Available online at www.sciencedirect.com
SCIENCE DIRECT®

Interacting with Computers xx (2005) 1–21

Interacting with Computers
www.elsevier.com/locate/intcom

# Better discount evaluation: illustrating how critical parameters support heuristic creation

Jacob Somervell[a,*], D. Scott McCrickard[b]

[a]*Department of Mathematical Sciences, University of Virginia's College at Wise, 221 Darden Hall, Wise, VA 24293, USA*
[b]*Center for Human Computer Interaction, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA*

Received 5 November 2004; accepted 4 March 2005

## Abstract

This paper describes a heuristic creation process based on the notion of critical parameters, and a comparison experiment that demonstrates the utility of heuristics created for a specific system class. We focus on two examples of using the newly created heuristics to illustrate the utility of the usability evaluation method, as well as to provide support for the creation process, and we report on successes and frustrations of two classes of users, novice evaluators and domain experts, who identified usability problems with the new heuristics. We argue that establishing critical parameters for other domains will support efforts in creating tailored evaluation tools.

*Keywords:* Heuristics; Evaluation; Notification systems; Critical parameters

## 1. Introduction

With the advent of non-traditional computing interfaces, evaluation is becoming more costly and difficult—largely because it is expensive and time-consuming to build or prototype interfaces intended for unique platforms and because it is difficult to devise realistic experiments for interfaces intended to blend into the environment. More broadly, evaluation methods that were effective for desktop interfaces may not be well suited for

---

[*] Corresponding author. Tel.: +1 2763764569; fax: +1 2763764589.
*E-mail address:* ollipevil@hotmail.com (J. Somervell).

0953-5438/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.intcom.2005.03.007

the platform, environment, and user goals and situations for ubiquitous, ambient, non-traditional displays.

For example, suppose you were asked to design and evaluate a large electronic poster to display important news, information, and upcoming activities within your local work area. The goal of the system is to allow the users to go about their daily tasks, stopping to study and discuss with others information and updates on the display when desired. Envisioning platforms and creating designs for such a display might be easy, but how would you assess how well candidate designs might perform? Creation of a prototype end product that relies on expensive and difficult-to-obtain hardware can be costly, and meaningful evaluations of an incomplete and unrealistic prototype can prove challenging as the effects of the system would only be measurable after extensive usage.

As a solution, we turned to discount usability methods, specifically Jakob Nielsen's heuristic evaluations (Molich and Nielsen, 1990). Since the early 1990s, Nielsen has touted heuristic evaluation as a flexible, inexpensive approach that can be employed at any time during design, including with early storyboards and incomplete prototypes. However, recent efforts have suggested that heuristics are poor at providing problem descriptions that capture the underlying user goals (Sauro, 2004; Cockton and Woolrych, 2002)—a problem certain to be exacerbated in the novel uses projected for non-traditional computing platforms.

This paper describes our efforts at creating heuristics tailored to specific system classes—types of systems that share commonalities that make them easy to compare. Evidence from Somervell et al. (2003a) and Mankoff et al. (2003) suggests that system-class level evaluation tools hold promise over more generic tools or even tools tailored for an individual system, due to increases in benchmarking performance and system comparison. We describe our experiences in creating a set of system-class level heuristics for large-screen information exhibits, a type of notification system. We outline our methodology—centered on the notion of critical parameters—in a way that can be applied by others. A comparison of our heuristics to others using a standard usability evaluation method (UEM) comparison technique (Hartson et al., 2001), described in Section 4.1, indicates that our tailored heuristics have measurable benefits when compared to more general heuristics like Nielsen's.

Next, this paper describes two applications of our heuristic set. Section 4.2 outlines the use of the heuristics by novice designers in a classroom setting. We reflect on the experience of the students in applying the heuristics to systems designed by others. Section 4.3 describes the experience of domain experts in using the heuristics to evaluate an early system prototype. While the non-technical domain experts were at first reluctant to comment on the system, the heuristics helped guide their discussion to produce guidance useful in the redesign of the prototype. The paper concludes with lessons learned, and some general conclusions and directions for future work.

## 2. Background and related work

The work described in this paper builds upon the notion of *heuristic evaluation*, a discount usability method devised by Jakob Nielsen and his colleagues. He described heuristic evaluation as a method for inspecting an interface guided by a set of usability principles (or heuristics) to identify problems. The most well-known set of heuristics was created and updated by Nielsen and his colleagues during years of experience designing and building systems (Molich and Nielsen, 1990; Nielsen and Landauer, 1993; Nielsen and Mack, 1994). One representative heuristic from Nielsen's set is:

*Recognition rather than recall*. Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

The complete set of Nielsen heuristics can be found in Nielsen and Mack (1994), and is listed or referenced in many human-computer interaction textbooks (Dix et al., 2004; Hix and Hartson, 1993; Preece et al., 1994, 2002).

More recently, there has been a trend to create heuristic sets for tailored system classes (though no specific creation process has been proposed or adhered to). The typical model seems to be to start with an established set of heuristics or similar guidelines (often Nielsen's), then use theories and experiences to tailor them to a specific system class. For example, Baker et al. modified Nielsen's original set to more closely match the user goals and needs associated with groupware systems (2002), basing their modification on prior groupware system models (the Locales Framework (Greenberg et al., 1999) and the mechanics of collaboration (Gutwin and Greenberg, 2000)). In another example, Mankoff et al. created a new set of heuristics for ambient displays (2003), deriving the new set of heuristics by eliminating some from Nielsen's original set, modifying the remaining heuristics to reflect ambient wording, then adding five new heuristics.

Both of these new heuristic sets were empirically compared to Nielsen's set by the respective authors. Perhaps not surprisingly, the comparisons suggested that the tailored heuristics perform better than the original set. It is important to note that both of these efforts reported creation methods that relied upon what the authors felt were the most important elements for the respective system classes, similar to the approach embraced by the work described in this paper. However, this paper describes a process *not* centered on existing heuristics or guidelines—we worry these may bias the resultant heuristics toward topics not important to the system class at hand. Instead, we focus on an approach centered on *critical parameters*, established and accepted figures of merit that predict or reflect how well a system will perform (Newman, 1997).

William Newman first put forth the idea of critical parameters for guiding design and strengthening evaluation as a solution to the growing disparity between interactive system design and separate evaluation (1997). For example, consider airport terminals, where a critical parameter would be flight capacity per hour per day. All airport terminals can be assessed in terms of this capacity, and improving that capacity would be positive for the airport. Newman argues that by establishing parameters for application classes, researchers can begin establishing evaluation criteria, thereby providing continuity in evaluation that allows us 'to tell whether progress is being made' (Newman, 1997). The key to successful evaluation tool creation requires focusing on the user goals associated with the target system class—the critical parameters for that system class.

By identifying critical parameters for an area of interest, designs within the area can be similarly evaluated, leading to comparisons, reuse, and sustained and incremental progress within the field. This approach was embraced in the study of *notification systems*, interfaces that provide interesting or important information to users who are busy with other tasks (McCrickard and Chewar, 2003; McCrickard et al., 2003). Familiar examples of notification systems include stock tickers, handheld reminder programs, and in-vehicle information systems. Emerging research, particularly in the field of ubiquitous computing, examines how ordinary objects like lamps and fans can be used as notification systems (Greenberg and Kuzuoka, 2000; Ishii and Ullmer, 1997; Ndiwalana et al., 2003). A set of heuristics has been tailored for notification systems, though in a somewhat ad-hoc fashion with no comparison to existing heuristics (Berry, 2003).

Three critical parameters that define the notification systems design space—interruption, reaction, and comprehension—where systems are modeled around high (1) and low (0) values for each parameter. As described in McCrickard et al. (2003), the eight permutations of high and low values for these three critical parameters create unique sub-classes of interfaces. For example, a system model for a business traveler using an in-vehicle information system might have a low interruption value to preserve attention for the primary driving task, high reaction to alert the driver of upcoming turns, and low comprehension as the driver has no desire to learn the city layout. In a contrasting example, a system model for the public large screen display from the introduction might have a high interruption value to attract attention to important news, low reaction as most news items are not critical, and high comprehension to help foster a shared knowledge base within the group.

The second example system described above and in the introduction can be classified as a *large screen information exhibit* (or *LSIE* in this paper), an application that continually runs on a large screen display, providing interesting and useful information. Despite being a sub-class of notification systems, the design of LSIEs is a rich and important domain itself, with applications in academic, military, government, and business domains (Elrod et al., 1992; Greenberg and Rounding, 2001; McCrickard et al., 2002; Russell et al., 2002; Skog and Holmquist, 2000; Abowd et al., 1998; Jedrysik, 2001). It is on this system class that we focused our heuristic creation and evaluation efforts.

## 3. Creating heuristics based on critical parameters

Our goal in outlining the heuristic creation effort was not only to describe our heuristic generation process (originally introduced and more fully described in Somervell et al., 2003b), but more importantly to understand better the heuristic creation process (described in this section) and to examine and reflect on the application of resultant heuristics in actual design situations (described in later sections). Broad, widely used heuristic sets emerged from years of work by established masters in the field (e.g. Molich and Nielsen, 1990; Shneiderman, 1998). The typical approach in tailoring heuristics has been to modify an existing heuristics set to a system class (Baker et al., 2002; Mankoff et al., 2003), leaving questions for researchers wishing to create heuristics of their own. Building on prior efforts, this section seeks to provide guidelines for heuristic creation based on our own experiences.

Before generating heuristics, we feel it is important to decide on three aspects of the design and evaluation approach: the system class, the design technique, and the knowledge storage approach. The system class bounds the design and evaluation space, and by considering these bounds initially, one can facilitate comparison and reuse throughout design. As discussed earlier, we identified critical parameters of high interruption, low reaction, and high comprehension to bound the large screen information exhibit system class.

The design technique we selected was *scenario-based design* (Carroll and Rosson, 1992; Rosson and Carroll, 2002), a popular and widely used design technique that leverages the story-telling nature of scenarios to raise questions throughout the flow of a problem or potential solution. Scenarios capture the stakeholders, setting, goals, decisions, and flow of actions and events that occur during a representative situation. Scenario-based design employs scenarios at all phases of design—from analysis to design to evaluation—to help situate and ground the system development.

Extending from our choice of scenario-based design as our design technique, we selected *claims* as our central knowledge storage unit (Carroll and Rosson, 1992; Rosson and Carroll, 2002). Claims, centered around an artifact of the system, encapsulate upsides and downsides of the artifact in the situation. For example, a shortened sample claim about large screen displays (taken from Payne et al., 2003) is:

Pulsing animations used to reflect information status:
+supports good peripheral detectability;
+can attract attention to new or special items;
BUT
–multiple pulsing items are confusing and difficult to perceive simultaneously;
–prolonged pulsing may be annoying and difficult to ignore

Claims are similar to heuristics in that they provide valuable insight into the design decisions that led to good (and possibly bad) designs. As such, they provide a good starting point for creating heuristics. However, as there can be dozens of claims even for a single system within a design class, and similar claims exist for many of the systems, steps for creating heuristics are needed. The following terse description of our steps reflects many of the key decisions made during the process (a more detailed description can be found in Somervell, 2004).

Identify example systems from the target class. Each system should be representative of the system class, and together the set of systems should encompass all features of the class. Current literature, successful applications, and highly visible interfaces are good places for finding example systems. Well documented systems, particularly those with usage data, are helpful for the latter steps.

In our work, we sought to identify systems that matched (or were intended to match) the critical parameter values important for LSIEs. We included several in-house systems, but were careful to include several developed externally to provide balance to our system set. When necessary, we contacted designers for additional materials or even brief interviews when necessary.

Extract design knowledge from each representative system. In our case, this process could be adopted from the scenario-based design processes, requiring designers to write scenarios and perform claims analysis in developing a system. We used professional papers, email exchanges with developers, system documentation, and other design artifacts to retroactively identify scenarios and claims for systems. We have found that anywhere from three to six scenarios are adequate for capturing the typical usage situations and tasks associated with a given system. (For a system with a greater scope of activities, more scenarios may be needed to cover typically supported tasks.)

Group and label heuristics. As claims are extracted from the scenarios, one needs to keep in mind the underlying psychological impact the design artifact has on the user goals associated with a system. Indicating how the claim impacts the associated critical parameters allows the researcher to group related claims.

Another technique that supports grouping of claims depends on the category in which the claim lies. In other words, different claims may deal with the same type of design element (like color or animation) or task (user feedback or system state information). By identifying the underlying design element, the researcher can group the related claims.

Derive heuristics from the design knowledge. This process requires that design knowledge be extracted from each group, worded in the form of heuristics that represent the collected set of knowledge.

For us, this process entailed inspection of the wordings of the claims to identify the category and classification that led to its inclusion in that particular group. This wording, coupled with the wordings of the other claims in that group, allow creation of one or more statements about the underlying design challenges captured in those claims. These statements about design, our early stage heuristics we call *high level design issues* (Somervell, 2004), have proven to be useful in their own right in guiding design. From this, we synthesized a small, manageable set of heuristics—investigating and analyzing the wordings and relationships among the various issues and identifying similar or common issues as well as issues that deal with similar critical parameters. This led to grouping and generalizing the issues into higher level heuristics. Reference to existing heuristics can help with understanding the level of generality needed in the wording, but care must be taken so that the new heuristics do not copy the model too closely.

An example heuristic that resulted from this process is:

*Judicious use of animation is necessary for effective design*. Multiple, separate animations should be avoided. Indicate current and target locations if items are to be automatically moved around the display. Introduce new items with slower, smooth transitions. Highlighting related information is an effective technique for showing relationships among data.

Note that, like the heuristics in Nielsen's set, it contains a representative opening statement, followed by several supporting and specifying high-level design issues that help a reader understand its meaning. This is one of only eight heuristics in our set (the complete set is available in Somervell et al., 2003b and in Appendix A).

## 4. Validation through use

How do heuristics that emerge from our creation process compare to existing heuristics, and what can we learn about the creation process through examination of the heuristics?

To answer these questions we provide information on three seperate efforts to directly explore the utility of the heuristics and indirectly explore the utility of our creation method. These efforts involve both formal comparisons with existing heuristics as well as instances of use.

### *4.1. Heuristic comparison*

We compared our LSIE heuristics to other types of heuristics to understand how well they perform against the existing alternatives. In addition to our own heuristics, we studied Nielsen's heuristics (Molich and Nielsen, 1990; Nielsen and Mack, 1994) and a set of heuristics tailored to notification systems (Berry, 2003). Using accepted usability evaluation method (UEM) comparison metrics from Hartson et al. (2001), we were able to determine that our new heuristics outperformed the other two heuristic sets through higher thoroughness, validity, effectiveness, and reliability scores.

It should be noted that the comparison study pitting our new heuristics to the more generic versions provided by Berry and Nielsen relied heavily upon the metrics provided by Hartson et al. (2001). We calculated these metrics based on the notion of a *real problem set* (Hartson et al., 2001). Specifically, we calculated the metrics by determining the numbers of problems found by each method in comparison to the real problem set.

We used structured presentation of identified problems in the target systems and asked 21 usability experts to indicate the level of applicability each heuristic held for the problem. This rating was provided through a 7-point Likert scale where the evaluator indicated their level of agreement with the heuristic when asked if the heuristic applied to the problem. An answer of seven would indicate strong agreement (highly applicable), while a one indicates strong disagreement (not applicable at all).

We used three example large screen information exhibits for the test and used pre-identified problem sets to serve as the 'real' problem sets:

The Notification Collage provides a communication mechanism for lab members upon which they can post various types of information, from personal communication to documents and even video clips (Greenberg and Rounding, 2001). Users are often busy with work at their desks but can choose to look up at the NC to check on postings and keep track of lab information.

The Plasma Poster performs similar tasks, but is placed in common areas like break rooms, kitchens, or atriums (Churchill et al., 2003). Users can find information on local events, user postings, and automatically generated content.

The Source Viewer is an LSIE system found in a local television station. Program control managers must ensure proper source switching between commercials and standard program content. This system allows the manager to see all of the upcoming sources simultaneously and thus facilitate source switching.

Each evaluator was randomly assigned to one of the three heuristic sets given the constraint of keeping equal numbers for each set. The evaluators then proceeded to rank the heuristics according to the problems for each of the three systems. System presentation order was completely balanced using a Latin Square ordering. Applicability scores indicate that a problem is ‘found’ by a heuristic if the average rating for the heuristic is greater than 5. The cutoff value of 5 was used because averages above this value indicate ‘agreement’, which suggests that the heuristic applies to the problem.

Thoroughness is found by dividing the number of problems found by a single method (set of heuristics) by the total number found by all methods. In other words, determine the total number of real problems found by all of the heuristic sets, then divide the number for each individual set by this total. Validity is found through a similar division but relies on the cardinality of the real problem set as the denominator instead of the union of problems found. Effectiveness is the product of thoroughness and validity. Reliability is found by calculating the average difference in the evaluators for each of the problems. This represents an accurate measure of the total difference in answers among the evaluators. Alternatively, one could measure the number of agreements among the evaluators, yielding a separate measure of reliability. To simplify calculations, all of the problems across all three systems were grouped together.

Results of this work indicate that the more specific heuristics held the best scores for the aforementioned metrics. There was also a general trend that the more specific heuristics were better suited to the large screen information exhibit system class, evident through the resulting ordering of the methods based on comparison metrics. However, to highlight the strengths and weaknesses of our heuristic creation process, we felt it was important to further examine the utility of the new heuristic set through investigation of instances of use. We examined two classes of heuristic users: novice designers who are learning evaluation methods and domain experts who are unfamiliar with HCI. Because of time concerns and other costs, these classes of users often rely on analytic techniques like heuristics, and examining their use of our heuristics helped us explore their utility and the utility of the heuristic creation process.

### *4.2. Novice designers*

An initial examination of heuristic use was gleened from novice designers: HCI students from an introductory undergraduate class taught during the summer of 2003. This course was beneficial to this research effort in that it provided a test bed for the new heuristic set. As part of the course requirements, the students were required to develop LSIE systems, providing an opportunity for applying the new heuristics in the development process. Student experience with heuristic evaluation was limited to such usability engineering concepts from the course content. The analytical evaluation stage occurred toward the end of the course, after empirical evaluation had been covered.

These students performed heuristic evaluation of several large screen information exhibits using the heuristics developed from the process described in Section 3. The goals of the evaluation were to help the students with the design of their systems (in relation to the class), and to gather feedback on the utility of heuristics for producing redesign guidance (in relation to this research effort). In addition to the evaluations, each student provided a critique in which they could give their opinions on the utility and usefulness of the heuristics for guiding an evaluation of large screen information exhibits.

### 4.2.1. Method

This test was conducted as part of course requirements for an Introduction to HCI class containing 16 students involved in five group project teams. These groups were tasked with creating new LSIE systems which displayed news content from CNN[1]. Each display was required to provide some subset of the daily news, presented on a large screen that would be situated in a lab or common area. There were no restrictions on how the display could look, as long as a user could gain an understanding of the daily news by looking at the display.

Development occurred over a 6 week period, with summative heuristic evaluation occurring in the fifth week. These students had learned about heuristic evaluation and other analytic evaluations through a class activity that used Nielsen's heuristics, but they were not familiar with our LSIE heuristics before the assignment.

These LSIEs were then used by the students in analytic evaluations involving the new heuristics. Each team was randomly assigned to a different team's interface. Each team member then *individually* performed an analytic evaluation on the interface using the heuristics. Once this part was completed, the teams reassembled as a *group* and produced a common problem list for the interface. This common list was a union of the individual problem sets found by each individual team member. These group-level problem sets were then returned to the development team and subsequently used to guide redesign efforts.

### 4.2.2. Results

Several measures were taken from the problem reports and critiques of the method. These measures help to assess the utility of the heuristic set for supporting formative usability evaluation. *Number of problems found* by each team is an early indicator that the method was successful in uncovering at least some of the issues with the various designs. Each team uncovered at least 10 problems, with an average of 16 problems found per team. Fig. 1b shows the distribution of the problems found by team. *Subjective opinion* was gathered from the critiques provided by these novice HCI students. The tone and nature of the critiques was easily discernible through the language and wordings used in their reports. These critiques provide unbiased feedback on the heuristics when used in traditional heuristic evaluations.

The majority of the students felt the heuristics were 'useful' and provided 'much needed guidance for evaluation effort.' In addition, students indicated that the heuristics were 'easy to understand', and application was 'straightforward'. Most of the students agreed that the majority of the heuristics were applicable to the designs they evaluated. As part of the critique, the student gave their *agreement* with the heuristic according to if they felt the heuristic applied to large screen information exhibits. Fig. 1a shows the percentage of students who agreed with each heuristic.

Also, 12 of 16 students explicitly stated that they would have liked to have had these heuristics available during the design phases of their projects. This information was voluntarily provided, as they were not prompted explicitly about this topic. These students indicated they would have used the information in the heuristics as design guidance and felt

---

[1] http://www.cnn.com.

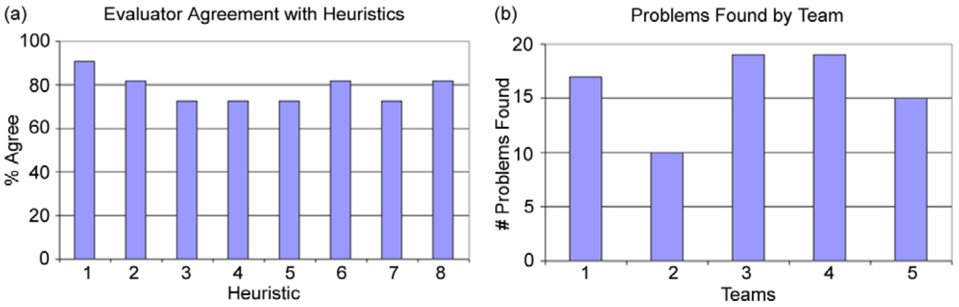


Fig. 1. (a) Percentage of student evaluators who agreed the heuristic was applicable to the large screen information exhibits. (b) Total number of problems uncovered with the heuristics, shown by team.

they would have produced better designs before doing any evaluation had they known about the issues contained in the heuristics. This finding suggests that the heuristics could provide useful information early in the design cycle, as predicted by Newman (1997).

### 4.2.3. Discussion

Clearly, these heuristics provided necessary guidance for the analytic evaluations performed by the novice HCI students. Considering the nature and intent of these particular large screen information exhibits, identifying 16 usability issues is encouraging. In fact, each of the solutions given by the student groups consisted only of a single screen, typically employing animation to show changes to information. Thus, 16 problems identified in these systems allows for substantial improvements to the design.

We acknowledge that the participants in this study were not expert HCI professionals, as is typically used in heuristic evaluation (Nielsen and Mack, 1994). Yet, given the nature and number of the problems found per system, we feel these heuristics provided essential evaluation guidance for the students. As such, the success of this study suggests that the heuristics were sufficient for evaluating a typical large screen information exhibit. The systems used in the evaluation were new and did not have any common design with the ones used in the creation method. This is an important distinction as we have shown that these heuristics are applicable across at least five different LSIEs. Hence, we believe that the creation method we developed produces usable heuristics that can be used in analytical evaluation.

Another interesting use for heuristics comes from the potential for design guidance provided from the heuristic sets. As seen in this study, most students felt these heuristics could serve as design guidelines to aid in the development and creation of the interfaces in the early design stages. This observation is powerful in that these heuristics have a second function beyond simply guiding evaluation effort—to guide design from the start of a project by identifying and illustrating potential trouble spots in the design of large screen information exhibits.

### 4.2.4. Post-analysis of problems

Because this summative-style evaluation occurred toward the end of the project development cycle in a shortened summer session, no significant changes were made to the students' designs. However, analysis of the problem reports can reveal how well the heuristics supported reporting of problems that are related to the critical parameters for the LSIE system class. These students had no prior knowledge of the critical parameters. All they knew about LSIEs was that they involved software running on large display surfaces. Hence, we felt it interesting to see if the problems they find with the heuristics can be traced back to the underlying critical parameters for the system class.

The majority of problems reported by students related to some specific design artifact within the evaluated system. To assess whether a problem related to interruption, reaction, or comprehension, the wordings of each problem were considered in relation to the artifact described therein. For example, the following problem refers to specific artifacts in a design:

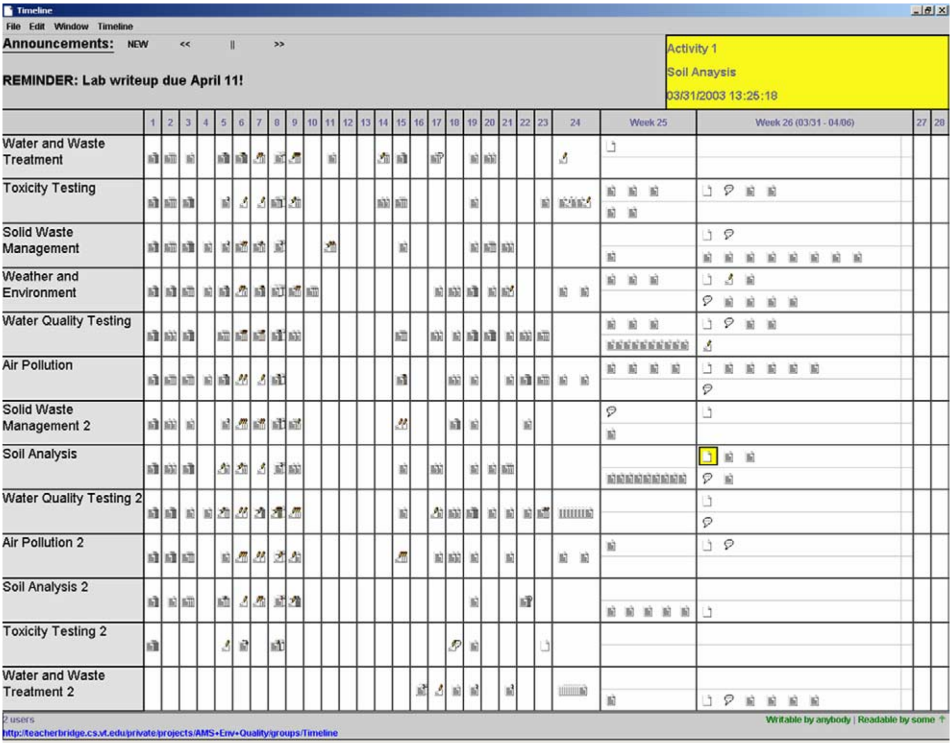
Temperature does not stand out well against the blue background.

Clearly this problem describes the representation of specific piece of information (the temperature), as well as the problem with that artifact (does not stand out well). However, inspection of this problem suggests a connection to the critical parameter of comprehension. Why? Assessment of the problem description implies that it will be difficult for a user to read the current temperature, hence, the user would experience decreased understanding of that information, or a lower comprehension. In addition, a user may experience increased interruption because it takes more time to decode temperature from the display.

Many of the problems reported by the students relate directly to one or more of the critical parameters. The 16 students reported a total of 183 problems across the five systems. This total includes multiple instances of the same problems because they were identified by separate evaluators. Of this 183 problems, 88 were related to the critical parameters—59 to comprehension, 20 to interruption, and nine to reaction. This breakdown reflects the desire for comprehension, willingness to be interrupted, and relative lack of desire for reaction—matching the critical parameters for LSIEs.

### *4.3. Education domain experts*

In a second application of our heuristics, domain experts used the heuristics in an analytical evaluation of the GAWK system. The GAWK is a classroom display that supports teachers and students in completing class-wide and cross-classroom projects. Specifically, the teachers involved in the Classroom BRIDGE effort (Ganoe et al., 2003) used the new heuristics to evaluate an updated version of the GAWK software, a part of Classroom BRIDGE. Fig. 2 provides a screenshot of an early prototype (before testing). We wanted to use domain experts because they have the unique ability to fully grasp the nature of the system and provide insight other evaluators may not have, and heuristics have been used successfully with domain experts in other investigations (Nielsen and Mack, 1994). Furthermore, these people can provide feedback on the format and wordings of the heuristics, illustrating that the heuristics are understandable and usable by a wide range of individuals.



Announcements: NEW << || >>

REMINDER: Lab writeup due April 11!

Activity 1
Soil Anaysis
03/31/2003 13:25:18

Fig. 2. Initial large screen design for the GAWK system. Work artifacts are shown by groups with time on the horizontal axis. Relative effort and progress can be determined by comparison across groups.

### 4.3.1. Method

Since these evaluators are not expert usability engineers, additional materials were provided to them for the evaluation. Specifically, scenarios were provided to illustrate the use of the display and to allow the teachers to get a feel for the display and how it worked. We felt this additional information was necessary for the teachers to understand the display, as they had not used it for about 5 months prior to this evaluation, and to put them in the mind set for assessing the display for usability problems. Additionally, we used structured problem reports (Lavery et al., 1997) to help the teachers capture a description of the problems they experienced or discovered through their inspection of the system. This choice was made because these evaluators had no experience with usability problems, and we felt these structured reports would help these evaluators codify and communicate the issues they found more effectively.

Using the scenarios as guides, each teacher performed the tasks outlined in the scenarios on the large screen. These tasks were done to ensure familiarity with the system so they could understand the interface and the information available in the interface components. After completing the scenarios with the software, they used the heuristics to determine problems with executing those tasks, filling out problem reports for each problem encountered.

There were two forms of data collected in this study: the problem reports and interview feedback on the heuristics. Evaluators provided their own problem reports, detailing the problem found, the applicable heuristic that led to the reporting of said problem, and the severity of the problem. After completing the evaluation, the two evaluators were interviewed jointly. Specific interview topics included how well the heuristics applied to the issues they found and their overall impressions of the heuristics. Interview data was informally captured through note taking.

### *4.3.2. Results*

These non-HCI professionals, who had never heard of heuristic evaluation before, were able to use these heuristics to identify several usability problems with the display. These two evaluators found a total of 23 problems with the system. Furthermore, these evaluators ranked all problems as being moderate to high in terms of severity or need to fix. This rating holds more weight with these particular evaluators, as these are the end users and thus would have a better understanding of the potential impact a problem may have on the actual usage situation. This means that the problems they identified were the most important problems to the actual users of the system.

In terms of using the heuristics, both evaluators stated that they could easily understand the heuristics. They also said they understood how the heuristics applied to the problems they identified with the systems. Neither evaluator suggested that the heuristics were difficult to read or understand, and they were able to relate all of the problems they came across to the heuristics in the set.

The heuristics even helped the teachers understand the purpose of the evaluation:

> “I don’t think I would have understood what you wanted me to do if you didn’t provide me that list.” [referring to the set of heuristics]

They also indicated that the heuristics applied to the system so well that they suggested problems that the teachers had not considered:

> “This list helps me identify what is wrong with the system. I didn’t think about the use of colors and what they mean till I read the list that talked about color.”

### *4.3.3. Discussion*

Analysis of the problems found in this effort shows that these two non-HCI people were able to identify about 40% (23 of 58) of the issues with the GAWK system. This statistic comes from comparing the problem reports in this study to the claims analysis performed on the original GAWK system as reported in Somervell (2004). Furthermore, if we examine empirical findings on heuristic evaluation (Nielsen and Landauer, 1993), we see that this number is right in line with the expected performance for a heuristic evaluation tool, using two evaluators.

It is interesting to see non-HCI people perform a heuristic evaluation of an interface. They start out not knowing what to do and seem frustrated by the seemingly overwhelming nature of the task with which they are faced. It only takes a few moments for them to recognize a problem with the interface, and identify the heuristic(s) that applies. Then they start identifying problems more easily and with more enthusiasm. This lends credit to the validity of this set of heuristics as genuine evaluation support for finding usability problems with large screen information exhibits.

### *4.3.4. Post-analysis*

Analysis of the problems reported by the teachers can reveal more information about the heuristics and how they support evaluation efforts. Specifically, it is worthwhile to consider how the heuristics may suggest re-design guidance for systems, and whether this guidance is directly tied to the critical parameters. This situation with the domain experts provides a unique opportunity to investigate how the heuristics support re-design because the system tested, GAWK, required continuing development. Hence, the results of the evaluation were applied in another development phase.

Inspection of the problem list generated by the teachers through the evaluation suggests that the heuristics support identifying problems that are pertinent to the underlying user goals of the system. This is evident in the nature of the problem, as well as the language used by the teachers when reporting the problems. For example, one teacher identified a problem with the icons used in the system and how it was ‘difficult to track group work over time’. Obviously this problem directly relates to the long-term understanding of the information in the display, clearly illustrating the connection to *comprehension*. Another example problem describes a ‘lack of separation in work icons’, suggesting lack of understanding of the icons and different bodies of work represented therein.

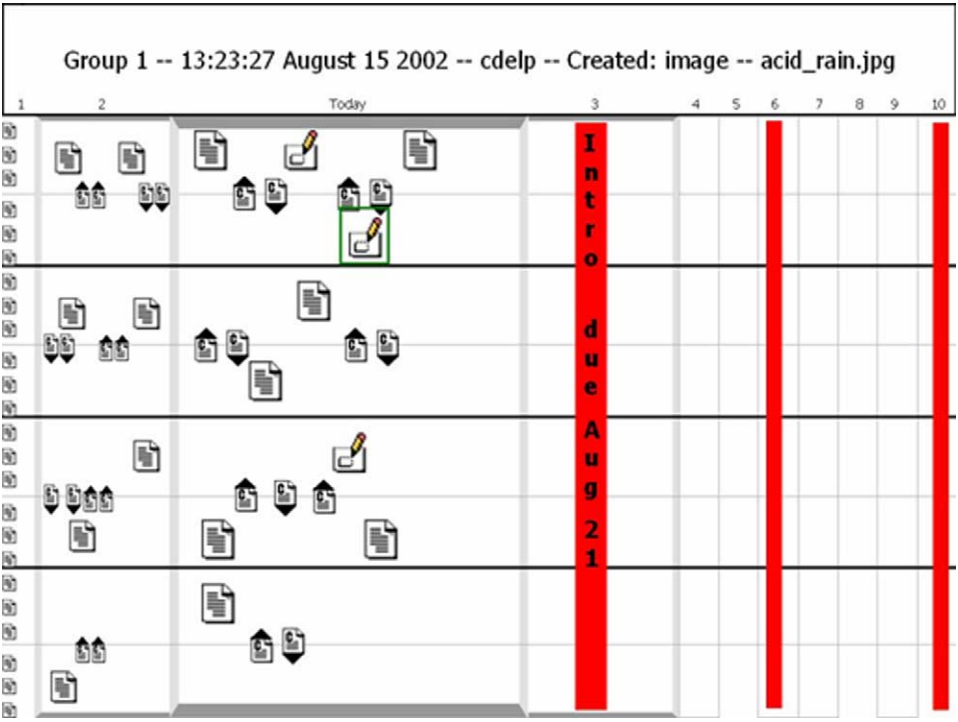
Assessing each of the 23 problems provides an indication as to how many were directly related to the critical parameters. Inspection of the wordings suggests which of the parameters are applicable, as in the previous examples. Nineteen of the 23 problems can be traced to one or more of the critical parameters associated with the GAWK display. Six of the problems were traced to interruption, seven to reaction, and 17 to comprehension. The sum is higher than the total number of problems because some problems were traced to multiple parameters. We believe that this high correlation between the problems and the parameters is a direct result of the critical parameter basis for the heuristic creation process. The implication is that the new heuristics address problems that pertain to the critical parameters for the system class, thereby providing important re-design considerations.

### *4.3.5. GAWK re-design*

Because the programmers involved with the Classroom BRIDGE effort did not have knowledge of the IRC framework, it was necessary to shield them from describing the problems in terms of interruption, reaction, and comprehension. Instead, the problems were described and discussed through language that referred to ‘supporting understanding’ or ‘preventing too much distraction’ or ‘allowing quick decisions’. These terms were understood by the programmers, and the pivotal concerns surrounding the GAWK display were addressed without reliance upon the IRC terminology.

It is further necessary, in supporting the programmers, to group problems into categories that correspond to artifacts within the interface. In the case of the GAWK system, there are distinct ‘parts’ of the display in which the problems occur. For example, there is a banner area near the top of the display. This banner area is created in a specific part of the code and any changes will have to be made in that part of the code. Grouping related problems into these parts can help the programmers as they address the problems and make changes. By providing the problems to the programmers in terms of interface artifacts, rather than in terms of interruption, reaction, and comprehension goals, the programmers are better able to make effective changes.

To understand how the GAWK display changed we provide Fig. 3. Comparing this new design to the earlier instance of the GAWK display (Fig. 2), we see some important changes. Overall the structure looks cleaner and the color scheme supports reading from a distance. As an example, consider the changes made to the banner design, including multi-line announcements, a new color scheme, separating announcements from artifact information, and more total space (Ganoe et al., 2003). These changes resulted from specific problems reported by the teachers, that impacted the comprehension and interruption parameters. However, these problems were reported to the programmers grouped in relation to the banner artifact, with wordings that required improvements for ‘user understanding’ or ‘decreasing distraction’. It is important that identified problems are given to the programmers in terms that they understand. In this case, the programmers did not have knowledge of the IRC framework and the critical parameters associated with the LSIE system class. However, these programmers did understand the underlying user goals associated with the display, in terms of supporting typical user tasks. It is



Group 1 -- 13:23:27 August 15 2002 -- cdelp -- Created: image -- acid_rain.jpg

Fig. 3. Redesigned GAWK. Overall color schemes changed. Announcements and artifact information is separated. Overall look and feel is polished. Compare to Fig. 2.

encouraging that the problems identified by the teachers through the evaluation were mappable to the underlying parameters associated with the LSIE class. This apparent connection between the problems and the underlying user goals could suggests that more robust techniques are possible, with which evaluators and designers could couple the results of heuristic evaluation to direct values for each parameter, facilitating the assessment of whether an LSIE system supports its intended purpose.

## 5. General discussion

Both examples of using the heuristics in system evaluation provided valuable insight into how the heuristics support these types of efforts. There are two key areas related to the development and use of these heuristics from which important ramifications arise: reporting problems to developers and assessing work/benefit tradeoffs.

### *5.1. Reporting problems to developers*

One of the most important aspects of usability evaluation is reporting the results to the developers in a format that is understandable. This implies a need for concisely worded statements that reflect specific changes to be made in the system. This can be difficult when evaluation is not focused on critical parameters, especially for analytic methods that rely upon experts who may not be familiar with the parameters for a specific system class. This problem reporting can be further blurred because the developers often do not know the specific terminology associated with the critical parameters for a system class. But, when a developer decides to build an LSIE system, without fully understanding the critical parameter concept, their design/evaluation cycle is guided by the parameters that they indirectly selected—the parameters associated with LSIE systems. Restricted knowledge of these critical parameters necessitates careful problem wording if effective design changes are to be achieved.

Fortunately, effective heuristics can remedy this situation. As seen in the case of the domain experts, problems are often closely tied to one or more of the underlying critical parameters for a system. It is natural and straightforward to describe problems in terms of user goals and the associated interface artifacts that hinder those goals, without explicit references to the critical parameters. Based on our comparison experiment and usage examples, our heuristics can guide developers toward a design that is more in line with the critical parameters, at least for LSIE systems.

Several researchers point out that, in general, heuristics are poor at providing problem descriptions that capture the underlying user goals (e.g. Sauro, 2004; Cockton and Woolrych, 2002). Critical parameter-based heuristics are a step in the right direction, because they are closely related to the critical parameters of the system class.

### *5.2. Balancing costs and benefits*

In considering the new heuristics and the critical parameter based creation process, one must confront the issue of development costs and long term benefits. It took two researchers 6 weeks of effort to come up with the final set of eight heuristics tailored to the LSIE system class. This effort consisted of both individual and group work and analysis. Typical work schedules involved 5–10 h per week by each individual in inspecting systems, creating scenarios, performing claims analysis, classifying claims according to critical parameter impacts, and categorizing claims into scenario based design categories. This was followed by a separate 1–3 h weekly meeting between the researchers, to assess each other's work and to reach consensus at every step. However, when one considers that the process was being refined and evaluated during the same time frame, the actual effort to produce heuristics from the process would be about one-half to three-fourths of this time (say 3–5 weeks) of part-time work.

Certainly this amount of work is not small. When is the work in creating heuristics worthwhile? There are three major considerations. First, targeted heuristics have higher thoroughness and validity scores in UEM comparison tests (Somervell and McCrickard, 2004). This suggests that the more specific heuristics can find more of the real problems with an interface in the first evaluation phase. This invariably reduces downstream development costs because problems are easier to fix earlier in the software development cycle. This benefit becomes even more valuable for entities who specialize in similar types of systems and perform many evaluations across multiple interfaces. A group that specializes in a user interface area with a well-defined IRC level could benefit from tailored heuristics. Long term reductions in evaluation costs can mean more projects because evaluation time is shortened.

Second, because targeted heuristics have high reliability scores (Somervell and McCrickard, 2004), this suggests that we can feel confident in our results with fewer evaluators. This is important for problematic domains like in-vehicle information systems, emergency response systems, and mission-critical systems where domain experts are rare or non-analytic evaluation techniques are costly. The higher reliability allows us to feel confident in our re-design guidance when faced with limited evaluation resources.

Third, these heuristics are tightly coupled with the critical parameters of the system class. This is important because design decisions and changes are made to address these pivotal concerns. Perfecting a system according to the appropriate levels of the critical parameters ensures that the system performs its intended function. Without a focus on the critical parameters, system developers are faced with the challenge of making design changes based on more simple usability metrics, which may or may not be adequate for improving design.

## 6. Conclusions and future work

This paper presents a method for creating heuristics, centered on the identification of critical parameters for a domain of interest. These focused heuristics showed promise in an analytic comparison to more generic ones, and we described two applications of the heuristics that found problems with large screen information exhibits (LSIEs). Thus, we hope this work strengthens the argument for critical parameters and supports our structured heuristic creation process. While our previous efforts have supported the critical parameter creation process (Somervell and McCrikard, 2004; Somervell et al., 2003b), our two examples of use (Section 4) more clearly illustrate the utility of a tailored evaluation tool. More importantly, they illustrate one of the strengths of critical parameters—allowing the creation of tailored evaluation tools that produce results focused on the primary intentions of the designer.

Mapping results from our heuristics back to the critical parameters is important for quantification of user comments. This mapping allows us to track levels of the parameters and feed into redesign, which in turn could further map to new user interface components and claims, providing a clear development history that could facilitate future reuse efforts. While our heuristics seemed to generate comments reasonably well connected with our critical parameters, certainly there is room for improvement. A possible way to improve this mapping is to augment our creation process to find which ratings correspond to each heuristic. Quantifying each of the heuristics in terms of the affected critical parameters would allow a designer to more readily assess how to correct problems found through the heuristics by focusing design effort on the most important aspects surrounding the critical parameters. That is, one could rate the heuristics for a given system with different scores for each critical parameter, such that a high score on one heuristic might indicate a need to change the interface to greatly increase one critical parameter, slightly decrease another, and keep the third about the same. Identifying these mappings could provide benefits to further address some of the downsides to traditional heuristic evaluation (e.g. Sauro, 2004; Cockton and Woolrych, 2002). However, great care must be taken to not limit the creative free responses of the evaluators—a strength of heuristic evaluation that often yields problems that the designer did not consider.

Building on the use of critical parameters as an enabler of effective design, our ongoing work considers their impact throughout the design process, including participatory design Ndiwalana et al. (2004), analytic evaluation Lee et al. (2004), and empirical evaluation Chewar et al. (2004b). By using a common set of critical parameters throughout these and other stages of design, we are exploring to what degree knowledge can be shared within and between design efforts. Many of these efforts are instantiated in a claims library, our LINK-UP design system, and accompanying visualization tools that provide constant access to problem and design claims throughout the stages of design (Payne et al., 2003; Chewar et al., 2004a; Wahid et al., 2004a,b). We seek to learn how a design approach based on critical parameters can help design professionals in various fields, domain experts with little design knowledge, and students seeking to learn design processes.

We chose to embrace critical parameters because of their promise in supporting successful design and evaluation. Our work has shown one important aspect that critical parameters bring to evaluation, that of focused effort on the important user goals surrounding a particular system type. There are other strengths of critical parameters (e.g. Newman, 1997) and we hope that the HCI community will commit to establishing parameters for all computing areas. We hope our work provides a springboard not only for establishing critical parameters and heuristics for other system classes, but also to pursue other rigorous, scientific approaches to HCI research.

## Appendix A. LSIE heuristics

*Appropriate color schemes should be used for supporting information understanding*. Try using cool colors such as blue or green for background or borders. Use warm colors like red and yellow for highlighting or emphasis.

*Layout should reflect the information according to its intended use*. Time based information should use a sequential layout; topical information should use categorical, hierarchical, or grid layouts. Screen space should be delegated according to information importance.

*Judicious use of animation is necessary for effective design*. Multiple, separate animations should be avoided. Indicate current and target locations if items are to be automatically moved around the display. Introduce new items with slower, smooth transitions. Highlighting related information is an effective technique for showing relationships among data.

*Use text banners only when necessary*. Reading text on a large screen takes time and effort. Try to keep it at the top or bottom of the screen if necessary. Use sans serif fonts to facilitate reading, and make sure the font sizes are big enough.

*Show the presence of information, but not the details*. Use icons to represent larger information structures, or to provide an overview of the information space, but not the detailed information; viewing information details is better suited to desktop interfaces. The magnitude or density of the information dictates representation mechanism (text vs icons for example).

*Using cyclic displays can be useful, but care must be taken in implementation*. Indicate ‘where’ the display is in the cycle (i.e. 1 of 5 items, or progress bar). Timings (both for single item presence and total cycle time) on cycles should be appropriate and allow users to understand content without being distracted.

*Avoid the use of audio*. Audio is distracting, and on a large public display, could be detrimental to others in the setting. Furthermore, lack of audio can reinforce the idea of relying on the visual system for information exchange.

*Eliminate or hide configurability controls*. Large public displays should be configured one time by an administrator. Allowing multiple users to change settings can increase confusion and distraction caused by the display. Changing the interface too often prevents users from learning the interface.

## References

Abowd, G.D., Atkeson, C.G., Brotherton, J., Enqvist, T., Gulley, P., LeMon, J., 1998. Investigating the capture, integration, and access problem of ubiquitous computing in an educational setting. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI’98). pp. 440–447.

Baker, K., Greenberg, S., Gutwin, C., 2002. Empirical development of a heuristic evaluation methodology for shared workspace groupware. In: ACM Conference on Computer Supported Cooperative Work (CSCW’02). New Orleans, LA, pp. 96–105.

Berry, B., 2003. Adapting heuristics for notification systems. In: Proceedings of 41st Annual ACM Southeast Conference (ACMSE’03). Savannah, GA, pp. 144–149.

Carroll, J.M., Rosson, M.B., 1992. Getting around the task-artifact cycle: how to make claims and design by scenario. ACM Transactions on Information Systems (TOIS) 10 (2), 181–212.

Chewar, C.M., Bachetti, E., McCrickard, D.S., Booker, J., 2004. Automating a design reuse facility with critical parameters: lessons learned in developing the link-up system. In: Proceedings of the 2004 International Conference on Computer-Aided Design of User Interfaces (CADUI '04). Island of Madeira, Portugal.

Chewar, C.M., McCrickard, D.S., Sutcliffe, A.G., 2004. Unpacking critical parameters for interface design: Evaluating notification systems with the irc framework. In: Proceedings of the 2004 Conference on Designing Interactive Systems (DIS '04). pp. 279–288.

Churchill, E.F., Nelson, L., Denoue, L., Girgensohn, A., 2003. The plasma poster network: Posting multimedia content in public places. In: Ninth IFIP TC13 International Conference on Human–Computer Interaction (INTERACT03). Zurich, Switzerland, pp. 599–606.

Cockton, G., Woolrych, A., 2002. Sale must end: should discount methods be cleared off hci's shelves? interactions september + october, 13–18.

Dix, A., Finlay, J., Abowd, G.D., Beale, R., 2004. Human Computer Interaction. Pearson/Prentice Hall, Harlow, England.

Elrod, S., Bruce, R., Gold, R., Goldberg, D., Halasz, F., Janssen, W., Lee, D., McCall, K., Pedersen, E., Pier, K., Tang, J., Welch, B., 1992. Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'92). ACM Press, pp. 599–607.

Ganoe, C., Somervell, J., Neale, D., Isenhour, P., Carroll, J.M., Rosson, M.B., McCrickard, D.S., 2003. Classroom bridge: Using collaborative public and desktop timelines to support activity awareness. In: Proceedings of the ACM Conference on User Interface Software and Technology (UIST '03). pp. 21–30.

Greenberg, S., Kuzuoka, H., 2000. Using digital but physical surrogates to mediate awareness, communication and privacy in media spaces. Personal Technologies 4 (1).

Greenberg, S., Rounding, M., 2001. The notification collage: Posting information to public and personal displays. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '01). Seattle, WA, pp. 515–521.

Greenberg, S., Fitzpatrick, G., Gutwin, C., Kaplan, S., 1999. Adapting the locales framework for heuristic evaluation of groupware. In: Proceedings of OZCHI. pp. 28–30.

Gutwin, C., Greenberg, S., 2000. The mechanics of collaboration: Developing low cost usability evaluation methods for shared workspaces. In: IEEE Ninth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE'00).

Hartson, H.R., Andre, T.S., Williges, R.C., 2001. Criteria for evaluating usability evaluation methods. International Journal of Human–Computer Interaction 13 (4), 373–410.

Hix, D., Hartson, H.R., 1993. Developing User Interfaces: Usability through Product and Process. Wiley, New York.

Ishii, H., Ullmer, B., 1997. Tangible bits: Towards seamless interfaces between people, bits, and atoms. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '97). Atlanta, GA, pp. 234–241.

Jedrysik, P., 2001. Interactive large-screen displays for command and control. online: retrieved Oct. 13, 2004. URL http://www.afrlhorizons.com/Briefs/Sept01/IF0012.html.

Lavery, D., Cockton, G., Atkinson, M., 1997. Comparison of evaluation methods using structured usability problem reports. Behaviour and Information Technology 16 (4/5), 246–266.

Lee, J.C., Lin, S., Chewar, C.M., McCrickard, D.S., 2004. From chaos to cooperation: Teaching analytic evaluation with LINK-UP. In: Proceedings of the World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education (E-Learn '04), Washington DC, November 2004, pp. 2755–2762.

Mankoff, J., Dey, A.K., Hsieh, G., Kientz, J., Lederer, S., Ames, M., 2003. Heuristic evaluation of ambient displays. In: Proceedings of the conference on Human factors in computing systems. ACM Press, pp. 169–176.

McCrickard, D.S., Chewar, C.M., 2003. Attuning notification design to user goals and attention costs. Communications of the ACM 46 (3), 67–72.

McCrickard, D.S., Wrighton, D., Bussert, D., 2002. Supporting the construction of real world interfaces (tech note). In: Proceedings of the 2002 IEEE Symposia on Human Centric Computing Languages and Environments (HCC'02). Arlington VA, pp. 54–56.

McCrickard, D.S., Chewar, C.M., Somervell, J.P., Ndiwalana, A., 2003. A model for notification systems evaluation: assessing user goals for multitasking activity. ACM Transactions on Computer–Human Interaction (TOCHI) 10 (4), 312–338.

Molich, R., Nielsen, J., 1990. Improving a human–computer dialogue: what designers know about traditional interface design. Communications of the ACM 33 (3), 338–348.

Ndiwalana, A., Chewar, C.M., Bussert, D., Somervell, J., McCrickard, D.S., 2003. Ubiquitous computing: By the people, for the people. In: Proceedings of the 41st Annual ACM Southeast Conference. Savannah, GA, pp. 24–29.

Ndiwalana, A., Kampanya, N., McEwan, I., Chewar, C.M., McCrickard, D.S., Pious, K., 2004. A tool for participatory negotiation: Linking-up participatory design and design knowledge reuse. In: Proceedings of the Eighth Biennial Participatory Design Conference (PDC '04). Vol. 2. Toronto, Canada, pp. 118–122.

Newman, W.M., 1997. Better or just different? on the benefits of designing interactive systems in terms of critical parameters. In: Proceedings of Designing Interactive Systems (DIS'97). pp. 239–245.

Nielsen, J., Landauer, T.K., 1993. A mathematical model of the finding of usability problems. In: Proceedings of the SIGCHI conference on Human factors in computing systems. ACM Press, pp. 206–213.

Nielsen, J., Mack, R.L., 1994. Usability Inspection Methods. Wiley, New York.

Payne, C., Allgood, C.F., Chewar, C.M., Holbrook, C., McCrickard, D.S., 2003. Generalizing interface design knowledge: Lessons learned from developing a claims library. In: Proceedings of the 2003 IEEE International Conference on Information Reuse and Integration (IRI '03).

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T., 1994. Human Computer Interaction. Addison Wesley, Reading, MA.

Preece, J., Rogers, Y., Sharp, H., 2002. Interaction Design: Beyond Human–Computer Interaction. Wiley, New York.

Rosson, M.B., Carroll, J.M., 2002. Usability Engineering: Scenario-Based Development of Human–Computer Interaction. Morgan Kaufman, New York.

Russell, D.M., Drews, C., Sue, A., 2002. Social aspects of using large public interactive displays for collaboration. In: Proceedings of the Fourth International Conference on Ubiquitous Computing (UbiComp 2002). Goteburg, Sweden, pp. 229–236.

Sauro, J., 2004. Premium usability: getting the discount without paying the price. Interactions 11 (4), 30–37.

Shneiderman, B., 1998. Designing the User Interface: Strategies for Effective Human–Computer Interaction. Addison-Wesley, Reading, MA.

Skog, T., Holmquist, L.E., 2000. Webaware: Continuous visualization of web site activity in a public space. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'00) Extended Abstracts. The Hague, Netherlands, pp. 351–352.

Somervell, J., 2004. Developing heuristic evaluation methods for large screen information exhibits based on critical parameters. Ph.D. thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA.

Somervell, J., McCrickard, D.S., 2004. Comparing generic vs specific heuristics: Illustrating a new UEM comparison technique. In: Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting (HFES'04), New Orleans LA, September 2004, pp. 2480–2484.

Somervell, J., Chewar, C.M., McCrickard, D.S., Ndiwalana, A., 2003. Enlarging usability for ubiquitous displays. In: Proceedings of the 41st Annual ACM Southeast Conference (ACMSE'03). pp. 24–29.

Somervell, J., Wahid, S., McCrickard, D.S., 2003. Usability heuristics for large screen information exhibits. In: Proceedings of the Ninth IFIPTC13 International Conference on Human Computer Interaction (INTERACT'03). Zurich, Switzerland, pp. 904–907.

Wahid, S., Allgood, C.F., Chewar, C.M., McCrickard, D.S., 2004. Entering the heart of design: Relationships for tracing claim evolution. In: Proceedings of the 16th International Conference on Software Engineering and Knowledge Engineering (SEKE '04). Banff, Alberta, Canada.

Wahid, S., Smith, J.L., Berry, B., Chewar, C.M., McCrickard, D.S., 2004. Visualization of design knowledge component relationships to facilitate reuse. In: Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration (IRI '04). Las Vegas, NV.