Lessons Learned: Using Contextual Inquiry Analysis to Improve PDA Control of Presentations

Karen Cross, Adrienne Warmack, and Brad Myers
Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA 15213
{kcross@andrew.cmu.edu, asw@cs.cmu.edu, bam@cs.cmu.edu}

ABSTRACT

Contextual Inquiry is a method developed by Beyer and Holtzblatt for grounding design in the context of the work being performed. In this paper, we adapted the method successfully for analyzing pre-existing videotaped presentations. Our goal was to find improvements for a slide presentation program currently in development, called 'SlideShow Commander.' Contextual Inquiry provided us with meaningful data on the structures and typical problems found in presentations, on which we based our design ideas. We then further analyzed the Contextual Inquiry data, beyond what Beyer and Holtzblatt suggest. This new step provided a means to prioritize the design suggestions, as well as a way to defend the potential commercial usefulness of the software. Deciding upon the value and direction of further effort is essential for software development; by using our adapted form of Contextual Inquiry, we were able to make and defend these decisions.

Keywords

Contextual Inquiry, Personal Digital Assistant, Palm Pilot, Usability Evaluation Method.

INTRODUCTION

The Pebbles SlideShow Commander [8] uses personal digital assistants (PDAs), such as the 3Com Palm Pilot and Windows CE devices, to control a presentation in Microsoft PowerPoint running on a PC. It provides a handheld-means for a speaker to navigate through PowerPoint slides, view notes associated with the current slide, view the list of slide titles, and point or draw on the screen using the PDA. The current system uses either a serial cable or infrared (IR) to enable two-way communication between the hand-held device and the PC. Future versions will take advantage of the improvements in IR and radio technologies for wireless communication between the PDA and PC.

The Pebbles group designed and implemented the current version of SlideShow Commander without using any formal evaluation of the domain for which it was intended — public presentations — or the usability of its proposed design. Further development and assessment of the software, however, was hampered by the difficulty of studying the type of impact of the software on the particular domain of formal presentations.

Submitted for Publication

We decided not to design a formal experiment to assess the usability of SlideShow Commander because of the inability to control conditions. Presenters must be familiar with both the content of the talk they are giving and the software to be tested. Otherwise, any difficulties with information flow affected by the software would be overshadowed by more severe presentation difficulties. If we relied on participants to use their own talks, the results could be confounded with order effects. Speakers improve with practice, navigating across familiar slides more easily, referring to notes progressively less, and incorporating questions occurring in front of practice audiences into the content of the talk. Even if we could balance for order across subjects, each talk is so individualized that comparing across subjects seemed like a poor measure of improvement. Thus, the nature of the domain made the experimental approach less likely to produce authentic and meaningful results.

Further, the nature of the software itself made quantitative assessment using any approach difficult. The impact of the software could potentially be an absence of sporadic interruptions in communication, rather than the presence of any positive affect. Initially, we had little idea of the common styles of presenters – their reliance upon the Notes page in Power Point, the frequency and means by which they changed slides, or their use of the cursor to point or draw on the screen. Background literature searches revealed a large quantity of information about how one *should* give a presentation (e.g. [4][5]), but little on what people *actually* did. Without a clear notion of genuine presentation styles, it was difficult to determine a clear measure of the potential impact of the software.

In order to gain an understanding of the ways people actually use PowerPoint, and to measure potentially subtle effects, we decided to study the domain both with and without the software. The method that seemed the best for this goal was Contextual Inquiry(CI), a technique originally developed by Beyer and Holtzblatt [1][2]. This method recommends observing work as it occurs in its own context, and using a graphical modeling language to describe the work process and to discover places where technology could overcome an observed difficulty. We chose this method because it would provide actual data about the detailed structure of the presentations themselves. This would, in turn, help us to: 1) predict how the proposed software would impact presentations; and 2) guide future development of the software.

CI, as advocated by Beyer and Holtzblatt, centers upon live interviews with current or potential users at the workplace, based on previously determined foci. We had been taught the technique from graduate lectures in combination with a textbook by Beyer and Holtzblatt[1], and had used the technique exactly as Beyer and Holtzblatt recommended in several class projects. However, for our study we had access to a large library of previously videotaped presentations. Rather than discard this valuable (and easily available) resource, we decided to adapt Beyer and Holtzblatt's technique.

Unfortunately, a literature search revealed an abundance of articles about how to perform the method (e.g. [1][2]), but not as much on the actual experience of normal, non-expert practitioners using it in the field. The best reference we found for these case studies [9] provided some insights into the variety of domains to which the technique can be successfully applied, but not many practical suggestions that we could specifically apply to our domain. We decided to use the technique nonetheless, agreeing with Beyer and Holtzblatt's argument on the importance of context, and the usefulness of their graphical representation of the information from the work – what they call 'work models.' The advantage of a live CI is that you can ask questions and prompt for explanations. However, this would not be appropriate for observing presentations, where an uninterrupted observation is actually preferable. In cases where interruptions are a distraction, Beyer and Holtzblatt recommend taking copious notes and interviewing the participants immediately after the observation period. Unfortunately, we did not have access to presenters immediately after their talks, so we decided to use what we had and piece together motivations and goals from the evidence of the videotapes themselves. These inferences were fairly easy to make based on the verbal comments and behaviors of the presenters when the problems occurred, although we may have lost some information about personal frustrations and desires.

We further differed from Beyer and Holtzblatt's method by performing additional analyses based on the problems identified in our work models. Because we had the advantage of videotape, we were able to measure the duration of each problem (where applicable) and rate the severity. These ratings enabled us to build evidence that the software authors were able to use to prioritize their development efforts.

We used our adaptation of CI to study a varied selection of presentations that did not use SlideShow Commander ("pre-software talks"). From this analysis, we produced sets of models that gave us an idea of the flow and characteristics of presentations we were studying. Further, these models revealed interruptions in the talks, henceforth called 'breakdowns.' Many of these breakdowns – including the most frequent ones – could potentially be remedied by SlideShow Commander. By analyzing the breakdowns, we found evidence that at least potentially, the software would produce smoother talks with fewer difficulties. To confirm this analysis, we introduced the software into two practice talks given by novice users, ("post-software talks") and performed CIs on these talks to see if the communication was indeed smoother. We complemented this analysis with a questionnaire designed to assess the user satisfaction of the software.

We found that CI using pre-existing videotapes produced an abundance of data that we used to develop our design ideas. Because we had videotapes, we saw problems that a single live viewing while taking copious notes would not have detected. Further, the videotapes allowed us to return to earlier talks to concentrate on detecting subtle problems that were only revealed in later tapes. Finally, the further analyses we performed on the severity and duration of the breakdowns, which could not have been determined without using videotapes, gave us the evidence with which we could convince the software developers of the importance of our design suggestions.

In this paper, we describe the technique we used, the results from analyzing pre-software talks, the confirmation of the results using the post-software talks, and our evaluation of the method we used.

METHODS FOR "PRE-SOFTWARE" TALKS

Types of presentations

Nine academic talks presented at Carnegie Mellon University between May 1998 and June 1999 were analyzed by videotape. In this paper, the terms 'presentation' and 'talk' will be used interchangeably to mean talks given with or without interruption to an audience, followed by a question and answer period. All of the presentations we examined through CI covered various topics in computer science. Presentations can be classified on several dimensions: remote or face-to-face talks; formal, informal, or practice talks; demonstrations, experimental results, or theoretical perspectives; and multiple or single, male or female, presenters. We selected talks that provided as wide a cross-section of these types as possible.

The description of each talk in terms of its group, task, and context characteristics is summarized in Table 1. These characteristics were selected from the analysis of group support systems presented in Nunamaker et. al. [7]. Values for some of these characteristics were obvious from the presentation. Others — the task, group composition, time pressure, and evaluative tone for each talk — were determined by analyzing the number and types of questions from the audience, as well as the time spent on demonstrations compared to straight lecture. PowerPoint was used in 7 of the 9 talks — these presentations contained an average of 35 slides each. One of the talks (CI2) was a formal presentation by a nationally renowned person in the computer science field to members of Carnegie Mellon University in a large lecture hall. Two talks, CI4 and CI8, were predominantly aimed at presenting demos of new applications and

	Presentations								
	CI1	CI2	CI3	CI4	CI5	CI6	CI7	CI8	CI9
Group Characteristics:									
Remote audience	No	No	No	No	No	Yes	Yes	Yes	Yes
Composition	Hierarchy	Hierarchy	Peer	Peer	Peer	Peer	Hierarchy	Hierarchy	Hierarchy
Audience size	<25	>50	25-50	25-50	25-50	<25	<25	<25	<25
Task Characteristics:									
Main goal of talk	Latest wrk	Visionary	Exp. res.	Demo	Exp. res.	Latest wrk	Prog. rpt	Demo	Prog. rpt
Context Characteristics:									
Time Pressure	No	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
Evaluative Tone	Critical	Critical	Support.	Support.	Critical	Support.	Support.	Support.	Support.
# Presenters	1	1	1	2	1	1	1	1	1
Gender	M	M	F	Both M	F	M	M	M	M
Power Point Used	Yes	Yes	Yes	No	Yes	Yes	Yes	No	Yes

Table 1. Characteristics of the Presentations Selected for Analysis. ‘Latest wrk’ stands for ‘Latest work;’ ‘Exp. res.’ stands for ‘Experimental Results;’ ‘Prog. rpt’ stands for ‘Progress Report;’ ‘Support.’ stands for ‘’Supportive;’ M’ stands for ‘male;’ and ‘F’ stands for ‘female.’

hardware to a small audience eager to see new technology. Other talks presented progress reports to other members of the same research group. Still others presented experimental and other usability studies to argue for the success of their ideas and designs. Across all of these talks, some presenters were welcomed with a supportive tone, few critical questions, and many shared experiences; others were interrupted frequently by criticisms or contradictory evidence.

Four of the talks analyzed (CI6-CI9) were presented simultaneously to a remote audience communicating over NetMeeting software and live video-conferencing, and to a local audience. The remote audience was located in Europe; the local audience was in the United States. The other six talks were presented locally only. Each presentation was divided between a 30 to 45 minute lecture period, and a question and answer period that ranged from 15 to 30 minutes in length. Both the lecture and the discussion periods were modeled using CI.

Contextual Inquiry

We created work models using the method of CI as described by Beyer and Holtzblatt [1][2]. In our study of formal presentations, ‘work’ consisted of giving a public presentation. As part of this method, we drew four types of work models for each presentation: sequence, cultural, flow, and physical. As Beyer and Holtzblatt explain, **sequence models** show “the detailed work steps necessary to achieve an intent;” **cultural models** illustrate “constraints on the work caused by policy, culture, or values;” **flow models** represent “the communication and coordination necessary to do the work;” and **physical models** illustrate “the physical structure of the work environment as it affects the work” [1, p. 86]. The final work model described by Beyer and Holtzblatt, the **artifact model** — a detailed view of the tools used by people for their work — was excluded because of the limited view we had of the user’s specific behaviors on their PC from the videotape.

While creating each model, we identified breakdowns in the flow of the information between presenter and audience. ‘Breakdowns’ were considered to be any interruption, confusion, or awkwardness in the flow of information between the presenter and audience. We used the breakdowns to inspire our design ideas. So far, except for the use of videotapes, we conformed to the proscribed method of CI.

At this point in our analysis, however, we added a final step to Beyer and Holtzblatt’s procedure. We took our master list of breakdowns, and classified each breakdown according to the severity ratings shown in Table 2. Breakdowns were judged to be the same across different presentations if they had the same potential fix. We also recorded the duration of each breakdown, to the extent possible. We then analyzed this master list to understand the severity and frequency of these problems, and the impact of the software we were analyzing.

RESULTS – PRE-SOFTWARE TALKS

Common characteristics

Our work models identified a variety of presentation activities and breakdowns, many of which were overlooked upon first viewing the presentation videotapes. For example, it was not until an audience member specifically asked for a reference to be written on the board (in the fifth talk we examined) that we realized the difficulty most audience members must have when references are given only verbally, or skimmed over in slides. None of the talks we studied used handouts, so audience members had no way to find the references after the talk. Once we noticed this breakdown in a single talk, having videotapes allowed us to return and look at the other tapes and notice places where this problem had been overlooked.

When the same types of models were compared across different talks, as Beyer and Holtzblatt recommend, common structures and problems were revealed. The roles of host, technician, advisor, and fellow research member were repeated across many of the talks. Audience members wanted information as clearly and as completely as possible. Presenters wanted to impress the audience, but they also wanted to stimulate interesting discussion and help their audience understand the material.

Rating	Explanation
5	Breakdown *cannot* be resolved. The presentation cannot be continued and must be postponed until later.
4	Parts of a presentation *cannot* continue. The presenter omits part of the presentation.
3	Breakdown causes the presenter to alter the presentation or style of speaking to continue. Causes serious delay in presentation (at least 8% of the speaking time, or five minutes in an hour presentation).
2	Breakdown causes disruption to the flow of the presentation.
1	Breakdown causes minor inconvenience to the speaker or audience.

Table 2. Severity Ratings of Breakdowns

Physical models for each presentation described the arrangement of the room, the positioning of the presenter and audience, and difficulties that arose because of these physical properties. Due to the nature of communication over a distance, the talks where audience members were distributed across two locations required a different physical arrangement from the purely local talks. In particular, remote talks used more artifacts, including cameras, screens, and sound control boards. All of the presentations, however, used a seated audience and a standing presenter; at least one screen that showed the changed slides, started demos, operated the VCR, and spoke. Physical breakdowns included presenters tripping over cords and having difficulty reaching controls for VCRs or slide advancement.

Sequence models illustrate the basic sequence of activities observed in a presentation. The steps include showing slides, demos, or video clips at appropriate times; and responding to questions from the audience. Examples of breakdowns we observed in the sequence models included applications not launching successfully, or presenters making errors while navigating either forward or backward through their slides.

Flow models reveal the spread of information across various artifacts and people. The flow models for the talks showed that the flow of information was often interrupted, either briefly (for a few moments), or significantly (for up to 10 minutes during the presentation). Further, some talks, in particular the remote ones, were overwhelmed with artifacts: multiple electronic devices that the speaker and technicians had to manipulate. Breakdowns we observed in the flow models were revealed most frequently when the information flow from one artifact to the presenter or the audience was interrupted, such as when extraneous application windows blocked part of the screen, or when sound controls were not adjusted properly.

Cultural models illustrate the wants and desires of participants, and how different roles interact. The presentations under study contained similar cultural models: in each, there were two main roles, the presenter and the audience. The presenter desired feedback, support, and questions; the audience desired clear and organized information. Each model, however, also included one or more other subsidiary roles: the technical support; the host; advisory committees; and members of the presenter's immediate research team. All of the people fulfilling these roles helped to make the communication between the audience and the presenter as smooth and as informative as possible. Most breakdowns in cultural models occurred when the desires of the presenter to give information, and the audience to receive information, were obstructed. For example, the behavior of several presenters indicated a desire to be near the audience, but this conflicted with the need of the audience to receive visual information from the slides and for the presenter to stay out of the line of sight of the screen.

Breakdowns

Thirty-eight unique types of breakdowns were found in the presentations by using Contextual Inquiries, with a total of 229 instances of breakdowns observed. Breakdowns were present during an average of 8.7 minutes of each talk. The most frequent breakdowns are summarized in Table 3. Each talk had an average of 34 instances of breakdowns, ranging from minor (Severity = 1) to moderately severe (Severity = 4).

The most frequent breakdown, as shown above, was the physical awkwardness of changing slides. Six out of nine presenters walked to one spot to talk, then turned and walked to a distance typically 3 feet away, positioned themselves, advanced slides using their PC, and then returned to the original spot from where they chose to talk. Often, the PC was on a low table, or difficult to reach, compounding the problem. This procedure wasted between 30 and 78 seconds during each talk in which it occurred,

Description	Model	% of Talks	Count	Average Severity	Average Duration (each time)
1. Changing slides is difficult and awkward because of the placement of the mouse or laptop.	Physical	67	166	1.2	2 sec
2. Presenter loses track of time, must ask for verbal update.	Sequence	44	6	1.5	55 sec
3. Reference provided is incomplete or skimmed over; audience members would be unable to find it after the talk.	Cultural	44	6	1	19 sec
4. Camera view is unclear or pointed at wrong information.	Flow	33	3	1.7	60 sec
5. Audio level for demos is not set correctly.	Flow	33	3	2	46 sec

Table 3. The most frequent breakdowns observed in talks. '% of Talks' is the percentage of talks in which the breakdown occurred at least once. Count is the total number of instances of the breakdown observed across all talks. Severity is the average severity rating across all instances of the breakdowns. Average Duration is the average length of a single instance of the breakdown.

with an average of 48 seconds. Another presenter found a less time-wasting solution: staying next to the slide control throughout the lecture part of their presentation, and then moving to a spot away from behind the podium and closer to the audience for the duration of the discussion period.

The second-most frequent breakdown was an inability of presenters to keep track of time or be aware of how much time they had remaining. Six of the presenters asked an audience member for a time check at some point during their lectures.

None of the observed breakdowns reached the highest severity — a permanent and premature end to the talk. However, three different talks had major breakdowns, requiring significant portions of the talk to be skipped. One had a demo that could not be shown because the PC lacked Shockwave. Two of the remote talks contained significant periods of time when the remote audience could not read the presentation slides, because of an unfocused camera and problems with the settings of the NetMeeting program.

SLIDESHOW COMMANDER

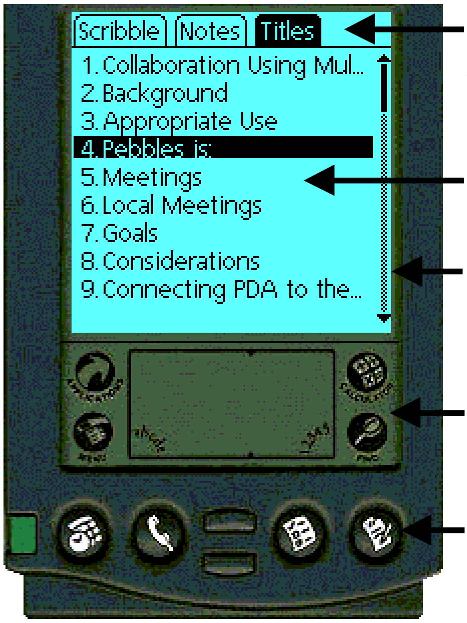
The software we evaluated, SlideShow Commander, is a PDA-based application designed to aid in the delivery of PowerPoint presentations. It provides tools for presenters to navigate among slides, access presentation notes, and draw and gesture on the displayed slide. A more complete description of the software can be found in [8].

The initial version of the application we studied, version 2.3, contains three main views: 'Notes,' 'Titles,' and 'Scribble' (Figure 1). The 'Notes' view displays the textual content of the PowerPoint Notes for the current slide. The 'Titles' view displays a list of the slide titles in a presentation. The presenter can navigate to a slide by tapping the stylus or a finger on the title. The 'Scribble' view allows the presenter to use the stylus to point or draw on the screen. Within any of these views, a presenter equipped with a PDA running SlideShow Commander can move linearly forward or backward through a series of slides by issuing "next" and "previous" commands on the PDA using simple gestures with a stylus. Finally, each of the four physical buttons is assigned to different modifier keys: <control>, <shift>, <alt>, <func>. These physical buttons could be used to send specific commands to the PC, such as F1, which displays a help screen.

RESULTS

After we analyzed all of the "pre-software" talks, we took the completed master list of breakdowns and determined which could have been eliminated using version 2.3 of SlideShow Commander; and which inspired us to design both short-term and long-term improvements to the software (version 2.4, 3.0, and x.0, see Table 4).

The initial version of the software we examined, 2.3, could have prevented certain breakdowns in the "pre-software talks." For example, we observed that presenters wasted time walking back and forth from behind the podium to the screen or audience, and returning to change slides (breakdown #1 on Table 3). Each time they returned, they had to reposition their hands on the keyboard, finding the correct 'forward' or 'backward' key. The original version of SlideShow Commander we studied, 2.3, would have prevented this breakdown by allowing the user to change slides away from the PC using the PDA. Further, the frequency of this breakdown made us realize that changing slides was a key goal for presenters, and should be supported by using physical buttons on the PDA, rather than requiring stylus gestures to advance. Thus, as an immediate fix to the software, we recommended that two of the hard buttons should be set as a default to 'next slide' and 'previous slide'. The developers took our suggestion and implemented this functionality in version 2.4 of the software.



1. Collaboration Using Mul...
2. Background
3. Appropriate Use
4. Pebbles is:
5. Meetings
6. Local Meetings
7. Goals
8. Considerations
9. Connecting PDA to the...

Tabs for switching between views.

Ordered list of slide titles for access to slides.

Scroll bars (the center physical scroll buttons also function)

Graffiti Stroke Pad

Physical buttons

Figure 1. Titles View of SlideShow Commander v. 2.3

Other breakdowns suggested important changes to the software that could be made in the short term. For example, many presenters lost track of the length of their talk, and had to ask verbally for the time remaining (breakdown #2 on Table 3). Not only was this an interruption in the flow of the presentation, but we knew from our own experience that it makes the audience aware of the time, and start looking at their own watches, further decreasing the attention they are paying to the presenter. If a presenter knows that she only has 30 minutes in which to talk, she should be able to set the timer to count down from 30, so she knows when she is near the end of her allotted time. The authors of the software implemented this suggestion in version 3.0 of the software.

Other breakdowns that we found suggested long term directions of development. For example, in every presenter's first attempt to play sound either locally through the PC or transfer it via NetMeeting software to a remote audience, the sound was not adjusted properly. One long-term development would be to allow the PDA to change the volume, and perhaps other controls, on the PC. The long term developments, combined under version 'x.0' in Table 4, have not yet been implemented by the software authors.

We used our measurements of the frequency and severity of breakdowns to argue for the importance of our design ideas (Table 5). Although the most frequent breakdowns are fixed by version 2.3 of the software, the most severe breakdowns (severity = 4) would not have been prevented by using either current or any foreseeable future version of SlideShow Commander. These breakdowns involved missing software on the laptop, wrong hardware for demonstrations, and remote audiences not being able to read the contents of slides using the cameras and NetMeeting. However, these breakdowns were rare – each being unique to only one presentation. We discovered that it is the small but common breakdowns in communication flow that would best be served by SlideShow Commander.

Version	Description
2.3	Initial version of SlideShow Commander.
2.4	Addition of flexibility to assign hard buttons to various PowerPoint controls, such as 'advance' and 'previous' slide.
3.0	Addition of a timer, which can display the time counting up or down, measure the duration of a presentation, and save times for later reference.
x.0	Future long-term developments, such as providing a way to control other devices such as VCRs and Projectors in conjunction with the PC; and including a 'task switcher' to enable fluid transfer between one running application and another, and launching of additional applications.

Table 4. Different versions of SlideShow Commander.

Version	Average % of Talks	Count (Breakdowns Prevented)	Average Severity	Total Time Saved Over All Talks
v. 2.3	20	178	1.5	8 min, 12 sec
v. 2.4	0	0	0	0 sec
v. 3.0	21	19	1.25	2 min, 1.8 sec
v. x.0	24	15	1.7	43 min, 9 sec
None	13	17	2.1	24 min, 31 sec

Table 5. Improvement from SlideShow Commander. All data refers to the number of breakdowns observed in 'pre-software' talks that would not have occurred had the presenter used the particular version of the software. 'None' refers to breakdowns that occurred in the presentations but were outside the bounds of current or future versions of the software.

Introduction of SlideShow Commander into presentations

We performed a preliminary test to confirm our CI-based predictions. To do this, we analyzed two presentations in which presenters used SlideShow Commander to control a PowerPoint presentation. Both presentations were practice talks in which the presenters were practicing before peers and advisors for an upcoming formal presentation. The first talk, T1, was given by three presenters, two of whom used version 2.3 SlideShow Commander to point, access their notes, and advance slides (the third presenter was only responsible for the demo portion of the talk). The second talk, T2, was given by a single presenter who used version 2.4 of SlideShow Commander. All of the presenters had prior experience with delivering PowerPoint presentations, and their self-rated PowerPoint presentation experience levels ranged from intermediate (5/10) to expert (10/10). None had used SlideShow Commander before. While all of the presenters had some prior experience using PDAs, their self-rated experience level on PDA use ranged from beginner (2/10) to expert (10/10). All of the users had a brief (20-30 minute) introduction to the software and the opportunity to practice with it several days their presentation. We analyzed their practice talks using CI as well as by questionnaires designed to assess the usability and satisfaction of the various features in the software. The questionnaires asked them to rate the ease of use of different features of the software on a scale of 1 to 10 (where '10' was 'easy to use'). They were also asked to rate the importance of each feature to them on a scale of 1 to 10 (where '10' was 'very important').

The only additional functionality that version 2.4, used in T2, contained over version 2.3, used in T1, was the assignment of the "forward slide" and "backward slide" commands to the PDA physical buttons (see Figure 1). In version 2.3 of the software, these four buttons were assigned to special application keys. None of the "pre-software" presenters that we observed, however, did anything more than start the presentation, move forward and backward, and view the 'thumbnail' view of the slides. While the ability to navigate through slides in included in version 2.3, we observed that the speakers in T1 had some difficulty with performing the stylus gestures. This observation provided additional support for the most immediate fix requested — assigning the physical buttons to promote a quick, easy way to navigate among slides.

The talks in which the presenters used SlideShow Commander revealed absences of: 1) awkward leaning to reach the slide control; 2) physically being constrained closer to the PC away from the audience, and 3) navigational difficulties that had been observed with the non-SlideShow Commander talks. Presenters stood straight throughout the presentation, rather than repeatedly bending down to the laptop, as had been observed in the 'pre-software' talks. The presenters also chose to speak physically away from the PC. This provides some confirmation of our earlier observation that a PC tethers the presenters away from their desired speaking location. None of the presenters advanced to the wrong slide by pressing the wrong buttons, which occurred in some of the non-SlideShow Commander talks.

Presenters rated the ease of use and importance of the software reasonably well (mean ease of use = 6 and importance = 5, where 10 = 'very easy/important'), and were most satisfied with the slide navigation feature of the software. The presenters' rating of slide navigation as 'easy to use' went from 7.5 (version 2.3) to 10 (version 2.4).

The different speaking styles of the presenters affected their ratings of the importance of the various features. For example, two presenters admitted that they were unfamiliar with their own slides, and relied heavily on the Notes feature of the software. Both of these presenters rated the importance of this feature as high (mean = 9). These results provide additional information that could be used in marketing the product and comparing it to its nearest competitors, which lack the 'Notes' feature (e.g. [3][6]).

EVALUATION OF THE METHOD

CI analysis of talks that did not use SlideShow Commander provided a successful means for predicting the usefulness of the software, as well as predicting what types of improvements should be made to the software. By using videotapes, we were able to make measurements that simple notetaking of observations (as recommended by Beyer and Holtzblatt) would not have made possible. We capitalized on the ability to return repeatedly to the videotape for confirmation of the identity, duration, and severity rating of the breakdowns. We used this data, as shown above, to more systematically classify each breakdown and its impact.

With our data, we could also assess the usefulness of drawing each of the types of work models (Table 6). We relied heavily upon the Beyer and Holtzblatt book [1] to refresh our knowledge of the technique. Certain models, however, were more difficult to draw than others, though they all became easier to draw with practice.

Physical models were the simplest models to create, requiring a view of the layout of the room and attention to overt indications of the smoothness, direction, and frequency of movement of the presenter. Rough drafts of these rarely took more than ten minutes to draw. The physical models, however, were the least useful in terms of identifying surprising breakdowns that we had not seen before and would not have predicted. For example, the breakdown where a presenter tripped over a cord was obvious from simply watching the tapes (and hearing the chuckles from the audience), and thus did not need a physical model to make it evident. Because of its low cost in terms of time and attention, we recommend drawing these models, but not spending much time on the fine details, unless this has a specific use for the type of software or hardware being developed.

Sequence models were also fairly easy to create and draw, given our heavily sequential domain. Although the steps were easy to identify, the modeling process still took a long period of time (up to an hour) in order to record interesting quotes and close details of the sequential process. Our notes for the sequence models also became useful when returning to the tapes later in the modeling process while trying to find (or re-confirm) the presence of particular breakdowns. This model type might have been more difficult to draw in other domains in which more work occurs in parallel, such as studying secretarial work with its boundless interruptions. The sequence model was, with our domain, the type that provided us with the most detailed, defensible breakdowns.

Flow models were the most complex of the models we drew. They required careful attention to the implicit ways in which information was transmitted between the audience, the presenter, screen, laptop, and other sources. Not only did they consistently require our full attention, but they were packed with information, and as a result required substantial effort to 'clean up' for a meaningful presentation of the data. However, they revealed the immense complexity involved in giving a smooth presentation that we would not have predicted given our own informal beliefs about this domain. A presenter must keep track of the many directions information can come from at all times, make sure applications are launched on time and without difficulty, ensure that the correct slide is on the screen, and continue to speak at the right pace. Difficulties in any of these activities caused breakdowns that were often highly relevant to the software we analyzed. For example, cameras and projectors being out of focus, and sound being incorrectly adjusted, led directly to one of our long-term suggestions for the software – the ability to control the audio/visual aspects of presentations.

Cultural models were the hardest to draw conceptually. Determining the desires and expectations of the presenters and audience required careful attention to overt behaviors, and the inference that these behaviors indicated certain mental states. We may have had difficulty because of the inability to interview presenters after their talks. Without this resource, we occasionally felt we were doing more inferring than observing, and were not absolutely confident of our cultural claims. Fortunately, often the presenters and audience members expressed their frustrations and desires during the course of the talk, probably because most of the talks were within a relatively small academic community that knew each other. Information from the cultural models would probably have been more detailed, with a greater number of desires and frustrations revealed, had we been able to interview the presenters after their talks.

FUTURE WORK

In our 'pre-software' contextual inquiries, we acquired a list of design suggestions inspired by our observations. By using our severity, duration, and frequency ratings, the value of further immediate development was ascertained. We also were able to present to the developers the strong long-term potential of a more general PDA presentation tool that would allow for control of PC sound, projection focus, and VCR operations from the PDA.

Further, observing the interaction between audience and presenter has given us ideas for the functionality of an 'audience mode' for SlideShow Commander. Currently, multiple PDAs can control the same presentation, but there is no way for one PDA to block input from other PDAs, so audience members could disrupt presentations. In the future, we hope to help design the interface of this mode.

Model	Average % of Talks	Count	Average Severity	Total Time Saved Over All Talks
Physical	33	169	1.2	5 min, 3 sec
Sequence	17	24	2.0	7 min, 21 sec
Flow	17	20	1.9	63 min, 37 sec
Cultural	16	16	1.2	2 min, 38 sec

Table 6. Characteristics of the breakdowns initially detected by different types of models.

CONCLUSIONS

Using CI, we analyzed various types of presentations and observed their common features and breakdowns. We predicted that some of these breakdowns would be remedied with our software, and introduced the software into two talks to test our predictions. These talks supported our predictions, in that SlideShow Commander talks revealed an absence of the most frequent breakdowns. Presenters could advance slides, point, and look at Notes with a minimum of physical movement. The best situations for using videotape are ones in which knowledge about the frustrations, beliefs, and expectations of the user are either openly evident or less important for the particular stage of design. Our adaptation of CI to capitalize on video to confirm and measure breakdowns is a useful addition that others should follow when trying to save time and provide more quantitative support for their design decisions.

REFERENCES

1. Beyer, H. and Holtzblatt, K. *Contextual Design: Defining Customer-Centered Systems.* Morgan Kaufmann Publishers, Inc., San Francisco CA , 1998.
2. Holtzblatt K. and Jones, S. Contextual Inquiry: Principles and Practice. *Participatory Design: Principles and Practice.* Lawrence Earlbaum, New York NY, 1993.
3. Logitech TrackMan Live! Description at http://www.logitech.com/us/trackballs/tb22_100.html
4. Mablekos, C. *Presentations that Work.* Institute of Electrical and Electronics Engineers, New York NY, 1991.
5. Mandel, S. *Technical Presentation Skills.* Crisp Publications, Los Altos CA, 1998
6. MindPath PocketPoint Description at http://www.mindpath.com/Products/pocketpointer.
7. Nunamaker, J.F., Dennis, A.R., Valacich, J.S. et. al. Electronic Meeting Systems to Support Group Work. *Commun ACM 34(7),* 40-61.
8. Pebbles SlideShow Commander. Available at http://www.cs.cmu.edu/~pebbles/
9. Wixon,D. and Ramsey, J. *Field Methods Casebook for Software Design.* Wiley Computer Pub., New York NY, 1996