter were not significantly different.

Table 3. Output Comparison Tests

Agile method, and additionally through the use of lean media.

Finally we assessed the maximum time wasted by team members that had completed gathering knowledge to meet their initial needs (RQ(3). The results indicate that the highest waste was by Agile teams in the lean media scenario (*waste* $\mu_{A\text{-}L}$ = 7.67 hours). This amount of time is 177% more than by teams with the least waste, Waterfall teams in the rich media condition (*waste* $\mu_{W\text{-}R}$ = 4.32 hours). While the amount of time represents only a fraction of the project duration, and that fraction is less on Waterfall projects because they tend to be longer projects than Agile projects, the important trend is that the wasted time by knowledge seekers is reduced when rich media is used.

DISCUSSION

The pattern of results does provide support that our model represents the way PM methods and communication media might be expected to unfold in two ways. First, in terms of speed, the results indicate that Agile teams finish the project faster than Waterfall teams. Since the tenet of Agile is to use short iterations to quickly iterate and produce a viable outcome (Henderson-Sellers et al., 2005), then the results meet expectations. Second, in terms of meetings, the use of rich media produces the fewest meetings, followed by mixed media, and lastly lean media. Given that rich media allows for the most knowledge to be shared in any one meeting (Daft et al., 1986), it makes sense to see the progression of meeting counts increase in the order shown. In summary, the pattern of results for organizational inputs suggests that communication activities are unfolding as we anticipate from projects using these PM methods and media. Next we look at how these inputs interactively affect knowledge seeking to provide insights for each RQ.

RQ(1) The Number of Team Members Exceeding Their Knowledge Need

Results indicate that teams in all conditions were almost equal in terms of the number of knowledge seekers exceeding their knowledge need, although slightly more knowledge seekers exceeded their knowledge need on Agile teams in the lean media scenario compared to Agile teams in the rich media scenario. Because members receive more knowledge with rich media (Chidambaram et al., 1993), perhaps members in the rich media scenario pace their knowledge gathering better than those using lean media. As members gain smaller amounts of knowledge using lean media, they may be gathering knowledge in small increments and this may lead more members to go over at the same time. Thus, for Agile teams, we recommend they rely more on rich than lean media to gather their knowledge because rich media allows for fewer members exceeding their knowledge need and better-paced knowledge gathering as they gain more knowledge at a time via rich versus lean media (RQ2). This insight corroborates other research promoting collocation for Agile teams despite organizations' push for distributed teams to reduce development timelines and leverage cheaper software development resources across the world (Green et al., 2010).

RQ(2) The Maximum Amount of Knowledge by Which a Team Member Exceeds His/Her Knowledge Need

Our results suggest that knowledge overage is reduced by utilizing a Waterfall rather than Agile method, and through the use of lean media. Because lean media provides smaller amounts of information (Boyle et al., 1994), it may prevent excessive knowledge overage because members do not gain large amounts at a time using lean media; thus, lean media can help mitigate knowledge overload as it can be more difficult to limit the amount of knowledge using rich media since you cannot cutoff the knowledge exchange. For example, using email as a lean medium, a member can delete the email without reading or processing the information, but when talking to someone face-to-face, the member cannot stop the communication as easily since it is a live interaction.

Furthermore, while it seems at first glance the results prefer teams use Waterfall methods with lean media, we must recall that Agile teams are purpose-built to respond quickly to change (Henderson-Sellers et al., 2005). Thus, due to Agile's iterative nature, members may be gaining more knowledge each iteration which compensates for the rate of change these teams incorporate into projects. Agile teams only plan for two-week iterations rather than the entire project like Waterfall teams, and they incorporate changes into such short-term planning so their knowledge needs will be greater and not necessarily an overage as they may use that additional knowledge when incorporating a later change. As such, Agile teams might consider a balance of rich and lean media to ensure they gain enough knowledge and are prepared for knowledge needs due to incorporating change regularly into their iterations.

RQ(3) The Maximum Amount of Time a Team Member Wastes After Filling His/Her Knowledge Need

The final RQ assesses the maximum time wasted by team members that had completed gathering knowledge to meet their initial needs, with the highest waste from Agile teams in the lean media scenario. We recommend teams use rich media to prevent waste as lean media does not provide as much useful knowledge as rich media. But while the results also suggest using a Waterfall method to prevent waste, we encourage teams to consider the length of overall project duration. Waterfall projects are much longer than Agile projects, so members waste less percentage of time on Waterfall projects than on Agile projects. Agile teams in the long-run waste less time gathering knowledge, freeing those team members to contribute to other value-add tasks. Thus, there is a trade-off between gathering too much extra knowledge which might help Agile teams prepare for change (RQ2) versus wasting time doing nothing. But if a member is doing nothing, perhaps the team can redirect that member to more value-add activities, thereby using time more efficiently.

LIMITATIONS

Our investigation utilized simulation procedures to assess knowledge sharing on projects. The model met expectations, and allowed us to examine 6000 project teams across six different experimental conditions that would have been cost and time prohibitive in other settings. To develop the simulation we relied on published parameters, theorized relationships, and insights from multiple studies to construct our simulation model. As such, we cannot validate our simulation on an independent set of data and settle for a pretest approach (Boudreau, Gefen, and Straub, 2001). The purpose of simulations is to create abstractions of reality (Harrison et al., 2007), and the model produced herein followed procedures (Rand and Rust, 2011) meant to represent a general model of theoretical relationships. As such, we were concerned with face validity, whether the certain constructs in the model produced expected effects (Boudreau et al., 2001), which we demonstrate through our preliminary analysis of project length for different communication media and PM methods. Given our limited validation applied, however, our results provide speculative initial insights about knowledge sharing over time on projects. Future researchers may seek additional validation using empirical data following established guidelines (e.g., Straub, Boudreau, and Gefen, 2004) to make more definitive implications.

CONCLUSION

In summary, our investigation of knowledge sharing provides at least three contributions. First, we consider inefficiencies to provide insights about the dark side of knowledge sharing under different organizational inputs. Second, PM methods like Agile and Waterfall provide the temporal structure for managing team interactions, and using different media types is often a mechanism for knowledge management directed in the project communication plan (Schwalbe, 2010). Through our study we suggest relative differences in PM methods and media use for diffusing the effects of knowledge sharing inefficiencies. Third, this paper describes the development of an agent-based simulation model of project management. As such, we provide a methodological contribution to the study of project team environments, knowledge sharing, and process outcomes. The utility of such an approach is to guide future research about complex parameters like knowledge sharing. Thus, this paper is the foundation for the authors to continue more complex study via simulation and ultimately with actual project teams in industry.

ACKNOWLEDGMENTS

We thank Randika Ekanayake for helping with initial set-up of the simulation and data collection.

REFERENCES

Alavi, M., and Leidner, D.E. (2001) Knowledge management and knowledge management systems: Conceptual foundations and research issues, *MIS Quarterly* 25, 1, 107-136.

Baker, G. (2002) The effects of synchronous collaborative technologies on decision making: A study of virtual teams, *Information Resources Management Journal* 15, 4, 79-93.

Ballard, G., and Howell, G. (2003) Lean Project Management, *Building Research & Information* 31, 2, 119-133.

Boudreau, M.-C., Gefen, D., and Straub, D. Validation in IS research: A state-of-the-art assessment, *MIS Quarterly* 25, 1, 1-24.

Boyle, E.A., Anderson, A.H., and Newlands, A. (1994) The effects of visibility on dialogue and performance in a cooperative problem solving task, *Language and Speech* 37, 1, 1-20.

Carmel, E., and Agarwal, R. (2001) Tactical approaches for alleviating distance in global software development, *IEEE Software* 18, 2, 22-29.

Chang, K., Yen, H., Chiang, C., and Parolia, N. (2013) Knowledge contribution in informaiton system development teams: An empirical research from a social cognitive perspective, *International Journal of Project Management* 31, 2, 252-263.

Chau, T., Maurer, F., and Melnik, G. (2003) Knowledge sharing: Agile methods vs. tayloristic methods, *Proceedings of the Twelfth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, IEEE International Workshops, 2003.*

Chidambaram, L., and Jones, B. (1993) Impact of communication medium and computer support on group perceptions and performance: A comparison of face-to-face and dispersed meetings, *MIS Quarterly* 17, 4, 465-491.

Cockburn, A., and Highsmith, J. (2001) Agile Software Development: The People Factor, *IEEE Computer* 34, 11.

Cusumano, M.A., and Smith, S.A. (1995) Beyond the waterfall: Software development at Microsoft, *Working papers from Massachusetts Institute of Technology (MIT), Sloan School of Management* Working Paper #3844-BPS-95.

Daft, R.L., and Lengel, R.H. (1986) Organizational information requirements, media richness and structural design, *Management Science* 32, 5, 554-571.

Davis, J.P., Eisenhardt, K.M., and Bingham, C.B. (2007) Developing theory through simulation methods, *Academy of Management Review* 32, 2, 480-499.

Drury, J., and Williams, M.G. (2002) A framework for role-based specification and evaluation of awareness support in synchronous collaborative applications, *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2002.*, WET ICE 2002. Proceedings. Eleventh IEEE International Workshops on, , pp. 12-17.

Drury, M., Acton, T., Conboy, K., and Golden, W. (2011) The role of experience in agile software development decision making, *The 6th International Research Workshop on IT Project Management , a pre-conference workshop for the 2011 International Conference on Information Systems (ICIS), hosted by the Special Interest Group for IT Project Management (SIGITProjMgmt) in the Association for Information Systems (AIS)*, Shanghai, China.

Drury, M., Conboy, K., and Power, K. (2012) Obstacles to Decision Making in Agile Software Development Teams, *Journal of Systems and Software* 85, 6, 1239-1254.

Dybå, T., and Dingsøyr, T. (2008) Empirical Studies of Agile Software Development: A Systematic Review, *Information and Software Technology* 50, 9-10, 833-859.

Edmunds, A., and Morris, A. (2000) The problem of informaiton overload in business: A review of the literature, *International Journal of Information Management* 20, 17-28.

Espinosa, J.A., and Carmel, E. (2003) The impact of time separation on coordination in global software teams: A conceptual foundation, *Software Process Improvement in Practice* 8, 4, 249-266.

Fitzgerald, B. (2000) Systems development methodologies: the problem of tenses, *Information Technology & People* 13, 3, 174-185.

Fitzgerald, B., Hartnett, G., and Conboy, K. (2006) Customising Agile Methods to Software Practices, *European Journal of Information Systems* 15, 2, 197-210.

Fong, P.S.W. (2003) Knowledge creation in multidisciplinary project teams: an empirical study of the processes and their dynamic interrelationships, *International Journal of Project Management* 21, 7, 479-486.

Fowler, M., and Highsmith, J. (2001) The Agile Manifesto, *Software Development* 9, 8, 28-32.

Gannon, A. (1994) Project management: An approach to accomplishing things, *ARMA Records Management Quarterly*28, 3-12.

Gray, P.H., and Meister, D.B. (2004) Knowledge sourcing effectiveness, *Management Science* 50, 6, 821-834.

Green, R., Mazzuchi, T., and Sarkani, S. (2010) Communication and Quality in Distributed Agile Development: An Empirical Case Study, *Proceedings of World Academy of Science, Engineering and Technology* 61.

Hackman, J.R. (1990) Groups that Work (and those that don't). Creating Conditions for Effective Teamwork, Jossey-Bass Inc. San Francisco, CA.

Hackman, J. R. (2003) Learning more by crossing levels: evidence from airplanes, hospitals, and orchestras. *Journal of Organizational Behavior* 24, 8, 905-922.

Harrison, J.R., Lin, Z., Carroll, G.R., and Carley, K.M. (2007) Simulation modeling in organizational and management research, *Academy of Management Review* 32, 4, 1229-1245.

Hass, K.B. (2007) The blending of traditional and agile project management, *PM world today* 9, 5, 1-8.

Henderson-Sellers, B., and Serour, M. (2005) Creating a Dual-Agility Method: The Value of Method Engineering, *Journal of Database Management* 16, 4, 1-23.

Hinds, P., and Kiesler, S. (Eds.). (2002) Distributed Work, Massachusetts Institute of Technology, Cambridge, MA.

Hsu, J.S., Shih, S., Chiang, J.C., and Liu, J.Y. (2012) The inpact of transactive memory systems on IS development teams' coordination, communication, and performance, *International Journal of Project Management* 30, 329-340.

Jurison, J. (1999) Software Project Management: The Manager's View, *Communication of the AIS* 2, 17.

Katzenbach, J.R., and Smith, D.K. (2005) The Discipline of Teams, *Harvard Business Review* 83, 7/8, 162-171.

Klein, K. J., and Kozlowski, S. W. J. (2000) From micro to meso critical steps: Conducting multilevel research. *Organizational Research Methods* 3, 3, 211-236.

Kozlowski, S. W. J., Chao, G. T., Grand, J. A., Braun, M. T., and Kuljanin, G. (2013) Advancing multilevel research design: Capturing the dynamics of emergence. *Organizational Research Methods* 16, 4, 581-615.

Larman, C. (2004) Agile & Iterative Development: A Manager's Guide, Addison-Wesley, Boston, MA.

LaToza, T., Venolia, G., and DeLine, R. (2006) Maintaining mental models: A study of developer work habits, in: *ACM ICSE'06*, ACM, Shanghai, China.

Lyytinen, K. (1987) A taxonomic perspective of information systems development: theoretical constructs and recommendations, *Critical issues in information systems research*, John Wiley & Sons, Inc. , pp. 3-41.

Mayer, R.C., Davis, J.H., and Schoorman, F.D. (1995) An Integrative Model of Organizational Trust, *Academy of Management Review* 20, 3, 709-734.

Natu, A.A., and Kennedy, D.M. (2012) A Simulation Study of Project Management and Collaborative Information Technologies, *Proceedings of the Eighteenth Americas Conference on Information Systems*, Seattle, Washington.

Noguiera, J.C., and Raz, T. (2006) Structure and flexibility of project teams under turbulent environments: An application of agent-based simulation, *Project Management Journal* 37, 2, 5-10.

Nonaka, I., and Takeuchi, H. (1995) The knowledge creating company, Oxford University Press, New York.

Nonaka, I., and Konno, N. (1998) The concept of "ba": Building a foundation for knowledge creation, *California Management Review* 40, 3, 40-54.

Park, J., and Lee, J. (in press) Knowledge sharing in information systems development projects: Explicating the role of dependence and trust, *International Journal of Project Management*, 1-13.

Patnayakuni, R., Rai, A., and Tiwana, A. (2007) Systems development process improvements: A knowledge integration perspective, *IEEE Transactions on Engineering Management* 54, 286-300.

Peltoniemi, M., Jokinen, T., and Mönkkönen, J. (2004) Project communications as a critical success factor in high technology projects, in: *IAMOT 2004*, Washington DC.

Rand, W., and Rust, R.T. (2011) Agent-based modeling in marketing: Guidelines for rigor, *International Journal of Research in Marketing* 28, 3, 181-193.

Royce, W.W. (1970) Managing the Development of Large Software Systems, *Proceedings of IEEE Wescon*, Los Angeles, CA, USA.

Salanova, M., Peiro, J.M., and Schaufeli, W.B. (2002) Self-efficacy specificity and burnout among information technology workers: An extension of the job demand-control model, *European Journal of Work and Organizational Psychology* 11, 1, 1-25.

Schwaber, K., and Beedle, M. (2002) Agile Software Development with Scrum, Prentice Hall, NJ.

Schwalbe, K. (2010) Information technology project management 6th Edition. , Course Technology Cengage Learning., Boston, MA.

Sommerville, I. (1996) Software process models, *ACM Computing Surveys (CSUR)* 28, 1, 269-271.

Straub, D., Boudreau, M.-C., and Gefen, D. (2004) Validation guidelines for IS posivist research, *Communications of the Association for Information Systems* 14, 380-426.

Straus, S.G. (1997) Technology, group process, and group outcomes: Testing the connections in computer-mediated and face-to-face groups, *Human-Computer Interaction* 12, 227-266.

Taylor, F. (1998) The Principles of Scientific Management, Dover Publications.

Wang, W., and Ko, N. (2012) Knowledge sharing practices of project teams when encountering changes in project scope: A contingency approach, *Journal of Information Science* 38, 5, 423-441.

Wilensky, U. (1999) NetLogo. http://ccl.northwestern.edu/netlogo/, Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL. .

Yuan, M., Zhang, X., Chen, Z., Vogel, D.R., and Chu, X. (2009) Antecedents of coordination effectiveness of software developer dyads from interacting teams: An empirical investigation, *IEEE Transactions on Engineering Management* 56, 3, 494-507.