19

Trigger Analysis: Understanding Broken Tasks

Alan Dix
Lancaster University

Devina Ramduny-Ellis
Lancaster University

Julie Wilkinson
University of Huddersfield

Why do things happen when they happen? Trigger analysis exposes the triggers that prompt subtasks to occur in the right order at the right time, and assess whether tasks are robust to interruptions, delays, and shared responsibility. Trigger analysis starts with task decomposition via any suitable method, proceeding to uncover what trigger prompts each subtask. The obvious answer: "because the previous subtask is complete" is often the precondition, not the actual trigger. Previous analysis has uncovered certain primary triggers and potential failure points. The complete analysis produces an ecologically richer picture, with tasks interacting with and prompted by their environment.

19.1 INTRODUCTION

Many task analysis and workflow techniques decompose the overall task into smaller subtasks, processes, or activities. The order of these subtasks is then typically specified or described (e.g., plans in hierarchical task analysis [Shepard, 1995; chap. 3], links between processes in a workflow, and temporal connectives in ConcurTaskTrees [Paternò, 1999; chap. 24]). For short-term tasks performed by one person without interruption, this may be the end of the story:

> *To photocopy a document*: (a) open copier lid; (b) put original on glass; (c) close lid; (d) select number of copies; (e) press copy button; (f) when copying complete, remove copies; (g) remove original.

But what happens if the number of photocopies is large and so you go for a cup of tea between (e) and (f)? Someone comes in and interrupts between (f) and (g)? Instead of a small photocopier, this is a large print machine, and different people are responsible for different stages?

Trigger analysis deals with exactly these issues. Why do things happen when they happen and do they happen at all? By exposing the triggers that prompt activities and subtasks to occur in the right order at the right time, trigger analysis allows us to decide whether tasks are robust to interruptions, delays, and shared responsibility (even across organizational boundaries).

Trigger analysis starts with a task decomposition obtained by any suitable method (and can therefore be used in combination with many task analysis and workflow methods). It then proceeds to uncover what trigger causes each subtask to occur. The initial answer is, "because the previous subtask is complete," but completion of that subtask is often merely a precondition, not the actual trigger.

Previous empirical and theoretical analysis has uncovered a small set of primary triggers, including the simple "previous task complete," timed events ("every hour I check the mail"), and environmental cues ("the document is in the in-tray"). For each class of trigger, there is a set of subsequent questions, for example, "What happens if you are interrupted between tasks?" "How do you know when it is the right time?" "Are there several tasks with the same environmental cues?"

Triggers are what make activities happen when they do. A closely related issue is knowing where in the task sequence you are. Environmental cues often act in both roles, as triggers saying that something should happen and as placeholders saying what should happen. Typically, the complete analysis produces a highly ecologically rich picture, and rather than cognitively driven tasks acting on the environment, we see tasks interacting with and prompted by their environment. Note that trigger analysis is not an alternative task analysis technique or notation but instead an additional concern that should be grafted on to existing analysis methods.

This chapter examines the trigger analysis method in depth, with particular reference to its grounding in empirical work. We start with a brief discussion of the theoretical underpinning for the fieldwork undertaken. We then suggest some reasons why prolonged interactions tend to break down in organizational contexts and provide an explanation of the five trigger types that have emerged during our investigations. We also present a second, related pattern, the 4Rs framework, in detail, as we believe the 4Rs form an important and fundamental unit of work. Finally, we describe a comprehensive application of the 4Rs and the benefits they provided for the analysis of work in a systems development project.

19.2 THEORETICAL BACKGROUND

The roots of trigger analysis lie in two principal theoretical foundations: the study of the pace of interaction (Dix, 1992, 1994) and status–event analysis (Abowd & Dix, 1994; Dix, 1991; Dix, Finlay, Abowd, & Beale, 1998). The primary basis consists of the issues surrounding pace—that is, the rate at which users interact with computer systems, the physical world, and one another. Thinking about pace makes one concentrate on the timescale over which interaction occurs, both the similarities between interactions of widely different pace and also the differences.

19.2.1 Status–Event Analysis

Status–event analysis is a collection of formal and semiformal techniques all focused on the differences between events (things that happen) and status (things that always have a value). Applications of status–event analysis have included auditory interfaces (Brewster, Wright, & Edwards, 1994), formal analysis of shared scrollbars (Abowd and Dix, 1994), and software architectures for distributed agent interfaces (Wood, Dey, & Abowd, 1997).

Status–event analysis allows the distinction between an actual event (some objective thing that occurs) and a perceived event (when an agent, human or machine, notices that an event has occurred). Sometimes this is virtually instantaneous, but more often there is a lag between the two. Many formal and informal analyses of events assume simultaneity between cause and effect. However, accepting that there is often a gap allows us to investigate what actually causes secondary events to occur when they do. The lag between the actual event and the perceived event (e.g., the time it takes to notice that e-mail has arrived) can in turn influence the pace of interaction.

Status–event analysis also looks at the effect of events on agents. An event may simply be intended to inform, but more often an event is intended to initiate actions, which in turn may cause further events. The actions of agents may change the status of the agent or the world, but changes in status are themselves events that may trigger further action.

Furthermore, most notations in computing focus primarily on events with little, if any, description of status phenomena. However, we shall see that environmental cues—things that are around us, such as piles of papers or notice boards—play an essential role in acting as triggers and placeholders.

The analysis of status and events has allowed us to see some common features of human-human, human-computer, and internal computer interactions. For example, it is common to see status mediation whereby one agent communicates an event to another by manipulating a status that will eventually be observed by the second agent. Moreover, polling, the periodic observation of a status phenomenon to detect change, is not just a low-level computational device but something people do as well, as described in more detail in chapter 13. The rich interplay of status and event phenomena is reflected in the ecological perspective that colors the analytic stance of our current study.

Because status–event analysis gives equal weight to status and event phenomena, it is able to address interstitial behavior—the continuous relationships between status phenomena in the interstices between events. This interstitial behavior is often what gives the "feel" to computer systems—you move the mouse and the window is dragged around the screen. Part of the power of the human motor system is its capability for near continuous control and response, as in sports, driving, manipulating objects, and so on, and paradigms such as direct manipulation and virtual reality use this power.

This continuous interaction is important in many nondesktop technologies, including aspects of ubiquitous computing (see Dix, 2002b). However, in this chapter we concentrate more on office style interactions and the discrete activities commonly dealt with in task analysis. Often the lowest level subtask or activity may involve continuous interaction, but here we look principally at the events that drive the transitions between these.

19.2.2 Pace of Interaction

The pace of interaction with other people or with a computer system is the rate at which one sends messages or commands and then receives a response. It varies from tens of milliseconds in a video game to hours or days when interacting by post.

The pace of interaction is influenced by and influences three principal factors (see Fig. 19.1):

1. the intrinsic pace of the channels through which the users communicate
2. the pace of the collaborative tasks
3. the users' own natural pace for different forms of mental activity.

Problems may occur when there is any significant mismatch between any of these and the resulting pace of interaction (Dix, 1992; Dix, Ramduny, & Wilkinson, 1998).

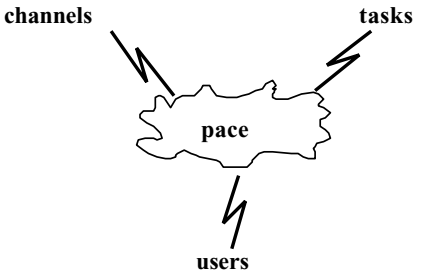

FIG. 19.1. Factors affecting pace.

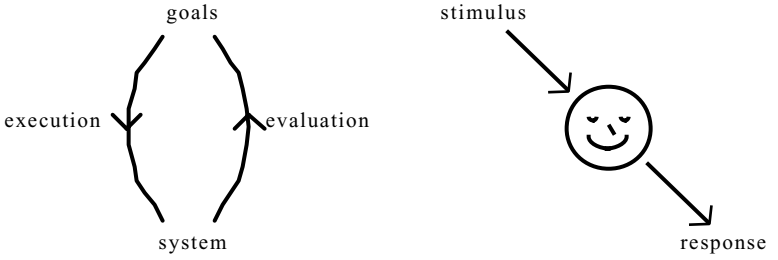

FIG. 19.2. (a) Norman's execution–evaluation cycle and (b) the stimulus–response cycle.

19.2.3 Prolonged Interaction

The study of pace helps us to understand interactivity in a wider context. A system or collaborative process is not interactive because it is fast or has instant feedback. Instead, interactivity is about the appropriate pace of interaction in relation to the task at hand. Indeed, in many collaborative situations the pace of communication may be over days or weeks.

The reason for the prolonged nature of these interactions varies: It may be because of the communication medium (e.g., normal postal delays) or the nature of the task (e.g., a doctor requiring an X-ray has to wait while the patient is wheeled to radiology and back). One of the key points is that models of interaction that concentrate on a tight cycle between action and feedback break down (Dix, 1992). This is typified by Norman's (1986, 1988) execution–evaluation cycle: A user has a goal, formulates actions that will further that goal, executes the actions, and then evaluates the results of those actions against the expected outcome and the goal (see Fig. 19.2a). This model effectively assumes that the results of the user's actions are immediately available. If the delay between executing actions and observing the results is greater than short-term memory times, then the evaluation becomes far more difficult. This problem has been called the "broken loop of interaction."

Another model of interaction, used in industrial settings, is the stimulus–response model (see Fig. 19.2b). Commands and alarms act as stimuli, and the effective worker responds to these in the appropriate manner. However, in a pure form, this model does not allow for any long-term plans or goals on the part of the worker; the worker is treated in a mechanistic manner, as a cog in the machine.

To incorporate both these perspectives, we need to stretch out the interaction and consider the interplay between the user and the environment over a protracted timescale. We use the term *environment* to include interaction with other users, computer systems, or the physical environment. Such interaction is typically of a turn-taking nature: The user acts on the environment, the environment "responds," the user sees the effects and then acts again.

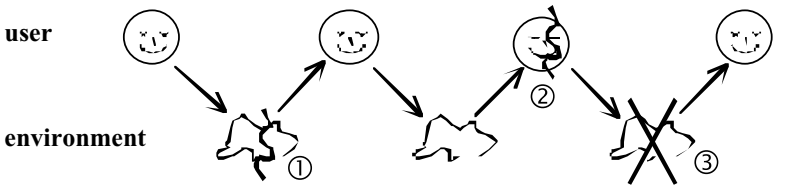

FIG. 19.3. Problems for prolonged interaction.

This process is illustrated in Fig. 19.3. Notice how the Norman loop concentrates on the user–environment–user part of the interaction whereas the stimulus–response model concentrates on the environment–user–environment part. We can see various ways in which long-term interaction affects this picture:

① *Action–effect gap*. The user performs an action, but there is a long delay before the effects of that action occur or become apparent to the user. For example, you send an e-mail and some days later get a reply. The problem here is loss of context: How do you recall the context when you eventually receive the feedback? When the reply comes, you have to remember the reason why the original message was sent and what your expectations of the reply were. The way in which e-mail systems include the sender's message in the reply is an attempt to address this problem. In paper communication, the use of "my ref./your ref." fulfills a similar purpose.

② *Stimulus–response gap*. Something happens to which the user must respond but for some reason cannot do so immediately. For example, someone asks you to do something when you meet in the corridor. The problem here is that you may forget. Hence the need for to-do lists or other forms of reminder (see chap. 14). In the psychological literature, this has been called *prospective memory* (Payne, 1993).

③ *Missing stimulus*. The user performs an action, but something goes wrong and there is never a response. For example, you send someone a letter but never get a reply. For short-term interactions, this is immediately obvious. You are waiting for the response, and when nothing happens, you know something is wrong. However, for long-term interactions, you cannot afford to do nothing for several days waiting for a reply to a letter. In this case you need a reminder that someone else needs to do something—a to-be-done-to list!

Trigger analysis focuses particularly on the second problem, the stimulus–response gap, but all three problems need to be considered when analysing the robustness of a process.

19.3 FUNDAMENTAL CONCEPTS OF TRIGGER ANALYSIS

19.3.1 Triggers: Why Things Happen When They Happen

Workflows and process diagrams decompose processes into smaller activities and then give the order between them. Similarly in HTA, plans give some specification of the order of subtasks, and in CTT, these temporal orders are made more specific using operators derived from LOTOS (Paternò, 1999; chap. 24).

Figure 19.4 shows a simple example, perhaps the normal pattern of activity for an office worker dealing with daily post. Notice the simple dependency: The post must be collected from the pigeonhole before it can be brought to the desk and before it can be opened. However, look again at the activity "open post"—when does it actually happen? The work process says it

FIG. 19.4. Simple work process.

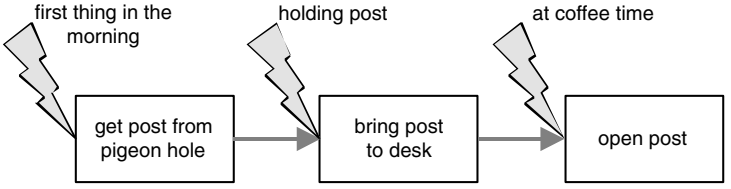

FIG. 19.5. Triggers for activities.

doesn't happen before the "bring post to desk" activity is complete, but does it happen straight away after this or some time later?

In previous work (Dix, Ramduny, & Wilkinson, 1996, 1998), we looked in detail at the triggers that cause activities to happen when they happen. In the case of opening post, this could easily be something like "at coffee time" rather than straight away. In our work, we identified a number of common triggers:

- *immediate:* straight after previous task
- *temporal:* at a particular time or after a particular delay
- *sporadic:* when someone thinks about it
- *external event:* some event occurs such as a phone call or the receipt of a message
- *environmental cue:* something in the environment prompts action.

We will look at a more detailed breakdown later.

Now we can augment the work process with triggers for each activity (Fig. 19.5). Notice how we have examples of several types of trigger, two temporal and one environmental (letters in the office worker's hand prompting her to carry them to her desk).

Triggers are important not only because they help us understand the temporal behavior of the task but because they tell us about potential failure modes. If two environmental triggers are similar, one might do parts of the task out of sequence. If a trigger may not occur or be missed (likely for sporadic triggers), activities may be omitted entirely. Triggers also help us assess the likelihood of problems due to interruptions. For example, immediate "just after" sequences are disrupted badly by interruptions but environmental cues tend to be robust (because they are still there).

Sometimes triggers are seen in the plans of HTAs and sometimes "waiting" subtasks are included for external events, but such cases are exceptional. The normal assumption is that tasks are uninterrupted. However, it is straightforward to add a trigger analysis stage to most task analysis methods.

In terms of the ecology of interaction, triggers remind us that tasks are not always performed to completion. In the television quiz program *Mastermind*, the time limit would sometimes buzz when the quizmaster, Magnus Magnusson, was in the middle of a question. He would calmly announce, "I've started, so I'll finish," and then continue the question. In the heat of the real world, it is rarely so easy to ignore interruptions and distractions (chap. 14). In practice, tasks are interleaved with other unrelated tasks or, potentially more confusing, different instances of the

same tasks, and may be interrupted and disrupted by other activities and events. Furthermore, the performance of the tasks is dependent on a host of sometimes fragile interactions with the environment and apparently unconnected events.

19.3.2 Artifacts: Things We Act on and Act With

Notice that one of the trigger types is an environmental cue—a thing in the environment that prompts us to action. Some years ago one of the authors received a telephone call reminding him to respond to a letter. He could not recall receiving it at all, but searching through a pile on his desk he found it and other letters, some several weeks old, unopened and unread. What had happened? His practice was to bring the post upstairs to his desk but not always read it straightaway. Not being a coffee drinker, it was not coffee time that prompted him to open the post but just the fact that there was unopened post lying on his desk. This process had worked perfectly well until there was a new office cleaner. The new cleaner did not move things around on the desk but did straighten higgledy-piggledy piles of paper. However, he had unconsciously been aware that unopened post was not tidy on the pile. This had effectively been acting as a reminder that post needed dealing with—this is a trigger. But with the new cleaner, post that for some reason got missed one day would then look as if it was tidily "filed" in a pile on the desk.

This story is not unique. The ethnographic literature is full of accounts of artifacts being used to manage personal work and mediating collaborative work. In some cases the effect has to do with the content of the artifacts, what is written on the paper, but in many cases it has to do with the physical disposition—by orienting a piece of paper toward you, I say, "Please read it." In the example above, the cue that said, "Post needs to be opened," was purely in the letters' physical orientation (not even their position).

Of course, artifacts do carry information and are often the inputs or products of intellectual work. Furthermore, in physical processes, the transformation of artifacts is the purpose of work.

One example that has been studied in detail in the ethnographic literature is air traffic control (see also chaps. 1, 13, and 25), and all these uses of artifacts are apparent in air traffic control (Hughes et al., 1995). Of central importance are flight strips (Fig. 19.6), small slips of card, one for each aircraft, recording information, such as flight numbers, current height, and headings. An aircraft's flight strip is essential for the controller managing the aircraft, but the strips taken together constitute an at-a-glance representation of the state of the airspace for other controllers. In addition, the strips are in a rack, and the controllers slightly pull out the strips corresponding to aircraft that have some issue or problem. This acts partly as a reminder and partly as an implicit communication with nearby controllers. Finally, the strips in some way represent the aircraft for the controllers, but of course the real purpose of the process is the movement of the aircraft themselves.

Task models often talk about objects, either implicitly in the description of subtasks or explicitly in the task model. However, the objects are always "second class"—users act on them, but they are not part of the task. CTT and most work process notations do talk about automated tasks but not about the artifacts, whether electronic or physical, included in the interaction.

FIG. 19.6. Air traffic control flight strip.

In Unified Modelling Language (e.g., Bennett, Skelton, & Lunn, 2001) and other object-oriented design methods (e.g., Jacobsen, 1992; see also chaps. 7, 11, 12, 22, 24, and 27), it is common to give a life cycle description of "objects." However, this is usually because the intent is to store and automate the object electronically. Also, workflow analysts study document life cycles, again largely because of the intention to automate.

The task analysis chapter of Dix, Finlay, et al. (1998) treats physical objects as "first class" in an example of entity-relationship style task analysis. The method used was based largely on the ATOM method (Walsh, 1989), but to our knowledge this type of method has not gained widespread acceptance.

There is no reason why most task analysis methods should not adopt some form of artifact tracking. The tracking may be as simple as recording which artifacts are triggers for, used by, modified by, or produced by any particular subtask. For tasks where artifacts are particularly central, more sophisticated artifact life cycles could sit alongside the task description. These life cycles may be mundane (letter closed → letter open), but this is the point: Users recruit their everyday knowledge and physical properties of the world to coordinate their activity.

19.3.3 Placeholders: Knowing What Happens Next

It is half past five in the evening. The busy office building is beginning to become quiet as people pack up to go home. One or two employees work late in their offices, but as the evening wears on, they too go home. Soon there is only the hum of vacuum cleaners and the clatter of waste bins as the office cleaners do their work, until eventually the last light goes out and the building sleeps. A few employees have taken papers and laptops home and continue to work, but eventually they too put aside their work and sleep.

It is three o'clock in the morning, and in the darkness and silence of the office and the deep sleep of all the employees, where is the *memory* of the organization? The next morning at nine o'clock the office is a flurry of activity. It has not forgotten and has restarted its activities, but how?

We have already discussed two aspects of this memory: information required to perform tasks, and triggers that remind us that something needs to happen. However, there is one last piece of this puzzle that we have only hinted at. As well as knowing *that* we need to do something, we need to know *what* to do next. In the complex web of tasks and subtasks that make up our job, *where* are we?

In fact, when looking at triggers, we have already seen examples of this. The post being untidy on the desk said, "Something needs to happen," but the fact that it was unopened said, "It needs to be opened." We have already noted that similar triggers could cause subtasks to be performed out of sequence. If we only have a small number of dissimilar tasks, moving out of sequence is unlikely to happen, as we can remember where we are in each task. However, as the number of tasks increases, especially if we are performing the same task on different things, we find it harder to remember where we are.

Let us look again at air traffic control. One of the controller's tasks is to manage the flight level of aircraft. A much simplified model of this activity is shown in Fig. 19.7. Because this is a shared task between the controller and the pilot, each box is labelled with the main actor (although Tasks 2 and 3 are both communications). Recalling earlier sections, we might ask what

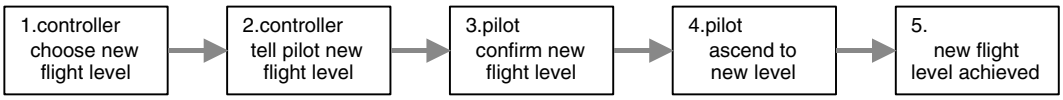

FIG. 19.7. Flight-level management task.

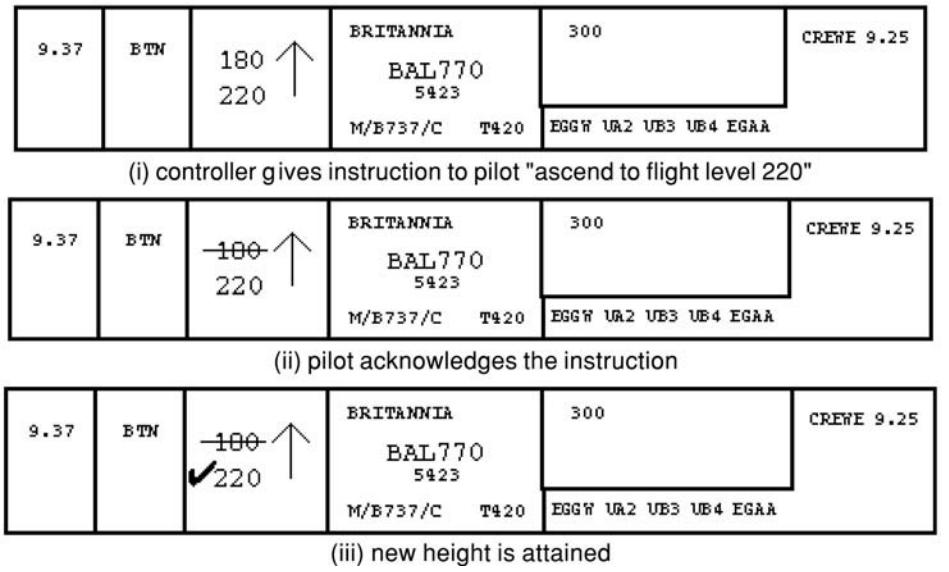

FIG. 19.8. Flight strip annotated during task.

information is required at each stage. For example, Task 1 would depend on radar, the locations of other planes, planned takeoffs and landings, and new planes expected to enter the airspace.

Note that Box 5 is not really a task but more a "state of the world" that signifies task completion. However, it is important, as the controller will need to take alternative action if it does not happen. Of course, without appropriate placeholders, the controller might forget that a plane has not achieved its target level. This may either cause trouble later, as the old level will not be clear, or even provoke potential conflicts between aircraft.

In fact, the flight strips do encode just such a placeholder (see Fig. 19.8). When the controller informs the pilot of the new height, he writes the new level on the flight strip (i). When the pilot confirms she has understood the request, the controller crosses out the old level (ii). Finally, when the new level has actually been reached, the new level is ticked (iii).

Virtually all task-modeling notations treat the placeholder as implicit. The sequence of actions is recorded, but not why the user should do things in the way proposed. Of course, one purpose of task analysis has been to produce training—that is, to help people learn what appropriate processes are—but learning this does not help them to actually remember where they are in the processes.

As with other forms of information, placeholders may be stored in different ways:

- in peoples' heads (remembering what to do next)
- explicitly in the environment (to-do-lists, planning charts, flight strips, workflow system)
- implicitly in the environment (Is the letter open yet?).

Although often forgotten, placeholders are crucial for ensuring that tasks are carried out effectively and in full. At a fine scale, it is rare to find explicit records, as the overhead would be too high. Instead, memory and implicit cues predominate. As users' memory may be unreliable when they are faced with multiple tasks and interruptions, it is not surprising to find that various forms of environmental cues are common in the workplace. However, electronic environments do not have the same affordances to allow informal annotations or fine tweaking of artifacts' disposition.

19.4 FINDING TRIGGERS

This chapter is focused on triggers and associated placeholders. However, these are important because of the general problem of missing stimuli, and they are linked to issues such as interruptions. The techniques we use are designed to expose all these problems as well. Part of the data we collect is on *what* is done. In traditional workflow fashion, we catalog the various activities performed and the dependencies between the activities. However, this is only intended as the superstructure of the analysis, not the focus. Instead, our focus is on *when* activities are performed and *whether* they happen at all. The central and distinguishing feature of our work is therefore the way we look explicitly for the triggers that initiate activities.

19.4.1 Standard Data Collection

Because of the similarity of our study to traditional task analysis, we can use many of the same sources for data collection: documentation, observation, and interviews. However, trigger analysis gives us an additional set of concerns.

19.4.1.1 Documentation

Documentation of long-term processes is likely to be relatively accurate, although it may omit the activities beyond organizational boundaries and also most of the triggers. However, we can use it as an initial framework that can be filled out by observation or during subsequent interviews.

19.4.1.2 Observation

Direct observation is a very effective technique, widely used in ethnographic studies or similar sorts of analysis. In many office situations, there are several instances of the same process at different stages of completion. For example, in an insurance office, many claims are processed, and each might be at a different stage. In these cases, using "a-day-in-the-life" observation may be sufficient. So long as we can see each activity during the study period, we can piece the activities together afterwards. Even if we never see a process run from end to end, we can reconstruct it from its parts. This is similar to observing a natural forest. The complete life cycle of a tree might be hundreds of years long, but by looking at trees at different stages of growth, we can build up a full picture over a much shorter period.

However, direct observation poses special problems when many of the processes of interest extend beyond the time frame of observation and are geographically dispersed. In particular, it may miss rare but potentially significant events. Observation usually fails in the following situations:

- Activities are sporadic and long term.
- Processes are in lockstep and long term.
- Unusual events occur.

19.4.1.3 Interviews

Where direct observation is impractical, interviewing can be used. Interviewing allows both prospective investigation (asking about what is currently going on) and retrospective investigation (asking about what has happened). However, interviewing is often regarded as problematic because the accounts people give of their actions are frequently at odds with what they actually do, although chapter 16 argues otherwise.

We are, however, in a strong position as we approach such interviews. Our analytic focus—the structure imposed by task analysis and the specific interest in triggers—allows us to trace

omissions and inconsistencies and obtain reliable results from interviews. This is important as, although we would normally expect some additional direct observation, practical design must rely principally on more directed and less intrusive techniques.

19.4.2 Types of Triggers

Based principally on our theoretical analysis and refined by the results of our previous case studies (Dix, Ramduny, & Wilkinson, 1996, 1998), we have classified the different kinds of triggers that occur. These include the following five broad categories:

1. *Immediate triggers* occur when one activity begins immediately after the previous activity reaches completion.
2. *Temporal triggers* include periodic actions that happen at regular intervals or actions that occur after a particular delay (e.g., the expectation of receiving a response by a certain date or the generic task of reminding people based on some time interval).
3. *Sporadic triggers* arise when an individual responsible for some action remembers that it must be done.
4. *External events* include alarms and other signals (e.g., a wristwatch or automatic calendar set to give a reminder at a specific time) and specific events (e.g., a telephone call, a face-to-face request, the receipt of a message, the completion of an automatic activity, or even an event in the world).
5. *Environmental cues* are things in our environment that remind us that something ought to be done, whether explicitly recorded (e.g., a diary entry or to-do list) or implicit (e.g., unanswered e-mails or a half-written letter in the typewriter).

Some of these triggers may be evident from observation, such as when a telephone rings and someone does something (external event). However, others are less clear, and this is where direct questioning helps ("Why did you do that then?"). In particular, it may be difficult to tell whether someone just remembers something (sporadic) or something acted as a reminder (environmental cue). The reason that these cases are difficult to determine is that both types of triggers relate to the user's perception and psychological state rather than observable actions.

After having identified the primary triggers, we need to assess their robustness by asking follow-up questions. Sometimes if the primary trigger fails, there is a secondary trigger that stops the process from failing. For example, if someone says that he or she always does two things one after the other (immediate), we can ask what would happen if there was an interruption. The answer may be that the person would need to remember (sporadic) or that the half-finished task would act as a reminder (environmental cue).

Sometimes the primary trigger is sufficiently complex to need its own secondary trigger. So, depending on the answers, we may have several levels of questions:

1. "Why did you send the reminder letter?" "Because it is 2 weeks since I sent the original letter" (temporal).
2. "How do you know it is two weeks?" "Because I wrote in my diary when I sent the letter" (environmental cue).
3. "What makes you check your diary?" "I always check it first thing each morning" (temporal).
4. "When you send the first letter, what happens if you are interrupted before entering the expected date in the diary?" "I always remember" (sporadic secondary to immediate primary).

Notice that both Questions 3 and 4 are suggested by the answer to Question 2. Question 3 is about chasing the chain of events that led to a particular trigger. In contrast, Question 4 arises

TABLE 19.1
Triggers, Failure Modes, and Follow-up Questions

Trigger Type	*Failure Mode*	*Follow-up Questions*
Immediate	Interruption	Does the second activity always proceed immediately? If there is any possibility of a gap, then look for secondary or backup trigger(s).
Temporal		
Periodic action	May forget	How do you remember to perform the action in the relevant period?
Delay	Poor memory	How do you remember the delay? This may be part of a routine, like consulting a diary every morning, but if it is an hourly activity, then how do you know when it is the hour? Perhaps the clock strikes, an external event—another trigger.
Sporadic		
Memory	May never happen	How do you remember to do something? If a request is made verbally, the recipient has to remember that the request is outstanding until either it can be performed or some record is made of the commitment, a reminder—another trigger.
External event		
Automatic	Reliability	How reliable is the medium of communication?
Communication	Delay	Are there likely to be any communication delays?
Environmental cue		
Explicit	Is moved	Can the cue be disrupted?
Implicit	Failure to notice	How do you remember to examine the cue? Perhaps a periodic activity or a note in a diary—a temporal trigger. Can the cue be missed?
	Ambiguity	Can the cue be confused with others, leading to skipped or repeated activities?

because the answer to Question 2 was an environmental cue and things in the environment need to have been put there by a previous activity. Note that this should also lead to a reexamination of the task associated with sending the original letter, as this should include the "write expected date in diary" activity.

We could continue asking follow-up questions indefinitely, but at some point we must stop. We can either believe that a trigger does always occur as specified or, if not, look at the whole process and assess the consequences should the activity fail to trigger at all and perhaps look at any delays associated with noticing it.

Table 19.1. lists some of the main ways in which different types of trigger can fail and the sort of follow-up questions that can be asked. It is not complete and is being amended as we learn more. An evolving and more detailed table with examples can be found on the trigger analysis web pages: http://www.hcibook.com/alan/topics/trigger-analysis/

19.4.3 Studying Artifacts: Transect Analysis

As we have seen, artifacts, both physical and electronic, are an inseparable part of an ecologically valid understanding of work and leisure. Tasks may be initiated to create an artifact (write a chapter abstract), tasks may occur because of artifacts (the memo requesting an action, the broken machine), artifacts may mediate tasks split between several people (patient records, whiteboard), artifacts may record where you are in a task (document in in-tray, office planner), and electronic artifacts may even control tasks (workflow systems; e.g., Winograd's [1988] Coordinator).

Studying artifacts can therefore give us a rich understanding of the tasks that they are part of, especially when the tasks are complex, are long lived, or involve different people.

One artifact-centered method that is particularly useful for uncovering triggers and placeholders is *transect analysis*. This method takes an ecologically rich approach to looking at artifacts in their physical context, and it views physical disposition as being as important as the artifact itself. It focuses on uncovering the task in praxis as performed in the actual work environment. Transect analysis looks at a snapshot of a work environment (desk, office, or potentially organization), either at a particular time (noon on Tuesday) or over a relatively short period (a day in the life).

Transect analysis is inspired by use of the transect in environmental studies to investigate a cross-section of an environment on a particular day. If the ecology is diverse and nonseasonal, it is possible to build up a picture of the life cycles of particular organisms and the interrelations between them even though they are not being studied over time. Similarly we can look at each artifact in the work environment and ask, "Who is using it?" "What for?" "Why is it here?" "What would happen if it were somewhere else?" "How does it relate to other artifacts?" In particular, we look for instances of the same kind of artifact in different places (e.g., several invoices in different stages of processing) within one person's immediate environment (in-tray, center desk, out-tray, at an angle on a pile) or between people. By piecing together these snippets of human–artifact interaction, we can create models of task processes and artifact life cycles.

This analysis can be done initially by the analyst but at some stage should be used as a prompt for the actual users. If asked questions away from the actual environment, they may not volunteer many of the unofficial procedures that make the workplace work. However, when shown a document in location and asked, "Why is it there?" they will usually be able not only to knit together the scattered threads of processes but also add details beyond the immediate workplace. If interviews are carried out in the workplace, then transect analysis can be woven seamlessly into the normal interviewing process.

19.5 LARGER STRUCTURES

Trigger analysis can be used in the context of virtually any task analysis. In the case studies that we have done, we have used a simple process diagram as a method of recording the task analysis and asked specific questions about the overall robustness of the process. However, we believe that the lessons and the broad techniques would be applicable to any type of process-oriented task analysis, such as HTA.

19.5.1 Processes and Activities

We record the processes as a series of circles or bubbles, one for each activity. Each bubble names the activity and the person or persons who perform it. Lines between the bubbles record dependencies, and arrows at the beginning of each bubble record the trigger for the activity (see Fig. 19.9).

There are plenty of methods for recording processes, and this is not the focus of our work, so we take a minimalist approach. We do not attempt to record all the complexities of real processes in a single diagram. Instead, we use many separate diagrams, often concentrating on specific scenarios. The crucial thing is to look for the trigger of each activity.

The level of analysis is also governed by this focus. In general, we place activity boundaries wherever there is the likelihood of a delay or gap. The most obvious such break occurs when people at different sites perform subsequent activities in a process. However, there are often distinct activities performed sequentially by an individual, as in the letter-reading example above. In principle, such analysis could go down to the full detail found in HTA. This would be reasonable if, for example, interruptions were possible in the middle of typing a letter.

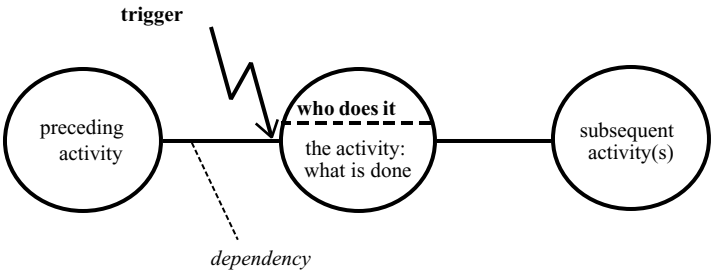

FIG. 19.9. Recording processes.

We deliberately use the term *activity* rather than *action* to emphasize that the lowest level of our analysis is far from atomic. Activities may be shared between individuals. For example, having a meeting or dictating a letter would be regarded as a single activity involving several people. Again, one could dissect such an interaction, but this would be the remit of conversational analysis. We may also ignore details of an activity because they are uninteresting or we do not have sufficient knowledge about them. For example, if we issue an order to an external organization and then wait for the goods to arrive, we may not be interested in the internal processes of that firm. Finally, we include some activities that would normally be omitted in a traditional process model. In particular, we often include the receipt of a message as a distinct activity. This is done to emphasize the gap that may occur between receipt and response.

19.5.2 Process Integrity

We can assess the reliability of the work process by asking questions about the triggers for activities. However, nothing is ever 100% correct, and it is inevitable that triggers will fail for some reason, activities will be missed, and perhaps the whole process will stop because something goes wrong. A process model combined with a well-founded assessment of the reliability of each activity can allow us to assess the robustness of the whole process. If someone fails to complete some activity and the next activity is never triggered, what happens? Does the whole process seize up or will the failure eventually be noticed?

Note that this is not an ad hoc procedure. We can *systematically* go to each trigger and ask what happens to the entire process if the trigger fails. Furthermore, by looking at the process as a whole, we can improve our assessment of the reliability of any trigger. For example, if the trigger for an activity is that a report is in someone's in-tray, we can examine the wider context and assess the likelihood that the report will indeed be there when required.

19.5.3 The 4Rs

Although our initial focus was on the individual triggers, we began to notice an emerging pattern as we recorded the processes during our case studies. We call this pattern the 4Rs: request, receipt, response, and release.

Figure 19.10 shows a simplified version of a case study on the operation of a conference office (Dix et al., 1996). We can see a general structure emerging:

Request: Someone sends a message (or implicitly passes an object) requiring your action.
Receipt: You receive it via a communication channel.
Response: You perform some necessary action.
Release: You file or dispose of the things used during the process.

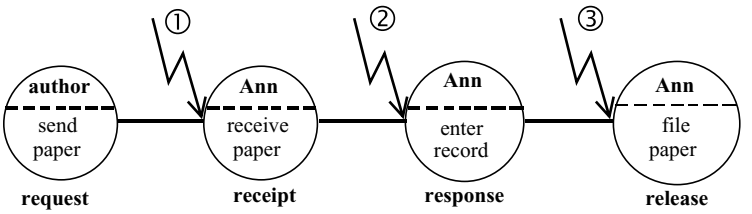

FIG. 19.10. The 4Rs.

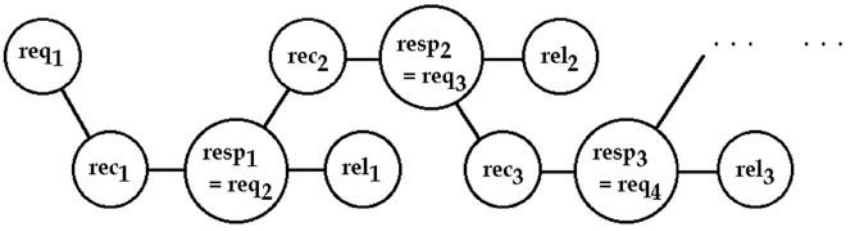

FIG. 19.11. A 4Rs chain.

At this point, if the functional goal has been achieved, the process can be considered to have reached completion.

The process in Fig. 19.10 is very similar to the process that one of the authors follows when dealing with e-mail. When the mail arrives, he reads it (or at least notes its arrival) but does not deal with it immediately. It stays in his in-tray until he has replied or otherwise dealt with it. Only at that stage does he file it in a folder or discard it. If he is interrupted after replying, the original message is still in the in-tray (secondary trigger). Once, whilst he was in the middle of replying to a message, the machine crashed (interruption). When some time later he again read his e-mail, he mistakenly (and unconsciously!) took the continued presence of the e-mail in the in-tray as signifying an interruption before filing (secondary trigger), and hence he filed the message without replying.

We believe that the 4Rs constitute a fundamental unit of long-term work. Not only does the pattern of activities occur among different processes, but we also see a similar pattern of triggers. ① is always simply some sort of communication mode and can be assessed for reliability and timeliness. The response activity is typically triggered by ②, the presence of a document or other object. The release activity triggered by ③, which is of the "immediately follows" kind, removes that cue but also relies on its existence as a secondary trigger. The problems with the author's e-mail will occur elsewhere! The existence of generic patterns makes it easier to uncover problem situations quickly and to take solutions found in one situation and adapt them to another.

Our case studies show that the 4Rs are normal: The same pattern recurs with similar triggers and similar failure modes. We have also seen that it is normative: If the 4Rs pattern is nearly followed but with some deviation, the deviation is an indication of possible problems. It is also frequently the case that the response of one 4Rs pattern forms the request activity initiating a new 4Rs pattern. A chain of such patterns constitutes a sort of long-term conversation (see Fig. 19.11). The 4Rs appear to make up a pervasive, generic pattern at a lower level than the patterns identified in Searle's speech-act theory (e.g., Winograd & Flores, 1986) and are perhaps the equivalent of adjacency pairs found in conversational analysis.

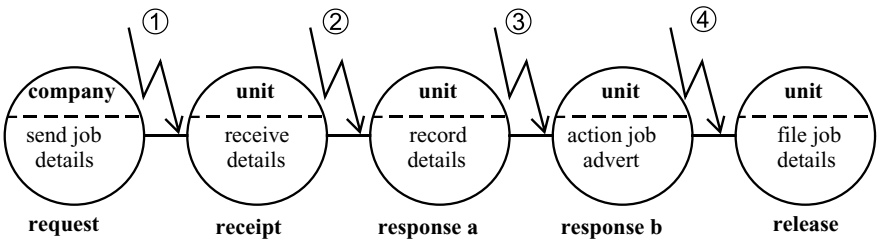

FIG. 19.12. Initial job advert.

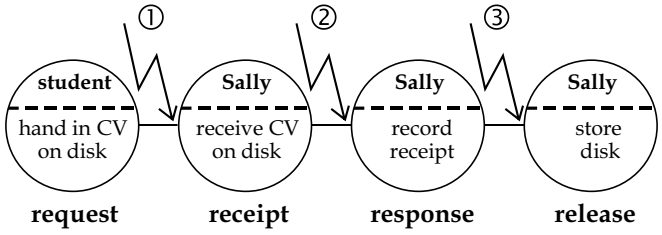

FIG. 19.13. Students submit CVs.

19.6 EXAMPLE: THE STUDENT PLACEMENT UNIT

A key aspect of validating the trigger analysis was to apply the framework to prolonged interactions in an extremely busy office environment—the Placement Unit at the School of Computing and Mathematics at the University of Huddersfield. The unit is staffed on a full- and part-time basis by administrative and academic staff respectively and is responsible for helping some 200 sandwich course students secure 1-year placements in industry every year. Besides dealing with students seeking placements (predominantly recruitment and skills-building processes), the unit also supports those already on placement (including processes in assessment and monitoring).

Also, the MaPPiT project[1] had just been launched at the university, and part of the project remit was to develop a process support system in Lotus Notes (e.g., Lloyd & Whitefield, 1996) for the placement unit, thereby automating many placement activities.

Contact with companies occurs via all media, the telephone being the most common and the most interruptive. Outright winners in the interruption stakes, however, are the students, for whom the unit has an "open door" policy between 10 a.m. and 4 p.m.

Figures 19.12 to 19.14 show that the activity triggers had the potential to be seriously delayed, sometimes indefinitely, if they had been left to be resolved by outside companies.

19.6.1 Case 1: Job Adverts

The establishment of a new placement starts with the initial request from a company for a placement student (see Fig. 19.12). This is a 4Rs pattern but with a two-stage response. One of the dangers of multistage responses is that the triggers at ② and ③ are often similar, which can lead to problems.

[1]MaPPiT stands for Mapping the Placement Process with Information Technology, a HEFCE-supported 2-year project funded by the Fund for the Development of Teaching and Learning. Details available at http://www.hud.ac.uk/scom/mappit/

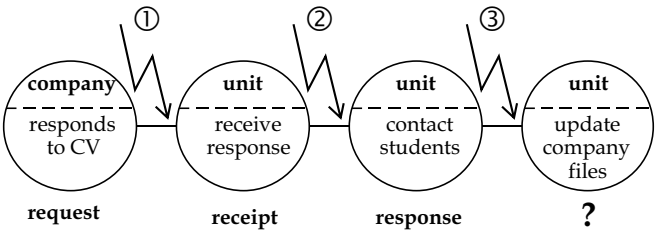

FIG. 19.14. Company decision.

In the advert process the company initiates the request. Many placement providers recruit annually, diarizing to send the unit their requirements. They drive the process by setting closing dates for applications. Another set of companies are triggered into providing placement details by a standard letter from the unit sent out fortnightly via a diary system when it is deemed most appropriate to make the contact. Looking at ①, we asked the administrative staff how they would know if details failed to arrive, thus breaking the chain of activities. At present, the only backup is the diary, so a time delay occurs between the failure occurring and the next fortnightly (sometimes monthly) check for responses from the previous month's companies. A follow-up question would be, "How do you remember to look in the diary?" The answer is that the paper-based diary remains highly visible on the placement officer's desk. However, as the year progressed, we noticed the diary being checked less and less.

The next activity, "record details," should ideally follow on directly from the first activity. This class of trigger is insecure and liable to interruption—a common occurrence in the unit. The staff member then has to remember what to do next. Usually, there is the environmental cue on the desk, such as a note jotted down from a phone call, a fax copy, a letter, or an open e-mail message. The follow-up question was, "What if you do not record the receipt of details in the diary?" Another staff member could check the diary, see the assumed nonreceipt of details, and annoy a company by chasing for details already sent in unless he or she checked the job adverts log (response b) first or checked the company file for the ad. All this checking seemed unnecessarily complicated.

With Lotus Notes in mind, the project team accepted these current problems as needing resolution. The diary could easily become electronic with built-in navigators (agents) that automatically trigger reminders to execute activities. Gone is the need to remember to check the diary, as reminders appear in individuals' to-do lists. Even to-be-done-to lists can be constructed. Receipt of a job description need only be recorded in one place, necessitating only one checking activity before chasing a company. Furthermore, the electronic record means that any inconsistency between recorded details can be displayed and thus act as a trigger at ③.

19.6.2 Case 2: Submission of CVs

A large proportion of the unit's placement providers happily accept standard CVs from students, so CVs are lodged with the unit and checked by placement tutors very early in the year. Figure 19.13 shows this process, and again we see a straightforward 4Rs pattern.

The first two activities usually happen face to face, so there is little risk of breakdown at ①. We noticed that some students, unfortunately, ignored the office hours of the unit and so "posted" CV disks under the door after staff had left for the day, creating a risk that the disk would be disposed of, damaged, lost, or misplaced. Similarly, the unit's activities here are all exposed to interruptions and therefore incompletion. Also, staff changes in the unit brought some new faces unaccustomed to the subprocess of receiving and recording students' disks.

In one case, a student made alterations to the CV and returned the disk, but a new member of staff promptly lost the update while being interrupted to do other more complex tasks. In other cases, a disk was found to be corrupt when the CV was required, leading to another set of interactions with the student and further delays.

The planned Notes implementation for this completely bypasses the current error-prone process. Students would fill in a CV template using a Web browser. The CV is then automatically submitted to a Notes database that logs the receipt and sends an e-mail to the unit to confirm that the CV has been submitted on time. Students can update the CV at any time without bothering the administrative staff and in the full knowledge that the latest version will be sent to potential placement employers.

19.6.3 Case 3: Company Decisions

After seeing the students' CVs, companies decide on students to shortlist for interviewing. Figure 19.14 demonstrates how the pace of interaction can really slow down when pursuing students to arrange interviews or to provide feedback if rejected.

At the response stage, the pace slows considerably once the students are on vacation and hard to track down. Much time can be spent trying to contact students, and assuming contact is made, the next activity can be stalled by a phone call or face-to-face enquiry. We rely on an individual's good short-term memory and/or an environmental cue to ensure the sequence is fulfilled.

Although the process appears at first to be a simple 4Rs pattern, we have put a question mark against the last activity. The release usually consumes or destroys the environmental cues that have prompted previous activity. It is not clear that this is the case for this process—what are the environmental cues? The company's decision will arrive in a letter or be recorded on paper, but the slow pace of the response means that the cue may be lost or grow stale, ceasing to be salient because it has been around too long.

Redesigning this sequence in Notes, we decided that it would be better to record the receipt of the company's contact electronically when the contact actually happens, invariably by means of a phone call, fax, or letter. All the information can therefore be displayed on screen, and so if incomplete because of an interruption, it cannot be discarded without a prompt. Note how this action has established an environmental cue within the electronic environment. In the revised process, this cue is removed when the contact details are complete, thus making the pattern a true 4Rs, with robust triggers throughout. Moreover, if the response stage becomes drawn out and relies on chasing student responses, there is the possibility of automatically signaling if the expected reply is not forthcoming, thus supplying a to-be-done-to facility.

From the three cases discussed above, we can see how different levels of automation have been suggested by the trigger analysis and the 4Rs pattern. At one extreme, the strategy involved the complete bypassing of the human process, but in the others only parts were automated. More importantly, the analysis has ensured that the Notes implementation does not hide existing triggers, as often is the case with electronic filing, but instead enhances the triggers with automatic reminders and electronic environmental cues.

19.7 CONCLUDING REMARKS

This chapter has considered the analysis of work processes with a view to understanding more clearly why things happen when they do. There is a strong indication that work activity is more prone to breakdowns and failures when the interaction is prolonged, when the pace of

interaction is slow, or when the activity is of a collaborative nature and crosses organizational boundaries. When all three conditions are met, failure becomes almost a certainty.

We recommend trigger analysis as a useful method for uncovering what prompts a task to happen. One particular advantage of trigger analysis is that weaker triggers can be identified swiftly and action can be taken to render the tasks more robust, as we saw with the student placement example. Moreover, the classification of the triggers by type allows a simpler analysis. A further benefit of this method is that individuals may conduct investigations with little or no expertise in knowledge elicitation techniques. The follow-up questions keep it simple, and the analysis can be used with virtually any other forms of task analysis.

Trigger analysis does not only stop at triggers; artifacts and placeholders also have their roles to play. Any activity in a workplace involves artifacts of work, and we recommend that current task analysis methods should accommodate some form of tracking for artifacts, placing them on a more equal footing with the user in the task context than is now common. Just as triggers remind us that *something* needs to be done, placeholders must be robust so that we know clearly *what* needs to be done.

The importance of environmental cues means there is another rich source of information: the work environment itself. We look at an office and see papers and files on the desk, Post-it notes, an in-tray, a wall calendar. Why is that file on the desk? What will happen to it? What would happen if it were not there? We know that environmental cues can be triggers for activities, and so we take each item in the environment and look for the activity it triggers or the coordinating role it fulfils. At the very least, a piece of paper left on the desk is saying, "File me please."

The 4Rs framework was found to be a fundamental unit of interaction and work—a handy "template" for relatively straightforward work analysis. A key feature of the persistent pattern of 4Rs is its ability to highlight weaknesses in prolonged interaction. We can establish whether the 4Rs are complete, as incomplete 4Rs have always revealed a weak trigger, environmental cue, and/or placeholder.

Finally, the study of work situations should not be isolated from the real context in which the work is conducted. Trigger analysis fits easily into such an environment, as it needs to examine and understand that environment. Most importantly, it can complement other task analysis methods that are also suitable for field investigations.

In summary, trigger analysis can augment many task decomposition methods. In addition to the description of *what* should happen found in most such methods, it captures *why* they occur *when* they do and *whether* they are likely to be missed or misordered. The strong emphasis on all types of triggers, including environmental cues, leads to an ecologically rich analytic approach.

19.8 FURTHER INFORMATION

The most complete theoretically rooted account of triggers and the 4Rs is the Dix et al. (1998) article in *Interacting with Computers*. For an analysis of how rich ecological features, including triggers and placeholders, can augment more formal task analysis models and notations see Dix (2002a). Finally, for new information and hyperlinked material, consult the trigger analysis web pages: http://www.hcibook.com/alan/topics/trigger-analysis/

REFERENCES

Abowd, G., & Dix, A. 1994. *Integrating status and event phenomena in formal specifications of interactive systems.* In *Proceedings of the 2nd ACM SIGSOFT symposium on foundations of software engineering* (pp. 44–52). New York: ACM Press.

Bennett, S., Skelton, J., & Lunn, K. (2001). *Schaum's outline of UML.* New York: McGraw-Hill.

Brewster, S. A., Wright, P. C., & Edwards, A. D. N. (1994). *The design and evaluation of an auditory-enhanced scrollbar.* In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence* (pp. 173–179). New York: ACM Press; Reading, MA: Addison-Wesley.

Dix, A. J. (1991). *Formal methods for interactive systems.* London: Academic Press.

Dix, A. J. (1992). Pace and interaction. In *Proceedings of HCI'92: People and computers VII* (pp. 193–207). Cambridge: Cambridge University Press.

Dix, A. (1994). Que sera sera: The problem of the future perfect in open and cooperative systems. In *Proceedings of HCI'94: People and computers IX* (pp. 397–408). Cambridge: Cambridge University Press.

Dix, A. (2002a). Managing the ecology of interaction. In *Proceedings of First International Workshop on Task Models and User Interface Design (Tamodia 2002)* (pp. 1–9). Bucharest: INFOREC Publishing House.

Dix, A. (2002b). Towards a ubiquitous semantics of interaction: Phenomenology, scenarios and traces. In *Proceedings of DSV-IS 2002* (pp. 238–252). Berlin: Springer-Verlag.

Dix, A., Finlay, J., Abowd, G., & Beale, R. (1998). *Human-computer interaction* (2nd ed.). Hemel Hempstead: Prentice-Hall.

Dix, A., Ramduny, D., & Wilkinson, J. (1996). Long-term interaction: Learning the 4Rs. In *CHI '96 Conference Companion* (pp. 169–170). New York: ACM Press.

Dix, A., Ramduny, D., & Wilkinson, J. (1998). Interaction in the large. *Interacting with Computers, 11*(1), 9–32.

Hughes, J., O'Brien, J., Rouncefield, M., Sommerville, I., & Rodden, T. (1995). Presenting ethnography in the requirements process. In *Proceedings of the IEEE conference on Requirements Engineering* (pp. 27–34). New Jersey: IEEE Press.

Jacobsen, I. (1992). *Object-oriented software engineering*. Wokingham: Addison-Wesley.

Lloyd, P., & Whitefield, R. (1996) *Transforming organisations through groupware: Lotus Notes in Action.* Berlin: Springer-Verlag.

Norman, D. A. (1986). New views of information processing: Implications for intelligent decision support systems. In E. Hollnagel, G. Manuni, & D. D. Woods (Eds.), *Intelligent decision support in process environments* (pp. 123–136). Berlin: Springer-Verlag.

Norman, D. A. (1988). *The psychology of everyday things.* New York: Basic Books.

Paternò, F. (1999). *Model-based design and evaluation of interactive applications.* Berlin: Springer-Verlag.

Payne, S. J. (1993). *Understanding calendar use. Human-Computer Interaction, 8*, 83–100.

Shepherd, A. (1995). Task analysis as a framework for examining HCI tasks. In A. Monk, & N. Gilbert (Eds.), *Perspectives on HCI: Diverse approaches* (pp. 145–174). London: Academic Press.

Walsh, P. (1989). Analysis for task object modelling (ATOM): Towards a method of integrating task analysis with Jackson system development for user interface software design. In D. Diaper (Ed.), *Task analysis for human-computer interaction* (pp. 186–209). Chichester, England: Ellis Horwood.

Winograd, T. (1988, December). Where the action is. *Byte*, pp. 256–258.

Winograd, T., & Flores, F. (1986). *Understanding computers and cognition: A new foundation for design.* Reading, MA: Addison-Wesley.

Wood, A., Dey, A. K., & Abowd, G. D. (1997). *CyberDesk: Automated integration of desktop and network services*. In *Proceedings of the 1997 Conference on Human Factors in Computing Systems (CHI '97)* (pp. 552–553). New York: ACM Press.