MEDINFO 2004
M. Fieschi et al. (Eds)
Amsterdam: IOS Press

A Framework of Interruptions in Distributed Team Environments

Jamie A. Rukab[a], Kathy A. Johnson-Throop[a,b], Jane Malin[b], Jiajie Zhang[a]

[a] *School of Health Information Sciences, The University of Texas Health Science Center at Houston, TX, USA;*
[b] *NASA Johnson Space Center, Houston, TX, USA*

Abstract

In this study, we developed a framework for the study of interruptions in distributed team environments from the perspectives of distributed cognition and activity theory. The core of this framework is the Action Coding System (ACS) that provides a language for the description, representation, categorization, and analysis of interruptions at the level of activities that are meaningful for team performance. We demonstrated the practical utility and theoretical significance of ACS in its application to a real-world, complex, dynamic, and mission critical environment - the Biomedical Engineer (BME) console in the Mission Control Center at NASA's Johnson Space Center. We discussed the potentials of our framework for the design of interruption management systems that could potentially eliminate some interruptions by information redesign, delegate others to autonomous agents, and help human agents to better manage the rest.

Keywords:

Interruption, team, activity theory, case study, distributed cognition, distributed information system, ethnographic study.

Introduction

Interruptions disrupt people's ongoing activities. As a result, they can not only decrease performance but can also cause human errors that sometimes lead to catastrophic events. The negative effects of interruptions are well documented: increasing performance time [1,2,3,4], disrupting emotional state [1,5], increasing perceived workload [6] and generating errors [3,4,7]. The effects of interruptions are especially significant in complex and dynamic domains such as the aviation, power plant and healthcare industries. In aviation, 50% of human errors were due to interruptions [8]; in nuclear power plants, interruptions accounted for 15% of incidents leading to shutdown [9]; and in healthcare, interruptions were highly frequent events (e.g., ten interruptions per hour for emergency physicians) [10] that increased clinicians' cognitive work load and which may contribute to medical errors [11].

Interruptions occur at the level of human activities. Despite the large number of empirical findings, theoretical studies, and application products about interruptions, systematic research frameworks of interruptions at the level of human activities have not been developed.

n addition, previous studies have focused on interruptions at the individual level, not at the system level where a team of individuals interact with each other and with system artifacts and artificial agents. Furthermore, there is hardly any research on study methods for interruptions in real-life operational settings (Zijlstra, Roe, Leonara, & Krediet, 1999).

The purpose of the current study is to develop and test the Activity Coding System (ACS) for interruptions in distributed team environments at the activity level in a live work environment. ACS is based on two theoretical foundations: distributed cognition and activity theory (AT). Distributed cognition is valuable in understanding the phenomena from the system level and concerns the dynamics of an activity or behavior specific to cooperative team work environments involving multiple human and artificial agents [12,13,14,15]. AT is useful for capturing the context in which interruptions occur at the activity level since team interruptions are essentially an effect of human activity [16]. With the use of the framework, practical and theoretical aspects of interruptions can be addressed. A few of these aspects include: How many work interruptions occur? Who gets interrupted? What proportion of time do they consume? How are they handled? How disruptive are interruptions to the task being carried out prior to the interruptions?

After developing the ACS, we will describe its application in a real-world, complex, dynamic, and mission-critical environment: the Biomedical Engineer (BME) console at the Mission Control Center at NASA's Johnson Space Center. Preliminary ethnographic observations revealed that interruptions between team members are frequent events for collaborative tasks within the BME environment. In the concluding section, we will discuss the implications of the ACS as a general framework for the study of human activities, workflow, and interruptions.

The Activity Coding System (ACS) For Interruptions

In this section, we will first identify the properties of interruptions that are (1) meaningful in the languages of distributed cognition and AT, (2) measurable, and (3) significant for the management of interruptions. Then we will develop the Activity Coding System (ACS) for addressing interruptions in distributed team environments at the activity level. Finally, we will describe the procedure of using ACS for data collection and analysis.

Properties of Interruptions

In the sequence of an interruption, some interrupting event causes the individual to suspend the current task (original task) and switch to a new task (interrupting task). For the purpose of this study, an interrupting task involves some sign of environmental change (a stimulus) that signals the switch or change in task. After handling and completing the interrupting task, an individual may switch tasks again to resume his/her original task prior to the interruption after some measurable lapse of time. An individual may also choose to ignore or resist an interruption and continue his/her original task. Furthermore, if they choose to handle the interruption, they still may choose to not resume their original task immediately after handling the interrupting task. They may decide to move on to another task that has higher priority or needs immediate attention. Or they may simply forget to return to their original task.

The following properties of interruptions can be measured or identified: (i) overall time parameters; (ii) types of interrupting stimuli; (iii) interrupted and interrupting tasks; (iv) prioritization among specific tasks; (v) interruption handling strategies; (vi) frequency of interruptions per individual; and (vii) other relevant characteristics within the scope of distributed cognition and AT. In order to obtain all these measurements, the properties of interruptions must initially be captured and represented through a common-language description specific to interruptions. This is the ACS we will describe in the next sub-section.

The Activity Coding System (ACS)

The ACS in the current study is tailored to measuring components of activities specifically related to interruptions. Every interrupting activity consists of the various components that make up an activity such as the subject that carries out the activity, the object or task that gives the activity its direction, and the actions that are the processes that are taken to accomplish the task. The system must address the "who, what, where, when, and how" questions about the different activities associated with interruptions within the system. For example, relevant questions may be who accomplishes the task, what is the task being disrupted and handled, where are people working on their activities, when are tasks being achieved and how are these interrupted tasks and activities handled or managed. These elements of the current coding system provide a vocabulary for describing all the properties necessary to understand interruptions and their effects on work flow and productivity. The basic utility of using this coding system is to empower and speed up analysis of information that a researcher may need in order to understand people's activities and activities related to interruptions.

The Activity Coding System (ACS) is specific to an individual interruption instance consisting of the tasks (objects), actions, and other components of an activity. It is general enough so that it is not content specific, but can be applied to any domain in which the codes can be developed inductively. The first column has a short descriptive label for each individual code (Table 1). The second column shows the actual codes for components of an interruption instance and the third lists the code definitions. The codes are ordered in the chronological order of an interruption.

Time is coded with each instance and measures the length of an interruption instance, which is from the time the subject reacts to a stimulus until the beginning of the post-interrupting activity (Task-P_N), where Task-P_N is the resumption of the original activity (Task-O).

Table 1: ACS for an individual interrupting activity

Code label	Code	Code definition	BME example
Rec	Receiver of interruption	Person or the individual who is interrupted.	BME1
Task-O	Original task	The interrupted task, otherwise known as the objective of the original or prior activity.	Log
Action-O	Original action	Original action that the subject is engaged in at the time of the interrupt.	Typing
Action-I	Interrupting action	Modality of interruption.	Telephone
Task-I	Interrupting task	Task that breaks the subject's attention on the original task.	Determine a medically-related event (status request)
Task-P_N	Post-interrupting task	Tasks following the interrupting task. N designates the numeric sequence of this task after the interruption.	Monitor a medically-related event
Action-P	Post-interrupting action	Action that the subject engages in after the interrupting action.	Listening

Procedure of data collection and analysis

Data collection is conducted during actual working hours by videotaping the work domain of interest. The focus is on naturally occurring events within an operational setting so that the phenomenon of interruptions is better understood in that particular domain. Data analysis consists of three steps: (1) encoding the data (2) displaying the encoded data and (3) extracting results to draw conclusions.

Case Study: NASA Biomedical Engineer Console

The domain

We have applied ACS to the task domain of the Biomedical Engineer Console, which is a part of the Flight Surgeon discipline at the Mission Control Center at NASA's Johnson Space Center. In this domain, Biomedical Engineers (BMEs) are responsible for providing the technical and operational support for all medically-related operations so as to ensure the health and safety of all mission crewmembers. Providing medical operations support for human space flight requires complex interactions and exchanges of information both synchronously and asynchronously across space and time [17]. It involves acquisition, transmission, distribution, integration, retrieval and archiving of significant amounts of data stored within disparate systems in a variety of formats using timely, reliable and secure measures.

Methods

Participants. Console BME (BME1) and oncoming BME (BME2) from the Mission Control Center at NASA's Johnson Space Center participated in the observational study. Informed consent was obtained from each of the research participants. The participants were advised that an observation was to be conducted unobtrusively and that it would not interfere with their job performance. The participants did not receive payment for their participation because they were civil servants.

Materials and Apparatus. A video camera was used to record the observation of the two BMEs within the BME console. Also, a wireless microphone was placed on the tabletop so that all conversations were audible at all times.

Design and Procedure. The BME console was selected for the observation because it is a distributed team environment consisting of multiple actors who cooperate and collaborate with each other. In addition to interaction between the two BMEs, the BMEs respond to and collaborate with multiple disciplines such as Environmental Control & Life Support, Electrical Power, Crew Health Care, etc. via voice loop, e-mail, and face-to-face interactions. The BMEs were observed and videotaped for two hours during the period of the BME shift handover. This particular observation period was selected due to it its time-critical, data intensive nature in which the two BMEs interact with each other to complete the handover procedure and at the same time perform routine tasks.

The ACS framework was used to guide the identification of interruptions instances from the video observation, coding of the data, and subsequent data analysis. An interruption event was defined as consisting of the following combined criteria: a stimulus and an associated task-switch that is subsequently performed by an individual.

Table 1 shows an example of how an interruption from the observation was coded according to the ACS. While BME1 (Rec) was logging routine events (Task-O) by typing on his computer (Action-O), she was interrupted by a telephone call (Action-I) for a status request on a medically-related event (Task-I). After taking the phone call, she monitored an ongoing event (Task-P_1) by listening (Action-P) then resumed her original task of logging (Task-P_2). Each interruption event was classified by the modality in which the interruption was delivered and then coded according to the ACS. In this specific case, the different modalities of communication for interruptions included face-to-face, telephone, or the DVIS voice loop. Preliminary user and hierarchical task analyses of the BME console environment were used to identify the different types of tasks performed in the domain. We measured up to Task-P_2 in sequence of tasks after the interruption unless the subject resumed to task immediately after the interruption. MacShapa, a Macintosh-based qualitative data analysis environment for sequential data was used to implement the coding scheme and subsequently carry out some of data analyses [18].

Results

Interruption Characteristics

Results extracted from the data analysis showed that during the two-hour handover period, the two BMEs were interrupted a total of 32 times (Table 2 displays the summary of interruption data for both BMEs). The average duration of an interrupting task was approximately 2 minutes and 28 seconds. Interrupting tasks consumed 46% of BME1's time and 21% of BME2's time during the two hour period.

Table 2: Summary of interruption data for the two BMEs in the study

	BME1	BME2	Total
Number of interruptions	15	17	32
By modality of interruption			
Face-to-face	2	0	2
Telephone	8	8	16
DVIS voice loop	5	9	14
Total interruption time (min)	55:18	25:35	80:53
By modality of interruption			
Face-to-face	4:49	0:00	4:49
Phone	36:19	9:54	46:13
DVIS voice loop	14:10	15:41	29:51

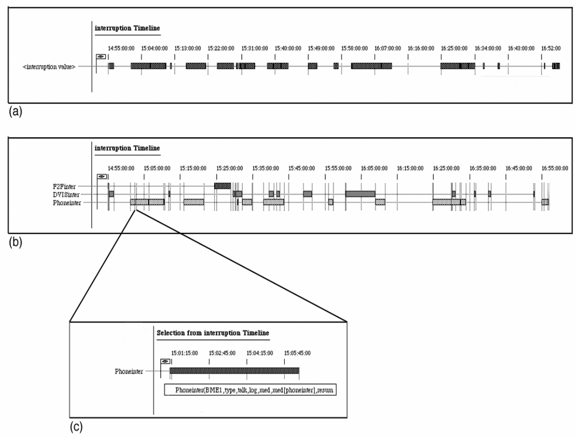

Figure 1 - (a)The timeline shows amount of time all interrupting tasks consumed for both BMEs during the observation. (b) Interrupting activity timeline for both BME 1 and BME 2 by the three modalities in which interruptions were delivered: F2Finter represents face-to-face interruptions, DVISinter represents voice loop interruptions, and Phoneinter refers to telephone interruptions. (c) Details of an interrupting activity.

The timeline in Figure 1a shows the amount of time that all interrupting tasks consumed for both BMEs during the two hour observation. According to this timeline, interruptions appear to be evenly scattered over the two hours. Figure 1b shows all interrupting activities for both BME 1 and BME 2 by the different interrupting modalities. Figure 1c shows the details of a specific interrupting event.

The timeline in Figure 1a shows the amount of time that all interrupting tasks consumed for both BMEs during the two hour observation. According to this timeline, interruptions appear to be evenly scattered over the two hours. Figure 1b shows all interrupting activities for both BME 1 and BME 2 by the different interrupting modalities. Figure 1c shows the details of a specific interrupting event

Figure 2 shows the distribution of interruptions across three modalities (Action-I) observed in the video. On average, BMEs spent more time handling interruptions on the phone than DVIS voice loop or face-to-face (38% of time spent on phone, 25% on the voice loop, and 4.2 % on face-to-face conversation).

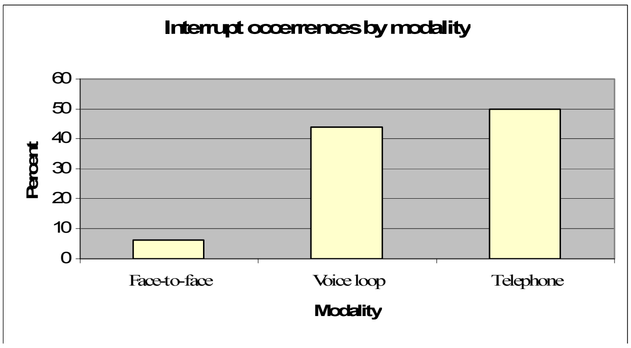

Figure 2 - Interrupting activities classified by three modalities in which they were delivered to the BME console.

Types of interrupting tasks

The types of interrupting tasks that took place during the observation were also measured (Figure 3). Non-work related interruptions consisted of requests for someone's pager number or other general contact information.

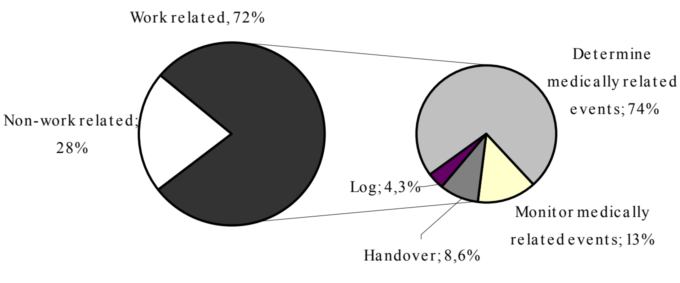

Figure 3 - Types of interrupting tasks.

Post-interruption activities

The data analysis also showed whether or not the subjects returned to their original task after the interruption took place. Figure 4 shows that BMEs resumed their original tasks right after the interruption 66% of the time (Task-P_1 = Task-O). 16 % of the time, the recipient of the interruption failed to return to their original activity within the extent of the observation (Task-O Task-P_1 or Task-P_2).

Discussion and Conclusion

In this paper we developed the Activity Coding System (ACS) for the study of interruptions. The ACS, when it is applied to a specific domain, specifically and systematically measures interruption-related activities and behaviors and provides a framework for understanding the nature and the dynamics of interruptions in collaborative work environments. The results of applying this methodology (ACS) to an extended observation in the BME domain indicate that interruptions can be highly frequent, unpredictable events which are sometimes redundant and non-urgent, consuming much of the BMEs' already full workload. The results indicate the extent of the disruptive effects that interruptions can cause to the work flow of BMEs during their shift. Such data can help identify opportunities to design systems and processes to eliminate or assist in managing interruptions. By understanding the phenomenon of interruptions within the BME environment, insight can be gained for the design of groupware applications to support collaborative activity. A preliminary analysis of the interrupting tasks indicated that 19% could be potentially eliminated completely by information redesign, 47% could be potentially delegated to autonomous agents, and 34% have to be handled by BMEs but could be potentially assisted by human-centered intelligent tools.

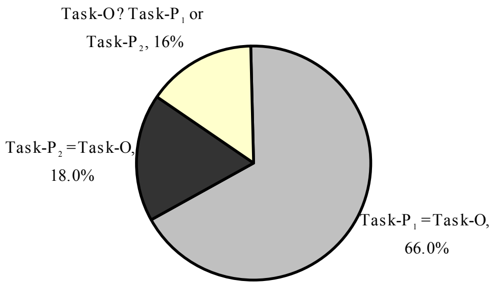

Figure 4 - Percent at which BMEs return to original task (Task-O) at Task-P_1, Task-P_2, or never return to original task within the extent of observation.

This level of analysis is the appropriate level of granularity and is unique in capturing crucial information for guiding system design. It provides answers to questions such as who gets interrupted, how many interruptions occur, what are the tasks being achieved or interrupted, and in what mode and how are interruptions handled in a team environment. Also ACS provides a vocabulary for describing not only the occurrence of an interruption but the sequences of activities that take place before, during and after the interruption. Understanding the dynamics of such group activity is important to designing computer systems supporting cooperative work. Finally, our framework is general enough to be practically applied to other interrupt-laden team environments such as emergency departments or other healthcare domains. The understanding of interruption dynamics can be used for designing more effective clinical information systems that may help reduce medical errors and improve quality and safety of healthcare delivery.

The ACS can be further improved by capturing such detail in team and collaborative environments. For example, our video observations demonstrated that the intended recipient of an interruption was not always the actual recipient. Instead, in our team environment, one team member would handle an interruption for her colleague on certain occasions when she was already busy with a specific task. Further research could investigate how best to capture such contextualized data from video observations. Also multiple different observational instances may be more effective than a single extended observation, as carried in this study, as it may provide more generalizable data for the particular domain.

Acknowledgement

This study was supported by a grant from the Human-Centered Computing and Intelligent Systems Program, National Aeronautics and Space Administration, Grant No. NCC 2-2-1234.

References

[1] Bailey, B. P., Konstan, J. A., & Carlis, J. V. (2000). *Measuring the Effects of Interruptions on Task Performance in the User Interface.* Paper presented at the IEEE Conference on Systems, Man and Cybernetics, Nashville, TN.

[2] Burmistrov, I., & Leonova, A. (1996). *Interruption in the Computer Aided Office Work: Implications to User Interface Design.* Paper presented at the Human-Computer Interaction: Human Aspects of Business Computing. Proceedings of EWHCI'96, Moscow.

[3] Gillie, T., & Broadbent, D. (1989). What makes interruptions disruptive? A study of length, similarity, and complexity. *Psychological Research, 50*(4), 243-250.

[4] Kreifeldt, J. G., & McCarthy, M. E. (1981). *Interruption as a test of the user-computer interface.* Paper presented at the 17th Annual Conference on Manual Control, California Institute of Technology.

[5] Zijlstra, F. R. H., Roe, R. A., Leonara, A. B., & Krediet, I. (1999). Temporal factors in mental work: Effects of interrupted activities. *Journal of Occupational and Organizational Psychology, 72*, 163-185.

[6] Kirmeyer, S. L. (1988). Coping with Competing Demands: Interruption and Type A Pattern. *Journal of Applied Psychology, 73*(4), 621-629

[7] McFarlane, D. C., & Latorella, K. A. (2002). The Scope and Importance of Human Interruption in HCI Design. *Human-Computer Interaction, 17*(3), 1-62.

[8] Dismukes, K., Young, G., & Sumwalt, R. (1998). Cockpit Interruptions and Distractions: Effective Management Requires a Careful Balancing Act. *ASRS Directline*(10), 4-9.

[9] Cellier, J.-M., & Eyrolle, H. (1992). Inference between switched tasks. *Ergonomics, 35*(1), 25-36.

[10]Brixey, J., J.Zhang, & Johnson, T. (2003). *Analysis of Interruptions in Healthcare.* Paper presented at the Proceedings of AMIA, in review.

[11]Coiera, E. W., Jayasuriya, R. A., Hardy, J., Bannan, A., & Thorpe, M. E. C. (2002). Communication loads on clinical staff in the emergency department. *MJA, 176*(9), 415-418.

[12]Hutchins, E. (1995a). *Cognition in the Wild*. Cambridge, MA: The MIT Press.

[13]Hutchins, E. (1995b). How a Cockpit Remembers Its Speeds. Cognitive Science, 19, 265-288.

[14]Zhang, J. (1998). A distributed representation approach to group problem solving. *Journal of American Society of Information Science, 49*(9), 801-809.

[15]Zhang, J., & Norman, D. A. (1994). Representations in distributed cognitive tasks. *Cognitive Science, 18*(1), 87-122.

[16]Leont'ev, A. N. (1978). *Activity, Consciousness and Personality*. Englewood Cliffs, NJ: Prentice-Hall.

[17] Rinkus, S., Johnson-Throop, K., & Zhang, J. (2003). *Designing a Knowledge Management System for Distributed Activities: A Human Centered Approach.* Paper presented at the Proceedings of AMIA 2003.

[18]Sanderson, P., Scott, J., Johnston, T., Mainzer, J., Watanabe, L., & James, J. (1994). MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA). *International Journal of Human-Computer Studies, 41*, 633-681.

Address for correspondence

Jamie A. Rukab
School of Health Information Sciences
The University of Texas Health Science Center at Houston
7000 Fannin st. 600
Houston TX, 77030
Email 1jrukab@hotmail.com;
Email 2: jamie.rukab@lmco.com
Phone: 281-723-2990