2013

IT Interruptions and Coordination Effectiveness in Software Development Groups: A Conceptual, Multilevel Model

Shamel Addas
Information Systems, IESEG School of Management, Lille, France., s.addas@ieseg.fr

Alain Pinsonneault
McGill University, alain.pinsonneault@mcgill.ca

IT Interruptions and Coordination Effectiveness in Software Development Groups: A Conceptual, Multilevel Model

Shamel Addas
IESEG School of Management (LEM-CNRS)
s.addas@ieseg.fr

Alain Pinsonneault
McGill University
alain.pinsonneault@mcgill.ca

ABSTRACT

Research abounds on software teams enhancing their processes via IT. However, the unintended group-level effects of interruptions triggered by such IT are rarely examined. This paper develops a conceptual, multilevel model that focuses on the paths linking individually experienced IT interruptions to group coordination outcomes. Drawing on coordination theory and the work interruptions literature, we propose that different IT interruption types exhibit different effects. IT intrusions create resource constraints that emerge to the group level via interdependencies and debilitate group coordination effectiveness. To mitigate these effects, groups engage in coordination by task organization. IT interventions facilitate coordination by group problem-solving (a cross-level effect), which enhances coordination effectiveness. This research extends the IT interruptions literature by focusing on the multilevel effects, and extends the IT impacts literature by unearthing the unintended differential effects of IT via interruptions of group members' work.

Keywords

IT interruptions; multilevel model; coordination effectiveness.

INTRODUCTION

Software teams rely on information technology (IT) to enhance their coordination, but such IT also interrupts their work (Rennecker and Godwin, 2005, Dabbish et al., 2007, Chong and Siino, 2006). Extant research focuses on isolated, individual-level interruptions (e.g., Adamczyk and Bailey, 2004, Cutrell et al., 2000, Speier et al., 1997, Avrahami et al., 2007). Little is known about the group-level effects of such interruptions on the group's coordination outcomes.

Drawing on coordination theory (Malone and Crowston, 1994) and the work interruptions literature (Jett and George, 2003), this paper proposes that IT interruptions are experienced individually but – due to interdependencies – spill over to the group level and exhibit differential impacts on coordination outcomes depending on interruption type.

Two contributions are made. First, we extend the IT impacts literature by focusing on the unintended impacts via group-level interruptions. Second, while interruptions research focuses on individual or dyadic outcomes, this research is an early attempt to examine multilevel impacts of IT interruptions on group coordination outcomes.

We first develop the notion of IT interruptions in group contexts. Then, we use coordination theory to build a conceptual multilevel model linking IT interruption types to group coordination effectiveness. We conclude with the discussion and research implications.

IT INTERRUPTIONS IN GROUPS

IT interruptions are perceived, IT-based external events that capture attention and break primary task continuity (Jett and George, 2003, Addas and Pinsonneault, 2010). Primary tasks represent individuals' core project activities. IT influences interruptions in two ways: (1) directly causing them (e.g., via software crashes/ issues); (2) mediating them (e.g., via email; SMS).

Types of IT Interruptions

Building on Jett and George (2003), we present two IT interruption types: intrusions and interventions. Intrusions compel one to turn attention away from primary project activities. Their contents can be related to activities in other projects, secondary activities in the current project, or non-project activities (e.g., a corporate meeting request interrupting a software developer's activity).

Interventions directly pertain to individuals' primary activities and motivate behavioral changes in them in response to perceived task performance discrepancies (Jett and George, 2003). Rather than trigger attention switching, interventions refocus attention to the source of the problem (e.g., an email from a software co-developer about resolving a programming issue).

IT Interruptions for Individuals Working in Groups

We scanned the literature on interruptions for individuals within groups. Three observations are made (Table with results excluded for space limitations). First, interruptions exhibit differential group-level effects, depending on interruption type: intrusions are generally negative (e.g., Dabbish and Kraut, 2008, Harr and Kaptelinin, 2007), while interventions produce positive outcomes such as better idea generation (Jessup and Connolly, 1993) and enhanced knowledge integration (Okhuysen and Eisenhardt, 2002).

Second, the literature hints only implicitly at the existence of multilevel interruption models. For example, some studies reflect single unit-level relationships (e.g., Okhuysen and Eisenhardt, 2002, Matsui et al., 1987), but these do not show how these group-level variables may be tied to micro-level aspects. Most reviewed studies depict cross-level relationships, but without conceptualizing the group-level outcomes as global, shared, or configural constructs (Kozlowski and Klein, 2000).

Third, the literature does not systematically specify task interdependencies, which are important to understand how the effects of individual-level interruptions aggregate to higher levels (e.g., Dabbish and Kraut, 2008).

In summary, prior interruptions literature elucidated some of the individual and group effects, but we lack a theoretical basis that explains the multilevel relationships between IT interruptions and group-level outcomes such as coordination effectiveness when tasks are tightly interlinked. In the following, we draw on coordination theory to develop a multilevel model of IT interruptions.

COORDINATION THEORY

Coordination theory focuses on interdependencies between a group's activities, which create constraints requiring effective coordination.

Coordination Problems

According to coordination theory (1994), any process involves *actors* (individuals or collectives) performing *interdependent activities* to achieve *goals*. Activities require or create *resources* (e.g., time; effort; skill; information). Coordination problems can result from three types of dependencies: (1) sharing dependencies (when multiple activities use the same resource); (2) fit dependencies (when multiple activities together produce a single resource); (3) flow dependencies (when one activity produces a resource that is used by another activity) (Malone et al., 1999).

Coordination Mechanisms

Coordination problems can be resolved via two primary coordination mechanisms: *task organization* and *group problem-solving*. Coordination via *task organization* is a structural arrangement involving programmed practices to manipulate tasks and resources (Gittell, 2002, van De Ven et al., 1976). These practices, while often predefined (March and Simon, 1958), can also arise in direct response to coordination problems (Malone and Crowston, 1994, Wittenbaum et al., 2002). For example, a reoccurring software bug issue can be addressed by reusing a previous solution (fit dependency).

Coordination via *group problem-solving* involves an organic process of intense communication and knowledge sharing between group members to resolve novel or discrepant task situations (Gittell, 2002, Okhuysen and Eisenhardt, 2002). Group problem-solving discussions may be formal or informal (Espinosa et al., 2004), written or verbal (Rico et al., 2008), and scheduled or unscheduled (van De Ven et al., 1976).

Coordination Effectiveness

Coordination effectiveness is defined as *the extent to which dependencies have been effectively managed* (Espinosa et al., 2004). It includes three dimensions: technical, temporal, and process. The technical dimension assesses how well technically-oriented dependencies are managed (e.g., when in a software project multiple software components are well integrated and work well together, Espinosa et al., 2004). The temporal dimension reflects the extent to which interdependent tasks are completed on schedule. Finally, the process dimension assesses the effectiveness in managing process dependencies (e.g., performing software project activities according to an established process) (Espinosa et al., 2004).

A MULTILEVEL MODEL OF IT INTERRUPTIONS

The multilevel model in Figure 1 suggests that IT intrusions trigger individual-level constraints that emerge to the group-level and debilitate group coordination effectiveness. This is mitigated by task organization coordination (moderation and partial mediation links). IT interventions elicit a group problem-solving coordination mode, which enhances coordination effectiveness.

IT Intrusions & Group Coordination Effectiveness

We propose that IT intrusions decrease group coordination effectiveness because of resource constraints triggered by interdependencies (sharing and flow).

Time Constraints (Sharing Dependencies)

Intrusions increase perceived time pressures for individuals (France et al. 2005; McFarlane 2002; Adamczyk et al. 2004). These individual perceptions propagate to the group-level (Karau and Kelly, 1992), especially when the group works on interdependent tasks for extended periods (Chong et al., 2011). Research on product development teams supports this effect (Gersick, 1989, Chong et al., 2011, Perlow, 1999).

Time-pressured groups attend less to coordinating their resources and outputs, and more to achieving "quick fix efficiencies" (Kelly and McGrath, 1985) and taking shortcuts (Alvero et al., 2001) to meet task demands (McGrath, 1991, Chong et al., 2011). This may not influence temporal coordination effectiveness (Abdel-Hamid et al., 1999). However, group members devote less time to ensuring adequate component integration and adherence to established development processes. They thus reduce promised functionality (Costello, 1984, Pries-Heje and Pries-Heje, 2011), customer involvement (Pries-Heje and Pries-Heje, 2011), and system testing (Brooks, 1979), and ignore maintenance issues (Pries-Heje and Pries-Heje, 2011).

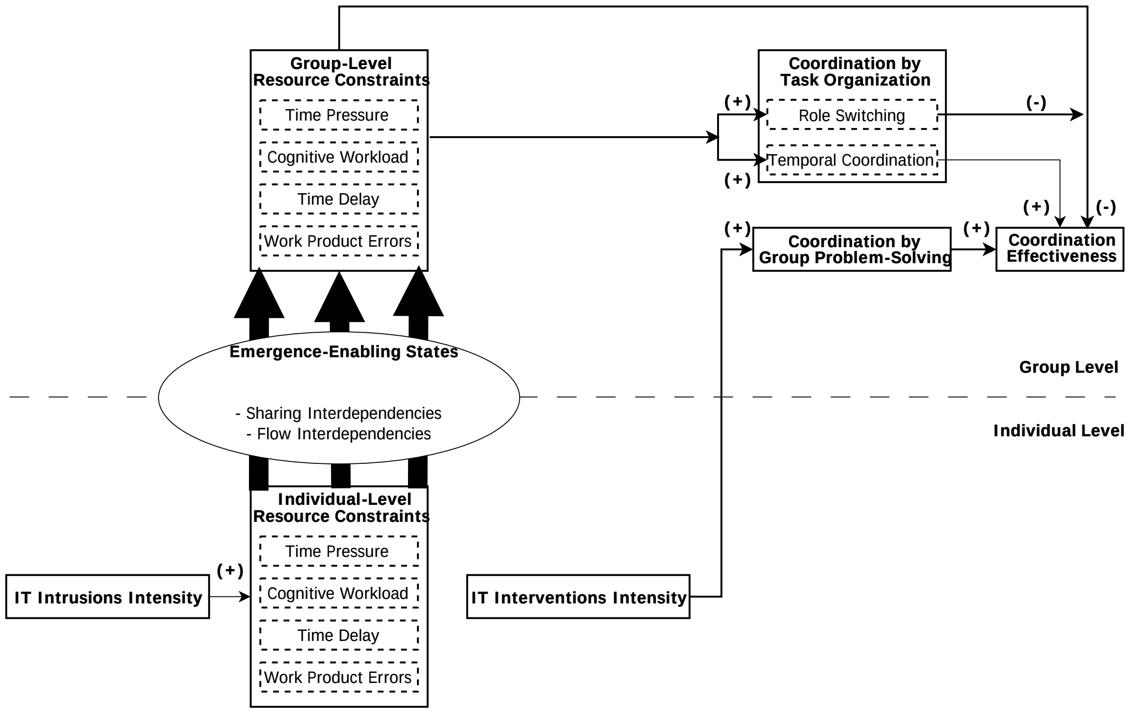

Figure 1. Conceptual Model

Proposition 1a: IT intrusions increase time pressure among software teams, which aggregates to the group-level because of sharing dependencies. This hinders effective task execution and diminishes group coordination effectiveness (technical and process).

Cognitive Load Constraints (Sharing Dependencies)

When cognitive effort required to complete a task exceeds available capacity, this results in high cognitive workload for an individual (Urban and Hauser, 1993, Bowers et al., 1997). Intrusions increase cognitive workload because of the additional effort that must be allocated and shared between the primary and intrusion activities (Basoglu et al., 2009, Gievska and Sibert, 2005). This detrimental effect emerges to the group-level due to sharing interdependencies. Team members have a finite effort capacity for managing their interrupted tasks (taskwork), communicating and coordinating such fragmented tasks (teamwork), and timesharing between the taskwork and teamwork (Bowers et al., 1997, Funke et al., 2012).

Group workload research found that increased cognitive workload diminishes coordination activities (Urban et al., 1996) and triggers coordination breakdowns as a result of shifting effort toward taskwork rather than teamwork (Bearman et al., 2010, Serfaty and Kleinman, 1990). This can trigger non-adherence to established procedures (process) (Bearman et al., 2010), decreased delivery performance (temporal) (Castaldo, 2010), and increased rates of component defects (technical) (Castaldo, 2010). Intrusions further complicate this effect by fragmenting the scarce cognitive effort of group members.

Proposition 1b: IT intrusions increase cognitive workload among software teams, which aggregates to the group-level because of sharing dependencies. This fragments effort and diminishes group coordination effectiveness (technical, temporal, and process).

Time Constraints (Flow Dependencies)

Intrusions create task resumption and completion delays at the individual level (McFarlane, 2002, Speier et al., 1997, e.g., Iqbal and Horvitz, 2007). Such delays can accumulate across the group's lifecycle activities and push the whole project behind, especially when some delayed activities reside on the project's critical path (Brooks, 1979). This can result from the sequential dependence between activities (Reichelt and Lyneis, 1999), or from interdependent modules calling upon each other (Williams, 1999). Trying to offset these slippages by throwing more resources at the tasks is difficult because of the added communication and coordination costs (Brooks, 1979), in addition to introducing quality and productivity problems (Reichelt and Lyneis, 1999).

Proposition 1c: IT intrusions increase time delays among software teams, which ripples to the group-level because of flow dependencies. This diminishes group coordination effectiveness (temporal).

Work Product Error Constraints (Flow Dependencies)

Because they trigger constant attention switching, intrusions incur cognitive costs and increase errors (e.g., Cutrell et al., 2001, Speier et al., 1997, Burmistrov and Leonova, 2003), especially in concentration-intensive environments such as software development (e.g., Smallwood et al., 2004, Robertson et al., 2004, Ko et al., 2006). Due to flow dependencies (Crowston, 1997), work product errors ripple through the entire lifecycle and affect the group's overall technical coordination effectiveness. Wohlin & Koemer (1990) showed that a single undetected error can cause four errors in the subsequent phase and up to 250 errors four phases after the phase where the error was introduced. Others found similar effects of errors propagating downstream, across modules, and across projects (Powell, 2001, Li, 2010).

Proposition 1d: IT intrusions increase work product errors among software teams, which ripples to the group-level because of flow dependencies. This diminishes group coordination effectiveness (technical).

Mitigating Effects of Task Organization Coordination

Two coordination mechanisms – based on manipulating resources or tasks – can be used to effectively cope with the disruptive effects of intrusions: (1) role switching (manipulating people resources), and (2) temporal task management (manipulating tasks and time resources).

Role Switching

Role switching relies on sharing dependencies (Crowston, 1997, Crowston, 1991). Software teams implement it to streamline their tasks and account for process disruptions (Strode et al., 2012). They can replace their interrupted teammates and generally compensate for the time and effort diverted into interruptive activities. (Faraj and Xiao, 2006, Chong et al., 2011, Strode et al., 2012).

Proposition 2a: Structural disruptions triggered by IT intrusions under sharing dependencies lead to the use of role switching as a task organization coordination mode.

Role switching can affect group coordination effectiveness by overcoming some constraints placed by IT intrusions on task execution. First, role switching can ease perceived time pressures by introducing additional members (or existing members with shared skills) who replace their interrupted teammates, keep the task activities on track, and restore attention on coordination activities (Ren et al., 2008). Second, it offsets effort fragmentation and smoothens effort allocation among the interrupted group members, such as to maintain a balanced attention between taskwork and coordination activities (Galbraith, 1973, Thompson, 1967, Bourgeois and Singh, 1983). Third, groups employing role switching can better verify one another's output to ensure conformity before passing it over to the next task (Crowston, 1997), which limits error propagation (Chong and Siino, 2006). Finally, role switching can limit the delays induced by intrusions through reallocating free actors or assigning more actors to the task; e.g., pairing programmers (Chong and Siino, 2006).

Proposition 2b: Role switching helps overcome the additional constraints placed by intrusions on task execution (time pressures, cognitive workload, work product errors, task performance delays) by allowing software group members to compensate for each other in response to IT intrusions. This will result in a negative moderation effect on group coordination effectiveness.

Temporal Coordination

Temporal coordination is the process of "complex matching of bundles of activities to particular periods of time" (McGrath, 1991, p. 163). It contains activities such as scheduling and deadlines, sequencing, prioritization, and synchronization (McGrath, 1991). Because of flow interdependencies between group member activities, temporal coordination can be used in response to disruptive situations to streamline tasks and put them back on track (McGrath, 1991, Waller, 1999, Espinosa et al., 2007).

Proposition 3a: Structural disruptions triggered by IT intrusions under flow dependencies lead to the use of temporal coordination as a task organization coordination mode.

Temporal coordination enhances coordination effectiveness. First, setting clear schedules and activity deadlines may act as a common leverage that reminds group members to return quickly to their primary activities (Kelly and McGrath, 1985, Gersick, 1988) and break out of the "chain of diversions" that is typically elicited by intrusions (Iqbal and Horvitz, 2007). Second, group members may cope with intrusions by sequencing and prioritizing their activities (Crowston, 1997, Malone and Crowston, 1994). Third, synchronizing activities by aligning the pace of effort across group members (McGrath, 1991) mitigates the temporal intrusion effects.

Proposition 3b: Temporal coordination (scheduling and deadlines, sequencing and prioritizing, synchronizing) has a beneficial direct impact on group coordination effectiveness (temporal) because it allows software teams to put their interrupted activities back on track.

IT Interventions & Group Coordination Effectiveness

An IT intervention (e.g., email about a software bug) produces a perceived discrepancy between actual and expected task performance (Jett and George, 2003). It redirects attention to the discrepancy source, and triggers a mindful information processing mode through which group members heedfully relate to their and their team members' actions (Louis and Sutton, 1991, Jett and George, 2003, Zellmer-Bruhn, 2003). Because of interdependencies, the mindful group members facing an intervention coordinate their efforts organically via a group problem-solving coordination mechanism (Grant, 1996, Malone and Crowston, 1994) in order to resolve the discrepancy (Okhuysen and Eisenhardt, 2002, Zellmer-Bruhn, 2003). Group members call upon each other to discuss the source of discrepancy, share knowledge about the problem scope (e.g., what other parts in the system are affected by it), ask questions, solicit opinions, summarize standpoints, vote on important issues, and orchestrate a collective approach to solve the problem (e.g., Boos et al., 2011). For example, discovered software bugs were shown to elicit group problem-solving coordination (Espinosa et al., 2004), especially for the high-priority or "showstopper bugs" that trigger a social process in which software team members use the tools to collectively conduct a "running dialog on the bug" (Bertram et al., 2010, p. 5) and figure out ways to resolve them.

Proposition 4a: IT interventions under interdependencies create a window of opportunity for members to collectively discuss and address task discrepancies and issues. This is expected to trigger a group problem-solving coordination mode.

Group problem-solving coordination enhances coordination effectiveness by providing a large capacity of information processing and a platform for coordinating the group's expertise to resolve discovered problems effectively (Nidumolu, 1995, Okhuysen and Eisenhardt, 2002). For example, intense discussions between software analysts and users during requirement analysis should allow the group to better identify the source of problems in capturing requirements and to specify functionalities that will be better aligned with each other (technical coordination effectiveness), and with user needs (process coordination effectiveness). Support for this relationship can be found in the literatures on knowledge integration (Grant, 1996, Okhuysen and Eisenhardt, 2002) and software development (Espinosa et al. 2004). This effect is not expected to hold for the temporal dimensions because while groups can work more productively together when in a problem-solving mode (Teasley et al., 2002), there are also process losses that diminish the expected efficiency gains (Pinsonneault et al., 1999).

Proposition 4b: Group problem-solving coordination is expected to enhance group coordination effectiveness (technical and process dimensions), because it helps integrate the knowledge resources of software teams and aligns their efforts around an established process to deal with the discovered discrepancies,

DISCUSSION AND IMPLICATIONS

This paper sheds light on the multilevel influence of IT interruptions on coordination effectiveness of software teams. Drawing on coordination theory, we argued that different IT interruption types create different coordination issues and leverage different coordination mechanisms that shape group coordination outcomes.

One key implication is to move beyond an individual-level focus of IT interruptions and to recognize that the effects of IT-based interruptions can be experienced individually yet spill over to the group level. This research also contributes to the IT impacts literature by highlighting the unintended effects of IT use, which remains an under-researched phenomenon (Orlikowski, 1996). An important practical implication is that managers can leverage different coordination mechanisms to mitigate the negative effects and reinforce the positive effects of different types of interruptions.

Future research can extend our framework by considering implicit coordination mechanisms, which are becoming recognized as an integral aspect of group performance (Espinosa et al., 2004, Rico et al., 2008). Another avenue would be to focus on subgroups to directly tackle the disparate interpretations of interruptions depending on the task roles of the interruption targets (cf. Carton and Cummings, 2012). Finally, empirical testing of the propositions developed in this paper is warranted to draw more valid conclusions.

CONCLUSION

IT interruptions elicit divergent, multilevel effects that extend beyond the individual being interrupted. To better understand the disparate effects of different interruption types, the notion of interruptions needs to be extended to the group context by looking at what dependencies exist, how these are affected by different interruption types, and how they are managed. Our study represents a step toward such extension of individual-level interruptions research to the group-level in order to explain group coordination outcomes. It is our hope that our proposed model is empirically tested and further extended.

ABBREVIATED REFERENCES

1. Chong, J. and Siino, R. (2006) Interruptions on software teams: a comparison of paired and solo programmers, *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*, Banff, Alberta, Canada.
2. Crowston, K. (1997) A Coordination Theory Approach to Organizational Process Design, *Organization Science*, 8**,** 2, 157-175.
3. Espinosa, J. A., Lerch, F. J. and Kraut, R. E. (2004) In *Team cognition: Understanding the factors that drive process and performance*(Eds, Salas, E. and Fiore, S. M.) American Psychological Association, Washington, DC, US, pp. 107-129.
4. Jett, Q. R. and George, J. M. (2003) Work Interrupted: A Closer Look at the Role of Interruptions in Organizational Life, *Academy of Management Review*, 28**,** 3, 494-507.
5. Kozlowski, S. W. J. and Klein, K. J. (2000) In *Multilevel Theory, Research, and Methods in Organizations*(Eds, Klein, K. J. and Kozlowski, S. W. J.) Jossey-Bass Inc., San Francisco, CA, pp. 3-90.
6. McGrath, J. E. (1991) Time, Interaction, and Performance (TIP): A Theory of Groups, *Small Group Research*, 22**,** 2, 147-174.
7. Okhuysen, G. A. and Eisenhardt, K. M. (2002) Integrating knowledge in groups: How formal interventions enable flexibility, *Organization Science*, 13**,** 4, 370-386.