A. Dix, C. Chisalita and G. van der Veer (2003). Moments of Significance - the meanings of event: enablement, initiation, completion. In Tamodia 2003, part of Universal Access in HCI, Volume 4 of Proceedings of HCI International 2003. C. Stephandis (ed.). Lawrence Erlbaum Associates, 2003. pp. 1519-1523
http://www.hcibook.com/alan/papers/moments2003

Moments of Significance – the meanings of event: enablement, initiation, completion

Alan Dix[1], *Cristina Chisalita*[2], *Gerrit van der Veer*[2]

[1] Lancaster University
Computing Department
Lancaster, LA1 4YR
UK

alan@hcibook.com

[2] Vrije Universiteit,
Department of Computer Science
de Boelelaan 1081 A
1081 HV Amsterdam
The Netherlands

cristina@cs.vu.nl, gerrit@acm.org

http://www.hcibook.com/alan/papers/moments2003/

Abstract

This paper examines the nature of events within task analysis. It draws on three roots: Groupware Task Analysis, Trigger Analysis and Status-Event Analysis. Each of these considers events and the way these enable or initiate activities, but they have subtly different understandings of these. Analysing both these distinctions and also the common features between them has lead to a deeper theoretical understanding of the ontological nature of events (and their relationship to activities/tasks), the attributes of these events (e.g. whether they originate internal/external to the task context), and the relationship between the events and tasks (e.g. different kinds of enablement/triggering).

1 Introduction

At the first Tamodia, Dix discussed various features that were important for task analysis to capture the rich ecology of work environments (Dix, 2002). One of these issues was the importance of events and triggers, which has been embodied in a task analysis 'add on' technique called Trigger Analysis. In fact Groupware Task Analysis already embodies a rich ontology including events and triggers, but during subsequent discussions between the authors it became clear that these notions have subtly different meanings in the two analysis techniques

In the rest of this paper we will examine the concepts of events and triggers as currently embodied in Trigger Analysis (TgA), Groupware Task Analysis (GTA), and Status–Event Analysis (S–E), which is one of the theoretical foundations of TgA. We will show how the analysis of their different treatments of events and triggers has lead to a deeper theoretical understanding of the ontology and attributes of events and of the way events act as triggers to enable, and initiate tasks.

2 Background

2.1 Trigger Analysis

Trigger analysis (Dix, Ramduny-Ellis & Wilkinson, 2003) is not a standalone method, but a series of concepts and techniques for using with other forms of task analysis or workflow analysis. Typically in a sequence of tasks there is an implicit assumption that the next subtask will happen once the previous one has completed. For example, in figure 1 we have three tasks “get post from pigeon hole”, “bring post to desk” and “open post”. However, you may not immediately open the post after putting it down on your desk, perhaps your morning pattern is to check email first. You cannot open the post before you have completed previous subtasks, but the actual commencement of the "open post" subtask may happen some time later. Trigger analysis focuses on what makes things happen when they happen. In this example, it may be you always open your post at coffee time. In figure 1 the 'lightening' symbols show the events that trigger each sub-task. In addition, trigger analysis asks the analyst to consider the placeholders that let the user know how far the process has got through the task sequence. For example, the fact that you are holding post in your hand implicitly tells you that you have already collected your post and need to take it to your desk.

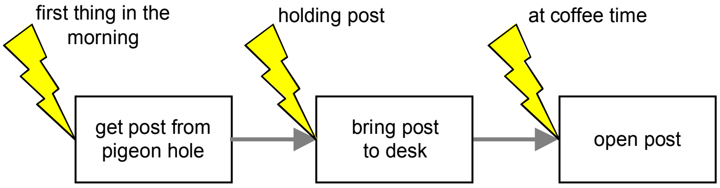

Figure 1. Triggers for activities

2.2 Groupware Task Analysis

GroupWare Task Analysis (GTA) is a broad task-analysis conceptual framework that is based on the integration of different approaches (van der Veer & van Welie, 2000). GTA is especially useful in complex design situations, which involves collaboration and communication. In the next paragraphs we briefly expose GTA and the elicitation techniques that can be used to extract information about the task world:

2.2.1 GTA conceptual framework

In order to model the task knowledge, GTA proposes a conceptual framework that is based on the integration of views from HCI and CSCW. What is important, especially in a complex situation, is to consider all the aspects related with performance of the tasks. Therefore, GTA describes the task world focusing on three different viewpoints:

1. *The agents and the roles.* Specifying the active entities in the task world: users and other stakeholders, systems or organizations and their role, as well as the organization of work (structure of actors, roles, and allocation rules).
2. *The work.* Specifying the decomposition of tasks and strategies (different informal ways, partly situated, of performing the tasks), goals and events (which trigger the tasks).
3. *The situation.* Specifying the objects used in the task world as well as their structure, the history of past relevant events and the whole social and physical work environment.

These three viewpoints form the basis for GTA ontology (figure 2).

Note how the ontology includes events that trigger tasks. In GTA an event is defined as:

> A change in the state of the task world at a point in time. The change may reflect changes of attribute values of concepts *internal* to the task world as modelled such as Objects, Task, Agent, Role or could reflect changes of *external* concepts like the weather or electricity supply (van Welie, 2001).

Currently GTA does not specify how the events are created or by whom.

2.2.2 GTA knowledge elicitation techniques

GTA intends to describe the task world in all its complexity and to generate a task model as complete as possible. This conceptual framework integrates analytical methods from HCI (individual oriented) with ethnographical methods from CSCW (group oriented). Consequently, GTA considers four sources of knowledge about the task that have to be considered in task analysis: implicit, explicit, individual and group knowledge. For each type of knowledge several specific elicitation techniques are used: e.g. for individual-explicit knowledge: interviews, psychological observation; for individual implicit knowledge: hermeneutics; for group-explicit knowledge: document analysis; for group-implicit knowledge: participant observation, interaction analysis.

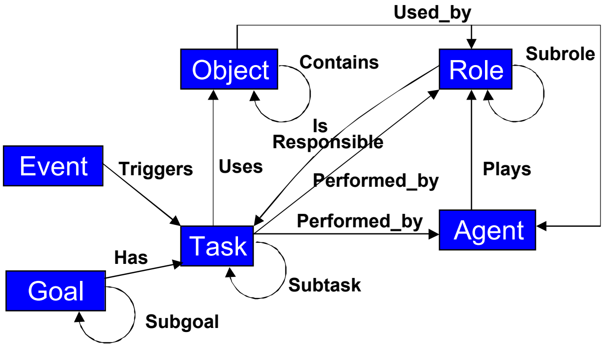

Figure 2. GTA ontology (van Welie 2001)

2.3 Status–Event Analysis

Trigger analysis has its origins in issues of pace and timeliness in interaction and also Status–Event analysis (S–E). In fact, this is not a single analytic method, but a collection of notations and techniques with a common underlying conceptual framework (Dix & Abowd, 1996). The key issue is the distinction between status and event phenomena. Status phenomena are those that always have some measurable value: for example the current weather, the current time, the current display on a computer screen. In contrast event phenomena have only a momentary existence: for example the alarm that wakes you up, the pressing of a button, the beeping of a computer.

Although this is a simple distinction there is a rich set of higher level phenomena associated with it that are common to human–human, human–computer and computer–computer interactions: for example, the distinctions between actual events (e.g. it is 9:30am when you have a meeting) and perceived events (when you belatedly notice!). Most computing formalisms and analyses focus exclusively on events, but in fact status phenomena are crucial for the smooth running of most interactions. A key issue in this paper is that status phenomena interact with event phenomena in that events often cause changes of status and conversely certain status changes can be seen as events. In trigger analysis this is crucial as the environmental cues that form the triggers for many tasks are typically status phenomena ‘left behind’ by previous tasks.

3 Understanding events and triggers

3.1 Events

Although both GTA and TgA (based on S–E) have clearly defined events, the meaning of these is slightly different. In GTA events are associated with changes in the state of some task world entity (object, task, agent, role). These are clearly the important events from the point of view of the task. S–E analysis takes a wider view and regards an event as any momentary phenomena, although these are usually either associated with the causes of change or are the effects of change. Of course, the world is in constant flux and only some changes are interpreted as events – those that have some significance for the tasks. Events are precisely these *moments of significance*.

Looking at GTA events we see that they may have two sources:

- internal – event inside the task world
- external – event outside from task world

An internal event is one caused by some element of the world of objects and agents included within the analysis, and external event is anything else. For example, if we had a library and our analysis may include librarians, library users, books, and the library computer systems. An internal events may be a library user returning a book, or the librarian noticing that there are recently returned books waiting to be put on the shelves; whereas an external event may be a user's dog eating a borrowed book. Note that this definition depends on the scope of our analysis and its associated task world. If we had taken a more restricted scope for our analysis and not included the library users then the user retuning the book would be regarded as an internal event. .

Typically the task world is largely within the target organisation, external events are those emanating from outside the organisation – crossing the boundary. TgA looks at inter- as well as intra-organisational flows of work, so effectively all events become internal events, but one of the reasons for the analysis is the possibility of interruptions – that is external events that disrupt the task sequence. Of course the points at which flows cross the organisational boundaries are crucial for coordination, in particular those points at which task flows start.

3.2 Triggers

In fact, this gets close to the key difference between triggers as used in GTA and TgA. In GTA triggers are things that set a task sequence in motion (e.g. new customer order arriving) or make a substantive change to the task sequence (for example the customer rings to cancel the order).

Triggers in TgA are concerned with the smooth running of tasks and what actually makes subtasks start at the moment they do. These differ in both level :

- major events that shape the task dynamics as a whole
- normal events that simply, keep the task on track

and also in the the way in which an event triggers a task:

- enablement – the event that makes a task necessary
- initiation – the event that makes a task/activity start at the moment it does

Figure 3 shows this schematically. The task may be a high level task with decomposition and has an enabling event and then some time later the initiation (trigger in TgA sense) that starts it going. Also marked is the completion event when the task finishes (this may or may not be marked in any way). These are linked as in simple linear task sequences the completion event of one subtask is implicitly the enablement of the next.

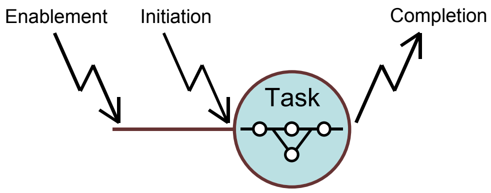

Figure 3. Significant task events

Of course, if the enablement of a task inevitably led to its execution then the initiation event would be uninteresting. However, due to frailty of human memory or interruptions that disrupt otherwise continuous task sequences. Of course, life experience tells us that this is not just the exception and ethnographic studies certainly reveal that interruptions are the norm (Rouncefield, Hughes, Rodden & Viller 1994). Interruptions are themselves another kind of event and it is precisely where interruptions can occur between the enablement and start of a task that initiation events become significant.

4 Conclusions and Future Work

By comparing the notions of event and task from methods that have developed in different contexts for different purposes we have found a far richer understanding than in either method alone. We are very aware that even this analysis is far from complete and we are continuing to study the rich interactions between events and tasks. As this develops we hope that it will feed back into practical task analysis techniques and into our understanding of real organisations and work processes.

5 Acknowledgements

This work was supported by EPSRC under the EQUATOR Interdisciplinary Research Collaboration (www.equator.ac.uk).

6 References

Dix, A. (2002). Managing the Ecology of Interaction. In *Proceedings of Tamodia 2002 - First Intnl Workshop on Task Models and User Interface Design*, Bucharest, Romania, 18-19 July 2002

Dix, A., Ramduny-Ellis, D., & Wilkinson, J. (2003). Trigger Analysis – understanding broken tasks. In D. Diaper & N. Stanton (Eds.), *The Handbook of Task Analysis for Human-Computer Interaction*. Lawrence Erlbaum Associates. (in press)

Dix, A. & Abowd, G. (1996). Modelling status and event behaviour of interactive systems. *Software Engineering Journal*, 11 (6),. 334-346

Rouncefield, M., Hughes, J., Rodden, T., & Viller, S. (1994). Working with "Constant Interruption" CSCW and the Small Office. In *Proceedings of CSCW'94*. Chapel Hill, North Carolina: ACM Press. pp. 275–286.

Van der Veer, G.C. & Van Welie, M. (2000). Task Based GroupWare Design: Putting theory into practice. In *Proceedings of DIS 2000*, New York, United States.

Van Welie, M.(2001). Task-Based User Interface Design. Thesis submitted to the Faculty of Sciences of the Vrije University of Amsterdam