DUO: A Discount Observation Method

Sari A. Laakso, Karri-Pekka Laakso
Interacta Design Oy
Vuorimiehenkatu 23 B
FI-00140 Helsinki, FINLAND
+358 9 6813 8520
{sari,karri}@interacta.fi

Colleen Phillips Page
Microsoft
Home Products Division
One Microsoft Way
Redmond, WA, 98052 USA
+1 425 706 4020
colleenp@microsoft.com

ABSTRACT

Field observations produce an overwhelming amount of data that must be organized for analysis and application. When qualitative field data is merely organized by tasks, we risk losing sight of the critical task interruptions, overlaps, and interdependencies that make up the fabric of communication and cooperation. We have found that a visual representation of events is needed, showing tasks in context with interruptions and temporal overlaps for reference during data analysis, scenario development, user interface design, system testing, and usability evaluation.

We have developed DUO (Discount User Observations), a discount method for recording qualitative field observations. The DUO format includes a sequential overview of the whole session as well as task-based sequential views of specific events. These high-level and detailed timeline views provide an easy way to track interrelated tasks throughout analysis, design, and testing.

Keywords

User observation, field study, qualitative data analysis, sequential data, DUO

INTRODUCTION

User observations produce an overwhelming amount of data that must be organized for analysis and application. The challenge is to find a discount method to document user observations in a meaningful format that preserves the full richness of data. We have researched alternative methods including video analysis, Contextual Inquiry [2,3], and sequential data analysis tools such as SHAPA [10]. However, sometimes we just need a fast, direct view of the sequential data, without a lengthy transcription and encoding process.

Videotaped observations provide very objective, detailed data, but this data format can be difficult and costly to use. The low data density of videotaped observations require time-consuming transcription and analysis to extract the important events and convert the material to a more usable format [1].

To avoid the analysis cost, the observer can try to analyze observation data during the observation and use a videotape only as a backup. This is extremely demanding, as well as risky. It can lead to the documentation of inferences and interpretations of what we see, rather than actual objective observations [8, p. 264-265]. If we generalize observations by mixing up similar or consecutive events, critical details are lost. Instead of generalizations we need to concentrate on objective tasks, goals, artifacts, and events [2].

Our discount observation method covers nearly as much detailed data of events as video analysis at a fraction of the cost. Instead of videotaping, the observer takes time-stamped notes and gathers data samples with a digital camera. The resulting data set is rich in detail and considerably faster to process than a videotape. The timeline-based documentation simultaneously shows an overview of the whole session, timeline views of a single task, detailed descriptions and pictures of the events and artifacts related to the task, and the other events that overlap with the task.

In the following sections, we describe our data collection method, followed by the process of documenting the findings and validation of the documentation by the observed participant[1]. Finally, we highlight the phases of a design project that benefit from our documentation of low-level sequential observation data.

GATHERING DETAILED OBSERVATION DATA

During a typical three to four hour observation session, two observers share the work of collecting data. One of them, the *scribe*, writes time-stamped freeform notes of low-level events and actions on paper, and is responsible for inquiring about ambiguous data afterwards. The other, the *photographer*, takes digital photos of observed situations and artifacts.

The scribe gathers information as detailed as possible – names of people and companies, dates, times, and places –

[1] All examples are based on real findings of observations, but due to confidentiality requirements, the original data has been replaced with corresponding fictious examples.

regardless of whether she understands them or not at that moment. She can ask some of the more critical missing details right away to understand the situation better, e.g. *"Is this Susan the same person who just called you?"*, but she does not interrupt the user's activity, and does not allow the observation to turn into an interview. Generally, the missing and ambiguous data soon becomes clear, and the remaining unclear data can be clarified afterward. For this purpose, the scribe marks the unclear or ambiguous data. Fig. 1 shows an excerpt of a scribe's notes; she was not sure who "Susan" is, and she could not see the sender's name on the third email message (question mark).

The photographer takes digital photos of screens and artifacts, e.g. printouts, business cards and hand-written notes. The scribe's note-taking effort is significantly reduced by the photography. A single photo can easily capture a complete email message in seconds, for example, and a picture of the user's calendar captures rich details for analysis. Digital camera images have automatic timestamps which along with overview pictures help in reconstructing event sequences. The timestamps of the photos and the manual timestamps on the scribe's notes contain adequate temporal data for later documentation.

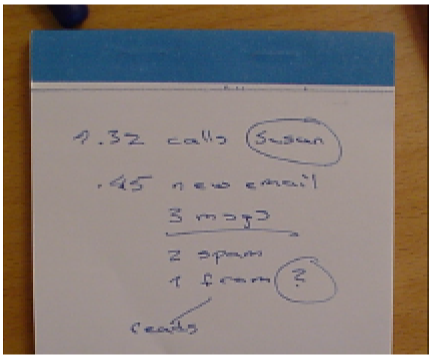

Figure 1. An excerpt from typical observation notes.

The observers do not focus on a certain piece of software, but instead collect data on everything the user is doing. If they ignored seemingly unrelated events, they might fail to note goals that current software does not support, thus missing a new product opportunity. In addition, tasks may have complex interdependencies, in which uninteresting tasks turn out to be very essential ones later on.

An indirect advantage of collecting low-level data is that it reduces the observers' cognitive load during observation. During intensive periods full of events, the scribe's working memory gets quickly filled with detailed data that he tries to write down, and he cannot afford cognitive resources on high-level processing of the data. If he tried, he would miss a lot of objective events and actions.

Typically, an observation session includes both short intensive periods full of events and relatively long periods of time when nothing happens. During the latter, the observers can process ambiguous connections between events.

We have tried observations with a single observer who is responsible both for note taking and photography, but the total cost is not significantly reduced. Two observers can of course collect more data, and we have also found that the data is more accurate. In addition to the quality of the data, the observation team can very quickly document the data after the session, and easily collaborate in the design cycle.

DOCUMENTATION OF OBSERVATIONS

After the session, observed events and actions are recorded on a timeline, and grouped by high-level tasks. We always document the observation immediately after the session, to ensure that the observers recall the details. A three to four hour observation with handwritten notes takes about four to five hours for two observers to document. The result is a timeline-based PowerPoint document with detailed data samples and photos.

Events on a Timeline

The observers begin by reading the scribe's notes and looking at photos. They produce a paper mock-up of the timeline with all the events attached, and create a descriptive title to label each event. The titles of events are based on the most interesting facts of the event, e.g. "two new messages" is more informative than "checking email". After drafting the timeline on paper, one of the observers draws it on the PowerPoint slide master (see the timeline on a gray background in Fig. 2 and 3; without black highlights).

Grouping the Events by Tasks

As a basis for task grouping, the observers print out the master timeline and manually group the events by tasks. Events are processed one by one, and for each event, a corresponding task is defined. After manual grouping, we create a new PowerPoint page for each task. The timeline that has been created on the PowerPoint slide master is visible on each page, and the events relating to the task are highlighted by adding new text boxes with reverse backgrounds on the timeline. In Fig. 2, events that belong to the *"Task 1: Product demo at Acme"* are highlighted on the timeline using four text boxes with black backgrounds.

In Fig. 2, two of the four events represent the same subtask on the background labeled *"Slides for demo"*. Typically, the user only has one background task (here, preparation for the demo at Acme), and it is no problem to repeat its event title on the timeline. Because the other two events, the phone call from Acme, and talking about the demo with a colleague, are very closely related to the general task of Acme's demo, they are grouped under the same task.

When grouping events by tasks, some odd orphan events may be discovered, e.g. *"the user took some papers (hotels, Paris) from the printer and threw them into the waste basket"* (in Fig. 3, the event straight after the lunch). They are usually segments of larger goals that extend beyond the observation session. Usually, the user can explain the goal of such an event or action, e.g. *"Ah, those papers. I printed them yesterday morning when I tried to find a suitable hotel in Paris, but in the afternoon, my boss told me that my trip will be cancelled due to changes on the project schedule."* To document as comprehensive a set of tasks as possible, we create titled pages for the orphan events as well as for the larger tasks.

Descriptions and Data Samples

To provide context, we connect the textual descriptions to the corresponding events on the timeline. The master timeline is visible on each page of the document to provide an overview as well as a structure for details [5, p. 285-305]. This visualization helps the reader construct an appropriate mental model of the situation as a whole. The reader can easily follow a single task from the first to the last action, and connect these to the bigger picture. For each task, the description of highlighted events with data samples is essentially a **task-based scenario**, grounded in real user data.

To document the use of physical artifacts, we insert artifact photos in context with task-based timelines and text descriptions. For example, during the preparation of slides, the user looked at a brochure of the previous product version on paper. The brochure is circled in the photo beside the text (Fig. 2).

Interrupts and Overlaps

Each interrupting and overlapping event is described in the context of the current task, e.g. in Fig. 2, the preparation of slides (current task, *Task 1: Product demo at Acme*) was interrupted by an incoming email message and a phone call. From the viewpoint of slide preparation (Task 1), the email and the incoming call are interruptions. However, both of them are important events in other tasks. For example, the email was about company training, and is therefore described in more detail on that task page (Fig. 3, *Task 2: Company training*).

User Verification of the Documentation

One of the advantages of our documentation is the ease of user verification. User verification of documentation is usually done by sending the document file or a hard copy of the document to the user, who reviews it and enters annotations. Alternatively, if there have been many users from a single company, we have a brief informal meeting with each of them to go through their comments. Then we modify the document. On average the verification of data and changes to the document take about an hour per user.

Users are generally eager to see the documentation; they are positively surprised of the number of details the observers have collected, and find it interesting to check, if the events have been connected to the corresponding tasks. They readily point out problems, e.g. *"I didn't make this phone call to solve my email problem but to ask for the tickets… however, lines were busy at the travel agency, and I decided to try again later."*

User verification is kept as objective as possible by basing it on the most detailed level of events that have been observed; the user's only task is to make sure that the events have been connected to correct tasks. If we asked the user to check a summary of the observations or a high-level description of his work flow, his task would require more subjective evaluation, and he would give comments like *"this is not exactly correct… usually I know this beforehand… this sounds too negative."* If for example we have documented that the user made a phone call at 10:23 and his boss came to talk him at 10:25 when he was still talking in phone, the user is not encouraged to create or verify generalizations, e.g. *"my phone calls are very seldom interrupted by my colleagues."* Generalizations may be biased to focus on the last few events, or subjective values, e.g. *"I do not want to say that my colleagues often interrupt me, because I like them."*

UTILIZING THE DOCUMENTATION

The documentation format of observation data has been designed to be usable throughout the project. For **user interface design** (interaction design), we derive the users' goals from observed tasks by interpreting and explaining the tasks. The resulting goals include very similar data to the "brief scenarios" presented in [8, p. 324-326]; a goal part ("task itself") and a status part ("attributes or data elements that are related to the task"). The primary goals will be those that have been deduced from the most frequently occurring tasks.

During the iterative design cycle, the design is **reviewed** by simulating it with the observed tasks and scenarios. In practice we take one goal at a time, refer to the timelines from several observation sessions to identify related tasks and events, and produce a representative scenario. These refined scenarios are similar to Cooper's key path scenarios [7], except that these have been derived from objective observation data, whereas Cooper's scenarios are based on more subjective user interviews. While simulating a scenario, we take into account the pauses between events and overlapping tasks. Reviewing the design with real observation data allows us to try it in a "real world" context during early design phases.

The tasks, goals, and scenarios also form the foundation for **usability testing**. The task timelines are used to create realistic tasks for usability tests of paper prototypes as well as functional systems. The usability tasks are presented in the context of a typical use situation derived from the refined scenarios.

Implementation design and **design reviews** benefit from realistic tasks and detailed situations. Object-oriented system analysis and design using CRC cards (Class, Responsibilities, and Collaborators), for example, is based on realistic situations [11]. In the CRC method, the designed system is reviewed by selecting one representative scenario at a time and "walking through" the design step by step.

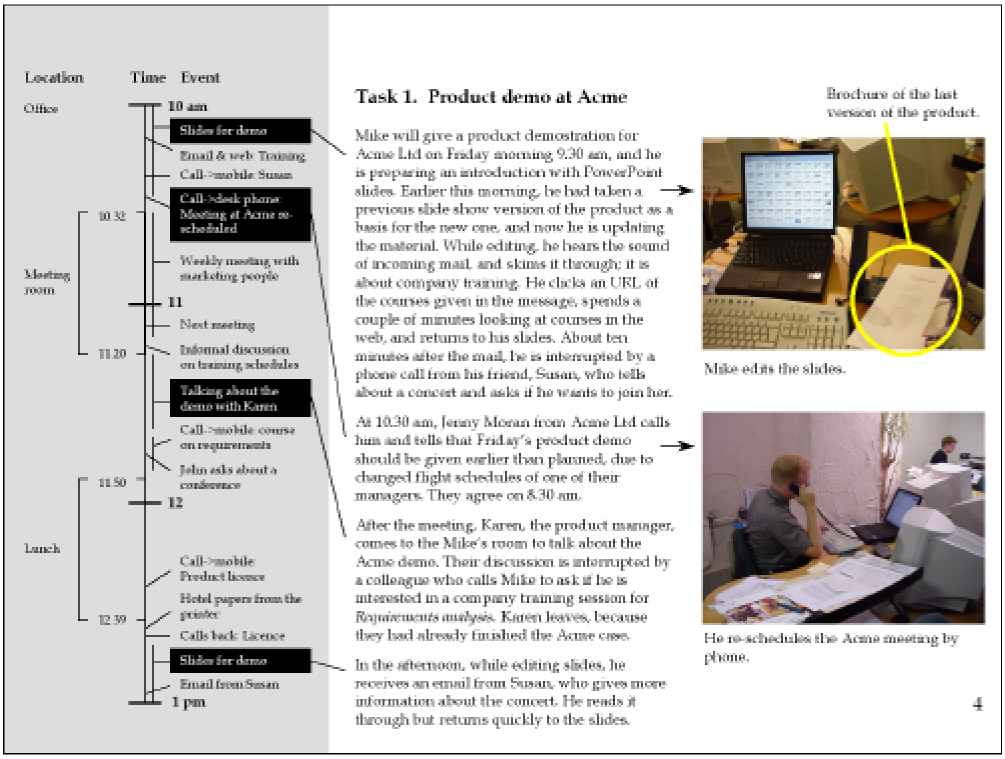

Figure 2. Timeline from the viewpoint of *Task 1: Product demo at Acme.*

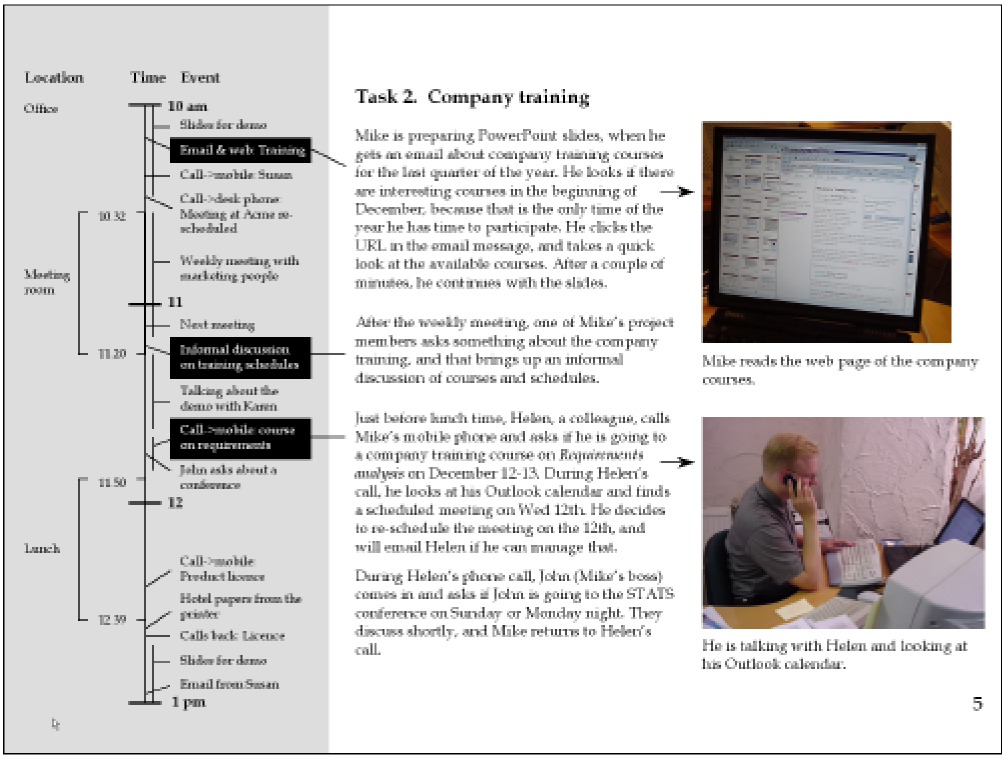

Figure 3. Timeline from the viewpoint of *Task 2: Company training.*

During **system testing**, we use the tasks and scenarios as test cases. Test cases based on real situations can reveal surprising bugs or unwanted features that have been missed during previous reviews. For example, observation of an image archive already put in operation revealed a scenario where the user started to insert a new photo in the archive and was filling in an image card. During fill-in, he answered to some questions of his colleagues working in the same room, went to take a cup of coffee, and talked with another colleague about 15 minutes in the corridor. Then, he came back to his room, saw the half-finished card, and tried to finish it. However, when he pressed the Ok button, he got an error message indicating that the database connection has been closed, and he had to create a new card. If the observation had been carried out earlier, and this realistic scenario or its variation had been run in system testing, the database timeout would have been discovered in advance. Of course, the time limit should have been included in the system requirements, but nobody considered it important during requirements specification; they could not guess that programmers would have to limit the duration of database connections.

The tasks and scenarios are also useful for creating user education materials. They provide the basis for task-based search and for a **user guide** view organized according to users' tasks [8]. The observation report contains examples of typical cases that can be adapted as examples for the user guide.

DISCUSSION

We have applied the observation technique in four different projects with a total of 26 observations. Two of the projects, including eight observations, were reported with a timeline. In the other two projects, the observations were reported either without the timeline, or the observation data was intended to only create coarse summaries of user behavior. However, we noted in the latter projects that even the coarse flow of events was difficult to reproduce a few months later. If the timeline view is not created immediately after the observation, we have found that data will be lost. It is almost impossible to decrypt observation notes even a few days after the session. If realistic task interdependencies are later needed, the only way to obtain them is to create fictional sequences or to arrange new observations. The cost of creating the timeline-based documentation right after the observation seems to be very low in contrast to reproducing it later.

Our observation and documentation method is especially suitable for subcontractors who carry out observations and report them to their customer. In some cases, the subcontractor cannot know precisely what is the intended use of the observation data. Our documentation can be used in various purposes as it is, or it can be processed further into summaries, task hierarchies, or affinity diagrams when required. However, when observing very large organizations, contextual interviews may be a more cost-effective way to start with (see an example in [6]).

FUTURE WORK

The biggest constraint of current DUO is that it only supports views of individual site visits. To support analysis across sites, we could move toward consolidation of timelines in the future. Our timeline resembles the sequence model of Beyer and Holtzblatt [3, p. 96-99], and they consolidate sequence models to model task structure and work strategies within a given segment and task [3, p. 171-178]. Their consolidation method would be a good starting point for analysis across sites.

Currently the timestamps are written manually by the observer, which is error-prone, unnecessary work for him. As a consequence of manual timestamps, also the timeline must be created manually, which is an unnecessary cost.

We have conducted some preliminary experiments using a handheld PC to write notes with automatic timestamps. This appears to be a promising technique, but there are some problems to be resolved. Taking notes with handwriting recognition is too slow for the observer. In addition, we need to draw images and make annotations connected to the notes with lines and circles. We have experimented with existing drawing programs and implemented a simple one of our own. However, even basic drawing response times were longer than acceptable for quick notes in a real observation session. The observer could draw fast enough, but the system could not keep up.

We have also considered implementing a timeline editor that would generate the first draft of the timeline automatically from time-stamped notes. The editor would make it much easier to modify the timeline than the PowerPoint line drawing capabilities we currently use.

In some cases, interrupts between tasks and overlapping tasks could be visualized more clearly by using a Gantt chart notation for tasks. Brown presents a graphical tool that shows a Gantt-like diagram of tasks, which helps to recognize work patterns over time [4]. Similar tools with Gantt-like timeline views have been developed for video analysis, as well, e.g. the Timelines system [9]. Separating the tasks into columns is valuable if we have repeating tasks in the context of a relatively structured workday.

CONCLUSIONS

We have described DUO, a discount observation method for collecting and documenting qualitative field observation data. Instead of videotaping the session and spending enormous amounts of time analyzing the tape, we have focused on gathering objective low-level data (events) with a pen, paper and a digital camera, and documenting it quickly.

Focusing on collecting detailed data increases objectivity of the documentation by guiding the observer to report what really happened during the observation instead of her own inferences or generalizations of the events. In addition, it reduces cognitive load of the observer and enables him to record more details.

To report the full observation data with partially overlapping tasks in a usable format, we have developed a timeline-based visualization that shows an overview of the entire observation period beside the detailed description of every task. The reader can follow the course of each task in a continuous flow, although the tasks are often interrupted during the observation. Because the report describes indisputable low-level events, the content can be verified by the field study participant. The report can be utilized to obtain and generate realistic day-in-a-life scenarios containing a complex mesh of entangled tasks. These scenarios are valuable in user interface design, usability testing, system testing and user manuals.

ACKNOWLEDGMENTS
Foremost, we would like to thank our colleague Asko Saura for valuable discussions and comments during the course of this work, and for the handheld device experiments. Our many thanks to Microsoft, and all the members of the virtual field study team for ideas and new viewpoints. We also thank Heikki Leinonen from Grafimedia, and Rami Saarela from University of Helsinki, for inspiring cooperation.

REFERENCES

1. Bauersfeld K., Halgren S. "You've Got Three Days!" Case Studies in Field Techniques for the Time-Challenged. In Wixon D., Ramey J. (Editors) Field Methods Casebook for Software Design. *John Wiley & Sons*, New York, 1996, p. 177-195.
2. Beyer H. R., Holtzblatt K. Apprenticing With the Customer. *Communications of the ACM*, 38 (5), p. 45-52.
3. Beyer H., Holtzblatt K. Contextual Design. Defining Customer-Centered Systems. *Morgan Kaufmann Publishers*, San Francisco, 1998.
4. Brown D. S. The Challenges of User-Based Design in a Medical Equipment Market. In Wixon D., Ramey J. (Editors) Field Methods Casebook for Software Design. *John Wiley & Sons*, New York, 1996, p. 157-176.
5. Card S. K., Mackinlay J. D., Shneiderman B. Readings in Information Visualization. Using Vision to Think. *Morgan Kaufmann Publishers*, San Francisco, 1999.
6. Curtis P. et al. Customer-Focused Design Data in a Large, Multi-Site Organization. *CHI '99 Conf. Proc.*, ACM, New York, 1999, p. 608-615.
7. Goodwin K. Goal-Directed Methods for Great Design. *CHI 2001 Tutorial Notes*, ACM, 2001.
8. Hackos J. T., Redish J. C. User and Task Analysis for Interface Design. *John Wiley & Sons*, New York, 1998.
9. Owen R. N., Baecker R. M., Harrison B. Timelines, a Tool for the Gathering, Coding and Analysis of Temporal HCI Usability Data. *CHI '94 Conference Companion*, ACM, New York, 1994, p. 7-8.
10. Sanderson P. M., McNeese M., Zaff B. Knowledge elicitation and observation in engineering psychology: MacSHAPA and COGENT. *Behavior Research Methods, Instruments, and Computers*, 26, 1994, p. 117-124.
11. Wilkinson N. M. Using CRC Cards: An Informal Approach to Object-Oriented Development. *SIGS Books*, 1995.